



DISSERTATION

Glass Formation of Colloids in Porous Media

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der Naturwissenschaften unter der Leitung von

Ao. Univ.-Prof. Dr. Gerhard Kahl

Institut für Theoretische Physik
Technische Universität Wien

eingereicht an der Technischen Universität Wien
Fakultät für Physik

von

Jan Kurzidim, M.Sc.

Matrikelnummer 0647153

Wien, im Juni 2013

Kurzfassung

Bringt man eine Flüssigkeit mit einem porösen Medium in Berührung, so wirkt sich dies tiefgreifend auf die Eigenschaften der Flüssigkeit aus. Dieses Phänomen spielt beispielsweise bei der Förderung von Erdöl oder der Bewegung von Eiweißmolekülen in Zellplasma eine maßgebliche Rolle. Bei zahlreichen Vorgängen dieser Art bewegt sich die Flüssigkeit „langsam“ hinsichtlich der Umordnung ihrer Teilchen, was von besonderer Bedeutung im Zusammenhang mit porösen Medien ist, da diese eine solche Flüssigkeit nicht nur verlangsamen, sondern unter Umständen auch beschleunigen können. Welche dieser beiden Möglichkeiten tatsächlich eintritt, hängt dabei von den Eigenschaften der Flüssigkeit und des Mediums sowie von den äußeren Bedingungen ab. In dieser Arbeit befassten wir uns mit diesem Effekt sowie darüber hinaus mit den dynamischen Eigenschaften von Flüssigkeiten in porösen Medien im Allgemeinen. Unsere Untersuchungen basierten auf umfangreichen Computersimulationen eines vereinfachten Modells, in dem sich die Teilchen einer Flüssigkeit durch ein poröses Medium aus ungeordneten, räumlich fixierten Teilchen bewegten. Die im Rahmen dieses Modells verwendeten Teilchen wurden dabei nach dem Vorbild kolloidaler Suspensionen modelliert und wechselwirkten mittels besonders einfacher Kräfte. Bei der Erforschung dieser Modellsysteme widmeten wir uns insbesondere dem sogenannten Glasübergang, bei dem das Abkühlen einer Flüssigkeit bewirkt, dass die Umordnung ihrer Teilchen bei einer Temperatur weit über dem Nullpunkt vollständig zum Erliegen kommt. Dies ist gleichbedeutend mit der Bildung eines amorphen Festkörpers. Der Glasübergang bot sich aus zwei Gründen als Studienobjekt an: Einerseits ist trotz jahrzehntelanger Forschung noch immer nicht genau bekannt, welche physikalischen Mechanismen diesen Übergang verursachen; andererseits trifft ein vor wenigen Jahren ausgearbeitetes theoretisches Konzept umfassende Voraussagen zu Glasübergängen in unserem Modell. Unsere Arbeit förderte zahlreiche aufschlussreiche Ergebnisse zutage: Unter anderem konnten wir ungewöhnliche Glasübergänge identifizieren, atypische Teilchenbewegungsarten beobachten und die Porenstruktur des Modellmediums im Detail charakterisieren. Darüber hinaus konnten wir wesentliche Einblicke in die physikalischen Zusammenhänge gewinnen, die unseren Beobachtungen zugrunde liegen, wodurch unsere Erkenntnisse signifikant zum Verständnis der dynamischen Eigenschaften von Flüssigkeiten in porösen Medien beitragen.

Abstract

Confining a liquid to a disordered medium may drastically change the physical properties of the liquid. This impact of confinement is crucial to processes like the extraction of mineral oil from porous rocks or the motion of proteins in cytoplasm. In those cases, as in many others, the relevant feature of the confined liquid is its slow dynamics, i.e., the slow propagation of its particles. It is therefore intriguing that depending on the details of the system, the introduction of confinement may *accelerate* a liquid's dynamics—or *decelerate* it. This work aims at elucidating this phenomenon in particular, and the dynamics of liquids in disordered media in general. We approach this challenge by means of extensive computer simulations of a model in which the particles of a liquid permeate a medium of immobile, disordered particles. We consider this model for particularly simple particles as they occur in nature in the form of colloids, i.e., nano- to micrometer-sized particles in a solute. In our study of the model, we devote special attention to its glass transitions, i.e., to the phenomenon that upon cooling or compressing a liquid, its particles entirely cease to propagate and a disordered solid is formed at a *nonzero* temperature. We focus on this issue for two reasons: firstly, the nature of the glass transition is still a matter of debate, and secondly, a recently-developed theoretical framework makes detailed predictions for this transition within our model. As the principal achievements of our work, we unveiled unconventional glass transitions in the model, exposed unusual types of particle propagation in it, and characterized in detail the pore structure of its confinement. Together with the insights we gained on the underlying physical mechanisms, our findings contribute significantly to a deeper understanding of the dynamics of liquids in disordered media.

“With four parameters I can fit an elephant,
and with five I can make him wiggle his trunk.”

John von Neumann

Contents

Preface	1
1 Background	5
1.1 Colloids	5
1.1.1 Coarse-grained picture	7
1.1.2 Colloids in experiments	8
1.1.3 Hard spheres	10
1.1.3.1 Definition	10
1.1.3.2 Basic properties	11
1.1.3.3 Significance	13
1.2 Glasses	15
1.2.1 History and applications of glass	15
1.2.2 Supercooling	17
1.2.3 Dynamic arrest	19
1.2.4 In the glass	20
1.2.5 A genuine transition?	22
1.2.6 Static features of slow liquids	24
1.2.7 Dynamic features of slow liquids	26
1.3 Porous confinement	29
1.3.1 Significance of fluids in confinement	30
1.3.2 Observed phenomena in confinement	32
1.3.3 Microscopic influence of confinement	34
1.3.3.1 General remarks	34
1.3.3.2 Frustration	36
1.3.3.3 Percolation	38
1.3.4 Modeling of disordered porous confinement	40
1.3.4.1 Characterization and modeling strategies	40
1.3.4.2 The quenched-annealed model	41
1.4 Theoretical predictions	44
1.4.1 Theories of the glass transition	44
1.4.2 Mode-coupling theory	45
1.4.3 Replica MCT and its predictions	48
2 Methods	55
2.1 Concepts	55

2.1.1	Overview	55
2.1.2	Statistical mechanics	58
2.1.3	Molecular dynamics	60
2.1.4	Monte Carlo	62
2.2	Event-driven molecular dynamics	65
2.2.1	Physical principle	65
2.2.1.1	Overview	65
2.2.1.2	Finding the next collision	66
2.2.1.3	Propagating the particles	69
2.2.1.4	Performing the collision	69
2.2.2	Optimization strategies	72
2.2.2.1	Periodic boundary conditions	72
2.2.2.2	Bucket lists	75
2.2.2.3	Reuse of previously-calculated events	79
2.2.2.4	Particle-local times	83
2.2.2.5	Efficient event list structure	85
2.3	Initial states of quenched-annealed systems	90
2.3.1	Overview	90
2.3.2	Established packing methods	91
2.3.2.1	Simple methods	91
2.3.2.2	Elaborate methods	92
2.3.3	Custom method	93
2.4	Geometry of the accessible volume	95
2.4.1	Traps and the percolating void	95
2.4.2	Definition of the Delaunay tessellation	100
2.4.3	Characterization of the accessible volume	102
2.4.4	Construction of Delaunay tessellations	105
2.4.4.1	Available packages and known algorithms	106
2.4.4.2	Auxiliary: Circumspheres and -circles	109
2.4.4.3	Auxiliary: Nearest-neighbor search	113
2.4.4.4	Custom construction algorithm	117
2.4.5	Percolation algorithm	121
2.4.6	Rastering algorithm	124
2.4.6.1	Given point within given Delaunay cell	124
2.4.6.2	Accessible volume within Delaunay cell	126
2.5	Observables	127
2.5.1	Observables in computer simulations	128
2.5.1.1	Phase-space density vs. observables	128
2.5.1.2	Time-dependent observables	129
2.5.1.3	Correlators	130
2.5.1.4	Logarithmic time scales	131
2.5.1.5	Quenched disorder	132
2.5.1.6	Reduced units	133
2.5.2	Static observables	133

2.5.2.1	Radial distribution function	133
2.5.2.2	Static structure factor	134
2.5.3	Dynamic observables	136
2.5.3.1	Mean squared displacement	136
2.5.3.2	Collective intermediate scattering function	137
2.5.3.3	Connected intermediate scattering function	138
2.5.3.4	Single-particle intermediate scattering function	140
2.5.3.5	Decay quantification and long-time value subtraction	141
2.5.3.6	Van Hove correlation function	142
2.5.4	Accessible volume quantifiers	144
2.5.4.1	Accessibility probability: grains, D-faces, D-cells	145
2.5.4.2	Probabilities for percolation and trapping	146
2.5.4.3	Trap-size distributions	147
2.5.5	Equilibration and error estimation	149
3	Results	155
3.1	General remarks	155
3.2	Full fluid	156
3.2.1	Mean squared displacement: Kinetic diagram	156
3.2.2	Paths and points in the parameter space	158
3.2.3	Static structure	159
3.2.4	Mean squared displacement: Paths	163
3.2.5	Intermediate scattering functions: Paths	166
3.2.6	Intermediate scattering functions: Kinetic diagrams	171
3.2.7	Total intermediate scattering function	173
3.2.8	Long-time value of the single-particle ISF	174
3.2.9	Self part of the Van Hove function	178
3.3	Structure of the accessible volume	182
3.3.1	Accessibility probabilities	186
3.3.2	Probabilities for percolation and trapping	188
3.3.3	Trap-size distributions	190
3.3.4	Moments of the trap-size distributions	192
3.4	Free and trapped particles	194
3.4.1	Mean squared displacement	194
3.4.2	Single-particle intermediate scattering function	200
3.4.3	Self part of the Van Hove function	203
3.4.4	Fluid–fluid trapping effects	209
	Conclusion	213
	Appendix	219
A.1	Network picture of the accessible volume	219
A.1.1	Voronoi picture	219
A.1.1.1	Voronoi tessellation	219
A.1.1.2	Voronoi network	221

A.1.2 Delaunay picture	225
A.1.2.1 From Voronoi to Delaunay	225
A.1.2.2 Network pictures: Delaunay \neq Voronoi	226
A.1.2.3 Additional Delaunay tessellation properties	228
A.1.2.4 Accessible volume in void: Delaunay = Voronoi	233
A.2 Finite-size effects in simulations	235
A.3 Validation and performance of numeric algorithms	238
A.3.1 Validation of molecular-dynamics algorithm	238
A.3.1.1 Conserved properties	238
A.3.1.2 Comparison with literature	240
A.3.2 Performance of molecular-dynamics algorithm	241
A.3.3 Performance of initial-states algorithm	245
A.3.4 Validation of Delaunay-tessellation algorithm	246
A.3.5 Performance of Delaunay-tessellation algorithm	247
References	251
Curriculum Vitae	267

Preface

The first thing to take care of when writing a thesis, they say, is the acknowledgments. Because once the work is done, you won't feel like thanking anyone anymore. I dared to schedule this task to the very end, and yet I am still widely aware of the fact that no thesis is the feat of just one person. Therefore, before commencing the scientific part of this work, I would like to credit those people without who this thesis would not have been possible.

Above all I would like to thank my parents, Elisabeth and Dieter, who made me the curious person that I am. On countless occasions, they encouraged my interest in so many different things—be it science, computers, languages, sports, music, or theater. Of my many memories of them, let me recount just this one: the nights when my father would go out with me to stargaze. I cherish those nights not because of the things we saw—we only had field glasses at our hands—but because of the shared fascination, the time spent together. I felt understood. I still do.

Then of course school. I am fortunate to have been educated, over the years, by many excellent teachers, particularly in physics, biology, geography, politics, history, and English. Pivotal in my eventual decision to take up the study of physics was Peter Stock, whose physics, mathematics, and computer science lessons and whose support I enjoyed for years. My serious advice to those seeking to attract pupils to science: look out for teachers knowing how to marvel and motivate like he does.

Following my undergraduate studies in Stuttgart, I moved to Lawrence, Kansas, where I spent an unforgettable time as a graduate student. It was there that I encountered teachers as terrific as Jack Shi, Doug McKay, Steve Shawl, and Tom Cravens, and it was also there that for the first time I was offered the opportunity to blend physics research with my other favorite pastime, computers.

Kansas also turned out to be the place where I would meet the most important person of my life—my wife Lina. In our years since then, we have stood side by side in dire times and have shared the blissful ones. It is in large parts because of your encouragement that I eventually found the will and strength to complete this thesis. You complete me. And you don't know how much I admire your resilience.

When we moved to Vienna, I decided I wanted to emphasize working with computers. For this purpose, I took the step to switch to the field of condensed-matter physics. My door-opener to the field was Franz Vesely, who I am glad to count among the reviewers of this thesis. I vividly remember his fascination with the algorithm that

later would wind up being at the heart of this thesis, and it is because of him that I became convinced to have chosen the right field.

In the further pursuit of my studies, fortune had it that I took up working with Gerhard Kahl, who at the time was just about to embark on the project that the lion's share of this thesis would be based on. Gerhard, who I have now been working with for over five years and who I immensely value, is not only masterful at establishing an environment of scientific appreciation (not just for me, but for everyone under his radar), but he also provided much moral support in times of personal hardship.

Joining Gerhard's project shortly after myself was Daniele Coslovich, who I consider my mentor in terms of glassy physics and of many other things. I profited so much from his extensive knowledge and critical thinking, and I greatly enjoyed sparring opinions with him over a vast range of topics, scientific or not. Hard-working and ever-open to new ideas and tools as he is, I have very much missed the dynamics in our office since he left.

Last but not least there are the many other members of the Soft Matter Theory group who have made the sometimes long hours at the TU bearable and enjoyable. Dieter Schwanzer, with whom at times I would spend hours chit-chatting about the latest computer stuff and who is the good heart of the group. Giannis Georgiou, Martina Lechner, and Ismene Kolovos, my dear office mates who were most always more industrious than I and who took good care of the plants when I happened to be away. Moritz Antlanger and Julia Fornleitner, with whom I just felt at ease to talk about just about anything. As well as all the others, Andi Tröster, Camille Jovvie, Cecilia Bores, Emanuela Bianchi, Gernot Pauschenwein, Günther Doppelbauer, Lukas Strauss, Mario Kahn, Marta Montes, Silvano Ferrari, and Ulf Pedersen, thank you for giving me that sense of belonging.

Now for some more technical issues. As alluded to in the above, much of this thesis was carried out in the framework of an externally funded project. The agency providing the funding was the FWF (the Austrian Science Fund), P19890-N16 having been the project number^{P.1}, and my position was that of a project assistant ("Projektsassistent"). Over the course of this project, I have had the privilege to be the principal author of four original scientific publications, which at the time of this writing have received a total of 59 citations according to the ISI Web of Science^{P.2}:

- (i) JAN KURZIDIM, DANIELE COSLOVICH, and GERHARD KAHL.
Single-Particle and Collective Slow Dynamics of Colloids in Porous Confinement.
Physical Review Letters **103**, 138303 (2009).
- (ii) JAN KURZIDIM, DANIELE COSLOVICH, and GERHARD KAHL.
Impact of random obstacles on the dynamics of a dense colloidal fluid.
Physical Review E **82**, 041505 (2010).

^{P.1}<http://www.fwf.ac.at/de/abstracts/abstract.asp?L=D&PROJ=P19890>

^{P.2}<http://apps.webofknowledge.com>

- (iii) JAN KURZIDIM and GERHARD KAHL.
Accessible volume in quenched-annealed mixtures of hard spheres: a geometric decomposition.
Molecular Physics **109**, 1331 (2011).
- (iv) JAN KURZIDIM, DANIELE COSLOVICH, and GERHARD KAHL.
Dynamic arrest of colloids in porous environments: disentangling crowding and confinement.
Journal of Physics: Condensed Matter **23**, 234122 (2011).

Further, I participated in a publication that was cited another four times:

- (v) DIETER SCHWANZER, DANIELE COSLOVICH, JAN KURZIDIM, and GERHARD KAHL.
Effects of porous confinement on the structural properties of the Gaussian core model.
Molecular Physics **107**, 433 (2009).

Finally, concurrent to the preparation of this thesis, I served as the co-author of a textbook for students of the Ukrainian language. While topically entirely unrelated to the thesis, I feel that mentioning the textbook is justified given the delay it caused to the completion of this work. For the unlikely case of potential buyers, the bibliographical information of the textbook reads:

- (vi) LINA KLYMENKO and JAN KURZIDIM.
Разом: Українськi для Anfängerinnen und Anfänger.
(Verlag Holzhausen, Vienna, 2012).

From the organizational point of view, the thesis is divided into four major parts:

- (1) a **background** chapter introducing all relevant physical aspects,
- (2) a **methods** chapter describing in detail the utilized procedures,
- (3) a **results** chapter largely recounting the findings of articles –, and
- (4) an **appendix** containing two vital proofs.

The appendix is included in the list for the reason that the proofs therein are crucial to the validity of parts of this work. The proofs address an issue that I uncovered only after all of the above-listed journal articles had already been published and that threatened the justification of some of the conclusions in the articles. Fortunately, according to the proofs, all conclusions in question eventually remain valid.

Over the process of compiling the thesis, I gradually evolved its design towards being self-contained. This essentially is a response to two impulses. On the one hand, to some degree I sought to vindicate the nonexistence of my Master's thesis. On the other hand, I simply gave in to my perfectionist nature and to my urge to be educational. The result is visible in the background chapter, and even more so in the methods chapter, in that those chapters are quite extensive—considerably more so than corresponding chapters in most comparable Ph.D. theses. Following the same impulses, I chose to include various references to Wikipedia articles so as to equip the reader with as much general background information as possible. Notably, these references are not intended to back up information directly relevant to the research of this work—this kind of justification is duly provided by numerous references to scientific journal articles.

So now, looking back at the past years, at all the programming and debugging, the conferences and classes, and the writing of articles and this work, I believe my time being a doctorate student has been productive and well spent. I hope that you, the reader, will agree, and that when studying this thesis you will find it sound if you review it, enlightening if you seek information in it, and entertaining if you just browse it. Or ideally, all of that.

Jan Kurzidim, in June 2013

Chapter 1

Background

In this chapter, we provide the inclined reader with an overview of what motivated the present work. Since the core part of this investigation is concerned with simulations—the natural habitat of which is a computer’s hardware—, it is vital to put the data therefrom into the context of physics. For this purpose, we will first introduce the concepts of colloids, glasses, and porous confinement from the perspectives of theory, experiment, and simulation; then, we will lay out the details of the system investigated in this work; and finally we will discuss the previous theoretical work on the basis of which the original goals of this work were formulated.

1.1 Colloids

Colloid, any substance consisting of particles substantially larger than atoms or ordinary molecules but too small to be visible to the unaided eye.

Encyclopedia Britannica^{1.1}

The first of the concepts underlying this work is that of “colloids”. Historically, the term was coined by the Englishman Thomas Graham, who in 1861 worked on the diffusion of substances through a membrane [105]. In his study, he associated the properties “low diffusibility” and “gelatinous” with materials such as “starch, dextrin and the gums, caramel, [...], gelatine, [and] vegetable and animal extractive matters”. Biased by his choice of materials, he identified the organic nature as their distinguishing feature, which prompted him to name the class of materials by a compound term of the two Greek words κόλλα (glue) and εἶδος (kind)—“sticky stuff” [172].

In the modern definition, the term colloid refers to a peculiar type of mixture of at least two materials: “dispersed substances” that form small particles, and a “dispersion medium” in which the particles are immersed [124, 172, 226]. The particles^{1.2} are

^{1.1}<http://www.britannica.com/EBchecked/topic/125898/colloid>

^{1.2}Depending on the context, the short-hand term “colloid” may refer to a colloidal substance as a whole or to a single colloidal particle.

		dispersed substance		
		<i>gas</i>	<i>liquid</i>	<i>solid</i>
dispersion medium	<i>gas</i>	<i>none</i> (<i>gases are miscible</i>)	liquid aerosol fog, mist, sprays	solid aerosol smoke, dust, granular media
	<i>liquid</i>	foam whipped cream, shaving cream	emulsion milk, mayonnaise, hand cream	sol blood, clay, ink, paint, proteins
	<i>solid</i>	solid foam styrofoam, pumice, bird bones	gel gelatin, jelly	solid sol colored glass

Table 1.1: Classification of colloidal systems. Common names and examples of colloidal systems for different thermodynamic phases of the dispersion medium (the host) and the dispersed substance (the small particles). (Derived from Ref. [124] and Footnote 1.3.)

usually considered colloidal if they are large enough for quantum effects to be negligible, but small enough for the mixture to be stable. Depending on the system at hand, this allows sizes of colloidal particles between 1 nm and 10 μm —much larger than atoms, but much smaller than the resolution of the naked eye (“mesoscopic”). The other key feature of colloidal systems is that at length scales *smaller* than the size of the colloidal particles, both the dispersed substance and the dispersion medium can be considered homogeneous.

In the general case, the dispersed substance and the dispersion medium can take on any of the classical states of matter, i.e., be gaseous, liquid, or solid [124]. Table 1.1 gives an overview over the possible combinations along with some prominent examples. As the table evidences, colloids are vital to a plethora of contexts such as the environment (pollution), the chemical industry (coatings, materials, cosmetics), the food industry, the geological sciences (volcanism, mineralogy), and last but not least biology, where it is safe to say that *we are colloids*^{1.4}. The definition of colloids also overlaps with the notion of “soft matter”, a wide class of solid materials characterized by its extraordinarily small resistance to macroscopic mechanical deformation. Over the course of the most recent decades, this material class has been the subject of a steadily-increasing level of scientific interest (see for instance the witty Nobel lecture given by de Gennes in 1992 [59])^{1.5}.

^{1.3}<http://en.wikipedia.org/wiki/Colloid>

^{1.4}In analogy to the astrophysical dictum “we are stardust”.

^{1.5}It is of little coincidence that “soft matter” also inspired the name of the scientific group that the author is a member of.

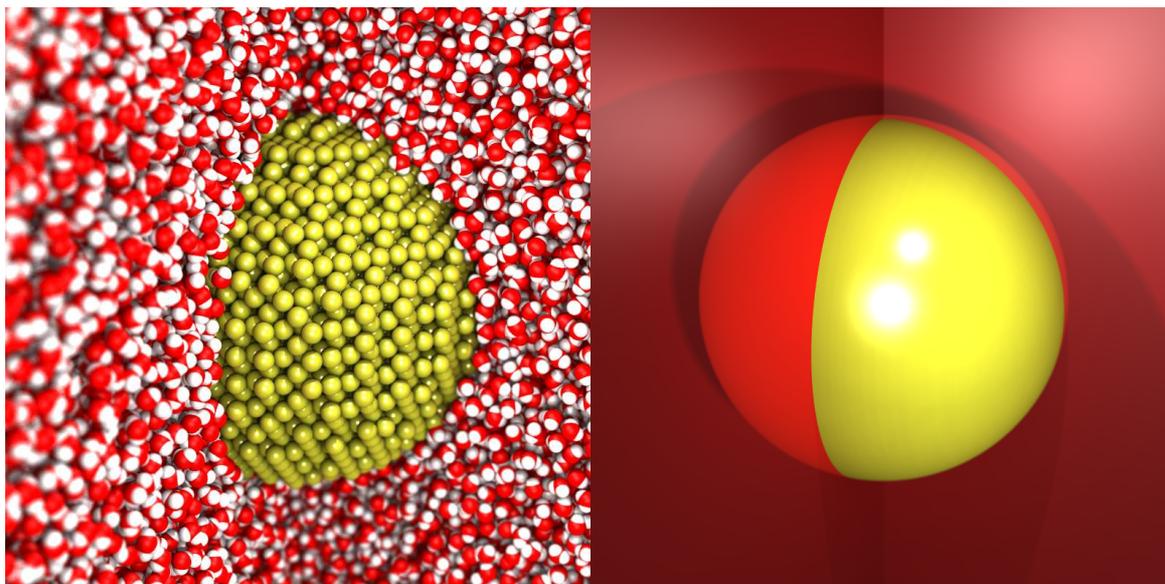


Figure 1.1: Coarse-graining schematic. Left: artist’s rendition of a ~ 2 nm diameter gold sphere immersed in water. Right: conception of a pertinent coarse-grained picture of the particle and the solvent. In both images, parts of the water are cut away to render the gold sphere visible. (Source: author’s work.)

1.1.1 Coarse-grained picture

So why are colloids interesting? They are because their macroscopic behavior is determined largely by their *mesoscopic* properties, i.e., by the statics and dynamics of the dispersed particles within the dispersion medium [124, 172, 226]. Therefore, it is sensible to attempt to describe a colloid using *only* these mesoscopic properties. The precondition for such a “coarse-grained” picture to be valid is that the dispersed particles do not merge or split, that they can be clearly distinguished from each other and from the dispersion medium, and that they (and the surrounding medium) are internally homogeneous on the length scale of the size of the particles. These conditions are met in particular by many sols, i.e., solid particles suspended in a liquid medium (cf. Table 1.1).

For such colloids, the atomic structure of the particles and of the solvent can be neglected, and each particle can be characterized by only a few attributes—a procedure also known as “averaging out the internal degrees of freedom” [172]. Depending on the specifics of the system, a colloidal particle may for instance be regarded as a spherically symmetric object (visualized in Fig. 1.1 for the example of a gold nanoparticle) which is then identified only by a mass and by the location and the velocity of its center of mass. In other cases, a more suitable description might be obtained by also taking into account properties such as an orientational vector (as for instance in “patchy particles” or “liquid crystals”; see [30, 60, 102]) or a magnetization vector.

The coarse-grained picture greatly facilitates investigations of the collective behavior of the colloidal particles, i.e., of their arrangement and their dynamics [5, 9,

214, 286]. A typical object of interest is the “thermodynamic phase” (solid, liquid, gaseous) that their arrangement corresponds to in analogy with atomic or molecular substances. Like for the latter substances, the thermodynamic phase of a colloid depends upon the external conditions (temperature, pressure, etc.), the properties of the constituent particles (e.g., their mass), and the *interactions* between these particles.

Interactions in colloidal systems may be very complex. As in atomic systems, the behavior of each colloidal particle depends in principle upon the properties all other colloidal particles [9, 115]. However, the interactions between colloidal particles may be very different from interactions between atoms or simple molecules, which is due to the internal structure of colloidal particles. Additionally, their interactions are significantly influenced by the dispersion medium. In brief, the most important types of forces on colloidal particles can be divided into two groups [124, 172, 226].

Firstly, there are forces between particles that are also known from atomic systems, for instance

- steric repulsion (“excluded volume”),
- electrostatic forces, and
- induced dipole (“van der Waals”) forces.

These forces are consequences of the Pauli exclusion principle^{1,6}, of charges on the particles, and of the distribution of the charges on the particles, respectively.

Interactions of the second group are original to colloidal systems and follow from the structure of the particles and in particular from the presence of the solvent. They include for example

- “kicks” from fast solvent molecules,
- viscous drag of the solvent, and
- hydrodynamic forces from solvent flow.

Here, molecule kicks and viscous drag are usually subsumed under the notion of “Brownian motion” [9], and solvent flow usually results from the movement of particles (but may also be externally imposed) [123].

1.1.2 Colloids in experiments

In experiments, the examining of colloidal systems is greatly facilitated by two features of their particles:

- (i) Colloidal particles are *large*—frequently large enough to scatter ordinary light, so that expensive observation techniques involving electron or neutron beams are superfluous.

^{1,6}http://en.wikipedia.org/wiki/Pauli_exclusion_principle

^{1,7}http://en.wikipedia.org/wiki/Scanning_electron_microscope

^{1,8}<http://weitzlab.seas.harvard.edu/gallery.html>

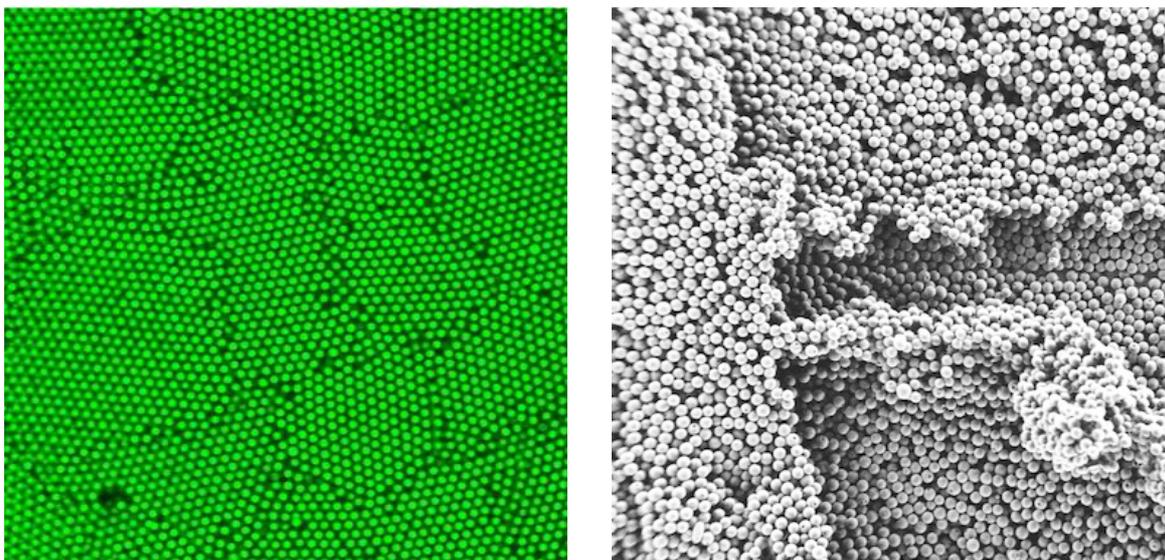


Figure 1.2: Microscope images of colloidal systems. Left: snapshot from a typical fluorescence confocal microscopy video. Right: example of a scanning electron microscopy view, which in comparison is more detailed but inherently static^{1.7}. (Both images: Experimental Soft Condensed Matter Group, Harvard University^{1.8}.)

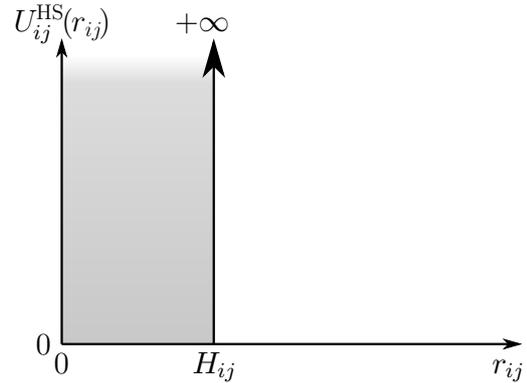
- (ii) Colloidal particles are *slow*, with relevant events taking place on the second or even minute scale (not nano- or picoseconds, as in atomic systems), meaning that it is viable to record ordinary videos [215].

This allows to observe colloids for instance using dynamic light scattering [23, 27, 36, 211, 215, 265] or confocal fluorescence video microscopy [56–58, 140, 277, 279]. By the latter, for example, hundreds of particles can be tracked simultaneously in real space, and the resulting information can be examined with the same methods as computer simulation data (cf. Sec. 2.5), thereby establishing an important link between simulation and experiment.

An even more appealing feature of colloids is the possibility to *modify the particles themselves* (and/or the solvent) on an atomic level in order to change the overall properties of a system. This way, if desired, it is even possible to *design* (“tailor”) colloidal systems to fit particular purposes [84, 172]. Such a changing of particles is entirely unknown in systems of atoms and simple molecules. The tailorability may be used for instance to mimic atomic systems using colloidal particles; however, its real appeal lies in the wealth of complex systems with exotic properties that can be created this way [59, 84].

Finally, in colloidal systems it is nowadays possible not only to modify the internal structure of all particles, but also to manipulate them *individually*. This is accomplished for instance using so-called “optical tweezers”—laser beams, essentially—which

Figure 1.3: The hard-sphere potential. Potential energy $U_{ij}^{\text{HS}}(r_{ij})$ of two isolated hard spheres i and j depending on their distance r_{ij} as given in Eq. (1.1). The hard-sphere potential implies that the distance between i and j cannot be less than H_{ij} , the latter being given by $H_{ij} = R_i + R_j$ in the case of additive hard spheres [Eqs. (1.2) and (1.3)].



allow for instance to relocate particles, or, more sophisticated, to measure the actual forces between colloidal particles [14, 56, 109]^{1.9}.

1.1.3 Hard spheres

In most theoretical and simulation investigations of colloidal systems, relatively-simple particle interaction models are employed to deduce a system’s overall behavior [172]. This reflects the coarse-grained picture of colloids. In this work, we made extensive use of the so-called “hard-sphere” (HS) model, which we shall introduce in the following.

1.1.3.1 Definition

In the HS model, as in many other models of colloids, a particle i at some time t is represented by a small set of properties, $\gamma_i(t)$. The $\gamma_i(t)$ ’s of HS particles include only a position vector $\mathbf{r}_i(t)$, a momentum vector $\mathbf{p}_i(t)$, and a time-independent mass m_i . The forces acting on HS particles—and therefore the evolution of their $\gamma_i(t)$ ’s—result from the following assumptions.

- (1) It is assumed that the force $\mathbf{F}_i(t)$ on particle i at the time t can be expressed as the gradient $\nabla_{\mathbf{r}_i(t)} U(\mathbf{r}, t)$ of a scalar potential that i moves in.
- (2) It is assumed that this potential arises solely from pairwise interactions between the particles, i.e., that $U(\mathbf{r}_i(t), t) = \sum_{j \neq i} U_{ij}(\gamma_i(t), \gamma_j(t))$.
- (3) It is assumed that the pair potential depends only upon the distance between i and j , meaning that $U_{ij} = U_{ij}(r_{ij})$ where $r_{ij} = r_{ij}(t) = |\mathbf{r}_j(t) - \mathbf{r}_i(t)|$. This implies translational and rotational invariance of the system.

^{1.9}One of the fathers of optical tweezers, Steven Chu, received not only the 1997 Nobel prize for this innovation, but is nowadays one of the most prominent voices of science due to his being the current Secretary of Energy of the United States^{1.10}.

^{1.10}http://www.energy.gov/organization/dr_steven_chu.htm

(4) It is assumed, as sketched in Fig. 1.3, that the potential follows the form

$$U_{ij}^{\text{HS}}(r_{ij}) = \begin{cases} +\infty & \text{for } r_{ij} \leq H_{ij} \\ 0 & \text{for } r_{ij} > H_{ij} \end{cases} . \quad (1.1)$$

Two particles i and j interacting according to these assumptions experience a force only if $r_{ij} = H_{ij}$. Since, moreover, achieving $r_{ij} < H_{ij}$ would require an infinite amount of energy, the involved particles represent infinitely-hard bodies of spherical shape—hard spheres.

The HS potential is particularly simple in that it involves only one parameter— H_{ij} . However, in a system of N particles, there are still as many as $N(N-1)/2$ such parameters. For further simplification, it is common to express the H_{ij} 's in terms of properties associated with the individual particles instead of particle pairs. This is achieved by assigning $N-1$ “radii” $R_i^{(j)}$ to each particle i , and by defining that

$$H_{ij} = R_i^{(j)} + R_j^{(i)} . \quad (1.2)$$

At first, this may seem a needless complication since it doubles the number of parameters. However, one may for instance require that

$$R_i^{(j)} = R_i , \quad (1.3)$$

i.e., that each particle be associated with just a *single* radius. Hard spheres obeying Eq. (1.2) and Eq. (1.3) are said to be “additive” [266]. A collection of additive hard spheres is called “polydisperse” if each R_i has a unique value, and “monodisperse” if all R_i 's are equal [229]. The monodisperse case is particularly appealing due to its exceptional simplicity, and was employed throughout this work.

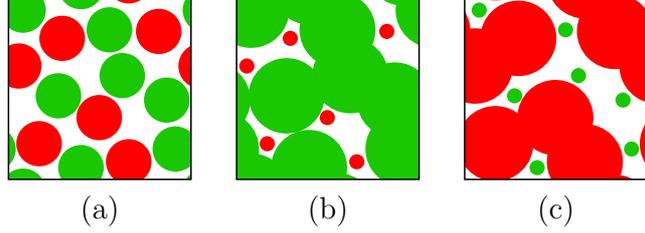
Finally, it is noteworthy that U_{ij}^{HS} does not depend upon an orientational vector. This is tantamount to considering nonrotating particles, which is important for the derivation of the dynamic interactions in the HS model (cf. Sec. 2.2.1.4). Also, since U_{ij}^{HS} does not depend upon any particle's momentum, it does not account for hydrodynamic effects.

1.1.3.2 Basic properties

In some HS systems, it is sensible to divide the particles into S “species” according to the values of the HS interaction parameters. In this case, it is common to associate each particle i with S radii $\{R_i^{(1)}, \dots, R_i^{(S)}\}$, and to re-express Eq. (1.2) as

$$H_{ij} = R_i^{(t)} + R_j^{(s)} \quad (1.4)$$

Figure 1.4: Redefinition of hard-sphere radii. Mixture of two hard-sphere species (red and green disks). For the interaction between red and green particles, only the *sum* of the radii is relevant, which renders panels (a)–(c) equivalent for the inter-species interaction. The hard-sphere radii of the intra-species interaction (red–red and green–green) are *not* depicted.



where particles i and j belong to species s and t , respectively. For interactions between particles of the *same* species, Eq. (1.4) then simply reverts to Eq. (1.3). For interactions between two *distinct* species s and t , however, the HS radii may be redefined as

$$\begin{aligned}\tilde{R}_i^{(t)} &= R_i^{(t)} + b^{(st)} & \text{and} \\ \tilde{R}_j^{(s)} &= R_j^{(s)} - b^{(st)} & ,\end{aligned}\tag{1.5}$$

the only restriction in the choice of $b^{(st)}$ being that all $\tilde{R}_i^{(t)}$ and $\tilde{R}_j^{(s)}$ have to be positive. Since the sums $\tilde{R}_i^{(t)} + \tilde{R}_j^{(s)}$ and $R_i^{(t)} + R_j^{(s)}$ are identical by construction, none of the H_{ij} 's are altered by the substitutions $R_i^{(t)} \rightarrow \tilde{R}_i^{(t)}$ and $R_j^{(s)} \rightarrow \tilde{R}_j^{(s)}$.

A geometric interpretation of this procedure is provided in Fig. 1.4, which shows an HS system with two distinct particle species encoded in red and green color. The only difference between panels (a), (b), and (c) lies in the inter-species radii of the particles. As is evidenced by the lack of overlaps between particles of distinct species in all panels, the radii of a green and a red particle always add up to the same value. The “overlaps” between particles of the *same* species are irrelevant since the green–green and red–red interactions are based on HS radii that are not shown. Figure 1.4 thus underscores the validity of replacing inter-species radii according to Eq. (1.5). This property will be made extensive use of in the context of characterizing the geometry of porous matrices (see Sec. 2.4 and Ref. [146]).

Another interesting feature of HS systems is their vanishing potential energy, $U = 0$. If the particles are subject to Newtonian motion, then the system can be described by a Hamiltonian of the form $\mathcal{H} = K + U = E$, where E is the system's total energy and K its kinetic energy. In this case, $U = 0$ yields $E = K = \sum_{i=1}^N \mathbf{p}_i^2 / 2m_i$, where m_i is the mass of particle i and \mathbf{p}_i is its momentum. This entails that the particles move along straight lines at all times, except for infinitesimal time spans during which “impulse interactions” take place (see Sec. 2.2.1.1). The physics of these interactions is invariant under a rescaling of time, as can be seen from Eq. (2.26) where prefixing a factor a to each velocity on the right-hand side is tantamount to multiplying the entire right-hand side with a . Moreover, using $\mathbf{p}_i = m_i \mathbf{v}_i$ from your physics 101 class, the relation $\sum_{i=1}^N m_i (a\mathbf{v}_i)^2 / 2 = a^2 E$ can be seen to hold, meaning that rescaling time

is equivalent to changing the total energy. Since E , via the equipartition theorem^{1.11}, also defines a temperature, $T = 2E/3Nk_B$ (where k_B is Boltzmann’s constant), this time rescaling invariance has gained HS systems the label “athermal” [227]^{1.12,1.13}.

Molecular dynamics (MD) simulations of hard spheres are typically carried out in the NVE —or “microcanonical”—ensemble (cf. Sec. 2.2). This means that E in these systems is one of the external parameters. However, since the actual value of E is irrelevant (see above), the only genuine control parameters in microcanonical HS systems are the particle number N and the volume V . Consider further that the actual size of the system is usually uninteresting, and that both N and V are proportional to the system’s size. This means that microcanonical HS systems are in fact controlled only by the *ratio* of N and V , i.e., only by *one* external parameter. In HS-MD simulations, this ratio is commonly represented indirectly by the volume fraction ϕ , i.e., by the fraction of volume covered if each particle i is identified with a sphere of radius R_i [see Eqs. (1.2) and (1.3)]. Since by definition hard spheres do not overlap, the volume fraction—or “packing fraction”—evaluates to

$$\phi = \frac{1}{V} \sum_{i=1}^N \frac{4\pi}{3} R_i^3 = \frac{\pi}{6} \rho \sigma^3, \quad (1.6)$$

where $\rho = N/V$ is the number density of the particles. For the second equality in Eq. (1.6), it was assumed that $R_i = \sigma/2$ for all i , meaning that the particles are monodisperse (cf. Secs. 1.1.3.1 and 2.5.1.6).

1.1.3.3 Significance

Monodisperse HS systems display a variety of features known from real colloids. Most notably, this includes a fluid and a crystal phase along with a fluid–crystal phase transition and the possibility for these phases to coexist [5, 201, 286]. Of particular importance to the present work is the fact that HS fluids also exhibit slow dynamics (i.e., glassy behavior; see Sec. 1.2) if prepared suitably [214].

^{1.11}The equipartition theorem holds in thermal equilibrium and reads $T = 2E/nNk_B$, where n is the number of degrees of freedom (DOFs) of one of the constituent particles. Typical DOFs are the translation, rotation, or vibration of a particle. The actual number of DOFs depends on the nature of the particles as well as on the overall energy of the system and its dimensionality [108, 238]. In the case of hard spheres it is $n = 3$ since only translation is relevant and the dimensionality is three.

^{1.12}The notion of athermality continues to persist when considering Brownian [9, 38, 85] instead of Newtonian motion. In this case, however, changing the total energy rescales the time coordinate only for time-averaged quantities (see Secs. 2.1.2 and 2.5.1). The reason for this is that Brownian systems are not described by a Hamiltonian, which in turn is owed to the fact that a given initial phase-space point does not give rise to a unique phase-space trajectory [9, 226].

^{1.13}In contrast to the total energy, thermodynamic variables such as the pressure or the chemical potential constitute genuine control parameters in HS systems. However, the definition of these quantities is nontrivial in HS systems [115].

The HS model is excellently suited to study the local particle arrangement in high-density (or high-pressure) condensed-matter systems [115]. This is due to the fact that under these conditions, the average distance between neighboring particles is short and steric repulsion—if present—is the dominant force (cf. Sec. 1.1.1). This was demonstrated famously in 1971 by Weeks, Chandler, and Andersen [115, 267, 278] for the Lennard-Jones (LJ) potential^{1.14}. Conversely, this implies that the HS potential is less suited for the modeling of systems in which other forces are significant (cf. Sec. 1.1.1).

HS systems are appealing also due to the availability of statistical-mechanical methods to theoretically evaluate their properties. One of these properties is the radial distribution function (cf. Sec. 2.5.2.1 and Ref. [280]), which can be approximated for instance using the “Ornstein-Zernike” (OZ) equations [202] in combination with the “Percus-Yevick” (PY) closure [210]. This approximation gives way to more sophisticated analyses concerned with the dynamics of HS systems (see Sec. 1.4.2). Further, it is possible to obtain the equation of state for HS systems—for instance via the OZ–PY scheme as above [280] or, to a higher precision, by the “Carnahan-Starling” approximation [44]. This fact will be used in Sec. 2.5.4.1.

Finally, the HS model is of significance due to the possibility to not only select but to also *engineer* HS-like colloids (cf. Sec. 1.1.2). Such colloids are typically prepared from a “latex”^{1.16}, i.e., from a suspension of solid polymeric particles in a liquid. In many studies [13, 40, 214], these particles are composed of poly(methyl methacrylate) (“PMMA”), a material also known as “Plexiglas”^{1.19}. The particles can be prepared such that they well approximate spheres and that they show diameter variations of less than 10%. The average diameter of the particles is typically chosen from the range of 50 to 300 nm, which allows for light scattering experiments (cf. Sec. 1.1.2).

^{1.14}The Lennard-Jones (LJ) potential, or 12–6 potential, is one of the most widely used model potentials in condensed-matter physics, with over 10 000 hits on Thomson–Reuter’s publication database “ISI Web of Science” (see Footnote P.2). The LJ two-body potential is defined as $U_{LJ} = 4\varepsilon[(\sigma/r)^{12} - (\sigma/r)^6]$, where ε is an energy parameter, σ a distance parameter, and r is the inter-particle distance. The potential features strong repulsion at short distances and mild attraction at longer distances, and is a good model for noble gases. Notably, “Lennard-Jones” is the surname of a single person, namely Sir John Lennard-Jones^{1.15}.

^{1.15}http://en.wikipedia.org/wiki/John_Lennard-Jones

^{1.16}In colloquial speech, the term “latex”^{1.17} commonly refers to an elastic material. In science and technology, however, the term designates a material that can be *processed* to yield an “elastomer”^{1.18}. Whereas nowadays most latices are chemically engineered, historically latices were harvested in the form of plant saps as occurring for instance in the Pará rubber tree or the dandelion (*German* “Löwenzahn”). In a side note, the plural of latex (“latices”) should not to be confused with the plural of lattice (“lattices”).

^{1.17}<http://en.wikipedia.org/wiki/Latex>

^{1.18}<http://en.wikipedia.org/wiki/Elastomer>

^{1.19}<http://en.wikipedia.org/wiki/Pmma>

1.2 Glasses

“The deepest and most interesting unsolved problem in solid state theory is probably the theory of the nature of glass and the glass transition. This could be the next breakthrough in the coming decade. The solution of the problem of spin glass in the late 1970s had broad implications in unexpected fields like neural networks, computer algorithms, evolution, and computational complexity. The solution of the more important and puzzling glass problem may also have a substantial intellectual spin-off. Whether it will help make better glass is questionable.”

Philip W. Anderson, 1977 Nobel laureate
[*Science* **267**, 1615–1616 (1995)]

The fundamental nature of “glass” has, for several decades now, been the subject of intense theoretical, simulational, and experimental research. Yet, despite these efforts, glasses continue to elude a coherent understanding. An important objective of this work is to contribute to the grand picture of what glass actually *is*. Since over the years a multitude of pertinent theoretical concepts have been developed, it is well beyond the scope of this work to elaborate on, or even touch upon, all of these concepts. In this background section, we will therefore first introduce some general information about glasses, and subsequently focus on the specific aspects that are relevant to the scientific approach of this work.

1.2.1 History and applications of glass

Let us start out with a brief summary of the history and applications of materials called “glass”. Glasses have been invaluable for mankind throughout almost its entire existence^{1.20}. In the stone age, for instance, the volcanic glass *obsidian*^{1.21} was treasured for its sharp and durable edges (upon fracturing) and its specular surface (upon polishing). Between 4000 and 3000 BC in Mesopotamia, glass was for the first time produced *artificially*, mainly to manufacture small vessels and jewelery. Around 1 AD in the Roman Empire, the technique of *glass blowing*^{1.22} was invented, which allowed to craft large vessels and, for the first time, glass windows. In medieval Europe, stained-glass windows came to be one of the main pictorial arts. The fabrication of window glass culminated in the mid-20th century with the *float glass*^{1.23} process, in which molten glass is floated over a bed of liquid metal to obtain extremely flat sheets of glass.

^{1.20}http://en.wikipedia.org/wiki/History_of_glass

^{1.21}<http://en.wikipedia.org/wiki/Obsidian>

^{1.22}<http://en.wikipedia.org/wiki/Glassblowing>

^{1.23}http://en.wikipedia.org/wiki/Float_glass

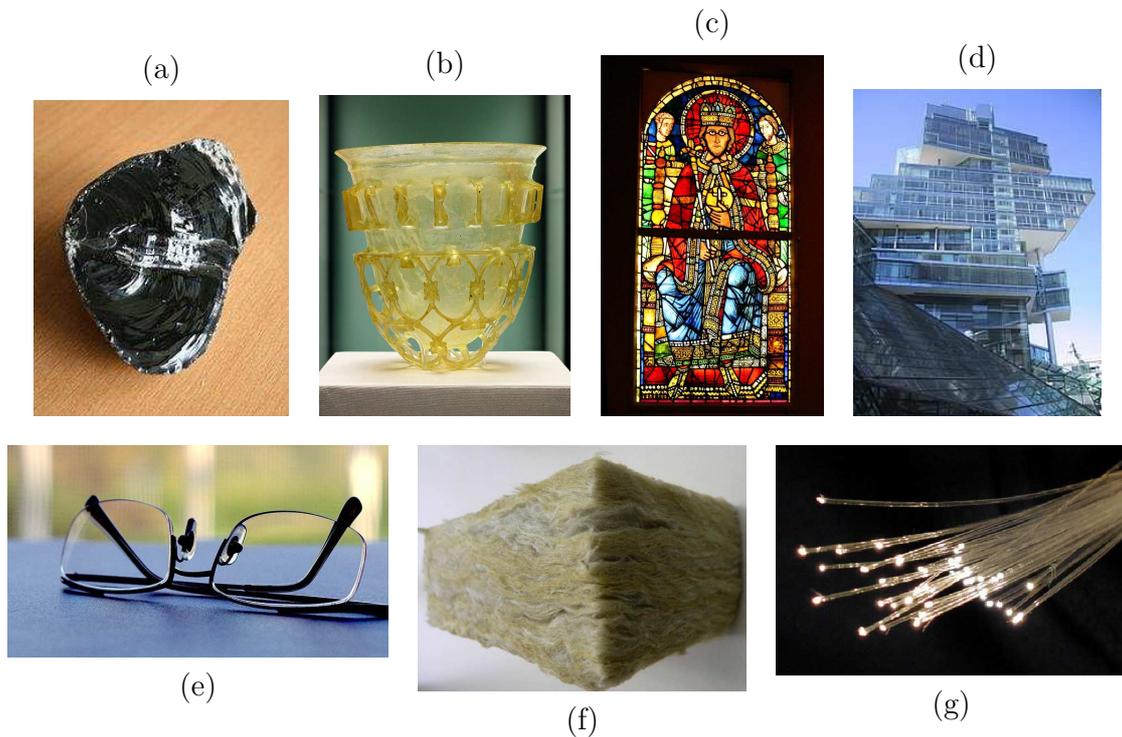


Figure 1.5: Historic and modern applications of glass. Panel (a): the volcanic glass obsidian, used in stone-age cutting tools. Panel (b): a Roman-age drinking cup. Panel (c): a medieval stained-glass window (Strasbourg Cathedral). Panel (d): a modern building façade of flat-glass panes. Panel (e): eyeglasses. Panel (f): glass wool, for fireproof thermal insulations. Panel (g): optical fibers, for high-speed data transmission. (All images: Wikimedia.)

As early as around 500 BC in Egypt and Greece, glass was used for *optical lenses*^{1.24}, initially probably as burning and/or magnifying glasses. Yet, only by the late 13th century AD in Italy, lenses were also used as eyeglasses. It was first in the early 1600s in the Netherlands that multiple lenses were combined into microscopes and telescopes. In the mid-19th century, J. Tyndall popularized *optical fibers*^{1.25}—light “conductors” by the principle of total internal reflection—, which nowadays are widely used to transmit digital information by modulating the conducted light. The 19th century also saw the invention of *glass wool*^{1.26}, a lightweight material in which thin glass fibers are arranged in a texture similar to wool. Glass wool is inflammable and a very poor heat conductor and therefore excellent for heat insulation of buildings and pipes. A mid-20th century invention, finally, is *fiberglass*^{1.27}, a class of extremely robust and lightweight compound materials in which a plastic polymer is reinforced by glass fibers.

^{1.24}http://en.wikipedia.org/wiki/Lens_%28optics%29

^{1.25}http://en.wikipedia.org/wiki/Optical_fiber

^{1.26}http://en.wikipedia.org/wiki/Glass_wool

^{1.27}<http://en.wikipedia.org/wiki/Fiberglass>

1.2.2 Supercooling

We now take on the question of what glass actually is. Let us start by considering the two materials “quartz”^{1.28} and “fused quartz”^{1.29}. Under standard conditions (0 °C, 1 bar) these materials are quite similar, but they differ slightly in various properties such as their density, thermal expansion, tensile strength, and refractive index^{1.30}. It is the very existence of this difference that is surprising since *both* materials are composed *purely* of SiO₂ (silicon dioxide)^{1.31}. How can this be? According to classical thermodynamics, for every set of external conditions (temperature, pressure, etc.) a substance in thermal equilibrium can assume only *one*^{1.33} thermodynamic “phase” [108, 238]. Therefore, it is impossible that both quartz and fused quartz represent stable states of SiO₂. Yet, both quartz and fused quartz exhibit unchanged properties over very long times—even over the course of centuries.

The key difference between quartz and fused quartz is that the latter results from a process called “supercooling” [101, 104, 288] while the former is in fact the stable phase of SiO₂ under standard conditions. So what is supercooling? Suppose that a substance is equilibrated and present in its liquid (or fluid^{1.34}) phase. Suppose further that in this situation the external conditions are changed in a way that the stable phase of the substance would now be the crystal phase. The substance will of course immediately start to react to this change, but since in any substance the velocities of the constituent particles are finite, this reaction cannot be *completed* immediately. From this it can be inferred that if the change of the external conditions occurs “fast enough”, the substance will remain liquid-like for some (possibly very extended) time before the crystalline state prevails. Since the changed conditions usually involve a decrease in temperature^{1.36}, the itinerant liquid state is called a “supercooled liquid”.

^{1.28}<http://en.wikipedia.org/wiki/Quartz>

^{1.29}http://en.wikipedia.org/wiki/Fused_quartz

^{1.30}<http://www.azom.com/article.aspx?ArticleID=1114>

^{1.31}Silicon dioxide constitutes the main component of most materials that are commonly referred to as “glass”. This in particular includes “soda-lime glass”, which is omnipresent in windows and food containers [104]^{1.32}. Fused quartz can therefore be considered a good representative of “glass”.

^{1.32}http://en.wikipedia.org/wiki/Soda-lime_glass

^{1.33}Under certain external conditions, it is also possible that multiple thermodynamic phases of a substance co-exist. In this case, each involved phase represents the single stable phase for some other set of external conditions, and the fractions of the substance occupied by the different phases are fixed by the external conditions [108, 238].

^{1.34}Some substances do not exhibit “liquid” and “gas” as distinct thermodynamically-stable phases. In such substances, which notably include hard-sphere systems (cf. Sec. 1.1.3), the thermodynamic phase least resistant to shear is commonly called the “fluid” phase^{1.35}.

^{1.35}<http://en.wikipedia.org/wiki/Fluid>

^{1.36}Depending on which external parameters are controlled with the least effort in the system of interest, supercooling may alternatively refer to an increase of the pressure or the density. In systems of hard spheres—the subject of this work—the relevant parameter is the *density* (cf. Sec. 1.1.3.2).

This obviously raises the question of *how* fast “fast enough” is. The answer to this question requires an understanding of the process by which a supercooled liquid is transformed into a crystal. In most substances, this process goes by the name of “nucleation and growth” [11, 32, 92, 141, 233]^{1.37}, which, in brief, involves a sequence of two phenomena: First, “nuclei” (small crystalline regions) are spontaneously formed and dissolved due to random positional fluctuations of the substance’s particles. Then, once “critical nuclei” (nuclei large enough to be stable) are formed, these nuclei grow since larger nuclei are energetically more favorable^{1.38}. This way, eventually the entire substance is transformed into crystalline structures.

In the present context, the most relevant aspect of nucleation and growth is the average time τ_{nucl} required to form a critical nucleus in the given amount of supercooled liquid. The reason for this is twofold: for times t shorter than τ_{nucl} there is no^{1.39} critical nucleus and therefore no growth, and the volume of the noncritical nuclei is typically tiny compared to the overall volume of the supercooled liquid. Therefore, the supercooled liquid is *effectively stable* (“metastable”) for $t < \tau_{\text{nucl}}$. This is the canonical notion of “quasi-equilibrium”, in which a metastable phase is considered *indistinguishable* from true equilibrium despite the exclusion of the actually stable arrangement (see, e.g., Refs. [1, 26, 31, 62, 71, 138, 154, 241], and compare with Sec. 2.5.5).

Finally, consider that a liquid is in a supercooled state only if it assumes a temperature below the melting temperature T_m . However, before this state can be attained, the liquid has to be prepared *above* T_m since only in that temperature regime it is thermodynamically stable. Therefore, the desired temperature T_{targ} of the supercooled liquid *always* has to be established by cooling, and cooling inherently requires some nonzero time τ_{cool} . Consequently, the liquid is supercooled also for some time *before* T_{targ} is reached, which means that for this time it is susceptible to the formation of critical nuclei. Thus, if a supercooled liquid is to be established at T_{targ} , then the cooling process needs to be controlled such that τ_{cool} is short enough for *no*^{1.39} critical nucleus to form during cooling.

In conclusion of the above, it is clear that supercooling is not limited to artificial environments or to specific materials, and that its establishing is a mere experimental challenge. In particular, it is also possible to supercool *colloids*, i.e., to retain colloidal particles in a liquid-like arrangement when a crystal-like arrangement is energetically preferred. In the specific case of the systems investigated in this work, the state of

^{1.37}<http://en.wikipedia.org/wiki/Crystallization>

^{1.38}Whether or not a crystal nucleus is stable depends essentially upon of two competing factors: On the one hand, below the melting temperature a crystalline particle arrangement is energetically *avored* over a liquid arrangement. On the other hand, the existence of nuclei entails the presence of interfaces with the surrounding liquid, which is energetically *disavored*. For more details on this highly-relevant and active field of research we refer the interested reader to the aforementioned sources [11, 32, 92, 141, 233].

^{1.39}In fact, since fluctuations are stochastic, it is only with some probability P that there will be no critical nucleus for $t < \tau_{\text{nucl}}$, and it is the definition of τ_{nucl} that determines the actual value of P .

supercooling is procured by the rather unusual procedure of increasing the diameters of the simulated colloidal particles, which effectively is tantamount to increasing the density of the particles (see Sec. 2.3 and Footnote 1.36).

1.2.3 Dynamic arrest

So how do supercooled liquids relate to glasses? After all, from the example of fused quartz in Sec. 1.2.2, it seems that glasses are more similar to *solids* than to liquids. As we shall see in the following, the desired connection is established by *another* process of cooling^{1.36}, where this time the relevant phenomenon is the effect of cooling upon a supercooled liquid’s *structural relaxation*.

On the macroscopic scale, the structural relaxation of a substance corresponds to its deformation response when subjected to a force, which in liquids is quantified by the viscosity η . The effect of cooling on η is generally the one observed for instance in honey, which is more viscous when refrigerated than when at room temperature. For a wide range of temperatures and materials, the increase in viscosity upon cooling obeys a so-called “Arrhenius law” [33, 101, 103, 104, 241],

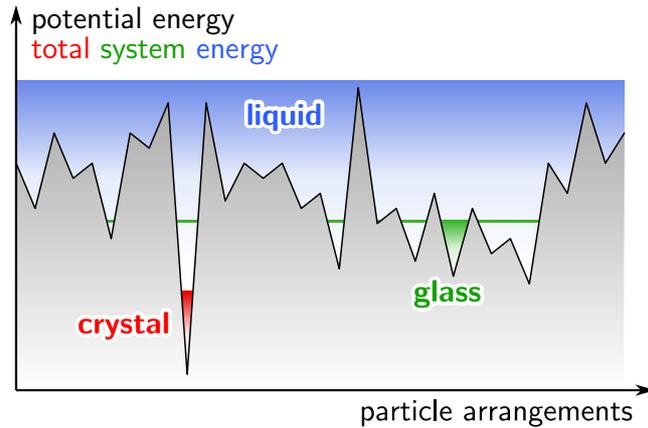
$$\eta(T) = \eta_0 \exp \left\{ - \frac{E_a}{k_B T} \right\} , \quad (1.7)$$

where T is the temperature, k_B is Boltzmann’s constant, and η_0 is a proportionality factor of dimension viscosity. In this law, which was originally observed for chemical reactions, E_a plays the role of an “activation energy”, i.e., an energy that needs to be surmounted (e.g., by fluctuations) before a reaction can take place. Rationalizing Eq. (1.7) requires to associate E_a with a “reaction coordinate”, which in the case of viscosity and the structural relaxation of liquids is the microscopic *rearrangement of particles*.

Cooling has two straightforward consequences for particle rearrangements. Firstly, it decreases the number of different configurational states that the substance assumes on average, i.e., it decreases the substance’s entropy. Secondly, it renders it more time-consuming for fluctuations in the reaction coordinate to overcome barriers between configurational states. If these fluctuations are *Gaussian*, then the rate by which the liquid elapses a sequence of configurational states depends *exponentially* upon the temperature. Since such a state sequence corresponds to a substance’s deformation, the aforementioned dependence is expressed by Eq. (1.7).

This, finally, allows to define the notion of “glass” as follows. The viscosity η of a liquid is essentially proportional to the time τ_α required for the same liquid to structurally relax [52, 71, 93, 103, 104, 241]. According to Eq. (1.7), the viscosity and therefore τ_α diverge at $T = 0$, which means that sufficient cooling may increase τ_α beyond any given time t_{obs} . If t_{obs} is the time over which a supercooled liquid is *observed*, and if $t_{\text{obs}} \ll \tau_\alpha$, then the substance will appear to be *structurally static* during the time of observation. This can be used to define the “glass transition temperature” T_g as the temperature at which the τ_α of the supercooled liquid of interest equals some

Figure 1.6: Potential energy landscape. Schematic of the potential energy of particle arrangements (black line) and of systems with different total energies (upper edge of colored areas) in this landscape: a liquid (blue), a crystal (red), and a glass (green). The glass’s nonergodicity is illustrated by the green line, which is visible for particle arrangements with a potential energy less than the glass’s total energy.



fixed, “large” value [11, 31, 62, 71, 178, 241]^{1.40}. Below this temperature, the substance is then referred to as “dynamically arrested”—a *glass*.

1.2.4 In the glass

Let us briefly review some properties of supercooled liquids below the glass transition. Figure 1.6 illustrates the phenomenon of dynamic arrest in terms of the potential energy of the arrangements of the considered particles (a “potential energy landscape” [62, 154, 230]). The upper edges of the colored areas represent the total energy of three different systems^{1.41}. The blue system is liquid, and it is ergodic since all allowed configurations can be reached. The red system is crystalline and likewise ergodic. The green system, however, cannot assume the configurational states beneath the green line although its total energy would suffice to do so, the reason being that the system is confined between potential energy barriers. This system, which corresponds to a glass, is therefore *nonergodic* (cf. Sec. 2.1.3).

Another implication of the green line in Fig. 1.6 is that in a substance at a sufficiently low temperature, there are *many* states that appear nonergodic to an observer. In practice, which of these states is assumed upon dynamic arrest is determined by the details of the preceding cooling process. To see this, consider for instance the (hypothetical) instantaneous cooling from a temperature T_{before} to a temperature at which the substance appears nonergodic. Such a cooling fully preserves the substance’s structure at T_{before} , which means that different values of T_{before} lead to different glass

^{1.40}In fact, T_g is conventionally defined via the viscosity η , namely as the temperature at which η attains 10^{13} Poise ($= 10^{12}$ Pa·s in SI units). In typical shearing experiments, liquids with this viscosity structurally relax within a few minutes [223].

^{1.41}In Fig. 1.6, the total energy $E = K + U$ is constant, whereas in typical experiments, the temperature T and therefore the kinetic energy K is constant while the potential energy U is allowed to fluctuate. Therefore, constant- T systems can surmount any potential energy barrier given sufficient time. Notably, in hard-sphere systems, $U = 0$ and thus $E = K$ (cf. Sec. 1.1.3.2), which, unfortunately, is difficult to visualize in Fig. 1.6 since it would require to replace U by an entropy-related quantity.

structures. This influence of the process upon the result is referred to as the “history dependence” of a glass [11, 26, 31, 62, 71, 167, 241]^{1.42}.

All glasses, however, exhibit by definition a frozen structure when observed over an appropriate time span t_{obs} . In this regard, glasses are similar to the thermodynamically-stable solid (crystal) phase. Microscopically, this similarity is reflected in the number of degrees of freedom (DOFs), which over t_{obs} is comparable for glasses and crystals but different for glasses and liquids^{1.43}. The number of DOFs in turn is determinative of various thermodynamic properties of a substance^{1.44}, which explains the fact that crystals and glasses frequently exhibit very similar heat capacities and thermal expansion coefficients [52, 62, 71, 101, 103, 104, 138]. It is, however, not surprising that the disparate structures of glasses and crystals (see Sec. 1.2.6) also render these states different—sometimes significantly—for other properties, a prominent example being the general lack of birefringence in glasses^{1.45}.

Finally, consider that nonergodic systems like glasses are by definition not in thermal equilibrium. Since statistical mechanics is applicable only to systems *in* thermal equilibrium^{1.47}, and since statistical mechanics forms the basis for computer simulations as applied in this work (see Secs. 2.1.2 and 2.1.3), it is difficult to characterize dynamically-arrested systems in simulations. Therefore, in the majority of simulational works on the glass transition, the transition is investigated only from the “ergodic side”, i.e., only for temperatures $T > T_g$. In the remainder of this work, we will pursue this approach and focus on systems in thermal equilibrium (cf. Sec. 2.5.5).

^{1.42}The history dependence of glasses is technologically exploited in the process of “annealing”, in which a glass is reheated to the vicinity of its glass transition temperature T_g . The purpose of this reheating is to stimulate limited structural relaxation and to thereby remove structural stresses that may have resulted from the initial cooling (“quenching”) below T_g [33, 97, 141]. A related effect is the partial structural relaxation of a glass if it is in use for times comparable to its structural relaxation time. This so-called “aging” is usually undesired in technological applications since it alters the material’s properties [11, 26, 31, 62, 71, 167, 241].

^{1.43}The difference in the number of degrees of freedom (DOFs) between crystals, glasses, and liquids is straightforward: in crystals and glasses of simple particles like hard spheres, the DOFs are limited to vibrations of the particles around their average positions, whereas the particles of liquids may also rearrange.

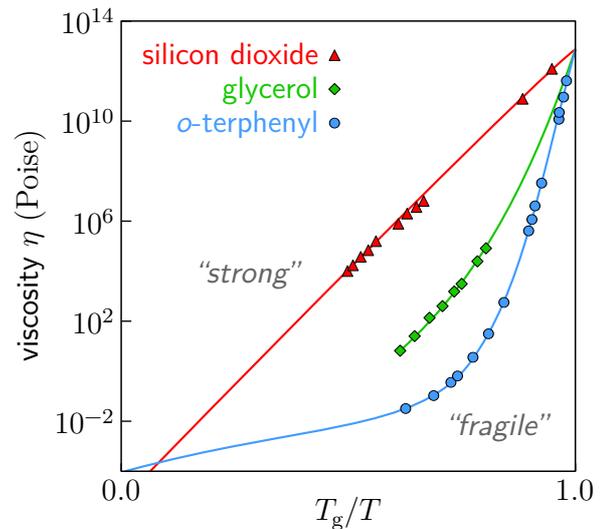
^{1.44}Strictly speaking, it is impossible to associate genuine thermodynamic properties with glasses since glasses are not in thermodynamic equilibrium. However, thermodynamic properties are experimentally determined in measurements of finite duration, and such experiments can also be conducted on glasses to measure “effective” thermodynamic properties [11, 138].

^{1.45}The effect of birefringence^{1.46} directly results from the anisotropy of the regular particle arrangement in crystals. In glasses, such a regular arrangement is absent (cf. Sec. 1.2.6).

^{1.46}<http://en.wikipedia.org/wiki/Birefringence>

^{1.47}In some cases, statistical mechanics is applicable also to out-of-equilibrium systems. Incidentally, one such case is the state of supercooling introduced in Sec. 1.2.2. Another case is that of systems in a “steady state”, i.e., systems that are subject to a constant external perturbation [194].

Figure 1.7: Angell plot. Variation of the viscosity η with the inverse temperature $1/T$ for three different liquids. Separately for each material, $1/T$ is normalized to the inverse of the material's glass transition temperature $1/T_g$. Classification by behavior of η towards $1/T_g$: approximately exponential increase = “strong”; superexponential increase for some temperatures = “fragile”. (Adapted from Ref. [11]; note the scale on both axes.)



1.2.5 A genuine transition?

In Sec. 1.2.4, we established that the “transition” at the temperature T_g (cf. Sec. 1.2.3) is not thermodynamic but merely reflects some arbitrary patience limit^{1.48}. The purpose of this section is to amend and eventually countermand this portrayal, namely by showing that there do seem to exist instances in which the cooling of a supercooled liquid causes some of its properties to change *discontinuously* even if the liquid remains *quasi-equilibrated* (cf. Sec. 1.2.2). Such changes are highly interesting since they constitute the hallmark of genuine phase transitions between thermodynamically-stable material phases [108, 238, 258].

Of the properties that seem to change discontinuously, let us first consider the one most relevant to this work: the structural relaxation time τ_α . As stated in Sec. 1.2.3, τ_α is equivalent to the viscosity η in liquids, and for many materials and temperatures, η obeys Eq. (1.7). However, η does *not* follow Eq. (1.7) for other materials and temperatures. To this end consider Fig. 1.7, which shows the dependence of η upon the inverse of the temperature T for three different materials. In each case, T is normalized by the material’s glass transition temperature T_g , and recalling that by definition $\eta(T_g) = \eta_g = 10^{13}$ Poise (see Footnote 1.40), this normalization means that all curves meet at η_g . A plot of this kind is called an “Angell plot” [11, 53, 62, 71].

As can clearly be seen from Fig. 1.7, there is considerable variation in the functional form of $\eta(1/T)$ for both different temperature regions and different materials. For silicon dioxide (red, cf. Sec. 1.2.2), an exponential behavior of $\eta(1/T)$ as described by Eq. (1.7) is observed over the entire depicted temperature range (note the scales

^{1.48}If the exponential law in Eq. (1.7) holds in the vicinity of T_g , then even large changes in the arbitrary threshold viscosity η_g cause only small changes in T_g (cf. Sec. 1.2.3 and Footnote 1.40). This renders T_g “almost a material property” [11, 71]. Also, according to Sec. 1.2.4, the “thermodynamic properties” are more similar for glass and crystal than for glass and liquid. Without knowledge of the underlying physics, these findings could be considered indicative of a phase transition at T_g .

on both axes). Liquids of this kind are called “strong” glass formers, and they include mostly substances in which all constituents are covalently bonded to form a large “network” [11, 31, 53, 62, 71, 138, 178, 193, 194, 288]. A vastly different functional behavior is observed for *o*-terphenyl (blue): here, cooling increases the viscosity only relatively slowly at first, but over a relatively-narrow temperature range, the dynamic slowdown is then enormously accelerated. Such liquids are called “fragile” glass formers, and include mostly substances with nonpermanently-bonded constituents like molecules [11, 62, 71] and notably also hard-sphere systems [104, 154, 240] (cf. Sec. 1.1.3). The third substance, glycerol (green), serves as an example for the many intermediary cases between the fragile and strong extremes shown in the figure.

The point in case is that for many glass formers, the acceleration of the slowdown upon cooling is *so* drastic that η (and therefore τ_α) seems to *diverge at some nonzero temperature* T_0 . In fact, already in the 1920s it was found that for most glass formers, η follows the empirical functional form

$$\eta(T) = \eta_0 \exp \left\{ - \frac{E_a}{k_B (T - T_0)} \right\} \quad (1.8)$$

instead of Arrhenius’s law [Eq. (1.7)], the latter being recovered from Eq. (1.8) for $T_0 = 0$. Equation (1.8) is today referred to as the “Vogel-Fulcher-Tammann (VFT) law” [1, 32, 62, 71, 73, 86, 103, 104, 254, 269] after its three independent discoverers. Since the functional form of Eq. (1.7) arises from Gaussian fluctuations (cf. Sec. 1.2.3), the form of Eq. (1.8) must be rooted in *non*-Gaussian fluctuations, which is interesting of its own right and has been the subject of many investigations [104, 125, 156]. Unfortunately, the limited nature of observation times does not permit to directly verify diverging relaxation times in simulations and experiments, which renders theoretical models of the utmost importance (cf. Sec. 1.4). However, experiments and simulations may well contribute to an understanding of the *mechanisms* that underlie the extraordinary slowdown, and it is largely this fact that motivated the work at hand.

Another indication of a genuine phase transition in supercooled liquids is the so-called “Kauzmann paradox”, which was devised in 1948 and, in brief, suggests that the “configurational entropy” in some substances *vanishes* upon cooling the liquid towards some nonzero temperature T_K [31, 62, 71, 138]. The configurational entropy represents essentially the number of different particle arrangements that correspond to a supercooled liquid state^{1.49}, and its vanishing is remarkable since it implies the existence

^{1.49}The configurational entropy of a supercooled liquid at some temperature T^* is defined as the difference between the entropy of the stable crystal, S_{cr} , and the entropy of the liquid, S_{liq} , at T^* . The term S_{cr} can be determined by a thermodynamic integration^{1.50} in the crystal phase from the reference point $T = 0$ (where the entropy vanishes) to T^* . The term S_{liq} first requires a thermodynamic integration in the crystal from the same reference point $T = 0$ to the melting temperature T_m . To this, the “entropy of fusion” (computable from the enthalpy of fusion) is added to obtain the entropy of the liquid phase at T_m . This then serves as the reference for another thermodynamic integration, now in the supercooled liquid state from T_m to T^* .

^{1.50}http://en.wikipedia.org/wiki/Thermodynamic_integration

of an “ideal” glassy state—much like an ideal crystal represents a thermodynamically-stable phase. However, a number of works indicate that T_K , if it exists, is always *below* the dynamic arrest temperature T_0 of the same substance (see above). For this reason, it is difficult for both simulations and experiments to verify the existence of Kauzmann’s paradox (cf. Sec. 1.2.4).

More recently, efforts have been intensified to identify diverging *length scales* in supercooled liquids. Such length scales would be considered airtight evidence for phase transitions, but are notoriously difficult to pinpoint. Only starting with the 1990s, the computational and experimental capabilities have allowed to investigate the properties that are nowadays considered “good candidates” for diverging length scales. One of these candidates is “dynamic heterogeneities” [25–27, 62, 66, 125, 140, 157, 241, 277, 282], which may correlate over large distances and which, among others, are commonly associated with collective particle rearrangements and the breakdown of the Einstein relation (see Sec. 1.2.7). Another candidate is the “point-to-set correlation”, in which the suitable freezing of particle positions creates boundary conditions that may expose growing correlation lengths [25, 26, 34, 42, 157, 231]. This approach bears certain similarities with the systems investigated in this work, as will become clear in Sec. 1.3.4.2.

1.2.6 Static features of slow liquids

In the remainder of this background section, we will give an overview over those features of supercooled liquids that are directly relevant to this work. Let us start by introducing the term “slow liquids” to refer to supercooled liquids cooled towards dynamic arrest. In the present section, we will present some simple *static* structural properties of slow liquids, and argue that these properties do not provide obvious indications of dynamic arrest.

Consider Fig. 1.8, which shows snapshots of three systems of hard disks (the two-dimensional equivalent of hard spheres, cf. Sec. 1.1.3): a thermodynamically-stable liquid (yellow), a slow liquid (green), and a crystal (blue). All systems are comparable in density, and their similar local arrangement is a consequence of hard particles being impenetrable. The only *qualitative* difference between the depicted systems is the periodicity that is present in the crystal but *absent* in the liquids. The fact, however, that the green liquid is close to dynamic arrest while the yellow one is not is essentially impossible to infer from the snapshots^{1.51}.

The similarity between normal and slow liquids is further substantiated by Fig. 1.9. The figure shows the radial distribution function $g(r)$, which encodes the normalized

^{1.51}Notably, disordered systems with slow dynamics may result not only from supercooling (cf. Sec. 1.2.2). The many alternative processes are for instance thermodynamic (compression, solvent evaporation, vapor deposition), mechanical (shocking, grinding), chemical (electrochemical deposition, polymerization, hydrolysis, sol-gel reactions), or irradiative (particle beams, X-rays) in nature [11, 31]. The results of such processes include for instance gels [120, 203], spin glasses [33, 73], amorphous thin films [3, 81, 187], and nanostructured materials [101, 194].

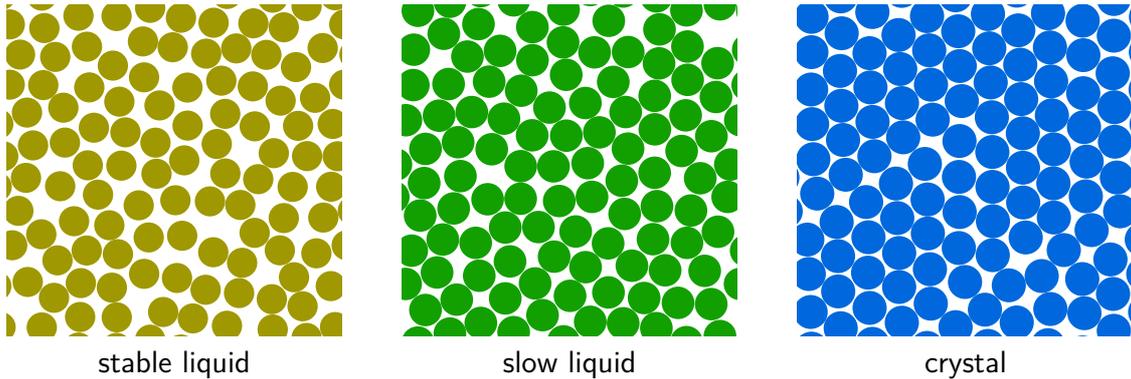


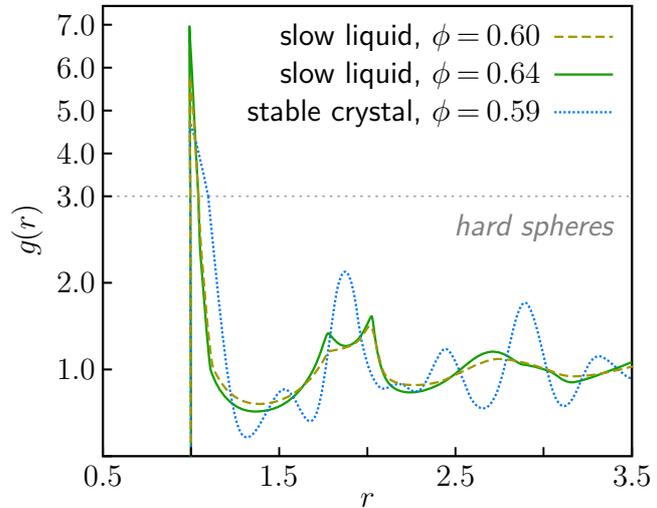
Figure 1.8: Structure of slow liquids. Snapshots of representative systems of hard-disk particles: a thermodynamically-stable liquid (yellow), a supercooled liquid close to dynamic arrest (“slow liquid”, green), and a stable crystal (blue). The yellow and green systems are similar in that they show only short-range order and are disordered otherwise, whereas the blue system exhibits long-ranged order (save a few defects).

probability of finding a pair of particles at a separation r (explained in detail in Sec. 2.5.2.1). The figure depicts $g(r)$ for three representative simulated systems of hard spheres (cf. Sec. 1.1.3): a moderately supercooled liquid (yellow), a strongly supercooled and therefore slow liquid (green), and a crystal (blue). Obviously, $g(r)$ differs substantially between the crystal on the one hand and the two liquids on the other. The $g(r)$ ’s of the two liquids, however, are remarkably similar, and in fact the few genuinely distinct features^{1.52} emerge continuously upon cooling. The static structure factor (essentially the spatial Fourier transform of $g(r)$, cf. Sec. 2.5.2.2) is similarly oblique to the dynamics. Therefore, a large number of authors deem it impossible to infer the dynamic arrest of a supercooled liquid from simple static structural features [11, 31, 54, 62, 71, 115, 138, 140, 261, 277].

Whether or not more complex structural features indicate dynamic arrest is the subject of a long-standing debate. In addition to the point-to-set correlation function mentioned in Sec. 1.2.5, the many candidates for such structural features include for instance correlations in nonaffine displacement fields [63, 199], medium-range crystalline order [255], “patch” correlations [231], spatial three-point correlations [54], and locally preferred structures [53, 54], the latter being tightly connected with the concept of frustration (see Sec. 1.3.3.2). However, none of these properties will be probed directly in this work since the foremost goal of this work is to characterize the *dynamics* of the investigated systems (cf. Sec. 1.4).

^{1.52}The only clear-cut discernible feature between weakly and strongly supercooled hard-sphere liquids is the splitting of the peak at $r \simeq 2$ (for unit-diameter spheres) in the strongly supercooled case. Its appearance, however, is limited to hard spheres, where it is typically associated with “random close packing”, i.e., disordered “jammed” arrangements at very high density (see Refs. [12, 22, 243] and cf. Sec. 2.3.2.2). This suggests a relationship between hard-sphere glasses and jammed systems, an idea that has been put forth by numerous authors [26, 173, 177, 237, 241, 260, 262].

Figure 1.9: Radial distribution function of slow liquids. Normalized probability $g(r)$ for the distance r of particle pairs in representative hard-sphere systems at different packing fractions ϕ . Yellow dashed line: moderately super-cooled liquid; green solid line: slow liquid in vicinity of random close packing; blue dotted line: stable crystal close to freezing point. The yellow and blue systems differ substantially, whereas the yellow and green systems are very similar and do not allow to infer the dynamic arrest of the green system. Note that $g(0 \leq r < 1) \equiv 0$ for hard spheres, and that the y axis scale varies. (Simulation data, reproduced from Refs. [51] and [263].)



1.2.7 Dynamic features of slow liquids

Investigating the dynamic properties of slow liquids constitutes the core of the present work. Of particular interest is the structural relaxation of slow liquids since, as discussed in the previous sections, it is precisely this process that governs their response to macroscopic stresses.

Many features of a slow liquid’s structural relaxation are reflected in intermediate scattering functions (ISFs). ISFs encode how correlations of particle positions evolve as the time t progresses. Figure 1.10 shows the specific case of the single-particle ISF, $F_s(k, t)$, for fixed wave vector $k = 2\pi$, the depicted system being an experimental realization of slightly polydisperse^{1.53} hard spheres at different packing fractions ϕ [36]. (For more details on ISFs see Sec. 2.5.3, and for hard spheres see Sec. 1.1.3.) As can be seen from the figure, the functional form of $F_s(k, t)$ depends decisively upon ϕ : whereas for the lowest depicted value of ϕ the ISF relaxes in a single step, for larger ϕ values it exhibits a distinctive *two-step relaxation* pattern. Highlighted in different colors are the β relaxation regime (red), the caging regime (green), and the α relaxation regime (blue), the significance of which we will explain in the following.

Figure 1.11 illustrates the physical mechanism underlying the regimes of β relaxation and caging (red and green in Fig. 1.10, respectively). The left panel of the figure shows the same slow hard-sphere liquid as the center panel of Fig. 1.8, while the right panel shows a zoom-in of the vicinity of a randomly-chosen particle (red). Highlighted in violet are the particles that neighbor the red particle and form a “cage” around it. Clearly, in slow liquids *each* particle is surrounded by such a cage, and it is plausible

^{1.53}The term “polydisperse” refers to random variations in the “size” of a substance’s particles. The significance of this property in the context of slow liquids will be explained in Sec. 1.3.3.2.

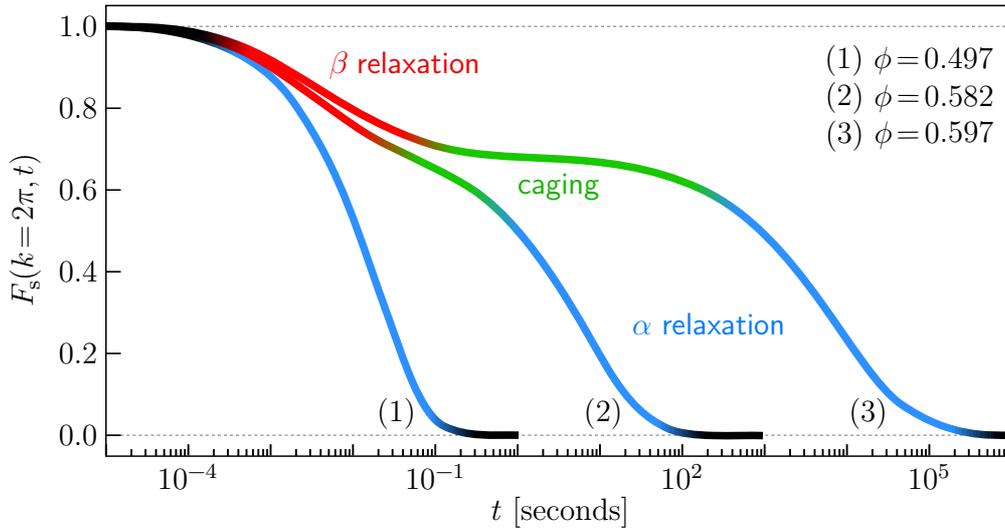


Figure 1.10: Intermediate scattering function of slow liquids. Structural relaxation for the representative case of supercooled polydisperse hard spheres at different packing fractions ϕ , quantified by the single-particle intermediate scattering function $F_s(k, t)$ for the wave vector $k = 2\pi$ and varying times t . (1) stable liquid, (2) moderately supercooled liquid, (3) slow liquid. Shown in red is the β relaxation regime, in green the caging regime, and in blue the α relaxation regime. For system (1), only α relaxation takes place. (Experimental data, reproduced from Ref. [36] with slight extrapolations.)

that random fluctuations of particle positions are quite unlikely to change this structure (see below). Therefore, each particle is confined to its cage for some time, and during this time it can merely undergo vibrational motion (indicated by the red circles and the black arrow, cf. Footnote 1.43). This phenomenon is called the “cage effect”, and it gives rise to the β relaxation regime highlighted in red in Fig. 1.10.

As laid out in Secs. 1.2.3 and 1.2.5, the structural relaxation time of a slow liquid increases tremendously within a small temperature and/or density interval (note also the values of ϕ in Fig. 1.10). Over this interval, the mean free path of a particle inside its cage does not change by much, and neither does the time τ_β required to complete the β relaxation. On the other hand, the time τ^* after which particle rearrangements occur in significant number depends strongly upon the degree of supercooling (see below). Therefore, τ^* may exceed τ_β by orders of magnitude. If this discrepancy holds in some system, then the structural relaxation of this system will be negligible for times $\tau_\beta < t < \tau^*$, and it is this time range that constitutes the “caging regime” (highlighted in green in Fig. 1.10).

Finally, for the structure of a substance to completely relax—i.e., for the α relaxation to take place—it is necessary that all of the substance’s particles relocate considerably. In slow liquids, however, each individual particle is left with only a marginal amount of space to move in by its cage (see above). For a particle to significantly alter its position, the surrounding particles have to vacate an equally significant

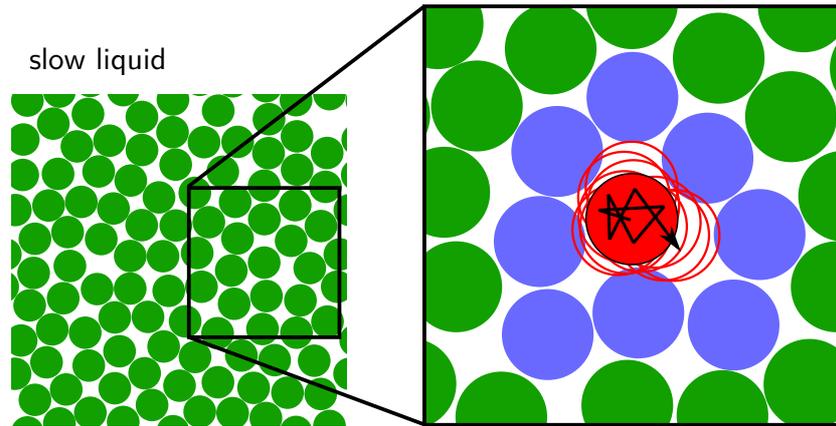


Figure 1.11: Cage effect. Left: same slow hard-disk liquid as in Fig. 1.8. Right: zoom-in centered about an arbitrarily-chosen particle (red). Red circles and black arrow: indicators of the vibrational movement of the red particle (slightly magnified movement extent). Violet: particles that “cage” the red particle and constrain its movement.

amount of space, which can be accomplished only if multiple particles relocate in a coordinated fashion [1, 71, 103, 104]. Due to the random nature of the particles’ movements, the probability for such “cooperative rearrangements” to occur decreases steeply with the number of participating particles. On the other hand, the number of particles required for such rearrangements to take place at all increases with the degree of supercooling. These dependencies match well with the observed α relaxation (the regime highlighted in blue in Fig. 1.10), which, as noted before, decelerates dramatically upon cooling a slow liquid^{1.54}. Notably, in liquids of lower density (such as the stable liquid in Fig. 1.8), structural relaxation is possible without cooperative rearrangements, which is reflected in the absence of the β relaxation and caging regimes in system (1) of Fig. 1.10.

Only relatively recently, it was determined that the most frequent type of cooperative rearrangement in slow liquids is the string-like or circular motion of groups of particles [62, 66, 277]. This process is visualized in Fig. 1.12, which shows again the slow hard-sphere liquid from Figs. 1.8 and 1.11. Highlighted in pink are five particles that may potentially move in a closed loop along the black arrows without being obstructed in that motion by other particles. The system contains several other groups of

^{1.54}The dynamics of collective rearrangements (CRs) is “energy landscape dominated” (cf. Fig. 1.6) in the sense that CRs overcome potential energy barriers between metastable states [62, 93, 103]. CRs are commonly associated with dynamic heterogeneities and non-Gaussian fluctuations (see Sec. 1.2.5), which, in turn, are responsible for the frequently-observed breakdown of the Stokes-Einstein relation (SER) in slow liquids [62, 104]. (According to the SER, the diffusion constant D of a liquid is inversely proportional to its viscosity η . The relation assumes that fluctuations of the particle positions are Gaussian, which means that non-Gaussian fluctuations break the SER. Also, D is dominated by the fastest particles while η is governed by the slowest ones, and these particles are equally fast on average if fluctuations are Gaussian while they move at different average speeds if dynamic heterogeneities are present.)

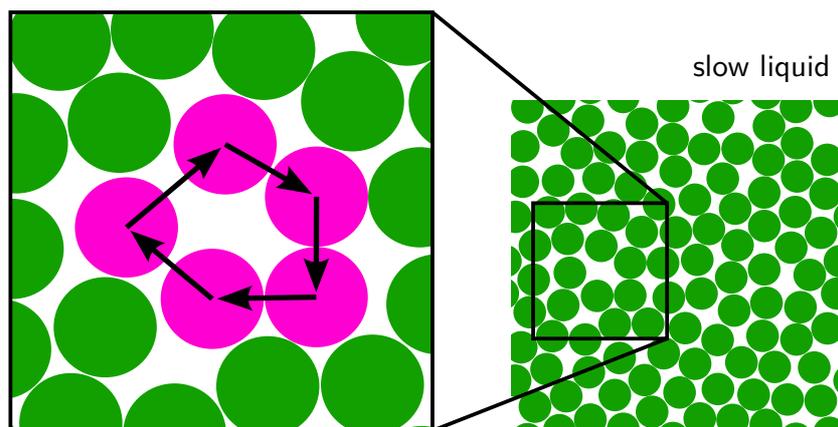


Figure 1.12: Cooperative rearrangements. Right: same slow hard-disk liquid as in Figs. 1.8 and 1.11. Left: zoom-in of five particles that circularly rearrange by randomly moving in the directions indicated by the black arrows.

particles (with varying member count) that are suitably positioned to undergo similar collective motions if their velocity vectors are appropriate.

Already in the mid-19th century, Kohlrausch [158] discovered that the α relaxation of certain properties in glassy materials are well described by a so-called “stretched exponential”,

$$F(t) = F_0 \exp \left\{ - \left(\frac{t}{\tau_\alpha} \right)^\beta \right\}, \quad (1.9)$$

where β is a heuristic “stretching exponent” unrelated to the β of the β relaxation process. The functional form of Eq. (1.9) was rediscovered in the 1970s by Williams and Watts [32, 73, 284], which gave rise to its contemporary name “Kohlrausch-Williams-Watts (KWW) law”. As indicated by the use of the symbol “ F ” in Eq. (1.9), the equation is particularly useful in the description of intermediate scattering functions of slow liquids (cf. Fig. 1.10), where in many cases it can be excellently fitted to the caging and α relaxation regimes [71, 93, 104, 155].

1.3 Porous confinement

Fluids may drastically change their behavior when subjected to spatial confinement. The changes may for instance be structural, thermodynamic, or dynamic in nature, and they are not limited to any specific geometries or materials, for neither the fluid nor the confinement. For two reasons, confinement is particularly interesting in the context of dynamic arrest (cf. Sec. 1.2). Firstly, confined slow fluids are widespread in nature and technology (Sec. 1.3.1), meaning that qualitative or even quantitative understanding of the effects of confinement (Sec. 1.3.2) may constitute a significant practical advantage. Secondly, and more fundamentally, confinement may elucidate the very nature of dynamic arrest: if the physics of fluids in confinement is understood

on a microscopic level (Sec. 1.3.3), then the presence of confinement might reveal the mechanisms of dynamic arrest in general. In this work, we chose to investigate the particular case of *disordered porous* confinement since this type of confinement gives rise to some particularly interesting phenomena (Sec. 1.3.3). Specifically, we investigated a model of fluids in porous confinement known as the “quenched-annealed” model (Sec. 1.3.4.2), and did so using computer simulations (cf. Chapter 2).

1.3.1 Significance of fluids in confinement

Fluids in confinement, and in particular in porous confinement, are abundant in nature, technology, and everyday life. A frequently-encountered example of a fluid with significant confinement effects is water immersed in an ordinary cleaning sponge, where the water remains confined to the sponge’s pores (and is even soaked up) despite the presence of gravity, and particles suspended in the water can enter the sponge’s pores only if they are small enough.

Of considerable technological and economic relevance are water and mineral oil—both in pure form and in combination with other substances—that permeate porous materials such as ceramics, cement, rock, clay, organic soil, or sand. Such combinations of fluids and confinement play essential roles in agriculture (e.g., water retention in soils) [48, 95], hydrogeology (e.g., replenishment of water reservoirs and aquifers) [39, 95, 192], the petrochemical industries (e.g., oil recovery) [10, 18, 39, 58, 95, 121, 232], and the geophysical sciences (e.g., lubrication in plate tectonics) [91, 95, 192, 235]. In other industries such as the chemical and pharmaceutical industries, porous substances—for instance in the form of powders—serve to expedite chemical reactions [95, 289].

Other industry-scale applications include the use of engineered porous materials for filtration and purification purposes (for instance for the removal of pollutants from water and thereby, indirectly, from soils) [10, 95, 206] or in the production of fresh water from sea water or contaminated water by the process of reverse osmosis [146]. More intricate technological applications of confinement include the separation of mixtures of liquids and/or gases by selective adsorption into a porous medium [95, 206, 232, 289], and, similarly, the construction of sensors in which particular porous media react to some substances but not to others [45, 289]. Also, host media with well-defined pore structures may serve as templates for the manufacturing of micro- or even nano-patterned devices used for instance in ordinary electronics or opto-electronics [95].

Further, porous environments are ubiquitous in biology, prominent examples being tissues such as bones and wood. Even more importantly, confinement is crucial to many processes on the cellular level. For instance, at any time biological cells are filled with dense collections of proteins floating in the cytoplasm [17, 18, 24, 50, 111, 118, 184, 188]. The simulational realization shown in Fig. 1.13(a) suggests that these proteins essentially represent a slow colloidal fluid [184]. The cryoelectron tomography of a living cell in Fig. 1.13(b), on the other hand, shows that the interior of cells is divided

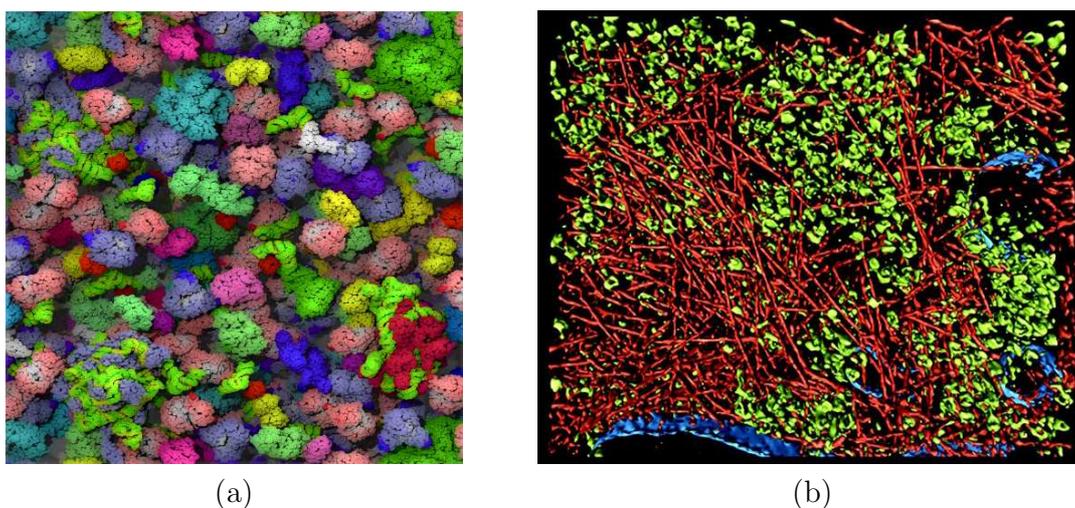


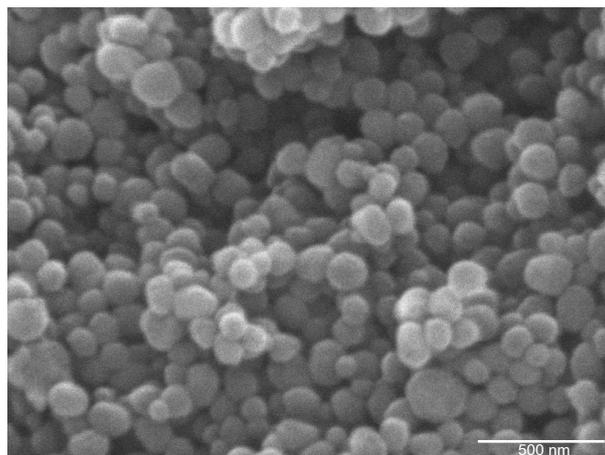
Figure 1.13: Proteins in cytoplasm. Panel (a): Snapshot of a computer simulation of the fifty most abundant protein species in the bacterium *Escherichia coli* under near-natural circumstances. Colors code different protein species. (From: McGuffee and Elcock [184].) Panel (b): Cryoelectron tomography of the interior of a cell of *Dictyostelium* (a single-celled slime mold) approximately $815 \text{ nm} \times 870 \text{ nm} \times 97 \text{ nm}$ in dimension. Red: actin fibers (cytoskeleton); green: macromolecular complexes (mostly ribosomes); blue: membranes. (From: Medalia et al. [188].)

into compartments by membranes (blue), and that there are numerous relatively-immobile cellular apparatuses such as actin fibers (which form the cytoskeleton; red) and macromolecular complexes (mostly ribosomes; green). These objects effectively render the inside of a cell akin to a porous medium for the fluid of proteins to move in [17, 18, 24, 45, 50, 111, 118].

Finally, designed porous materials with well-characterized properties are instrumental in experimental, simulational, and theoretical investigations of the effects of confinement upon a fluid's properties. Such investigations, empirical and mathematical, have been undertaken in considerable number in an effort to attain an enhanced understanding of the physical mechanisms in confined fluids [3, 95, 185–187]. Popular model materials in these investigations are in particular porous silica, or aerogels [3, 10, 95, 101, 143, 145, 160, 161, 163, 207, 232]. These materials are obtained from silica gels by a “sol-gel” process [95, 120], i.e., essentially by replacing the liquid component of the gel by a gaseous one. Porous silica are appealing for theoretical and simulational modeling since, as is evident from Fig. 1.14, their microscopic structure is approximately equal to that of a collection of partly-fused, solid, spherical particles frozen in place (cf. Secs. 1.1 and 1.3.4.2) [51, 95, 229, 266, 289]. Other popular model materials include porous Vycor and controlled pore glass, which differ from porous silica primarily in the size and the distribution of the pores [3, 10, 95, 160, 161, 187, 232]. In experiments, these materials are then injected with fluids (for instance with fluids

^{1.55}http://en.wikipedia.org/wiki/File:Mesoporous_silica_SEM.jpg

Figure 1.14: Porous silica. Scanning electron microscope graph of mesoporous silica. Clearly, the substance resembles an aggregate of spherical, solid, partly fused particles with large interstitial volumes. (Source: Wikipedia^{1.55}.)



of colloidal particles) in order to study the effects of confinement.

1.3.2 Observed phenomena in confinement

This section provides a brief review of the many peculiarities of fluids in confinement. The review is subject to two limits: Firstly, we are interested primarily in the properties of the *fluids*, not those of the confinement. The confinement’s properties will therefore be mentioned only if they explain some particular fluid behavior. Secondly, the present work focuses on systems in thermal *equilibrium*. Thus, the effect of confinement upon fluid flow [39, 48, 68, 152, 179, 260] will not be discussed directly, but instead only indirectly in the context of percolation (see Sec. 1.3.3.3).

In the context of thermal equilibrium, it is first of all due to note that in neither experiment nor simulation it is trivial to *establish* the state of equilibrium for confined fluids [95] (see also Sec. 2.5.5). Provided that equilibrium *has* been reached, one effect—found for instance in ordinary Helium (^4He) or molecular nitrogen (N_2)—is that the liquid–vapor coexistence region (cf. Footnote 1.33) is significantly narrowed by confinement [10, 95, 207]. In some substances (e.g., again, in ^4He), in addition to that the temperature and pressure of the critical point^{1.56} have been observed to be shifted considerably by confinement [10, 95]. Analogous findings hold for mixtures of different substances; for example, confining ^4He and ^3He into porous gold may entirely suppress the liquid–liquid critical point of that mixture [10, 95, 207]. This means that confinement may forcibly separate substances that without confinement are miscible under the same external conditions.

As mentioned before in Sec. 1.2.5, processes of general interest in equilibrated substances are phase transitions such as freezing/melting and condensation/evaporation [108, 238, 258]. Remarkably, confinement may induce phase transitions that do not

^{1.56}The term “critical point” denotes the maximum pressure and temperature for which the liquid and the gaseous phase of the same material can be distinguished^{1.57}.

^{1.57}http://en.wikipedia.org/wiki/Critical_point_%28thermodynamics%29

occur at all in the absence of confinement. A pertinent example is the “wetting” transition, i.e., a transition between states in which a liquid does or does not tend to form films on the inside surface of the confinement [3, 10, 39, 95, 143, 232]. Also, confinement may prompt the particles of the immersed substance to undergo a “layering” transition [3, 95, 187]. Of significance to this work is the fact that confinement may also considerably alter “ordinary” phase transitions. With confinement, for instance, the condensation of a gas into the corresponding liquid phase occurs at a lower temperature and/or pressure than without confinement. This effect is widely known as “capillary condensation” [95, 143, 186, 187, 232, 234]. In the context of supercooling (cf. Sec. 1.2.2), it is particularly interesting that confinement may also influence the melting temperature T_m , the most striking finding being that in some systems T_m may be *raised* by confinement whereas in others it may *lowered*, while in yet other systems the freezing transition may completely *vanish* [3, 95, 186].

In view of the latter, it is only a small step to suspect that confinement may also alter the properties of liquids with slow dynamics (“slow liquids”). Indeed, it has been found that the glass transition temperature T_g as defined in Sec. 1.2.3 is subject to variations similar to those of T_m if confinement is applied. The observed increases or decreases of T_g for various systems are succinctly summarized in Table 1 of Ref. [186] and Table 1 of Ref. [3], and have been designated by these (and many other) authors as probably the most puzzling effect of confinement upon slow liquids. It has even been suggested that confined slow liquids undergo *two* glass transitions—one for fluid particles close to the confining medium, and one for the bulk of the fluid [3]. This confirms that confinement may qualitatively alter not only the structure but also the (equilibrium) dynamics of a fluid.

Aside from the viscosity—which defines T_g —, the dynamic alterations by confinement are probably most prominent in the (self-)diffusion of fluids. As elaborated on in Sec. 2.5.3.1, the phenomenon of diffusion microscopically refers to the mean squared displacement (MSD), which according to Einstein [74] obeys the “diffusion relation” $\lim_{t \rightarrow \infty} \delta r^2(t) = 6Dt$. For the diffusion constant D , it has been found that—like T_m and T_g —it may change drastically in the presence of confinement, with both an increase and a decrease having been observed with respect to the case without confinement [3, 10, 16, 45, 89, 95, 111, 113, 146, 179, 196–198, 252].

Most strikingly, however, under confinement the MSD may *not at all* obey the diffusion relation and may instead conform only to the more general relation $\delta r^2(t) \propto t^z$. In this notation, normal diffusion corresponds to $z = 1$ and dynamic arrest to $z = 0$, while all other values of z represent “anomalous diffusion”. In confinement, the notion of anomalous diffusion usually implies $z < 1$, meaning that the particle propagation is *slower* than ordinary diffusion, i.e., “subdiffusive”. As detailed in Sec. 2.5.3.1, the exponent z equates to the “logarithmic derivative” of $\delta r^2(t)$ and is well defined locally in time. In fact, transient subdiffusion has long been known to occur even without confinement, and it does so for instance in simple slow liquids in the caging regime (cf. Sec. 1.2.7). Yet, in most systems and under most external conditions, the long-time

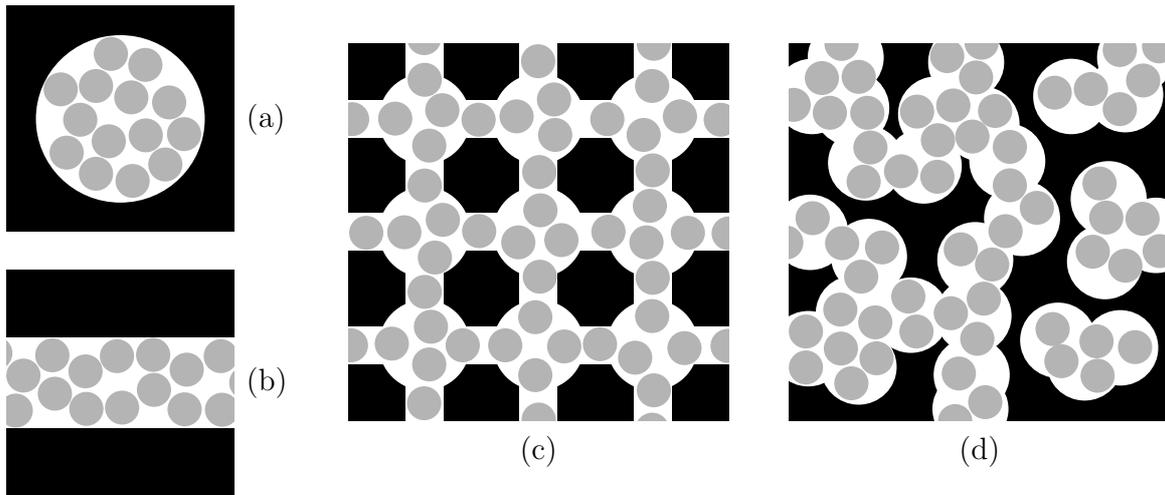


Figure 1.15: Confinement types. Two-dimensional schematic examples for types of confining media (black areas) with an immersed dense fluid (gray disks). Panel (a): single circular pore. Panel (b): single linear pore, corresponding to a single capillary or a single slit in three dimensions. Panel (c): multiple regularly-arranged circular pores with interconnections. Panel (d): multiple disordered and interconnected circular pores.

value $\lim_{t \rightarrow \infty} z(t)$ reverts to either zero or unity, implying either dynamic arrest or normal diffusion.

There exist, however, also systems with genuine long-time subdiffusion, i.e., with $0 < \lim_{t \rightarrow \infty} z(t) < 1$. In fact, the research on the various incarnations of anomalous diffusion constitutes a veritable scientific field of its own. Subdiffusion has been observed in a wide range of experimental systems, as for instance for the movement of particles in turbulent liquids, charge carriers in amorphous semiconductors, reptating polymers, beads in polymeric networks, lipids and proteins in cell membranes, colloids in quasi-one-dimensional setups, and most notably liquids in porous media [3, 83, 118, 119, 188, 191, 279]. On the theoretical side, equally many concepts have been developed to account for the experimental findings, including fractional Brownian motion, continuous-time random walks, Lévy flights, Lévy walks, time-fractional dynamics, and single-file diffusion [2, 17, 134, 165, 191, 192]. Last but not least, the phenomenon has been identified and investigated in detail in numerous computational studies [16, 18, 45, 55, 75, 96, 110, 111, 113, 150, 184, 246, 251, 252, 273].

1.3.3 Microscopic influence of confinement

1.3.3.1 General remarks

This section summarizes the most important microscopic mechanisms that underlie the confinement effects presented in Sec. 1.3.2. To understand these mechanisms, it is vital to take into account the nature of the confinement since some mechanisms

are specific to particular confinement types. One of the most influential properties of the confining medium is its geometry [3, 17, 95, 185, 186, 249]. Figure 1.15 shows various confinement geometries, each time with an example fluid immersed. As is evident, the confined fluid particles (gray) see an environment that is very different from that in a bulk fluid (compare Fig. 1.8). In each depicted case, a considerable fraction of the fluid’s particles are in proximity to the confining medium, and the same particles interact with much fewer neighboring particles than they would in a bulk fluid [95]. In the general case, the fraction of particles that are directly influenced by the confinement depends decisively upon the confinement’s geometry, with smaller pores entailing more influence of the confinement [3, 67, 95, 146, 187].

The pore size is important also for another reason. Any type of confinement is characterized by one or more length scales. Upon confining a fluid, these length scales are superimposed onto the inherent length scales of the immersed fluid (e.g., the diameter of the fluid particles, their local ordering, or their spatially correlating dynamics in the case of slow liquids). Thereby, the confinement’s length scales affect the arrangement and the movement of the fluid particles [3, 29, 42, 62, 67, 95, 160, 186]. For example, if a fluid is confined to an isolated pore [panel (a) of Fig. 1.15], then no property of the fluid can be associated with a length scale exceeding the size of that pore, thereby inducing “finite-size effects” [3, 95]. Similarly, confining a fluid between two parallel plates [panel (b) of Fig. 1.15] may effectively reduce the *dimensionality* of the fluid by one if the distance between the plates is only little more than one particle diameter [3, 95]^{1.58}.

Further effects result from the immediate spatial proximity between the confinement and some of the fluid particles. First of all, there is direct interaction between the particles and the confining medium. For instance, chemical reactions may take place, including in particular the possibility of the confining medium *catalyzing* a reaction between constituents of the fluid [10, 17, 67, 95, 101, 146, 196, 232, 289]. Catalysis is crucial in technology and a vast array of biological processes since it allows to greatly expedite and control the rate of chemical reactions. However, the present work will not focus on this topic, with the exception of the particle movements that may eventually lead to such reactions. Secondly, the confinement exerts *forces* upon the fluid’s particles. These forces usually are repulsive (thereby effecting the spatial confinement of the fluid) but may also be attractive. If the latter is the case, and if the attraction is comparable to or even greater than the force between the fluid particles, then the fluid tends to *wet* the surface of the confinement [3, 39, 48, 95]. This aspect is typically expressed in terms of the “hydrophobicity” or the “hydrophilicity” of the confinement. Thirdly, confinement may strongly affect the *arrangement* of fluid particles close to the confining medium. If those particles are relatively immobile due to the spatial proximity of both neighboring particles and the confinement, then their arrangement

^{1.58}Depending on the nature of the fluid particles, it may or may not be sensible to speak of the “diameter” of a particle. In the case of hard spheres, the definition of this property offers itself naturally (cf. Sec. 1.1.3).

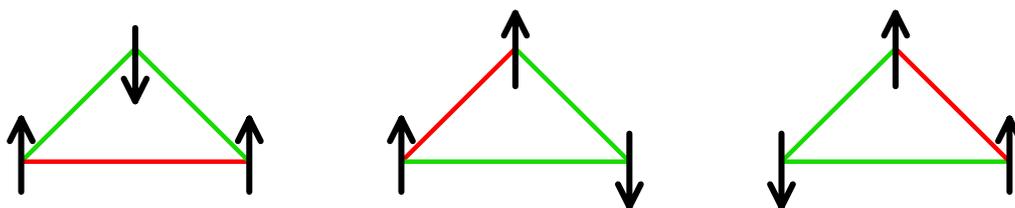


Figure 1.16: Frustration. Illustration of a frustrated ground state using the example of an antiferromagnetic three-spin system. Arrows: spins, each of which can point either up or down. Lines: interactions between spin pairs (green: energetically favored; red: energetically disfavored; interactions of the same color are identical). Interactions are energetically favored if the spins of a pair are antiparallel. Shown are three of the six configurations with the same lowest energy; the other three configurations mirror the depicted ones, i.e., up spins become down spins and vice versa. (Adapted from Ref. [33].)

is essentially imprinted by the geometry of the confinement. In the case of simple confinement geometries such as those in Fig. 1.15(a) and (b), this may lead to a “layering”, also known as a “wall effect” [3, 95, 187].

In the following, we will spotlight the two aspects of confinement that are most important to the context of slow liquids: frustration, and percolation.

1.3.3.2 Frustration

In many condensed-matter systems, there exists a unique particle arrangement (save translations and rotations of the whole substance) which represents the lowest-possible potential energy that can be realized in the system under the given external conditions. This arrangement is called the system’s “ground state” [33, 73, 108, 223, 238, 258]. If the external conditions include the vanishing of temperature—i.e., if the particles do not move—, then the ground state constitutes the single stable configuration of the substance. Usually, in condensed-matter systems the ground state is ordered, i.e., crystalline^{1.59}.

Other substances under other external conditions may *not* have a unique ground state, but instead exhibit multiple particle configurations with the same lowest-possible potential energy. These configurations are commonly subsumed under the term “frustrated ground state” [3, 33, 53, 62, 95, 154, 225, 241], and their existence implies that the corresponding system always retains a nonzero entropy. A simple system with a frustrated ground state is depicted in Fig. 1.16 [33]. It consists of three “spins” (black

^{1.59}In some substances, the number of distinct realizations of the *ground* state is equal to the number of *allowed* states. In these substances, a unique ground state exists only for a specific choice of external conditions. This is the case in particular for systems of bulk monodisperse hard spheres, where the potential energy is always zero and the only genuine thermodynamic control parameter is the density (cf. Sec. 1.1.3). In this system, the only unique state is attained at the maximum density in the crystalline phase where the particles cannot move.

arrows), each of which can assume either an “up” or a “down” state. Each spin interacts pairwise with all other spins, so that there are three interactions in the system (red and green lines). The interactions are “antiferromagnetic”, i.e., antiparallel spins are energetically favored over parallel spins. Neither the distance between the spins nor their identity plays a role. Obviously, it is impossible for all spin pairs to be antiparallel; instead, there is always at least one parallel pair. Therefore, there are three lowest-energy configurations in which two spins point up and one down (those depicted in the figure) and three more in which two spins point down and one up. This means that in this system there exist six equivalent ground state configurations—the system is frustrated.

In the context of condensed-matter systems with mobile particles, the term “frustrated” is often used more loosely, namely to refer to supercooled liquids that show only a marginal tendency to crystallize. Although most materials of this kind do not exhibit an actual frustrated ground state, they are valuable for elemental studies of slow liquid behavior^{1.60}. A prominent example for materials with a *genuinely* frustrated ground state are “atactic polymers”, i.e., polymers with randomly-varying bond types and/or side groups [3, 71, 101, 107, 154]. More generally, any substance with particles of random constitution is a good candidate for the existence of a frustrated ground state. In the case of hard spheres, for instance, crystallization can be avoided by making the particles *polydisperse*, i.e., slightly random in diameter (cf. Sec. 1.1.3.1). This is exploited in both simulation and experiment, the latter typically (and often unavoidably) involving diameter variations of as much as ten percent [62, 93, 104, 140, 211, 214, 227, 229, 277].

One of the fundamental motivations of the present work is the fact that frustration may also be introduced into a condensed-matter system by subjecting it to suitable boundary conditions. It is natural to suspect that if the boundary conditions are spatially random, then the ordered particle arrangement of a crystal might be disfavored or even entirely suppressed. Such boundary conditions are naturally provided by *random porous media* as depicted for instance in Fig. 1.15(d). This constitutes one of the reasons why liquids—and in particular slow liquids—in porous media are interesting from a fundamental point of view [10, 95, 100, 144–146, 189, 224, 266, 289]. However, not all random porous media actually suppress crystallization: only if a sufficient fraction of fluid particles is directly influenced by the random geometry of the medium, the randomness is translated to the overall arrangement of the fluid particles. This means that the porous medium in question has to be random on a length scale similar to the size of a fluid particle^{1.61}.

^{1.60}In particular, frustration in slow liquids has been associated with the arranging of a substance’s particles in “locally preferred structures” (LPSs). LPSs may impede crystallization if they do not tile the given space without gaps, as is for instance the case if the LPS of a substance is an icosahedron (a twenty-faceted platonic solid). Also, LPSs may play an important role for the slowdown of substances in the approach of dynamic arrest at nonzero temperatures (cf. Sec. 1.2.5) [51, 53, 62, 225].

^{1.61}Recently, approaches have been developed in which disordered boundary conditions reveal growing static correlation lengths. For instance, in order to compute the “point-to-set” corre-

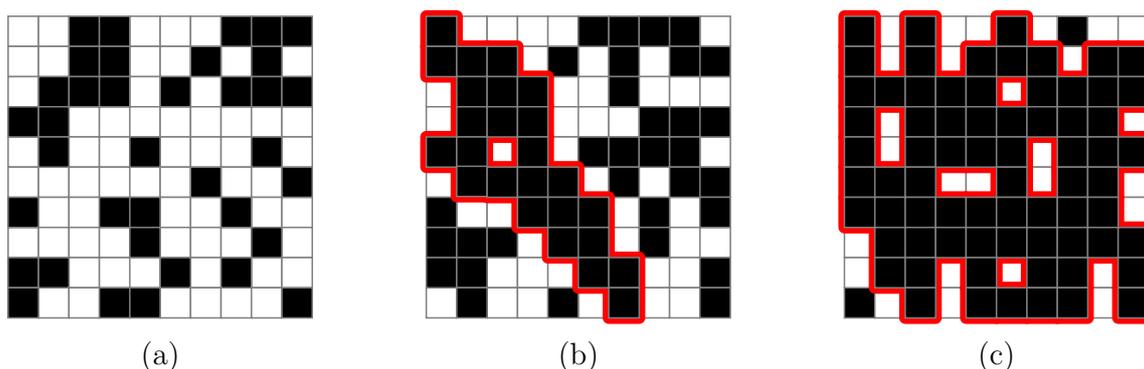


Figure 1.17: Percolation. Example system in which percolation may or may not occur. Each element of the square lattice is occupied (black) with probability P and not occupied (white) otherwise. Depending on P and on the realization, there may exist a percolating backbone, i.e., a cluster of edge-sharing occupied squares spanning across the system (here from top to bottom, red outline). Panels (a), (b), and (c) show realizations for $P = 40\%$, $P = 59\%$, and $P = 80\%$, respectively. In an infinitely-large system, percolation occurs for $P \gtrsim 59\%$ [276].

1.3.3.3 Percolation

One of the most important aspects in the behavior of a confined fluid is the following: Assuming that the system of interest is infinitely large, does there exist a pore in which fluid particles can propagate infinitely far? This issue is crucial to phenomena like flow and diffusion, which are ultimately limited if such a pore does not exist [39, 45, 107, 152, 179, 229, 248]. In simple geometries such as in Fig. 1.15(a) and (b), whether such a pore exists is usually straightforward, and likewise is it in periodically-replicated geometries such as in Fig. 1.15(c). For *disordered* porous media, however, is difficult to determine whether the medium is of the “open pore” or the “closed pore” type, i.e., whether or not a “percolating backbone” of infinitely-many interconnected pores exists. The reason for this is evident from Fig. 1.15(d), which shows that the pore sizes and shapes in disordered porous media may vary strongly. Specifically, some of the pores may be small and disconnected from all other pores; others, in contrast, may constitute a considerable fraction of the overall volume of the confining medium and thus be infinitely large if again it is assumed that the confining medium is infinitely large.

From a theoretical point of view, the challenge is to predict the existence of a percolating backbone from properties of the overall material [122, 152, 175, 179, 248, 276, 291]. For this, it is instructive to consider model substances such as the one shown

lation length (cf. Sec. 1.2.5), some of the particles of a substance in a fluid configuration are “pinned” [25, 26, 29, 34, 42, 46]. Depending upon the aim of the investigation (e.g., for deductions based on the “mosaic theory”), the pinned particles may for instance reside outside a spherical shell, outside a slit geometry, or simply be chosen at random. Notably, the latter procedure is closely related to the porous media model employed in this work (cf. Sec. 1.3.4.2).

in Fig. 1.17. The substance in the figure consists of “sites” arranged on a square lattice, which are “occupied” (black) with probability P and “not occupied” (white) otherwise. The occupied squares form “clusters”, i.e., groups in which each member is connected to at least one other member by sharing an edge. A percolating backbone in this system is a cluster that stretches across the entire system (top to bottom or left to right), and it may exist depending on P and on the realization. As can be shown (see references above), if the system depicted in Fig. 1.17 were infinitely large, then there would exist a unique value $P = P^*$ above which a backbone always exists and below which it never exists. This behavior is akin to *phase transitions* in condensed-matter systems, with P being the control parameter, P^* being its critical value, and the existence of the percolating backbone being the order parameter. This suggests that statistical-mechanical methods may be useful in determining P^* (usually called the “percolation threshold”), and in fact there does exist a large body of literature on this topic.

Of particular importance in the context of porous media are *continuum* percolation models [107, 113, 142, 152, 248, 252]. Such models differ in two aspects from lattice models like the one in Fig. 1.17: Firstly, the sites may be spatially disordered and connected to any number of neighboring sites. Secondly, each site may assume a value from a continuum of possible values, meaning that there may be varying degrees of connectivity. A popular continuum percolation model is the “random resistor network”—electrical resistors with random strengths and connections—, the question being whether or not (and how much) electrical current may flow across the network [113, 152]. In view of Fig. 1.15(d), it is intuitive that continuum percolation models provide a more natural description of porous media than lattice percolation models, and it is precisely the former class of models that we will use to characterize the model porous medium investigated in this work (see Sec. 2.4). The caveat of this class of models is that pertinent theories are significantly more complicated than lattice percolation theories.

Of the research conducted on percolation, a significant fraction involves simulation methods. As mentioned above, porous media are typically modeled as networks. In a computer’s memory, any such network is necessarily finite; therefore, whether it contains a percolating backbone is in principle a mere bookkeeping problem [122, 276]. It is two aspects that render computational percolation analyses of disordered porous media challenging. Firstly, it is nontrivial to express the geometry of a porous medium in terms of an appropriate network. This issue will be discussed in detail in Sec. 2.4 and in Appendix A.1. Secondly, “percolation” is usually understood to refer to a phenomenon of infinite size (the percolating backbone), which cannot be represented in a finite-sized system. In order to nevertheless evaluate accurate percolation threshold values in simulations, one may examine properties that are particularly insensitive or particularly sensitive to finite-size effects (see Sec. 2.5.4), or, if possible, make use of finite size–scaling methods [2, 33, 175, 213, 248].

1.3.4 Modeling of disordered porous confinement

1.3.4.1 Characterization and modeling strategies

In view of the manifold microscopic mechanisms at work in fluids in disordered confinement (Sec. 1.3.3) and the resulting contradictory phenomena (Sec. 1.3.2), it is challenging to devise a pertinent model that captures the essential physics and all the while is tractable by simulation and theory [3, 57, 58, 95, 185–187]. Ideally, such a model should allow to disentangle the various mechanisms in order to assess their respective contributions to the changed fluid properties. For example, a reduced available volume should decrease the entropy of a confined fluid and therefore *increase* its glass transition temperature T_g , while T_g is expected to *decrease* as there is an increase of hydrostatic effects at surfaces [3].

The principal problem is the considerable variation in the composition and constitution of porous media [3, 95, 185–187], which may require many parameters for a meaningful description. Among the relevant features—all of which may decisively influence the structure and dynamics of the fluid—are

- the presence of a percolating backbone,
- the fraction of volume accessible to an immersed fluid (the “porosity”),
- the ratio between internal surface area and volume,
- the distribution of pore locations and sizes,
- the distribution and strength of inter-pore connections, and
- the interactions between fluid and porous medium.

In experiments, porous media are typically characterized on the basis of the arrangement and chemistry of their constituent particles, popular methods being X-ray diffraction [95], neutron diffraction [95], positron annihilation [127], and photochromic labeling [127].

From a theoretical point of view, porous media are essentially tantamount to an external potential. However, for *disordered* porous media this potential may be highly complicated, with the possibility of chemical interactions adding further to the complexity. Therefore, intelligent modeling strategies are required. A class of particularly simple models is that of lattice geometries such as the one mentioned in Sec. 1.3.3.3 in the context of percolation. In this class, once settling for a specific lattice geometry, sensible porous media models can typically be obtained using only a single parameter (for instance an occupation probability) [122, 152, 248, 276, 291]. However, unfortunately, lattice models are *not* suited to represent *liquids* since they cannot account for effects such as packing and layering [154, 224].

A different modeling strategy is inspired by isolated pores as in Fig. 1.15(a), namely to consider a solid homogeneous substance into which many such pores are bored at random locations. Upon defining the geometry of an individual pore, the parameters in this strategy merely need to account for the density and the spatial distribution of the pores. A two-dimensional example for porous media of this kind is depicted in Fig. 1.15(d), where the pores are circular (with fixed diameter) and

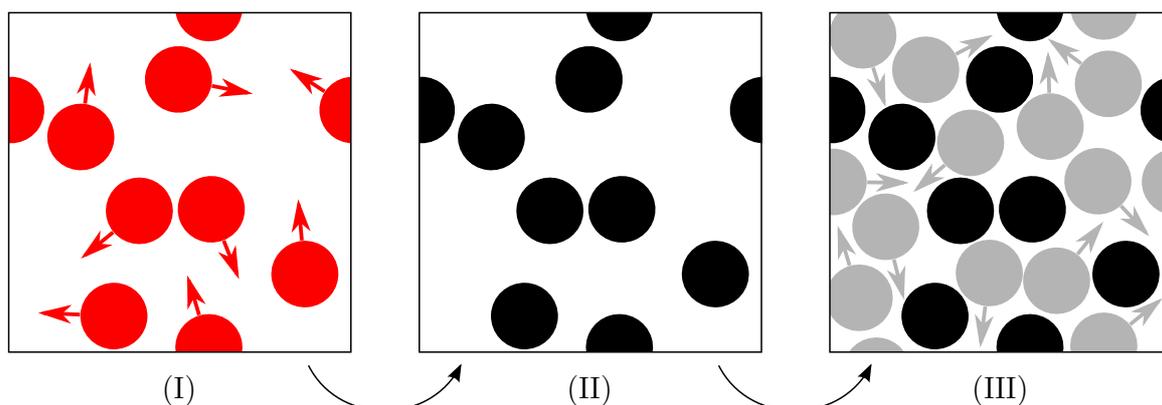


Figure 1.18: The quenched-annealed model. Construction of a disordered porous medium and of an immersed fluid according to the QA protocol. Details: see text. (I) Equilibrium configuration of matrix precursor fluid with mobile interacting particles. (II) Matrix of fixed particles. (III) Mobile fluid particles inserted into the matrix at random positions. Arrows indicate the movement of particles.

randomly distributed (i.e., possibly overlapping). This particular model has also come to be known as the “inverse Swiss cheese model” (cf. Sec. 1.3.4.2 and Refs. [67, 290]).

Probably most inspiring for the modeling of porous media is the fact that these media, like any substance, ultimately consist of particles. Accordingly, some modeling strategies regard a porous medium as a collection of particles that are spatially disordered and restricted in their movement (e.g., by chemical bonds). For further simplification, these particles may for instance be modeled as entirely immobile and/or as relatively large so that their interstitial volumes constitute the pores for the smaller particles of a fluid to move in [39, 51, 57, 58, 97, 98, 146, 152, 229]. Models of this kind are intuitively sensible considering aggregates such as sand or powders, but they are also representative of other porous media like silica gels (recall Fig. 1.14) [51, 95, 229, 266, 289]. In combination with the simplest choice of particles—spheres—, this model features the advantage that the pertinent arrangements (in particular “random close packing” structures) are very well investigated^{1.62}.

1.3.4.2 The quenched-annealed model

The quenched-annealed (QA) model has been designed specifically to investigate the combination of disordered porous confinement and an immersed fluid. The model is based on the idea of representing a porous medium by particles that are artificially fixed in space. These particles are usually referred to as “obstacles”, “pinned particles”, or “matrix particles”, and collectively as the “matrix” [98, 100, 174, 189, 266]. The

^{1.62}Sphere packings are well investigated due to the fact that they are representative also of many other physical systems such as colloidal liquids and glasses (cf. Footnote 1.52), and due to the fundamental interest in random close packing structures (cf. Sec. 2.3.2.2) [39, 77, 130, 132, 243, 260, 261, 292].

QA model shares the idea of fixing particles with a number of other models [29, 41, 42, 49, 135, 147, 148, 150, 163, 164, 252], its distinguishing feature being the protocol for obtaining the distribution of the matrix particles. This protocol is visualized in Fig. 1.18, and can be summarized in the following three steps (see also Sec. 2.3):

- (i) Into the space that is to host the QA system, first insert all the particles that are to constitute the matrix. (At this stage, the matrix particles represent a precursor fluid, i.e., they are not yet fixed but rearrange and interact as in an ordinary condensed-matter system. This means that an interaction between the matrix particles needs to be defined.) Equilibrate the matrix precursor fluid.
- (ii) Choose an arbitrary configuration of the matrix precursor fluid that is representative of the state of equilibrium. Rigorously fix the particles at the positions corresponding to that configuration to obtain the matrix.
- (iii) Into the matrix, insert all particles that are to represent the immersed fluid. In the insertion, place the particles at random positions irrespective of the pore structure of the matrix. Forbidden are only positions that correspond to an infinite potential energy of the system.

Notably, according to the protocol, the external parameters (temperature, pressure, etc.) at which the matrix precursor fluid is equilibrated are entirely independent of the external conditions at which the immersed fluid is observed. Likewise, the interactions among the matrix precursor particles, the interactions between fluid and matrix particles, and the interactions among the fluid particles are independent of each other, and in each case the interaction potential can be chosen freely.

The name “quenched-annealed” for this protocol derives from techniques used in metallurgy and in the fabrication of glasses. There, a molten substance is first cooled rapidly (“quenched”) to obtain an essentially-solid material^{1.63}. Subsequently, the substance is reheated to temperatures close to a liquid state (“annealed”) in order to allow for the relaxation of stresses induced by the quenching^{1.64,1.65}. In a warped sense, these techniques translate to the components of a QA system, the matrix being the product of a quench, and the fluid component resulting from annealing.

From an experimental point of view, a rigorous fixing of particles is clearly artificial; however, it can be justified as being *approximative* of various real systems. For instance, it may represent particles that are very massive and therefore very slow [45, 75]. However, in true thermal equilibrium such massive particles would still be influenced by the fluid particles, which is not the case in the QA protocol. Alternatively, the fixed particles could be thought of to be held in place by external forces. In two-dimensional systems, this can be accomplished for instance by optical tweezers [29],

^{1.63}<http://en.wikipedia.org/wiki/Quenching>

^{1.64}http://en.wikipedia.org/wiki/Annealing_%28glass%29

^{1.65}http://en.wikipedia.org/wiki/Annealing_%28metallurgy%29

or by the clamping of particles between two glass plates [57, 58]. In this case, the QA picture does not account for the fact that the fixed particles undergo residual oscillations whenever the external force is finite. Likewise in two dimensions, matrix particle configurations can be emulated by lithography [236], which, however, is tedious if an averaging over different samples is desired (see below).

From a theoretical point of view, the QA model presents tremendous opportunities which more than compensate for its moderately artificial nature. In fact, the model has been *designed* to suit the needs of “integral equation theory”, where on the basis of the “Ornstein-Zernike” (OZ) equation it is possible to infer the equilibrium structure and thermodynamics of a system [115, 170, 208]^{1.66}. Specifically, via the “replica trick”, the OZ equation has been extended to a set of “replica Ornstein-Zernike” (ROZ) equations in order to account for the rigorous fixing of the matrix particles [97–100, 180, 181]. Among others, the replica trick reconciles the requirement of thermal equilibrium with the fixing of particles, namely by imposing the condition that in addition to the usual thermodynamic averaging there be an average over matrix realizations (“replicas”). Based on the ROZ equations, many properties of the fluid component of QA systems have been investigated in detail by theoretical methods. This includes for instance the pair distribution function [174, 189, 266], the Gibbs-Duhem equation, the virial, and the pressure [144, 224], as well as phase diagrams [145] for one-component QA fluids, along with similar investigations for two-component QA fluids [206, 207, 232]. The most recent major development—a theoretical method to investigate the *dynamic arrest* of QA fluids—will be presented in detail in Sec. 1.4.

From a simulational point of view, investigations of QA systems are almost as simple and even more efficient than investigations of two-component systems since the particles of the matrix species do not need to be propagated. However, if simulation data are to be compared to the predictions of integral equation theory, then additionally the averaging over replicas has to be performed. Even if in practice the number of replicas is limited, this entails a computational effort well exceeding that of the simulation of ordinary multi-component systems. (For simulational details see Chapter 2.)

A much investigated special case of QA systems is the “Lorentz gas” [16, 18, 83, 104, 107, 111, 113, 175, 246, 247, 252]. In brief, the three-dimensional version of

^{1.66}In an isotropic and homogeneous one-component liquid system, the Ornstein-Zernike (OZ) equation reads $h(r) = c(r) + \rho \int c(|\mathbf{r}-\mathbf{r}'|) h(r') d\mathbf{r}'$, where ρ is the number density, $h(r) = g(r) - 1$ with $g(r)$ being the radial distribution function (cf. Sec. 2.5.2.1), and $c(r)$ is the “direct correlation function” [98–100, 115, 144, 145, 174, 189, 202, 266]. Evaluating the OZ equation presents two principal problems: (i) the equation does not include system-specific information, and (ii) its right-hand side corresponds to an infinite integral series that cannot be approximated by simple truncation. Both problems are taken care of by “closure relations” (CRs), which provide controlled approximations. For instance, the “hypernetted chain” CR assumes that $c(r) = h(r) - \ln[h(r)+1] - \beta\Phi(r)$, where β is the inverse temperature, and $\Phi(r)$ is the inter-particle potential and therefore introduces system-specific information [115, 145, 189, 206, 239]. Although CRs typically eliminate $c(r)$ or $h(r)$ from the OZ equation, in most cases the remaining equation is analytically unsolvable. Therefore, the final step usually involves converging numerical evaluation schemes.

this model encompasses a matrix of monodisperse hard spheres that are randomly distributed (i.e., they may overlap) and a fluid that consists of a single hard-sphere tracer particle of the same size. This is tantamount to a QA system with (i) an immersed fluid of infinitesimally low density, (ii) no interactions between the particles of the matrix precursor fluid, and (iii) monodisperse hard-sphere interactions between the matrix particles and the fluid particle. This model is also referred to as the “Swiss cheese model” [107, 175, 252], and has a long-standing and successful history of investigations in the fields of both static percolation [107, 175] and particle dynamics [18, 83, 104, 111, 113, 246, 247].

1.4 Theoretical predictions

One of the primary motivations of this work is the fact that for the phenomena and the systems investigated in this work—dynamic arrest in hard-sphere quenched-annealed systems—there exist detailed theoretical predictions. In this section, we provide a brief overview over established theories of the glass transition, describe in basic terms the theory of interest to this work, and present the pertinent relevant predictions.

1.4.1 Theories of the glass transition

Describing the glass transition is a notoriously hard problem for theoretical physicists. Among others, this is due to the following fundamental issues:

- (i) Via supercooling and dynamic arrest, the glass transition involves nonequilibrium physics.
- (ii) It is uncertain whether the nature of the glass transition is ultimately dynamic or thermodynamic (the latter implying an equilibrium phenomenon).
- (iii) It is unclear which quantity—if any—is the “correct” indicator of the glass transition.
- (iv) Depending on the material and the considered quantities, “the glass transition” might refer to multiple independent phenomena.

Over the past decades, numerous material properties have been suggested to indicate the glass transition, and similarly many theories have been developed on their basis. In line with point (ii) above, these theories generally fall in one of two categories: theories of dynamic arrest, and theories of thermodynamic glass transitions.

In the thermodynamics corner, probably the most widely recognized representatives are mean field theories [33, 46, 205, 241] and the “random first order” (or “mosaic”) theory [26, 34, 178, 199, 241], which like other theories of this class relate to thermodynamic quantities (e.g., configurational entropy [1, 62, 71, 103, 138, 154]) and/or sophisticated structural features such as point-to-set correlation functions [25,

26, 34, 42, 157, 231], nonaffine displacement fields [63, 72], or local order parameters [26, 53, 178, 225, 231, 255]. Fundamental questions pertinent to this view are “What symmetry does the transition break (i.e., what is the order parameter)?” and “Can a simple substance have an amorphous ground state?”

In the dynamics corner, which this work is set in, focus is instead on quantities like relaxation times, fragilities [11, 62, 71, 104], dynamic heterogeneities [25–27, 62, 140, 157, 282], the traversal of energy landscapes (“hopping”) [62, 154, 178, 241], and relations between these. This class of theories includes the early approach of “cooperatively rearranging regions” by Adam and Gibbs [1, 66, 103, 104, 277] and the more recent theories pertinent to “kinetically constrained” models [26, 117, 223, 241] as well as the “self-consistent generalized Langevin equation” (SCGLE) theory [131, 204, 217]. Most notably, however, the celebrated “mode-coupling theory of the glass transition” [71, 93, 104, 154, 211, 220, 223, 241, 242, 270, 271] which we will describe in more detail in Sec. 1.4.3, is a dynamic theory.

In the case of hard-body, or “granular”, systems (a class that includes hard spheres, cf. Sec. 1.1.3), the concept of “jamming” [132, 154, 173, 223, 237, 259, 260, 262] additionally allows for the formulation of theories that link structural with dynamic features. As early as 1959, for example, Cohen and Turnbull suggested that the dynamics of such systems is determined by the amount and/or the percolation of “free volume” [52, 103, 104, 196, 241, 252], whereas more recent works suggest for instance that the glass transition of hard-body systems corresponds to a “rigidity percolation” [81, 260]. Therefore, such systems are promising study objects in the context of the glass transition—despite the absence of temperature as a control parameter (cf. Sec. 1.1.3.2).

1.4.2 Mode-coupling theory

The mode-coupling theory of the glass transition (MCT) is arguably the most successful theory for describing the dynamic slowdown and eventual arrest of glass-forming systems, and it has been shown to make reasonable predictions—both qualitatively and quantitatively—for a number of different systems and conditions [51, 66, 80, 93, 140, 155, 156, 211, 240, 242, 265]. In the following we will outline its general ideas, combining the introductions by Kob [154], Sciortino and Tartaglia [241] and Voigtmann [271].

MCT was developed in the mid-1980s by Götze, Bengtzelius, and Sjölander [20], whose motivation—at least in hindsight—was to mathematically describe the dynamics pertinent to the cage effect, the reason being that this effect is central to the structural relaxation of slow liquids. As discussed in Sec. 1.2.7, the breaking of cages requires the coordinated motion of multiple particles at different times and in different locations. MCT considers the motion of particles indirectly in terms of local particle densities, which transformed to the Fourier domain read

$$\rho_{\mathbf{k}}(t) = \sum_{i=1}^N e^{i\mathbf{k}\cdot\mathbf{r}_i(t)}, \quad (1.10)$$

where $\mathbf{r}_i(t)$ is the location of the i^{th} of N particles at time t , and \mathbf{k} is the considered wave vector. The central idea of MCT is to construct an “equation of motion” in which the $\rho_{\mathbf{k}}(t)$ are the degrees of freedom and which accounts for the influence of the $\rho_{\mathbf{k}}(t)$ onto themselves across space and time^{1.67}. The most important tool in this endeavor is the famous Zwanzig-Mori projection-operator formalism, which provides a means to re-express the time evolution of correlation functions [20, 104, 115]. The formalism predates MCT by two decades and has been instrumental for instance in the description of critical phenomena and of Brownian motion [154]. In the case of MCT, the relevant correlation function involves the densities from Eq. (1.10),

$$F(k, t) = \frac{\langle \rho_{\mathbf{k}}(t) \rho_{-\mathbf{k}}(0) \rangle}{S(k)}, \quad (1.11)$$

where $S(k) = \langle \rho_{\mathbf{k}} \rho_{-\mathbf{k}} \rangle$ is the static structure factor (cf. Sec. 2.5.2.2), and only the modulus $k = |\mathbf{k}|$ is relevant in the correlator if the system is isotropic. $F(k, t)$ is usually called the (collective) intermediate scattering function (ISF; cf. Sec. 2.5.3.2). By a rather tedious twofold application of the Zwanzig-Mori formalism—once on $F(k, t)$, and once on a subterm arising in the first application—, it can be shown that for the case of Newtonian dynamics, $F(k, t)$ obeys the exact relation

$$\ddot{F}(k, t) + \Omega^2(k) F(k, t) + \Omega^2(k) \int_0^t M(k, t-s) \dot{F}(k, s) ds = 0, \quad (1.12)$$

where $\Omega^2(k) = k^2 / [m \beta S(k)]$, with β being the inverse temperature and m being the particles’ mass. The term $M(k, t)$ is the so-called “memory kernel”, which via the enclosing integral provides the desired feedback mechanism for $F(k, t)$ to influence itself at different times.

Unfortunately, the memory kernel expression emanating from the Zwanzig-Mori formalism does not allow to solve Eq. (1.12). In order to obtain an evaluable term, $M(k, t)$ is first split into two contributions named the “regular” and the “slow” memory kernel. The regular memory kernel by definition determines only the short-time evolution of $F(k, t)$ and can thus be dropped in the context of dynamic arrest^{1.68}. This leaves the task of obtaining a usable expression for the slow memory kernel, $M^{\text{slow}}(k, t)$, which is accomplished by projecting it onto a subspace of known quantities. MCT’s subspace of choice for this procedure is that of “density pairs”, which is chosen ad-hoc and only due to its being the simplest subspace nonorthogonal to $M^{\text{slow}}(k, t)$. This projection has come to be known as the “MCT approximation”. In a second ad-hoc

^{1.67}The name “mode-coupling theory” derives from the fact that particle densities in the Fourier domain are also called “modes”, and from the assertion that these modes “couple”, i.e., influence each other and themselves.

^{1.68}As can be shown, the predictions of MCT are in fact insensitive to the microscopic dynamics (Newtonian, Brownian, etc.) in the sense that a different microscopic dynamics merely shifts the slow dynamical features by a constant factor in the time domain. It is this finding that warrants the neglecting of the regular memory kernel, and it is likewise this finding that lends meaning to the use of Newtonian dynamics in the simulation of the slow dynamics of colloidal particles (cf. Sec. 2.2).

simplification, four-point correlation functions arising from this projection are replaced by products of ISFs (the so-called “convolution approximation”), and three-point correlation functions are neglected altogether. The resulting expression for the memory kernel reads

$$M(k, t) \approx M^{\text{MCT}}(k, t) = \int V^{(2)}(\mathbf{k}, \mathbf{q}, \mathbf{p}) F(q, t) F(p, t) \frac{d\mathbf{q}}{(2\pi)^3} \quad (1.13)$$

with the “quadratic vertex”

$$V^{(2)}(\mathbf{k}, \mathbf{q}, \mathbf{p}) = \frac{n}{2k^4} S(k) S(q) S(p) [\mathbf{k} \cdot \mathbf{q} c(q) + \mathbf{k} \cdot \mathbf{p} c(p)]^2 \quad (1.14)$$

where n is the particle number density, $c(k) = [1 - 1/S(k)]/n$ is the Ornstein-Zernike direct correlation function, and $\mathbf{p} = \mathbf{k} - \mathbf{q}$. Equations (1.12), (1.13), and (1.14) together with the boundary conditions $F(k, 0) = 1$ and $\dot{F}(k, 0) = 0$ (see Ref. [241]) represent a closed equation for $F(k, t)$ in which all other quantities [namely m , n , β , and $S(k)$] are assumed to be known. The only remaining obstacle is the fact that the equation is analytically unsolvable; however, it is well possible to numerically determine approximate solutions to any desired degree of precision.

All things considered, MCT is therefore capable of determining the *time evolution* of $F(k, t)$ using only *static* structural properties as input parameters. Of the input parameters, the only nontrivial case is $S(k)$. If a completely theoretical description of a system is desired, then $S(k)$ can be obtained from integral equation theory (see Footnote 1.66), which, however, introduces approximations associated with closure relations in addition to the MCT approximation. The alternative is to experimentally or simulationally measure $S(k)$ in the system of interest, which likewise introduces uncertainties.

The principal capability of MCT, and the one most relevant to this work, is its power to predict from the input information whether or not the pertinent system is dynamically arrested. The latter is manifested primarily in the “nonergodicity parameter” $f(k) = \lim_{t \rightarrow \infty} F(k, t)$, which is nonzero in the arrested case and zero otherwise. MCT provides a host of further predictions, many of which have been tested and verified^{1.69}. The predictions of MCT relevant specifically to this work will be discussed in Sec. 1.4.3.

^{1.69}Notwithstanding its success, MCT also has well-known shortcomings [51, 93, 154, 241, 265]. For instance, MCT originally did not include predictions for single-particle properties such as self diffusion. Also, the conditions for which MCT predicts dynamic arrest are usually quantitatively off by at least 20%. More fundamentally, MCT has been criticized for its prediction of “ideal” dynamic arrest, which is most likely unphysical. The latter two problems have been attributed mainly to the projection on the subspace of density pairs, which has been labeled “uncontrolled” since the choice of subspace is arbitrary and no “small parameter” is involved as in perturbation theories. In response, additional subspaces like density currents have been considered, which resulted in the avoidance of ideal transitions and in increased quantitative accuracy, but came at the cost of vastly more complex expressions.

1.4.3 Replica MCT and its predictions

Over the past years, Krakoviack published a series of articles in which he adapted MCT to fluids with quenched disorder and in which he investigated the pertinent predictions for dynamic arrest [159–164]. The theory considers quenched disorder in the form of pinned particles that are arranged according to equilibrium configurations of another fluid. This is precisely the premise of quenched-annealed (QA) systems (cf. Sec. 1.3.4.2), and in fact hard-sphere (HS) QA systems are among the systems for which Krakoviack investigated his theory.

The MCT equations resulting from Krakoviack’s efforts are remarkably similar to those of the original MCT equations, the main differences being:

- Any involved quantity that is supposed to be spatially homogeneous and isotropic has to be averaged not only thermodynamically but also over realizations of the quenched disorder (indicated by the symbol $\overline{\cdots}$, compare Secs. 1.3.4.2 and 2.5.1.5). Owing to this requirement, Krakoviack’s MCT for fluids with quenched disorder has come to be known as “replica MCT” (RMCT).
- In RMCT, all quantities refer exclusively to the fluid (i.e., nonpinned) particles. This includes the particle mass m , the number density n , the inverse temperature β , the Fourier-space density $\rho_{\mathbf{k}}(t)$, the static structure factor $S(k)$, and the (collective) ISF $F(k, t)$.
- In the presence of quenched disorder, $F(k, t)$ can never fully relax. This is a purely static phenomenon. However, $\lim_{t \rightarrow \infty} F(k, t) > 0$ indicates dynamic arrest in MCT, which would imply QA fluids to always be arrested. Since this is not a useful prediction, in place of $F(k, t)$, RMCT considers the “connected” ISF,

$$F_c(k, t) = \frac{\overline{\langle \delta \rho_{\mathbf{k}}(t) \delta \rho_{-\mathbf{k}}(0) \rangle}}{S_c(k)}, \quad (1.15)$$

which involves $\delta \rho_{\mathbf{k}}(t) = \rho_{\mathbf{k}}(t) - \langle \rho_{\mathbf{k}} \rangle$, i.e., only the *fluctuations* of the local density (cf. Sec. 2.5.3.3). These density fluctuations also define the connected structure factor $S_c(k) = \overline{\langle \delta \rho_{\mathbf{k}} \delta \rho_{-\mathbf{k}} \rangle}$. Except for the changes detailed in the next point, the RMCT expressions are obtained from the MCT expressions [Eqs. (1.13) and (1.14)] by the straightforward substitution $F(k, t) \rightarrow F_c(k, t)$ and $S(k) \rightarrow S_c(k)$.

- The RMCT memory kernel is more complex than the MCT memory kernel from Eq. (1.13). It reads

$$\begin{aligned} M(k, t) &\approx M^{\text{RMCT}}(k, t) \\ &= M^{\text{MCT}}(k, t) + \int V^{(1)}(\mathbf{k}, \mathbf{q}, \mathbf{p}) F(q, t) \frac{d\mathbf{q}}{(2\pi)^3} \end{aligned} \quad (1.16)$$

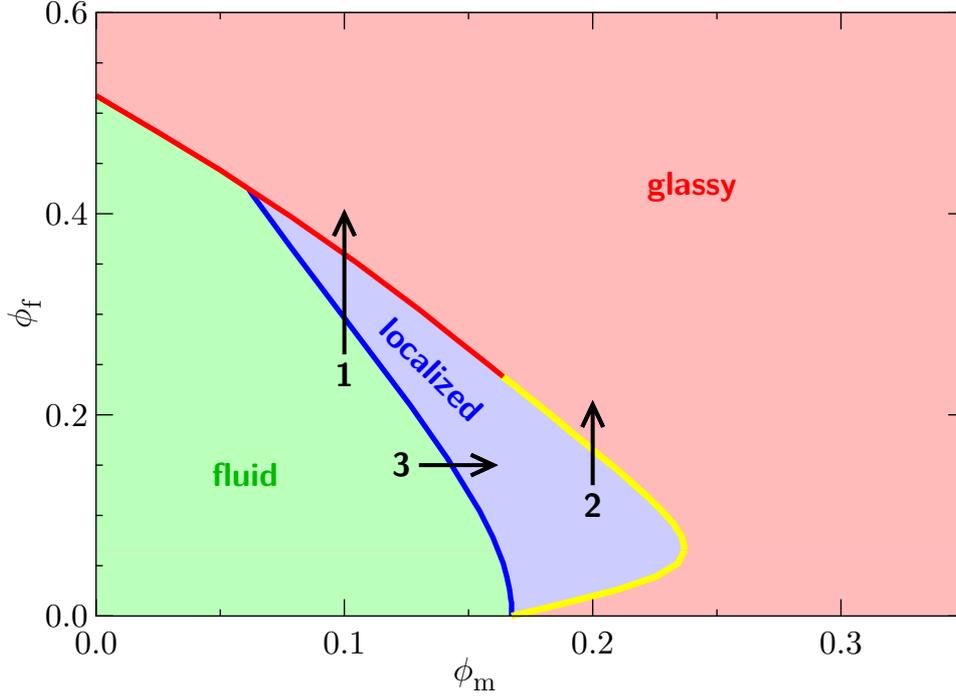


Figure 1.19: RMCT HS-QA kinetic diagram. Depending on the packing fractions of the matrix, ϕ_m , and of the fluid, ϕ_f , RMCT predicts HS-QA systems to assume either a fluid (green area), a localized (blue area), or a glassy (red area) state. The pertinent parameter-space regions are separated by type-B (red line), type-A (yellow line), and localization (blue line) transitions. Black arrows indicate parameter-space paths followed in Figs. 1.20, 1.21, and 1.22. (Reproduced from Ref. [162], Fig. 2(a).)

with the “linear vertex”

$$V^{(1)}(\mathbf{k}, \mathbf{q}, \mathbf{p}) = \frac{n}{2k^4} S_c(k) S_c(q) S_b(p) \left[\mathbf{k} \cdot \mathbf{q} c_c(q) + \mathbf{k} \cdot \mathbf{p} n^{-1} \right]^2, \quad (1.17)$$

where $S_b(k) = \overline{\langle \delta \rho_{\mathbf{k}} \rangle \langle \delta \rho_{-\mathbf{k}} \rangle}$ is the “blocked structure factor”, and $c_c(k) = [1 - 1/S_c(k)]/n$.

In consequence, the input quantities of RMCT are m , n , β , $S_c(k)$, and $S_b(k)$. Remarkably, this means that none of the matrix properties are referenced directly by RMCT; instead, the influence of the matrix upon the fluid is represented only indirectly via $S_c(k)$ and $S_b(k)$.

RMCT makes several intriguing predictions for the dynamic arrest of fluids with quenched disorder, most of which are reflected in the case of monodisperse HS-QA systems. In the following, we focus primarily on those predictions that later in this work will be examined using computer simulations (see Chapter 3). For an overview, let us start with the “kinetic diagram”, i.e., with the set of information about a system’s dynamic state for all (relevant) points in its control parameter space. As shown in

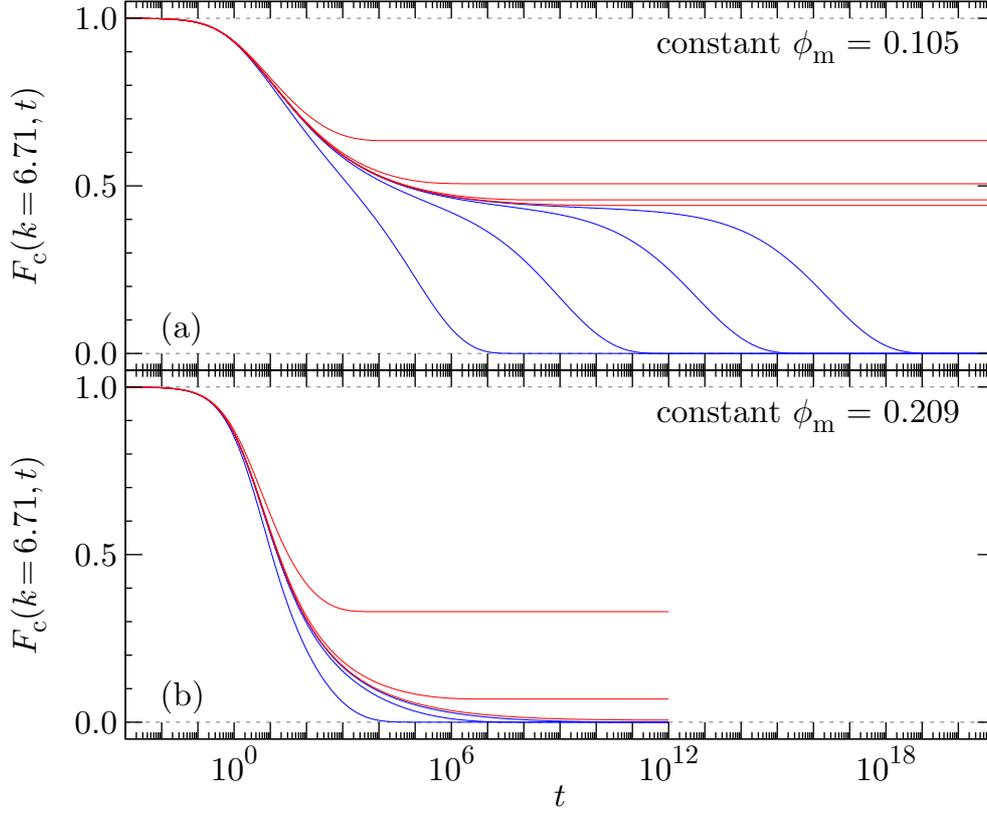


Figure 1.20: RMCT HS-QA connected ISF. RMCT predictions for the relaxation of the connected ISF in HS-QA systems. In both panels, the wave vector is $k = 6.71$, the matrix packing fraction ϕ_m is constant, and the fluid packing fractions are $\phi_f \in \{0.9\phi_f^g, 0.99\phi_f^g, 0.999\phi_f^g, 0.9999\phi_f^g, 0.99999\phi_f^g, 1.00001\phi_f^g, 1.0001\phi_f^g, 1.001\phi_f^g, 1.01\phi_f^g, 1.1\phi_f^g\}$. Panel (a): $\phi_m = 0.105$ and $\phi_f^g \simeq 0.352$. Panel (b): $\phi_m = 0.209$ and $\phi_f^g \simeq 0.146$. Colors correspond to regions in Fig. 1.19; curves are listed bottom left to top right. (Reproduced from Ref. [160], Figs. 1(a) and 1(b).)

Fig. 1.19, the parameter space of HS-QA systems is spanned by the packing fractions of the matrix, ϕ_m , and of the fluid, ϕ_f . In this space, RMCT predicts HS-QA systems to assume either a “fluid” (green area), a “localized” (blue area), or a “glassy” (red area) state. These states are separated and defined by “type-B” (red line), “type-A” (yellow line), and “localization” (blue line) transitions, which, as explained below, refer to features of ISFs. Probably the most striking feature of the RMCT HS-QA kinetic diagram is the presence of a “reentrant region” for $0.17 \lesssim \phi_m \lesssim 0.24$, where a decrease of ϕ_f from >0.25 towards zero first leads from glassy to localized states and then back to glassy states.

The difference between type-A and type-B transitions is illustrated in Fig. 1.20. The figure depicts the connected ISF, $F_c(k, t)$, at state points with $\phi_m = 0.105$ in panel (a) and $\phi_m = 0.209$ in panel (b), with $k = 6.71$ being the pertinent wave

vector^{1.70}. In both panels, the ϕ_f values of the shown state points are in the vicinity of the fluid packing fraction ϕ_f^g at which RMCT predicts dynamic arrest to occur, with $\phi_f^g \simeq 0.352$ in panel (a) and $\phi_f^g \simeq 0.146$ in panel (b). The state points therefore reside along paths 1 and 2 in Fig. 1.19, and the curves are colored according to the regions in Fig. 1.19.

The relaxation patterns of $F_c(k, t)$ are obviously very dissimilar between panels (a) and (b). In panel (a), the nonarrested (blue) curves relax in *two steps*, which—as explained in Sec. 1.2.7 using the cage effect—is typical for simple glass-forming liquids. If this relaxation pattern is present in some parameter-space region, and if dynamic arrest is approached in that region, then the second (and only the second) relaxation step is delayed beyond any limit. This means that at the transition, the ISF’s long-time value $f_c(k) = \lim_{t \rightarrow \infty} F_c(k, t)$ exhibits a *jump*. Transitions of this kind are referred to as “type-B”.

In panel (b), in contrast, $F_c(k, t)$ relaxes in a *single* step. If in some parameter-space region this relaxation pattern is present, and if the dynamic arrest transition is crossed in that region, then $f_c(k)$ exceeds zero just like in type-B transitions. However, in the depicted case $f_c(k)$ varies *continuously* with the external parameters, i.e., it does *not* jump upon crossing the transition. Such transitions are referred to as “type-A”, and it is one of the aims of this work to elucidate the physical origin of transitions of this type.

Transitions of collective correlators such as $F_c(k, t)$ define the distinction between glassy or nonglassy systems. However, there also exist single-particle (“self”) quantities, which further allow to differentiate nonglassy systems into “localized” and “fluid” systems. One of these self quantities is the single-particle ISF, $F_s(k, t)$, as defined in Sec. 2.5.3.4 and as shown in Fig. 1.21. The figure differs from Fig. 1.20 in that now $k = 7$ and that the constant quantity in panel (b) is ϕ_f instead of ϕ_m , with $\phi_m = 0.1$ in panel (a) and $\phi_f = 0.15$ in panel (b). The figure contains state points with ϕ_f in the vicinities of ϕ_f^g and ϕ_f^l in panel (a), and analogously with ϕ_m in the vicinities of ϕ_m^g and ϕ_m^l in panel (b). Here, ϕ_f^l and ϕ_m^l are the fluid and matrix packing fractions at which RMCT predicts the occurrence of localization (see below). The state points thus reside along paths 1 and 3 in Fig. 1.19, and again the colors correspond to the regions in Fig. 1.19.

As can be seen from Fig. 1.21, the relaxation patterns of $F_s(k, t)$ are more complex than those of $F_c(k, t)$. Panel (a) clearly reflects the (collective) dynamic arrest at $\phi_f = \phi_f^g$ with a two-step relaxation for $\phi_f \lesssim \phi_f^g$ as in Fig. 1.20(a). However, $\phi_f < \phi_f^g$ is *not* sufficient for $f_s(k) = \lim_{t \rightarrow \infty} F_s(k, t) = 0$, which instead is the case only for $\phi_f < \phi_f^l$. Panel (b) is similar to Fig. 1.20(b) in that a two-step regime is absent, and it resembles panel (a) in that $f_s(k) = 0$ only for $\phi_m < \phi_m^l$. The transition at $\phi_m = \phi_m^g$ —i.e., the change from blue to red curves—is virtually featureless in panel (b). In the remainder of this work, we will refer to the transition at ϕ_m^l or ϕ_f^l as the “localization transition”,

^{1.70}For the RMCT transition predictions, it is irrelevant which wave vector k is considered, the reason being that according to the theory, dynamic arrest occurs simultaneously for all k .

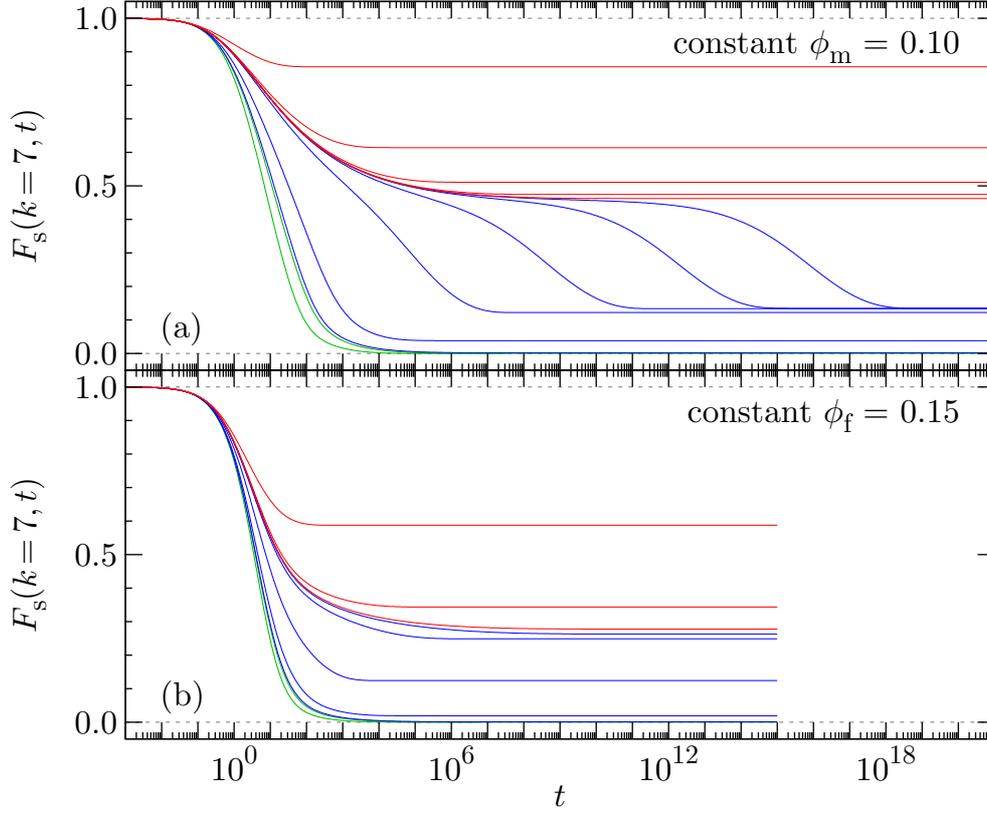


Figure 1.21: RMCT HS-QA single-particle ISF. RMCT predictions for the relaxation of the single-particle ISF in HS-QA systems. In both panels, the wave vector is $k = 7$. In panel (a), the matrix packing fraction is constant at $\phi_m = 0.1$, and the fluid packing fractions are $\phi_f \in \{0.9\phi_f^l, 0.99\phi_f^l, 1.01\phi_f^l, 1.1\phi_f^l \simeq 0.9\phi_f^g, 0.99\phi_f^g, 0.999\phi_f^g, 0.9999\phi_f^g, 0.99999\phi_f^g, 1.00001\phi_f^g, 1.0001\phi_f^g, 1.001\phi_f^g, 1.01\phi_f^g, 1.1\phi_f^g\}$. In panel (b), the constant quantity is $\phi_f = 0.15$, and $\phi_m \in \{0.9\phi_m^l, 0.99\phi_m^l, 1.01\phi_m^l, 1.1\phi_m^l, 0.9\phi_m^g, 0.99\phi_m^g, 0.999\phi_m^g, 1.001\phi_m^g, 1.01\phi_m^g, 1.1\phi_m^g\}$. Colors correspond to regions in Fig. 1.19; curves are listed bottom left to top right. (Reproduced from Ref. [162], Figs. 8 and 11.)

although according to the previous terminology it is of type A. Doing so highlights its occurrence only in self properties, and distinguishes it from type-A transitions in collective quantities.

The localization transition is also reflected in the mean squared displacement (MSD), $\delta r^2(t)$, as defined in Sec. 2.5.1.2. Figure 1.22 shows the MSD for the same constant quantities as Fig. 1.21— $\phi_m = 0.1$ in panel (a) and $\phi_f = 0.15$ in panel (b)—and likewise for state points with ϕ_f in the vicinities of ϕ_f^g and ϕ_f^l in panel (a) and ϕ_m in the vicinities of ϕ_m^g and ϕ_m^l in panel (b). As can easily be seen, in both panels the MSD's long-time value tends to infinity only in the fluid regime, while in the localized and the glassy regimes it attains a finite maximum value. This behavior is well known from simple glass formers, where the localization and the dynamic arrest transitions coincide. The separation of the latter in QA systems is reflected in panel (a), where

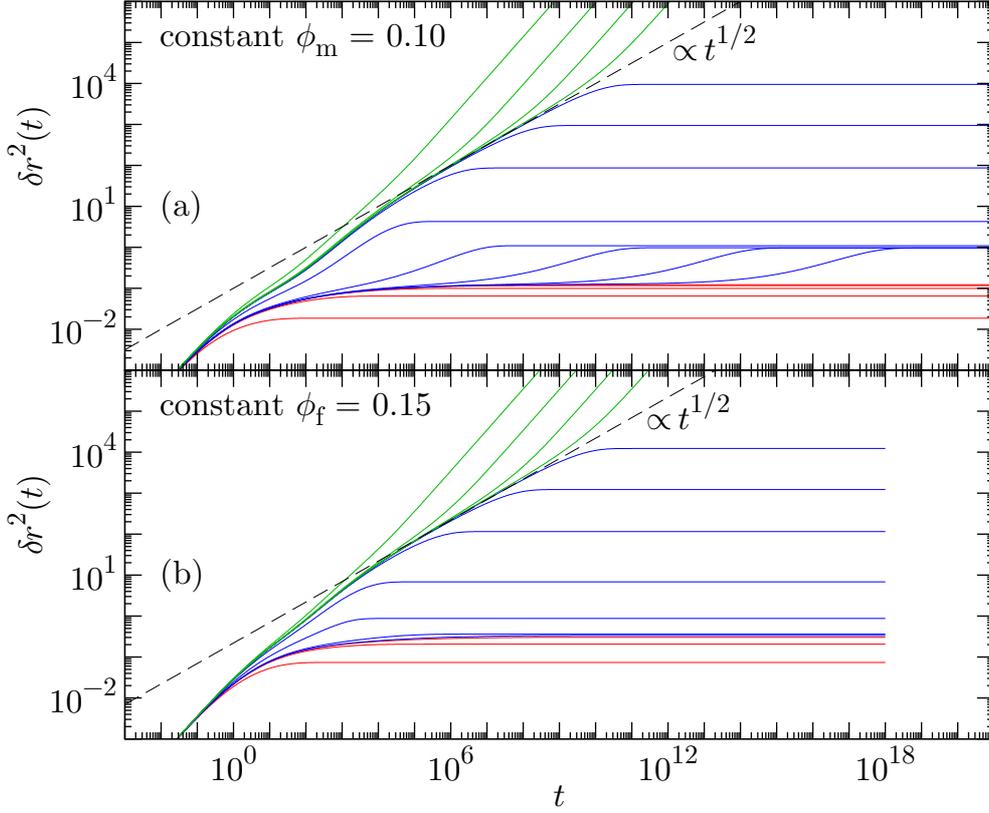


Figure 1.22: RMCT HS-QA MSD. RMCT predictions for the mean squared displacement in HS-QA systems. In panel (a), the matrix packing fraction is constant at $\phi_m = 0.1$, and the fluid packing fractions are $\phi_f \in \{0.9\phi_f^l, 0.99\phi_f^l, 0.999\phi_f^l, 0.9999\phi_f^l, 1.0001\phi_f^l, 1.001\phi_f^l, 1.01\phi_f^l, 1.1\phi_f^l \simeq 0.9\phi_f^g, 0.99\phi_f^g, 0.999\phi_f^g, 0.9999\phi_f^g, 0.99999\phi_f^g, 1.00001\phi_f^g, 1.0001\phi_f^g, 1.001\phi_f^g, 1.01\phi_f^g, 1.1\phi_f^g\}$. In panel (b), the constant quantity is $\phi_f = 0.15$, and $\phi_m \in \{0.9\phi_m^l, 0.99\phi_m^l, 0.999\phi_m^l, 0.9999\phi_m^l, 1.0001\phi_m^l, 1.001\phi_m^l, 1.01\phi_m^l, 1.1\phi_m^l, 0.9\phi_m^g, 0.99\phi_m^g, 0.999\phi_m^g, 1.001\phi_m^g, 1.01\phi_m^g, 1.1\phi_m^g\}$. Dashed lines: long-time subdiffusion at $\phi_f = \phi_f^l$ in panel (a) and at $\phi_m = \phi_m^l$ in panel (b). Colors correspond to regions in Fig. 1.19; curves are listed bottom left to top right. (Reproduced from Ref. [162], Figs. 7, 10, and 13; dashed line in panel (b) added.)

upon crossing the transition at ϕ_f^g the long-time value of $\delta r^2(t)$ features a jump.

Upon closer inspection, the MSD does not merely corroborate the findings of Fig. 1.21 but additionally evidences the presence of long-time subdiffusion (compare Sec. 1.3.2). Specifically, RMCT predicts this phenomenon to occur for all state points at which a localization transition takes place. For these cases, the precise value $z = 1/2$ is analytically inferred from RMCT^{1.71}. In Fig. 1.22, the long-time subdiffusive behavior at $\phi_f = \phi_f^l$ in panel (a) and at $\phi_m = \phi_m^l$ in panel (b) is highlighted by black dashed lines. It is one of the foremost goals of this work to verify the existence

^{1.71}Recently, the value $z = 1/2$ deduced from RMCT for the long-time subdiffusion has been contested; however, the general prediction of long-time subdiffusion has been found to hold [235].

of this phenomenon in HS-QA systems and, if present, to elucidate its underlying mechanisms.

RMCT makes several predictions in excess of those discussed above. However, in this work we did not attempt to verify those predictions since either

- (1) the computational effort would have been exceedingly large, or
- (2) we found the preconditions of the prediction to differ from reality.

Among others, point (1) ruled out the verification of glass–glass transitions and of state points with higher-order singularities, while point (2) rendered it impossible to verify the predicted dependence of $f_c(k)$ and $f_s(k)$ upon the wave vector and/or the state point in large parts of the kinetic diagram.

Chapter 2

Methods

2.1 Concepts

2.1.1 Overview

After fast electronic computing machines emerged in the middle of the past century^{2.1}, computer simulations have grown to be an invaluable tool in science and technology. Popular contemporary examples for fields in which simulations play a major role are weather forecasting^{2.2}, machine and building construction^{2.3}, cosmological modeling of the universe^{2.4}, and thermonuclear reactions (*vulgo* atomic bombs)^{2.5,2.6}. A striking common feature of this case selection is that all of the named simulations consider their respective system as a large collection of constituent units that behave and interact according to a prescribed set of relatively-simple rules. Although there are many ways to put fast computing machines to use, the repeated evaluation of simple instructions is a task at which calculation machines truly excel—and this is precisely what the quoted cases exploit.

So wherein lies the (scientific) value of computer simulations? Classically, the scheme of progress in the natural sciences is that of the mutual stimulation of theory and experiment or, put differently, the co-development of more refined observations of and control over real-world systems on the one hand, and the search for patterns in the behavior of those systems on the other. Naturally, difficulties have to be overcome to make this scheme work. Experiments need to be controlled and reproducible, and reasonable assumptions have to be made as to what circumstances are relevant to

^{2.1}http://en.wikipedia.org/wiki/History_of_computing_hardware

^{2.2}http://en.wikipedia.org/wiki/Numerical_weather_prediction

^{2.3}http://en.wikipedia.org/wiki/Computer-aided_engineering

^{2.4}http://en.wikipedia.org/wiki/Millennium_simulation

^{2.5}http://en.wikipedia.org/wiki/Los_Alamos_National_Laboratory

^{2.6}http://en.wikipedia.org/wiki/Lawrence_Livermore_National_Laboratory

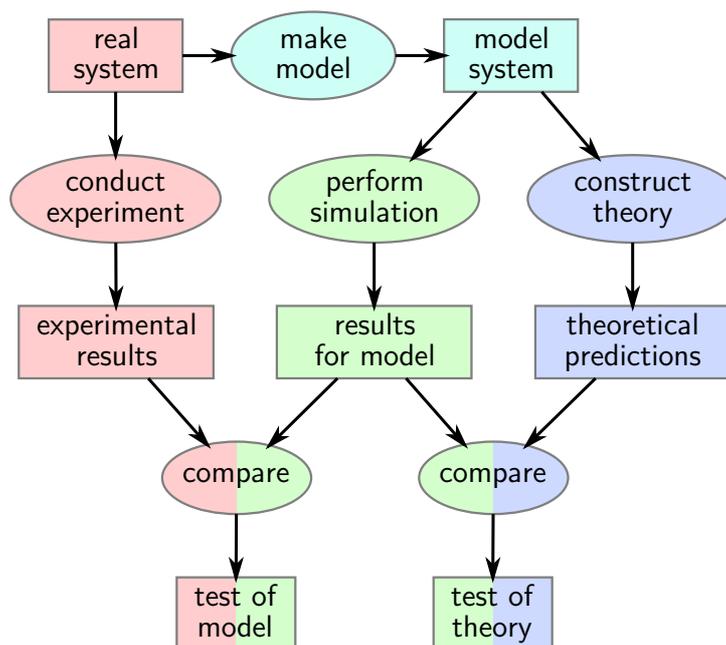


Figure 2.1: Theory, simulation, experiment. Flowchart of scientific fact finding, highlighting the significance of computer simulations (reproduced from Allen and Tildesley [9]). Red: experiment branch; turquoise: model branch, the latter splitting up into sub-branches for simulation (green) and theory (blue).

measurements. The complexity of mathematical theories can typically be controlled by the initial assumptions, yet even if the latter are simple, complex—or even unsolvable—expressions are common to arise so that approximations (which may or may not be justified) are frequently required to arrive at predictions for the behavior of a real-world system. In this age-old scheme, computer simulations assume an *intermediary* position between experiment and theory, and owing to their rapid acceptance and prowess (as, hopefully, this work demonstrates) they are nowadays even called the “third pillar of science”^{2.7}.

Let us consider the specific case of condensed-matter systems, i.e., large collections of distinct micro- or mesoscopic particles that form a material. In this case, a theory may for instance make assumptions about the properties of a set of particles (such as their composition, their arrangement, or their interaction) to predict the characteristics of a material is constituted by them. The theoretical predictions may then be compared with experimental observations conducted on the same material. A straightforward application of computer simulations in this case is to *simulate* this material using as input exactly the assumptions about the particle properties that were made to derive the theoretical predictions. This information is then processed according to some algorithm (such as the one detailed in Sec. 2.2) and yields a set of results. As sketched in Fig. 2.1, these computer simulation results then provide the

^{2.7}<http://www.grs-sim.de/education/master>

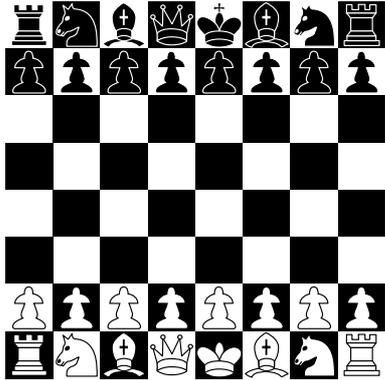


Figure 2.2: Chess. The conventional set-up of chess, with 32 pieces on a 64-field checkerboard. For the simple consideration in Sec. 2.1.1, all pieces are regarded to be equal, and likewise are all fields.

distinct possibility to carry out *two* comparisons of outcomes—as opposed to merely one in the classical experiment/theory scheme. Firstly, it is possible to compare simulation results with the observed physical behavior of a system, which allows to assess whether the properties of the model particles are well chosen. Secondly, if theoretical evaluations require approximations, the results from computer simulations for a given model choice allow to judge the appropriateness of those approximations.

Of course, the two new comparisons are prudent only if the results of computer simulations are generally more similar to experimental observations than theoretical predictions are. This is by no means trivially fulfilled, with the main problem being as simple as severe: limited computing power. A slightly unrelated problem, which is nevertheless seriously considered by scholars (see, e.g., Ref. [165]), may clarify just *how* limited even the most powerful present computers are. Consider a system in which 32 “particles” of differing types move according to various convoluted rules on a two-dimensional square with 8×8 lattice sites—i.e., consider the game of *chess*. If we want to fully characterize this “system” without a-priori knowledge about the rules, then we need to find every permitted constellation of pieces along with its probability to occur. If for a moment we make the (significant) simplifications that all pieces are indistinguishable, that every piece may occupy every field, and that capture is impossible, then we arrive at the astronomically large number

$$\frac{64!}{(64-32)!} \simeq 4.8 \times 10^{53} \quad (2.1)$$

for the number of possible arrangements^{2.8}. For comparison, the fastest supercomputers currently execute $\sim 10^{15}$ low-level operations per second^{2.9}, i.e., even if it took

^{2.8}If all chess rules were properly accounted for, then the quoted number of configurations would have to be enlarged by a factor $12! \simeq 5 \times 10^6$ due to the 12 distinct types of chess figures, but also reduced by a factor that accounts for the fact that some pieces can never occupy certain fields (which is difficult to quantify). To account for capture, the procedure would then have to be repeated for less than 32 pieces and summed over the respective outcomes. Since in condensed-matter systems particles are generally indistinguishable and subject to comparatively simple rules, the number of configurations quoted in the text is more meaningful to the present context.

^{2.9}<http://www.top500.org/list/2012/11>

only one such operation to construct one arrangement^{2,10}, it would take 10^{31} years (!) to compute them all even on a supercomputer. Considering that physical systems typically consist of *much* more than 32 particles which are moreover not constrained to a lattice, it is clear that computer simulations in condensed matter physics *cannot* operate in “brute force” mode, i.e., attempt to find each and every configuration of a system in order to describe it.

2.1.2 Statistical mechanics

Fortunately, there exists a well-established theoretical framework that can be used—amongst others—to approach this unfortunate deficiency of condensed-matter systems: *statistical mechanics* (see, e.g., Refs. [108, 238, 258]). The fundamental concept of this theory is to link a macroscopic descriptor of a system to its detailed microscopic representations (“microstates”). Obviously, statistical mechanics is one of the theories alluded to in the previous subsection that predict a system’s macroscopic behavior based on a set of underlying assumptions. However, due to its mathematical structure it is not only suited to make predictions, but also to develop approximations in a meaningful way and to assess their accuracy.

Let us go about this a bit more formally. The central quantity in statistical mechanics is the so-called “partition function”; its German name *Zustandssumme* (“state sum”) reflects the fact that it represents a sum over all possible states of the system. However, it is not a simple sum but a *weighted* one in which each state is represented by its probability to occur. In the chess example, if we count every arrangement with unity weight, then the partition function is nothing but the figure $64!/32!$ that we already quoted. In the general case we have N particles instead of 32, and one state of the system is characterized by the set of the (continuous) coordinates $\mathbf{r}^N = (\mathbf{r}_1, \dots, \mathbf{r}_N)$ and the momenta $\mathbf{p}^N = (\mathbf{p}_1, \dots, \mathbf{p}_N)$, and possibly by other properties of the particles such as their magnetic moment or their orientation. A general expression for the partition function is then given by

$$\Omega_{\mathbf{Q}}(t) = h^{-3N} \int_{\{\Gamma\}} w_{\mathbf{Q}}(\Gamma; t) d\Gamma . \quad (2.2)$$

Here, the function $w_{\mathbf{Q}}(\Gamma; t)$ denotes the weight of a microstate $\Gamma = (\mathbf{r}^N, \mathbf{p}^N)$, i.e., the relative probability that a certain combination of \mathbf{r}^N and \mathbf{p}^N occurs at a particular time t under the premise that the external conditions $\mathbf{Q} = \{Q_1 = \text{const}, \dots\}$ are fixed (see further below). The integration is carried out over the entire range of possible Γ , the so-called “phase space” $\{\Gamma\}$. The factor h^{-3N} only occurs in the integral form of the partition function and serves to render $\Omega_{\mathbf{Q}}$ dimensionless, with the actual value of h being insignificant since the weights are only determined up to a common constant

^{2,10}In reality, it is much more likely that finding one particular arrangement would take several dozen operations *per piece*, that is, perhaps 1000 operations per arrangement—excluding the computational effort required to determine the probability of that arrangement.

factor^{2.11}. If the weight function (and thus the partition function) does not explicitly depend upon time, then by definition the respective system is in *thermal equilibrium* [115]; throughout this work we will assume this condition to hold. The delicate issue of *establishing* thermal equilibrium will be touched upon in Sec. 2.5.5.

Before we discuss the external conditions \mathbf{Q} and the associated weight $w_{\mathbf{Q}}$ in more detail, we should wonder why the rather strange object of the partition function $\Omega_{\mathbf{Q}}$ is so important. The answer is: because it is the very tool that connects the microscopic with the macroscopic description of the system. Suppose we want to measure some physical property A of the system (A should of course not be one of the external conditions \mathbf{Q} since those are imposed and constant). Examples for A are the density, the pressure, the heat capacity, the viscosity, or the structure factor of a system. Obviously, each microstate $\mathbf{\Gamma}$ of the system may have its own value of A —in other words $A = A(\mathbf{\Gamma})$. The average value of A observed in a measurement therefore depends upon which $\mathbf{\Gamma}$'s the system assumes during the time of measurement, which in turn is determined by the probability of the $\mathbf{\Gamma}$'s to occur. If we want to evaluate the “true” average value of A under the conditions \mathbf{Q} , then we need to consider *all* microstates $\mathbf{\Gamma}$, i.e., the entire phase space $\{\mathbf{\Gamma}\}$. This is formally encoded by the expression

$$\langle A \rangle_{\mathbf{Q}}^{\text{ps}} = h^{-3N} \int_{\{\mathbf{\Gamma}\}} A(\mathbf{\Gamma}) \frac{w_{\mathbf{Q}}(\mathbf{\Gamma})}{h^{-3N} \int_{\{\mathbf{\Gamma}\}} w_{\mathbf{Q}}(\mathbf{\Gamma}') d\mathbf{\Gamma}'} d\mathbf{\Gamma} , \quad (2.3)$$

where the superscript “ps” serves to remind about the phase-space integration. The term in the denominator of the fraction in Eq. (2.3) purports an important physical purpose: since the system has to occupy *some* microstate, the sum of the probabilities has to be unity. A quick check against Eq. (2.2) reveals that this denominator is nothing but the partition function $\Omega_{\mathbf{Q}}$. We can therefore introduce the so-called “phase-space density”

$$f_{\mathbf{Q}}(\mathbf{\Gamma}) = \frac{w_{\mathbf{Q}}(\mathbf{\Gamma})}{\Omega_{\mathbf{Q}}} \quad (2.4)$$

to rewrite the observed average of the quantity A as

$$\langle A \rangle_{\mathbf{Q}}^{\text{ps}} = h^{-3N} \int_{\{\mathbf{\Gamma}\}} A(\mathbf{\Gamma}) f_{\mathbf{Q}}(\mathbf{\Gamma}) d\mathbf{\Gamma} . \quad (2.5)$$

This formula represents the core of every application of computer simulations in the context of statistical mechanics. However, unfortunately, Eq. (2.5) alone does not yet relieve the investigator from the task of calculating every possible state of a system to evaluate an observable since all integrals involved run over the entire range of their arguments.

Before we address this problem in the next subsection, one issue remains to be discussed: the fixed external conditions \mathbf{Q} and the associated weight $w_{\mathbf{Q}}(\mathbf{\Gamma})$. There

^{2.11}It is only in quantum statistics that h attains an actual meaning, being then interpreted as Planck’s constant.

are many properties of a system that one may decide to keep constant, for instance the number of particles N , the volume V , the pressure P , the total internal energy E , the temperature T , or the chemical potential μ , just to name a few. Each choice of \mathbf{Q} determines a so-called “ensemble” and entails a particular functional form of $w_{\mathbf{Q}}(\Gamma)$. As will become clear in Sec. 2.2, for the investigations carried out in this work the natural choice is $\mathbf{Q} = \{N=\text{const}, V=\text{const}, E=\text{const}\}$, or short $\mathbf{Q} = NVE$. This corresponds to a system that is completely isolated from its surroundings (no transfer of particles, volume, or energy into or out of the system) and is commonly termed the “microcanonical ensemble”. The *fundamental postulate of statistical mechanics* states that in such isolated systems, every microstate is equally likely [108], i.e., w_{NVE} is a constant^{2.12}. For our purpose this means that evaluating the partition function formally reduces to “counting” the microstates (similar to the chess example), which, however, remains a highly-nontrivial task.

2.1.3 Molecular dynamics

So how does statistical mechanics help us eventually? As mentioned before, the framework allows to work out meaningful approximations. One concept in this direction is to consider real-world systems, where the constituent particles are subject to physical laws. As a consequence of these laws, as time progresses a system will evolve in a sequence of closely-related microstates. The careful reader will notice that the concept of the *time evolution* of a system has not played a role so far. On the contrary: in Sec. 2.1.2 we explicitly disregarded all dependence of our ensembles upon time (and thereby assumed our systems to be in thermal equilibrium). However, in reality time does play a major role. For instance, one could imagine that given an infinitely-long time span, a system would explore its entire phase space, i.e., realize all possible microstates at least once. In fact, always bearing in mind that the phase space of a system is dependent upon the external conditions \mathbf{Q} , already Ludwig Boltzmann postulated that the time average

$$\langle A \rangle_{\mathbf{Q}}^{\text{time}} = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^{\tau} A(\Gamma(t)) dt , \quad (2.6)$$

where now $\Gamma(t) = (\mathbf{r}^N(t), \mathbf{p}^N(t))$, is equivalent to the ensemble average as defined in Eq. (2.5), i.e.,

$$\langle A \rangle_{\mathbf{Q}}^{\text{time}} = \langle A \rangle_{\mathbf{Q}}^{\text{ps}} . \quad (2.7)$$

^{2.12}Notably, if the system of interest consists of *hard spheres* (cf. Sec. 1.1.3), then the microcanonical ensemble is *equivalent* to the “canonical” ensemble (see below). This is due to the “athermal” nature of hard spheres (cf. Sec. 1.1.3.2), the effect of which in the present context is the following. In the canonical ensemble, $\mathbf{Q} = NVT$ and $w_{NVT} = \exp\{-\beta \mathcal{H}(\mathbf{r}^N, \mathbf{p}^N)\}$, where $\mathcal{H}(\mathbf{r}^N, \mathbf{p}^N) = \sum_{i=1}^N \mathbf{p}_i^2 / (2m_i) + U(\mathbf{r}^N)$ is the Hamiltonian of the system, m_i is the mass of particle i , the potential energy is represented by $U(\mathbf{r}^N)$, and $\beta = 1/(k_B T)$ with k_B being Boltzmann’s constant. Since in hard-sphere systems it is always $U(\mathbf{r}^N) = 0$, and the total momentum of the particles $\sum_{i=1}^N \mathbf{p}_i^2 / (2m_i)$ is conserved, the Hamiltonian reduces to a constant. Resulting from this, w_{NVT} is simply constant for systems of hard spheres—just as is the case for w_{NVE} regardless of the system.

The statement expressed in Eq. (2.7) is the (in)famous “ergodic hypothesis”, which, despite all controversy, will be assumed to hold in the remainder of this work^{2,13}.

This reinterpretation of the averaging of an observable can be exploited, namely by noticing that in the real world, any measurement time τ_{obs} is finite. It has been established in a plethora of instances that reliable and reproducible measurements can be obtained in experiments despite finite observation times. This implies that the limit $\tau_{\text{obs}} \rightarrow \infty$ is not compulsory, and that neither is considering the entire phase space of a system. Of course, the truncation of the observation time comes at the cost of a systematic measurement uncertainty since part of the phase space has been neglected. The immediate question naturally is: if a finite τ_{obs} *can* suffice, how long *is* “sufficient”? Unfortunately there is no general answer to this question. However, it is common practice to conduct repeated measurements of a certain length τ_{obs} (each of which will yield a different value), and to evaluate statistical uncertainties for their outcomes, which are then required to remain below a predefined value (e.g., 10% of the mean of the measured values). A more succinct answer to how long τ_{obs} has to be chosen involves the study of so-called time correlation functions (discussed in more detail in Sec. 2.5), which usually show features on various time scales: if all important time correlators have been considered, then the largest of their time scales can be interpreted as a lower bound for the observation time.

In the same context, another important point calls for care by the investigator. In any measurement—be it in a simulation or in an experiment—, the sample has to be prepared in some way, i.e., it has to be assigned an initial microstate. (The delicate issue of finding such initial states for quenched-annealed systems is the subject of Sec. 2.3.) Unfortunately, the value of the observable of interest A in this initial state may be far from its ensemble average. On the other hand, it can be shown that a reasonably large system in thermal equilibrium (and far away from a phase transition, but this is a different topic) is subject to only little fluctuations [108], i.e., that almost all microstates occurring under these conditions yield values of A close to $\langle A \rangle_{\mathbf{Q}}^{\text{ps}}$. Therefore, if the ergodic hypothesis holds, then the system must sooner or later enter a region of phase space in which “equilibrium microstates” are abundant. The time τ_{eqlib} a system requires to arrive at this phase-space region is called the “equilibration time”, which clearly depends on the initial microstate. Since the microstates that the system assumes in the physical evolution subsequent to its preparation usually have a value of A similar to the initial state, it is generally advisable to let a time $t > \tau_{\text{eqlib}}$ elapse before commencing a measurement. As mentioned before, for the systems investigated in this work, the procedures and issues connected to establishing thermal equilibrium are discussed in Sec. 2.5.5.

^{2,13}Unfortunately, as the name indicates, the ergodic hypothesis has remained a hypothesis to the present day. A proof of it would provide the long-sought classical-mechanical basis to statistical physics, which is in fact based solely on the mathematics of probability. Despite this radiant appeal, the problem has turned out too adamant even for minds as brilliant as Henri Poincaré, Paul Ehrenfest, Enrico Fermi, John von Neumann, and Boltzmann himself. Nevertheless, the named personages and many more after them were able to demonstrate that in slightly modified versions or under additional conditions, the hypothesis indeed holds in numerous cases [258].

Summarizing this ansatz, one approach for computer simulations in condensed-matter physics is to evolve a digital representation of a system according to a set of physically meaningful laws. Indeed, algorithms of this kind have found widespread application and are commonly termed “molecular-dynamics” (MD) simulations [6, 9, 85, 216, 219]. The fact that it imitates a system’s physical development renders the MD concept particularly appealing since it allows for the distinct possibility to determine time-dependent properties of a system. In fact, it is owed to this capability of MD simulations that major parts of this investigation are based on this technique. The particular incarnation of MD underlying this work (which is a relatively rarely used one) shall be discussed Sec. 2.2, where details on its algorithmic implementation as well as on its physical justification will be given.

2.1.4 Monte Carlo

In addition to MD, there exists another very successful approach for computer simulations in condensed-matter physics: the “Monte-Carlo” (MC) technique. In the present work, a MC-type method is used for only one specific purpose, namely for generating initial configurations of the systems of interest. Since a very simple version of MC suffices to fulfill this task, and since no calculation of observables is required in this context, the present section shall merely outline some general ideas that provide the basis of MC. The description of the actual algorithm as well as of its application are then detailed in Sec. 2.3.

Essentially, the term “MC technique” describes the concept of using mathematical methods—instead of physical insight—to obtain an (approximate) value for the central quantity of statistical mechanics: the partition function $\Omega_{\mathbf{Q}}$. The most basic of the mathematical considerations involved is the following: suppose we have a one-dimensional function $f(x)$ that is defined on some interval $[x_a, x_b]$. Then, even if it is impossible to find an analytic solution to the integral $\int_{x_a}^{x_b} f(x) dx$, it is in principle possible to approximate its value—for instance by calculating $f(x)$ at K regularly-spaced points $\{x_i\} = \{x_1, \dots, x_K\}$ separated by a distance $\Delta x = (x_b - x_a)/(K - 1)$ and by then summing over these values, i.e.,

$$\int_{x_a}^{x_b} f(x) dx \approx S_{\text{reg}} = \Delta x \sum_{i=1}^K f(x_i), \quad (2.8)$$

where the subscript “reg” indicates that the $\{x_i\}$ are regularly spaced. In fact, in the limit $\Delta x \rightarrow 0$ this expression provides the conventional definition of the Riemann integral^{2.14}. There are two main obstacles that thwart the use of Eq. (2.8) to evaluate the partition function $\Omega_{\mathbf{Q}}$ with its integrand $w_{\mathbf{Q}}(\mathbf{\Gamma})$. Firstly, $\mathbf{\Gamma} = (\mathbf{r}^N, \mathbf{p}^N)$ lives in a $6N$ -dimensional space, meaning that the number of interpolation points in this case is as large as K^{6N} —as opposed to only K in the one-dimensional case. Secondly, the functional form of $w_{\mathbf{Q}}(\mathbf{\Gamma})$ is typically such that K needs to be relatively large to obtain

^{2.14}http://en.wikipedia.org/wiki/Riemann_integral

an acceptable accuracy in S_{reg} . As an example, if we consider merely 10 particles and for their loci and momenta we allow only *two* interpolation points per dimension (a basically useless characterization), this results in the astonishing figure $2^{60} \simeq 10^{18}$ for the number of interpolation points^{2.15}.

In the 1940s, von Neumann and Ulam realized that any Riemann-type integral can be approximated to any desired accuracy by means of stochastic methods [9]. While at first this seems to be a needless complication, this finding provides the foundation to the MC-simulation method. In Eq. (2.8), for $K \gg 1$ the sum S_{reg} represents nothing but the average of $f(x)$ over the interval $[x_a, x_b]$, multiplied by $(x_b - x_a)$. Now who is to say that for determining some average, the points $\{x_i\}$ have to be picked on a regular lattice? The mathematics of probability states that an average can just as well be determined from the values of $f(x)$ at *random* positions $\{x_j\} = \{x_1, \dots, x_M\}$. Due to its probabilistic nature, this scheme, expressed by

$$\int_{x_a}^{x_b} f(x) dx \approx S_{\text{uni}} = \frac{(x_b - x_a)}{M} \sum_{j=1}^M f(x_j), \quad (2.9)$$

is called ‘‘Monte Carlo integration’’. The index ‘‘uni’’ signals that for this equation to hold, the $\{x_j\}$ have to be sampled *uniformly* from $[x_a, x_b]$, i.e., every value in that interval has to be equally likely to be selected. In fact, Eq. (2.9) is very similar to Eq. (2.8), so what do we gain by using it? One can show that the uncertainty in S_{uni} scales proportionally to $1/\sqrt{M}$, which means that approximating an integral can be commenced with a small number of evaluations M , and that S_{uni} can subsequently be refined continuously by considering more $f(x_j)$. However, just as in the deterministic scheme (where S_{reg} is meaningless if the grid is too coarse), the number of addends M in S_{uni} is subject to a lower bound: it must be chosen such that the integrand does not contain maxima narrower than $(x_b - x_a)/M$, so that each maximum is likely to be sampled. Unfortunately, since $w_{\mathbf{Q}}(\mathbf{\Gamma})$ is often highly nonuniform, this condition requires very large M to reliably estimate $\Omega_{\mathbf{Q}}$, which means that for this purpose the simple MC-integration scheme is almost as useless as deterministic integration.

However, MC integration provides the basis for a more sophisticated scheme pioneered by Metropolis and coworkers [190]. The first fundamental idea of these authors was to employ a *nonuniform* probability $p(x)$ for selecting a particular x_j , and to modify the sum S_{uni} in Eq. (2.9) such that in the limit $M \rightarrow \infty$ it remains invariant despite the different sampling of the $\{x_j\}$. This can be achieved by dividing each addend of S_{uni} by $p(x)$, i.e.,

$$\int_{x_a}^{x_b} f(x) dx \approx S_{\text{imp}} = \frac{(x_b - x_a)}{M} \sum_{j=1}^M \frac{f(x_j)}{p(x_j)}. \quad (2.10)$$

^{2.15}In fact, the simple ‘‘rectangular’’ discretization scheme in Eq. (2.8) effectively converts the continuous problem to a lattice problem of the chess type, for which we have already demonstrated in Sec. 2.1.1 that a brute-force approach is futile. However, the current approach additionally evidences that more sophisticated deterministic approximation schemes such as the trapezoidal scheme or Simpson’s rule are also bound to fail since in any case the number of interpolation points tends to be astronomically large due to the high-dimensional nature of the integration space.

If $p(x)$ is suitably chosen, then the error of S_{imp} may decrease *much* faster with increasing M than does the error in S_{uni} . Obviously, the x_j that are favorably selected by such a $p(x)$ are more significant, which is why this method is also called “importance sampling”—hence the index “imp”. If for every addend we have to generate a new x_j without any additional information, then the probability $p(x)$ has to be *normalized* since for each addend we have to select some x_j . In this situation, we cannot use $p(x) = f(x)$ for the selection function (which would seem a promising choice) since the normalization for this function would be precisely the integral that we set out to determine in the first place.

Here, the second fundamental idea of Metropolis and his collaborators comes into play, which is to avoid the necessity of normalizing $p(x)$ altogether by only evaluating *relative* probabilities. This can in principle be achieved by generating a so-called “Markov chain” [9], which is essentially a state automaton with a (possibly infinite) number of states, as well as with probabilities to change from one state to another that depend only on a finite number of previous states. Since there are infinitely-many possibilities to prescribe a Markov chain, it is nontrivial to choose a set of rules for which the $\{x_j\}$ are selected according to $p(x)$ as $M \rightarrow \infty$. One prescription that suffices this criterion^{2.16} was originally proposed by Metropolis et al. [190]: Suppose the automaton is in state x_j . To go to state x_{j+1} we first choose some $x_k \in [x_a, x_b]$ at random. If $p(x_k) > p(x_j)$, then we set $x_{j+1} = x_k$ immediately. If on the other hand $p(x_k) < p(x_j)$, then we first draw a random number $R \in [0, 1]$; if $R < p(x_k)/p(x_j)$, then we set $x_{j+1} = x_k$ as above, and $x_{j+1} = x_j$ otherwise. (Notably, the latter does not mean *rejecting* the trial but rather that the new state of the chain is equal to the old one.) In addition to evaluating only relative probabilities, this algorithm requires to know at any time merely two states of the system: the current and the proposed one.

The final consideration is concerned with the actual values of the probability ratios. It is of paramount importance that there occur sufficiently many trials in which $p(x_k)/p(x_j)$ is below unity but large enough to allow the system to change to the dispreferred state x_k . This is necessary for the MC algorithm to surmount barriers that separate regions of preferential states, which is an obvious requirement for the algorithm to explore the entire range $[x_a, x_b]$. The frequent occurrence of probability ratios below but close to unity can be achieved by selecting in each step a trial x_k that is in a relatively-small vicinity δx of the current x_j . However, this vicinity should not be chosen too small since it takes at least $(x_b - x_a)/\delta x$ trials to explore the entire interval. In the high-dimensional integration spaces of condensed-matter systems this requirement is not easily fulfilled and requires adaptive techniques as well as some

^{2.16}While it is reasonably simple to prove that the described state-change prescription fulfills the Markov-chain criteria, the proof is slightly lengthy [9, 190]. It is based on solving the eigenwert problem of the stochastic matrix that describes all possible state transitions. Since the criteria defined by this problem are insufficient to devise a simulation algorithm, Metropolis and his coworkers introduced the additional (actually overly strict) criterion of “detailed balance”, which constitutes their third fundamental idea. On this basis, the same authors succeeded in deriving the so-called “asymmetric solution” described in the text.

(limited) insight into the physical mechanisms that are at work in the investigated system.

Since the MC method does not require a system to evolve physically correctly, it allows for an abundance of optimizations. For this reason (and of course because it is very efficient), the MC technique is widely used in the scientific community. The method also constitutes an ideal starting point to tailor custom algorithms for unusual purposes. In the present work, an MC-type algorithm was employed to generate initial conditions for fluids in porous media; the custom scheme developed to suit this objective is described in Sec. 2.3.

2.2 Event-driven molecular dynamics

2.2.1 Physical principle

2.2.1.1 Overview

A popular exercise in elementary college physics textbooks reads as follows:

Two marbles roll in a straight horizontal groove such that one hits the other. The marbles have the same diameter, their masses are m_1 and m_2 , and their velocities before the collision are v_1 and v_2 . The collision is fully elastic, and friction and rotation do not play a role. What are the velocities w_1 and w_2 of the marbles after the collision?

In such an exercise, the student is supposed to realize the significance of conservation laws (namely those of momentum and energy) which have to be employed to obtain the solution. Of course, the conditions described in this exercise are somewhat artificial: nothing is known about the actual collision (it is just assumed that “before” and “after” the collision are well defined), and in reality there will always be some energy dissipation. Also, the objects move along a groove only for the purpose of rendering the problem one-dimensional and therefore free of vectorial quantities. Nevertheless, the three-dimensional analog of such collisions represents a reasonable approximation for a variety of interactions between macroscopic objects, examples being tennis balls, curling rocks, and bumper cars.

An important motivation of the present work is the fact that collision-like interactions can take place also among mesoscopic colloidal particles, especially if they are *designed* to have this property (see Sec. 1.1). The major part of this work is devoted to investigating large collections of such particles, which in the framework of statistical physics are commonly called “hard spheres” or “billiards”. As described in more detail in Sec. 1.1.3, aggregates of hard spheres are widely considered (proto)typical condensed-matter systems due to their resembling of many key properties of other materials [115]. In a nutshell, this feature is owed to the fact that hard spheres embody the essence of excluded-volume interactions.

In 1959, Alder and Wainwright took up the concept of collisions to devise a computer-simulation method for finding the *exact* time evolution of a collection of ideal hard spheres [5, 6]. In fact, their scheme constitutes the very first molecular-dynamics (MD) algorithm in history, and has remained popular to the present day. The central ingredient of this scheme are particles that never overlap and that move at constant velocity along straight lines except for infinitesimally short times during which momentum and energy are mediated via “impulse interactions”. The physics of these interactions—a.k.a. collisions—can be derived from the additional conditions that the system is invariant under time shifts, translations, and rotations (i.e., that energy, momentum, and angular momentum are conserved), and that no external forces or torques are exerted on any part of the system. The particle trajectories resulting from all of these assumptions are obtained by iterating the following scheme:

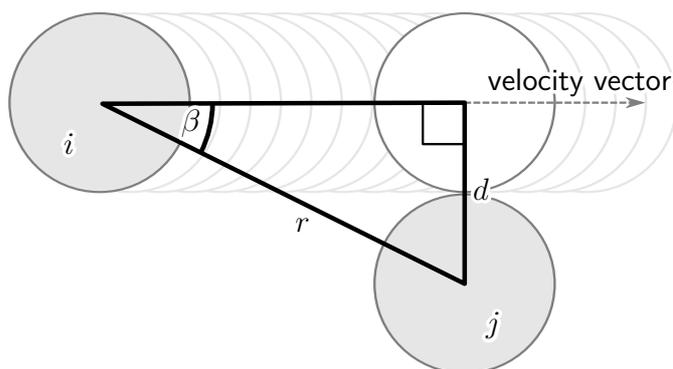
- (I) find the time of the next collision between any two particles in the system,
- (II) propagate all particles to that time along straight lines according to their current velocity vectors, and then
- (III) change the velocities of the colliding particles in a way that conserves energy, momentum, and angular momentum.

The procedure therefore does *not* involve the notion of a fixed time step (as do other MD schemes [9, 85, 216, 219]) but rather evolves a system in a sequence of discrete physical events—hence the name “event-driven molecular dynamics”. As shall be demonstrated shortly, all of the above points are reasonably simple from the mathematical point of view. The nontrivial task is to *optimize* this scheme such that the event computing rate is sufficient. This is particularly important if a system consists of a large number of particles, if collisions are frequent due to a short mean free path, or if the slow nature of the system’s dynamic features requires the simulated time to be long. Since the systems discussed in this work are subject to at least the latter two conditions, we implemented a considerable body of optimizations. The latter will be described in detail in Section 2.2.2, after introducing the physical basis to points (I)–(III) in the above scheme.

2.2.1.2 Finding the next collision

Finding the time at which the next collision event will take place is the most difficult task in the three-step algorithm outlined in Sec. 2.2.1.1. The problem actually consists the following sub-tasks:

- (1) Define the set $\{P_{ij}\}$ of pairs of particles that *might* collide.
- (2) For each particle pair in $\{P_{ij}\}$, check *if* the particles will collide.
- (3) For each remaining pair, determine *when* the particles will collide.
- (4) From the times resulting from step (3), find the *smallest* one.

**Figure 2.3: Will a collision occur?**

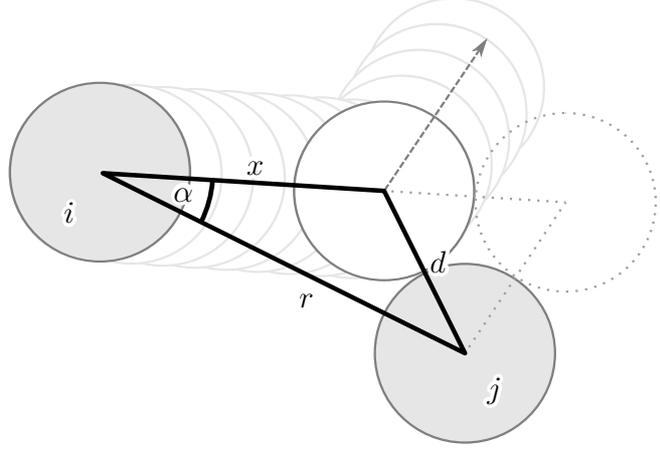
Sketch of two identical particles of diameter d separated by a distance r . Particle j is fixed at its position, and particle i moves such that at closest approach it touches particle j . For a collision to occur, the angle enclosed by the vector connecting the centers of i and j and the velocity vector of particle i must be smaller than β .

In each iteration of steps (2) and (3), exactly two particles are considered and all other particles are neglected. This is a valid approach since there exists a definite “next event” that depends only upon the current properties of exactly two particles. The sole precondition for finding this event is that in step (1) the corresponding pair of particles be included in $\{P_{ij}\}$. Unless there is further insight (see Sec. 2.2.2), performing the next event then renders all other events invalid since the physical conditions from which they were calculated have changed. In this section, we will focus on steps (2) and (3) since they contain the major part of the physical and geometric considerations. Steps (1) and (4) will be discussed in detail in Sec. 2.2.2 since they offer themselves to numerous optimizations. For step (1), the reader may temporarily assume the simple (yet inefficient) approach in which the set $\{P_{ij}\}$ comprises *all* possible pairs of particles [i.e., $N(N-1)/2$ elements] and thereby trivially includes the next event. Step (4) is then accomplished equally simply by updating scalar variables for the time and the collision partners each time a collision time is computed.

For step (2), consider two particles i and j . Whether or not a collision will take place between i and j depends on merely six quantities: the current loci of the two particles, \mathbf{r}_i and \mathbf{r}_j , their current velocities \mathbf{v}_i and \mathbf{v}_j , and their diameters d_i and d_j . For notational convenience, we have suppressed all time dependencies since here we focus on a fixed instant of time. The problem is invariant under Galilean transformations, which means that the relevant quantities reduce to the relative location of the particles $\mathbf{r} = \mathbf{r}_i - \mathbf{r}_j$, their relative velocity $\mathbf{v} = \mathbf{v}_i - \mathbf{v}_j$, and their diameters d_i and d_j . We may for instance consider the frame of reference in which particle j is fixed at the origin and particle i is located at \mathbf{r} and moves with velocity \mathbf{v} ; this situation is depicted in Fig. 2.3.

Suppose that particle i would *touch* (but not interact with) particle j in the fly-by depicted in Fig. 2.3. In this case, the particle’s velocity vector and the vector connecting the particle centers at closest approach would be orthogonal, and the length of the latter vector would be $d = (d_i + d_j)/2$. The angle β in Fig. 2.3 therefore serves as a threshold for the angle α enclosed between \mathbf{r} and \mathbf{v} , with a collision occurring only if $\alpha < \beta$. This consideration can be expressed using planar trigonometry since the problem is of cylindrical symmetry about the vector connecting the particle centers. In terms of the known quantities (and using the notation $r = |\mathbf{r}|$ and $v = |\mathbf{v}|$), the

Figure 2.4: When will the collision occur? Sketch of two identical particles of diameter d separated by a distance r . Particle j is fixed at its position, and particle i moves along a path tilted by an angle $\alpha < \beta$ relative to the vector connecting the particle centers (cf. Fig. 2.3). The path is such that a collision between i and j occurs, with the time of the collision being determined by the length x and the velocity of particle i .



condition for a collision to take place can hence be expressed as

$$\begin{aligned}
 \alpha &< \beta \\
 \cos \alpha &> \cos \beta \\
 \cos \alpha &> r^{-1} \sqrt{r^2 - d^2} \\
 r^2 v^2 \cos^2 \alpha &> v^2 (r^2 - d^2) \\
 a^2 &> b \quad ,
 \end{aligned} \tag{2.11}$$

where simply the definition of the cosine and Pythagoras's theorem were used. In the last step, the shortcuts

$$\begin{aligned}
 a &= r v \cos \alpha = \mathbf{r} \cdot \mathbf{v} \quad \text{and} \\
 b &= v^2 (r^2 - d^2)
 \end{aligned} \tag{2.12}$$

have been introduced for the reason that in a computer program the values associated with a and b can be stored and reused, the benefit of which will become clear shortly.

The collision time t_{coll} can be evaluated from an equivalent trigonometric consideration. In this case, the primary object of interest is not the threshold angle (as above) but rather the distance that particle i has to travel from its current position to the position of closest approach. In Fig. 2.4 this distance is labeled with x , while the quantities r , d , and $\alpha = \sphericalangle(\mathbf{r}, \mathbf{v})$ are the same as above and therefore known. Using the cosine rule, the distance d can be expressed as

$$d^2 = r^2 + x^2 - 2r x \cos \alpha \quad , \tag{2.13}$$

and employing the quadratic formula, the solution of this equation for x is

$$x_{1,2} = \frac{1}{2} \left\{ 2r \cos \alpha \pm \sqrt{4r^2 \cos^2 \alpha - 4(r^2 - d^2)} \right\} . \tag{2.14}$$

The “+” solution of this expression corresponds to the dotted circle in Fig. 2.4, which is clearly unphysical since particle i would have to pass through particle j to reach

this position. We therefore get

$$\begin{aligned}
 x &= r \cos \alpha - \sqrt{r^2 \cos^2 \alpha - (r^2 - d^2)} \\
 &= \frac{1}{v} \left\{ v r \cos \alpha - \sqrt{v^2 r^2 \cos^2 \alpha - v^2 (r^2 - d^2)} \right\} \\
 &= \frac{1}{v} (a - \sqrt{a^2 - b})
 \end{aligned} \tag{2.15}$$

where in the last step we used a and b as introduced in Eq. (2.12). Dividing x by v , we finally get the time of collision t_{coll} as

$$t_{\text{coll}} = \frac{1}{v^2} (a - \sqrt{a^2 - b}) + t_{\text{curr}}, \tag{2.16}$$

where t_{curr} is the current time of the system. It is readily verified that in accordance with Eq. (2.11), the resultant t_{coll} is a real number only if $a^2 > b$.

2.2.1.3 Propagating the particles

Moving the particles from their current location to their location at the time of the collision is trivially performed by evaluating the simple equation

$$\mathbf{r}_i(t_{\text{coll}}) = \mathbf{r}_i(t_{\text{curr}}) + (t_{\text{coll}} - t_{\text{curr}}) \mathbf{v}_i(t_{\text{curr}}) \tag{2.17}$$

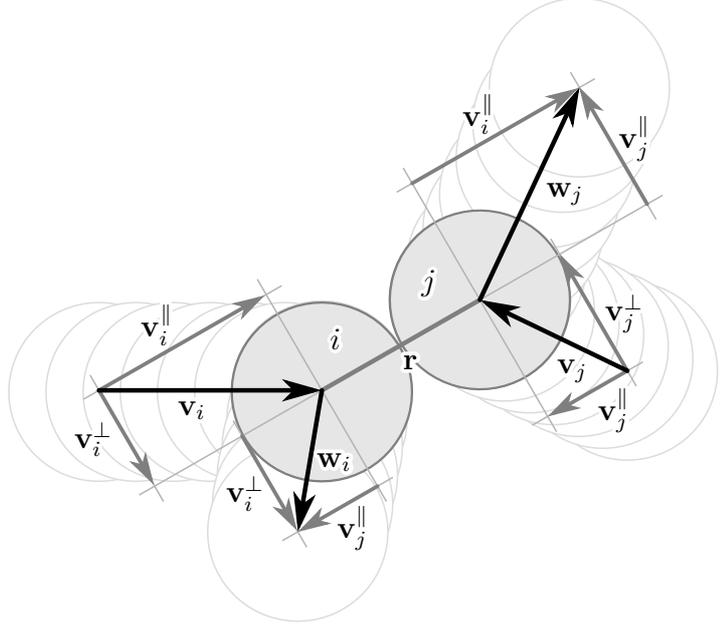
for all particles. Here, t_{curr} is the current time of the system (i.e., the time at which the previous event occurred), t_{coll} is the time at which the next collision takes place, and $\mathbf{r}_i(t)$ and $\mathbf{v}_i(t)$ are the location and the velocity of particle i at time t . Surprisingly, it is possible to optimize this task, as will be explained in Sec. 2.2.2.4.

2.2.1.4 Performing the collision

Once steps (I) and (II) specified in Sec. 2.2.1.1 have been performed, the colliding particles i and j are directly at contact^{2.17}. The ensuing collision event is fully defined by the particles' locations, \mathbf{r}_i and \mathbf{r}_j , their velocities *before* the collision, \mathbf{v}_i and \mathbf{v}_j , and their masses m_i and m_j . The quantities to be determined are the velocities *after* the collision, which shall be denoted by \mathbf{w}_i and \mathbf{w}_j . As in Sec. 2.2.1.2, all time dependencies are notationally suppressed since the time instant is fixed. To solve the

^{2.17}In fact, the particles-to-collide are rarely precisely at contact after conducting the propagation step in a computer simulation. This is due to unavoidable numerical round-offs, which regularly lead to minute overlaps of (or spacings between) the respective particles. While this does not affect the validity or the stability of the MD algorithm, a routine has to explicitly account for the existence of these small errors if it relies on identifying particle overlaps. This is the case for instance when verifying the proper functioning of the MD algorithm (see Sec. A.3.1).

Figure 2.5: After-collision velocities. Decomposition of the particles' velocities into two components: one *parallel* to the vector \mathbf{r} that connects particle i and j , and one *perpendicular* to this vector.



task, it is necessary to decompose the velocity of each particle $k \in \{i, j\}$ into the two components

$$\mathbf{v}_k^{\parallel} = \hat{\mathbf{r}} (\hat{\mathbf{r}} \cdot \mathbf{v}_k) \quad (2.18a)$$

$$\mathbf{v}_k^{\perp} = \mathbf{v}_k - \mathbf{v}_k^{\parallel}, \quad (2.18b)$$

that is, in a component parallel to $\hat{\mathbf{r}}$ and a component perpendicular to $\hat{\mathbf{r}}$, where

$$\hat{\mathbf{r}} = \frac{\mathbf{r}}{r} = \frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|} \quad (2.19)$$

is a unit-length vector parallel the vector connecting the particle centers, and Eq. (2.18b) simply results from requiring

$$\mathbf{v}_k = \mathbf{v}_k^{\parallel} + \mathbf{v}_k^{\perp}. \quad (2.20)$$

The decomposition is visualized in Fig. 2.5; note that despite the sketch being two-dimensional, the problem is actually fully three-dimensional since \mathbf{v}_i^{\perp} and \mathbf{v}_j^{\perp} need not be parallel to each other. If we assume the spheres to be *frictionless* (which we shall do in the remainder of this work), then no momentum is mediated perpendicular to $\hat{\mathbf{r}}$ during the collision, which immediately yields

$$\mathbf{w}_k^{\perp} = \mathbf{v}_k^{\perp}, \quad (2.21)$$

meaning that the velocities perpendicular to $\hat{\mathbf{r}}$ are left unchanged. This is precisely tantamount to the assumption that torques are absent during collisions, i.e., that angular momentum is conserved (cf. Sec. 2.2.1.1). Eq. (2.21) therefore leaves us to be concerned solely with vectors parallel to $\hat{\mathbf{r}}$, which renders the nature of the task

one-dimensional and in fact recovers the elementary problem described in Sec. 2.2.1.1. The quantities involved are m_i , m_j , $v_i^\parallel = |\mathbf{v}_i^\parallel|$, and $v_j^\parallel = |\mathbf{v}_j^\parallel|$ and, as alluded to, if no energy is dissipated in the collision event (which we will assume to hold throughout this work), then the problem can be solved by exploiting the conservation of momentum and energy

$$m_i v_i^\parallel + m_j v_j^\parallel = m_i w_i^\parallel + m_j w_j^\parallel \quad (2.22a)$$

$$\frac{m_i}{2}(v_i^\parallel)^2 + \frac{m_j}{2}(v_j^\parallel)^2 = \frac{m_i}{2}(w_i^\parallel)^2 + \frac{m_j}{2}(w_j^\parallel)^2 . \quad (2.22b)$$

We can rearrange this to

$$m_i \left[v_i^\parallel - w_i^\parallel \right] = -m_j \left[v_j^\parallel - w_j^\parallel \right] \quad (2.23a)$$

$$m_i \left[(v_i^\parallel)^2 - (w_i^\parallel)^2 \right] = -m_j \left[(v_j^\parallel)^2 - (w_j^\parallel)^2 \right] \quad (2.23b)$$

which upon dividing Eq. (2.23b) by Eq. (2.23a) reduces to

$$v_i^\parallel + w_i^\parallel = v_j^\parallel + w_j^\parallel . \quad (2.24)$$

Substituting the solution for w_j^\parallel in Eq. (2.22a) gives

$$\begin{aligned} m_i v_i^\parallel + m_j v_j^\parallel &= m_i w_i^\parallel + m_j (v_i^\parallel - v_j^\parallel + w_i^\parallel) \\ w_i^\parallel &= \frac{1}{m_i + m_j} \left[v_i^\parallel (m_i - m_j) + 2m_j v_j^\parallel \right] \end{aligned} \quad (2.25)$$

which, together with Eqs. (2.18b) and (2.20) yields

$$\mathbf{w}_i = \mathbf{v}_i^\perp + \frac{\hat{\mathbf{r}}}{m_i + m_j} \left[v_i^\parallel (m_i - m_j) + 2m_j v_j^\parallel \right] . \quad (2.26)$$

The after-collision velocity \mathbf{w}_j pertaining to particle j is trivially obtained by swapping the indices i and j in Eq. (2.26). Two special cases of the mass m_j play a prominent role in this work, namely $m_j = m_i$ and $m_j \rightarrow \infty$. In these cases, Eq. (2.26) takes on the particularly simple forms

$$\mathbf{w}_i^{m_j=m_i} = \mathbf{v}_i^\perp + \mathbf{v}_j^\parallel \quad \text{and} \quad (2.27a)$$

$$\mathbf{w}_i^{m_j \rightarrow \infty} = \mathbf{v}_i^\perp - \mathbf{v}_i^\parallel . \quad (2.27b)$$

In the quenched-annealed systems discussed in this work (cf. Sec. 1.3.4.2), the case $m_j = m_i$ occurs frequently since all fluid particles are taken to be identical. The case $m_j \rightarrow \infty$ is no less important since it provides the rule for the case in which fluid particle i collides with matrix particle j . It is worth noting, however, that this infinite-mass limit may entail subtle difficulties, for instance when using the conservation of momentum for validating the MD algorithm (see Appendix A.3.1.1).

2.2.2 Optimization strategies

An important characteristic of an MD algorithm is its change in performance upon altering the number of particles N in the system while leaving all other simulation parameters unchanged. For instance, the unoptimized algorithm in Sec. 2.2.1 requires to check for N^2 possible collisions in each attempt to find the next event. Making use of the so-called “Landau” or “Big O” notation^{2,18}, this can be expressed as $\theta_{\text{coll}} \propto \mathcal{O}(N^2)$, where θ_{coll} denotes the computing time required to determine just the next collision. Consider now the average number of collisions that *one* particle participates in during a fixed interval Δt of simulated time, which is *independent* of N . However, there are N particles in the system, meaning that $\mathcal{O}(N)$ collisions have to be computed for Δt to elapse, which in turn implies the proportionality $\theta_{\Delta t} \propto \mathcal{O}(N^3)$, with $\theta_{\Delta t}$ standing for the computing time required to progress the simulation by the interval Δt .

Since condensed-matter systems generally require a large number of particles for measurements to be reliable, the scaling with N usually represents the primary optimization target. Notwithstanding, enhancements that merely decrease the constant of proportionality in the above-quoted formulae are equally welcome since contributions to the overall performance of the MD algorithm are usually cumulative. As will be detailed in the following, the event-driven MD algorithm introduced in Sec. 2.2.1 offers several targets for optimization, some of which can be taken on with physical insight while others have to be tackled with methodologies developed in the computer sciences. In Appendix A.3.2 we shall see that the reward of those—sometimes intricate—optimizations is an MD algorithm that requires merely $\mathcal{O}(N \log N)$ operations to advance a system of N particles by a constant interval of time. In order to ensure that despite these optimizations the algorithm still is sound and robust, we will in Appendix A.3.1 introduce a number of suitable validation methods.

2.2.2.1 Periodic boundary conditions

The first optimization approach that we shall discuss is somewhat of an oddball since its main objective is not to enhance the actual computing algorithm but rather to minimize the number of particles in the simulated system while still being able to conduct meaningful observations. The method is motivated by the fact that if a condensed-matter system is small, then surface effects play a significant role. Since the available computing power generally puts severe limits to the number of particles that a simulated system can consist of, a means to suppress this effect is highly desirable.

To understand the graveness of the surface problem, consider that macroscopic condensed-matter systems generally comprise an extremely large number of particles—usually comparable to Avogadro’s number, $N_A \simeq 6 \times 10^{23}$. Unless specific conditions are fulfilled (such as a particular arrangement of the particles), the overwhelming majority of the particles in such a system is not part of an interface. For instance, if N_A particles are arranged in a simple cubic lattice and form a cubic body, then only one

^{2,18}http://en.wikipedia.org/wiki/Big_O_notation

out of $N_A/[N_A - (N_A^{1/3} - 2)^3] \simeq 1.4 \times 10^7$ particles is located directly at one of the cube's faces. In computer studies, the number of particles that can be efficiently simulated is *much* smaller than N_A . If we consider a million particles (which is a huge number for simulations), then in the same arrangement almost 6% of the particles reside at one of the side areas, and for $N = 10^3$ this figure increases to an astonishing 49%. From the investigator's point of view this is unfortunate since in many cases the object of interest is the behavior of a material's *bulk* rather than that of its surface.

The basic idea to remedy this problem is to provide interaction partners to the surface particles without having to increase the number of particles in the system. This can be accomplished by applying "periodic boundary conditions" (PBCs; the reason for the naming shall become clear below), a technique which introduces interactions *between particles located at "opposite" surfaces*. This rather abstract and seemingly unphysical concept has first been described in the year 1912 by Born and von Karman in the context of crystal lattices [35] and has been employed in the pioneering works by Metropolis et al. [190] and Alder and Wainwright [6], i.e., from the very beginnings of computer simulations^{2,19}.

We shall illustrate the technique by considering a specific implementation, keeping in mind that the individual features of a given system may suggest—or even require—the realization to be different. If the particles of a system assume continuously-distributed positions (as is the case in this work), then it is convenient to first define a finite spatial region in which all particles are placed. The shape of this region should be reasonably simple while tiling the entire space, and its volume should of course fulfill possibly-present external conditions. While the best choice generally depends on the system and the aims of the investigation, the most popular shape in three-dimensional systems (and the one that we shall adopt) is a cube^{2,20}.

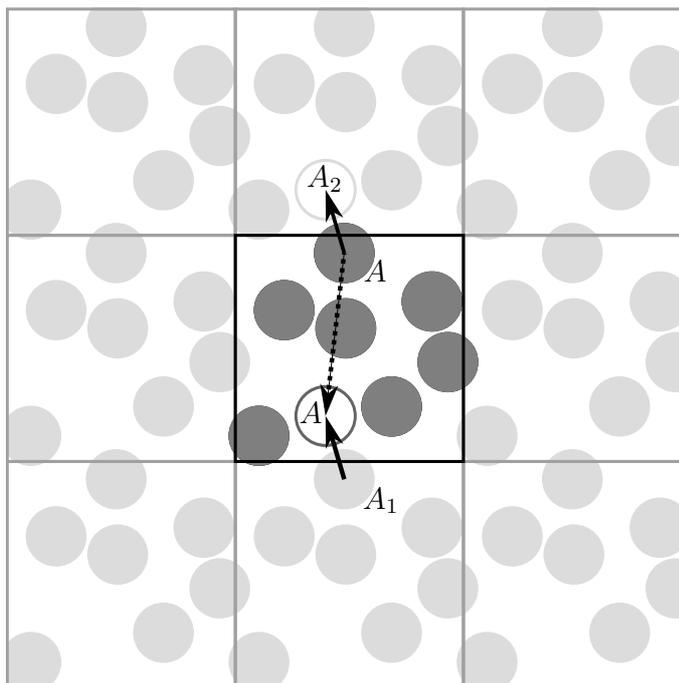
Consider now the particles close to one of the cube's faces. Providing these particles with interaction partners from the opposite face of the cube is fact equivalent to placing a *copy* of the simulation box in a position at which the considered face of the original cube and the opposite face of the copy coincide. This interpretation is visualized in Fig. 2.6, in which a two-dimensional hard-disk system within a quadratic simulation box is portrayed. The figure illustrates that particles extend across the cube's borders even if their centers reside inside the cube; therefore, neighboring copies have to be present not only at the edges but also at the vertices (meaning that 26 copies have to be considered in three dimensions). Since in the bulk of a material there should not exist any artificial barriers, particles are free to leave the original cube. Tracking the full arrows and the empty circles in Fig. 2.6, it is clear that if a particle leaves the central box through some face, then a copy of the same particle enters the same box through the opposite face. Therefore, at any time at least one

^{2,19}Another famous use of PBCs is in one of the two solutions to the one-dimensional Ising model, both of which were provided by Ising himself in 1925 [128].

^{2,20}Other commonly-used shapes of the simulation volume range from cuboids over prisms to more complex bodies which may for instance reflect the real-space base cell of a relevant crystal structure (see, e.g., Ref. [9]).

Figure 2.6: Physical significance of periodic boundary conditions.

Two-dimensional example system of seven hard disks residing in a quadratic area. Dark disks: “original” particles. Light disks: periodic images of the particles. Full arrows: movement of particle A and of its copy A_1 . Dotted arrow: effective movement of particle A . The latter results from the fact that after the movement, A_1 is treated as the original A , whereas the former original is afterwards regarded to be the copy A_2 .



copy of each particle is present in the central cube. This is usually exploited by always defining the copy in the central box to be the “original”, which means that the particle effectively “jumps” to the opposite end of the cube upon crossing one of its faces (indicated by the dotted arrow in Fig. 2.6). This effective behavior facilitates the identification of interacting particle copies, and provides an opportunity to verify the algorithm’s correct functioning by requiring all particles to reside inside the central box.

Obviously, the term “using periodic boundary conditions” is reasonably descriptive of this procedure. As a variation, occasionally the word “toroidal” is used in place of the term “periodic” [106]; this alternate naming is motivated in Fig. 2.7, which shows a two-dimensional system that is warped in a third dimension such that it assumes the topology of a torus and that opposing pairs of its edges (pictured in red and green) eventually coincide^{2.21}.

PBCs in fact achieve the initial aim of reducing surface effects by removing surfaces altogether. Since a surface-less body cannot be less than infinitely large, this method has the welcome side effect of mimicking a system with infinitely-many particles. However, this capability comes at the cost of artificially introducing periodicity, which not only means that no physically relevant information is contained in distances larger than the smallest extent of the simulation volume, but also that the simulation cannot account for any physical effects extending beyond this length. In particular, if conditions are simulated under which a large system would develop a large length

^{2.21}Note that the warping only correctly reflects the topology but *not* the geometry. The three-dimensional analog would be to wrap a cube around a four-dimensional torus, which may mathematically be straightforward but definitely escapes this author’s imagination.

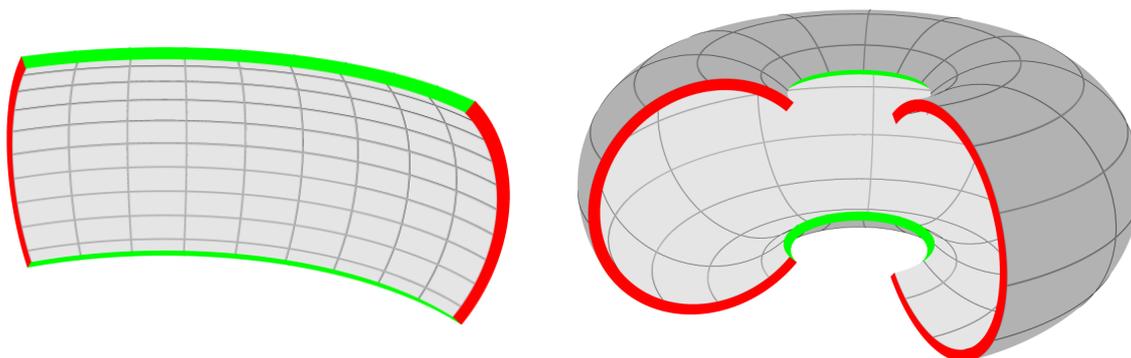


Figure 2.7: Toroidal topology of periodic boundary conditions. A two-dimensional simulation area is distorted in a third dimension such that it assumes the topology of a donut.

scale, then the simulation will fail to exhibit this length scale if only few particles and PBCs are used. This actually reflects the mere fact that the simulated system is really no larger than the original simulation box, and the investigator must be constantly aware of this subtlety of PBCs.

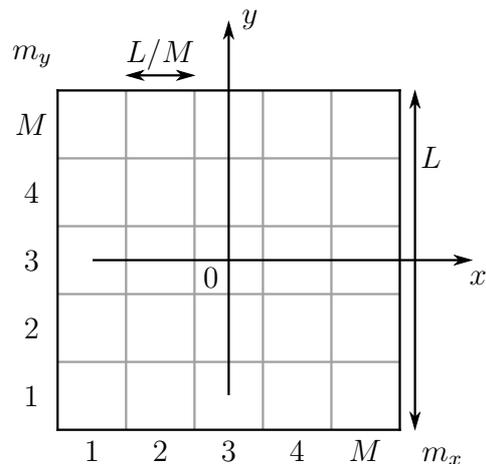
Finally, periodic boundary conditions are commonly complemented by the “nearest image convention” (NIC), the significance of which is the following. Consider a particle i and its interaction with another particle j . Focusing on the periodic image of i that is considered the “real” one (usually the one in the central box), i could in principle interact with *all* periodic images of j . As mentioned before, however, no physical information is contained in the presence of periodic replicas of particles. Therefore, it seems reasonable to disregard all images but one^{2.22}, and in fact if the inter-particle potential emphasizes interactions over short distances, then it is customary to consider only the image of j that is *nearest* to i [9, 85, 219]. In systems in which particles interact through continuously-varying potentials, the NIC is commonly used in combination with a potential cutoff, compensated by additionally considering a homogeneous background [9, 85, 219]. In the case of hard-sphere interactions, however, the NIC is exact in this regard in the sense that by construction, an interaction between i and j takes place only if i and one of the images of j are in immediate spatial proximity of each other.

2.2.2.2 Bucket lists

Historically, the bucket-list method was first implemented *later* than the concept of reusing previously-calculated events (the latter being introduced in Sec. 2.2.2.3). This is due to the fact that bucket lists require substantial memory resources, which were not readily available in the early days of computer simulations [6]. The event-reuse

^{2.22}Notably, in the case of “long-range interactions” it is not valid to consider only one image since in that case the combined influence of all images of a particle j on some particle i always outweighs the influence by any particular image of j (see, e.g., Refs. [9, 85, 172, 219]).

Figure 2.8: Geometry of the buckets. Two-dimensional schematic of the geometry choices for the buckets and the simulation volume, with M being the number of cubic buckets in one dimension and L being the edge length of the cubic simulation box.



concept, on the other hand, entails subtle difficulties if it is used in conjunction with periodic boundary conditions (for details see Sec. 2.2.2.3). Although there does exist a well-established technique to solve these problems in the absence of bucket lists [6], this approach is entirely obsolete if bucket lists are suitably implemented since the associated modifications of the MD algorithm intrinsically alleviate the event-reuse problems. Therefore, we shall first introduce the notion of bucket lists, and only afterwards turn to the event-reuse concept.

Bucket lists directly tackle one of the primary problems of determining interactions in condensed-matter systems, namely that in a system of N particles there exist $N(N-1)$ pairs of particles that—depending on their distance—may interact. Bucket lists greatly facilitate the identification of pairs in which the respective particles are separated by less than a given distance. In event-driven MD, this kind of selection assumes a central role since collision events solely involve particles that are precisely at touching distance.

The basic concept of bucket lists is to partition the simulation volume into sub-volumes (“buckets”^{2.23}) and to assign each particle at each instant of time to exactly one of these buckets. Obviously, this requires the buckets to cover the entire space without overlapping each other. For each assignment of a particle to a bucket, a piece of information is stored not only in the particle’s data structure but also in a data structure pertaining to the bucket. This procedure provides the basis for an efficient identification of all particles that reside in some given region of space: if we know all buckets that overlap this region, then all particles in question are contained in the data structures of these buckets. If the bucket boundaries follow some pattern, then the relevant buckets can be determined from computationally inexpensive rules, with less complicated boundary patterns being obviously favorable.

^{2.23}The entities that we call “buckets” here are referred to as “cells” in most works [9, 85, 219]. However, since we reserved term “cells” for the geometric regions in tessellations (see Sec. 2.4), we chose to instead use the term “buckets”, the latter being reasonably common in the information sciences (see Footnote 2.64).

Since in Sec. 2.2.2.1 we opted for the simulation volume being cubic, for the present work we will accordingly employ cubic buckets (which are therefore arranged on a cubic lattice). As is visualized in Fig. 2.8, assigning a particle to one of the buckets is then particularly simple: if the simulation cube of edge length L is centered about the origin of the coordinate system, and a bucket is to be identified by the indices $m_\alpha \in \{1, 2, \dots, M\}$ with $\alpha \in \{x, y, z\}$ denoting the dimension, then

$$m_\alpha = \left\lceil M \left(\frac{r_\alpha}{L} + \frac{1}{2} \right) \right\rceil \quad (2.28)$$

if the particle in question is located at $\mathbf{r} = \{r_\alpha\}$ with $r_\alpha \in [-L/2, L/2]$. The notation $\lceil a \rceil$ symbolizes the ceiling operation, which yields the smallest integral number that is larger than a .

For reasons that will become clear later, in the context of event-driven MD it is vital to guarantee that at any given simulation time, all particles are assigned to the correct bucket. This can be achieved by introducing a new type of event, namely the “bucket change” event. This event is treated in the same way as collision events, in the sense that the three-step scheme in Sec. 2.2.1.1 is modified to read

- (I') find the time of the next event,
- (II') propagate all particles to that time along straight lines according to their current velocity vectors, and then
- (III') for particles that participate in the event, change the velocities according to the laws of the event,

with an event now being either a bucket change or a collision. At first sight, this introduces additional computational effort since there are now more events to be calculated per simulated time interval. However, the computational gain that can be achieved by employing bucket lists usually clearly outweighs this disadvantage (see below).

For each particle, at any given time there is precisely one future bucket-change event. In the case of cubic buckets, determining the time of this event is particularly simple: since the inside of each bucket is defined by six planes, it is sufficient to evaluate the times at which the particle crosses each of these planes, with the smallest positive one of these times defining the bucket-change event. The time of the next crossing is then

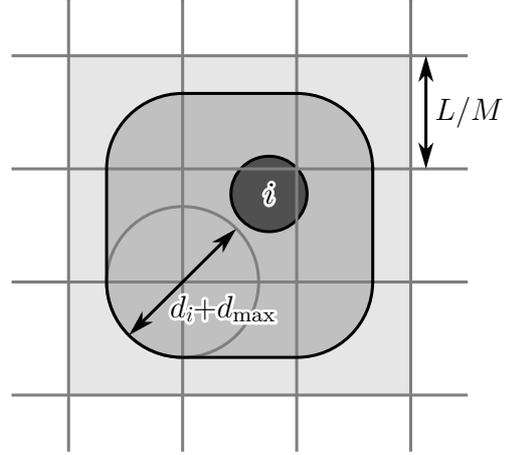
$$t_{\text{cross}} = \min \left\{ t_\alpha^\pm \mid t_\alpha^\pm > 0 \right\} + t_{\text{curr}} \quad , \quad \text{where} \quad (2.29)$$

$$t_\alpha^\pm = \frac{1}{v_\alpha} \left[\frac{L}{M} \left(m_\alpha - \frac{M}{2} - \frac{1}{2} \pm \frac{1}{2} \right) - r_\alpha \right] ,$$

if we use the same geometry descriptors as for Eq. (2.28), and let t_{curr} denote the current time of the system.

For determining collisions in event-driven MD, the bucket list structure can then be exploited as follows. Suppose that we are given a particle i , and that we want to

Figure 2.9: Buckets relevant for collisions. Two-dimensional sketch of the procedure to determine the bucket set $\{B_{\{k_\alpha\}}\}$. Dark gray: particle i , located in the central bucket. Medium gray: “rounded square” area within which a particle of diameter d_{\max} has to be placed to allow for the possibility of touching particle i . Light gray: buckets that overlap the rounded square.



determine the next collision involving i . For this collision to take place, the collision partner needs to be located within one of the buckets $\{B_{\{k_\alpha\}}\}$ that allow it to touch particle i . Any particle that is not located in $\{B_{\{k_\alpha\}}\}$ would first have to enter one of the buckets in $\{B_{\{k_\alpha\}}\}$, which would cause a bucket-change event that would precede the collision of the respective particle with i . Consider now that particle i is mobile itself: as long as it moves only within its current bucket, it will *not* cause a bucket-change event; nevertheless, by propagating this way it might approach other particles such that a collision takes place. This suggests to extend the set $\{B_{\{k_\alpha\}}\}$ such that the buckets therein include all particles that could touch particle i if the latter were located *anywhere* in its current bucket. Provided this, collisions between particle i and particles outside $\{B_{\{k_\alpha\}}\}$ are always preceded by bucket-change events, which means that none of these collisions are candidates for the next event in the system, which in turn means that all buckets not contained in $\{B_{\{k_\alpha\}}\}$ can be *neglected* when seeking the next collision partner of particle i .

The remaining task is to pinpoint which buckets constitute $\{B_{\{k_\alpha\}}\}$. As explained above, this selection is performed based on the distance that allows other particles to touch particle i . Since any particle might be i 's next collision partner, we need to specifically consider the largest of all particles with diameter d_{\max} . According to the rules compiled in the preceding paragraph, the region that needs to be covered by the buckets in $\{B_{\{k_\alpha\}}\}$ is that of a “rounded cube”, the two-dimensional analog of which (a rounded square) is visualized in Fig. 2.9. If particle i measures d_i in diameter, then the rounded cube is obtained by forming the spatial union of the following bodies: the bucket that particle i resides in, a sphere of diameter $d_i + d_{\max}$ centered at each of its eight vertices, a cylinder of height L and diameter $d_i + d_{\max}$ aligned along each of its twelve edges, and a cuboid of the dimensions $L \times L \times (d_i + d_{\max})/2$ attached to each of its six faces. If the condition $L/M > (d_i + d_{\max})/2$ holds, then $\{B_{\{k_\alpha\}}\}$ comprises simply the bucket that particle i is located in and the 26 buckets that directly border it, that is,

$$k_\alpha \in \{(m_\alpha - 1 \bmod M), m_\alpha, (m_\alpha + 1 \bmod M)\}, \quad (2.30)$$

where m_α is determined according to Eq. (2.28). The notation “mod” signals the

modulo operation, which is required if periodic boundary conditions are employed since in this case the condition $k_\alpha \in \{1, 2, \dots, M\}$ needs to be adhered to.

The choice of M significantly impacts the performance of the MD algorithm. If it is chosen such that $L/M < (d_i + d_{\max})/2$, then $\{B_{\{k_\alpha\}}\}$ consists of more than 27 buckets, with larger M entailing more buckets. If, on the other hand, M is selected such that $L/M \gg (d_i + d_{\max})/2$, then only few buckets can be neglected when seeking the collision partners of particle i . Hence, usually a value of M is employed for which $L/M \gtrsim (d_i + d_{\max})/2$ holds. Further, the value of M is the same for all particles since there is only one bucket grid that the particles are assigned to. If we want to ensure that $\{B_{\{k_\alpha\}}\}$ comprises 27 buckets irrespective of the particle, then d_{\max} has to be used in place of d_i . Finally, considering that M has to be an integral number this results in

$$M = \left\lfloor \frac{L}{d_{\max}} \right\rfloor \quad (2.31)$$

for the choice of M . Here, similar to the ceiling operation in Eq. (2.28), the notation $\lfloor a \rfloor$ describes the floor operation, which returns the largest integral number that is smaller than a .

Using the bucket-list method, the computational effort for finding the next collision of particle i is reduced from $\mathcal{O}(N)$ to $\mathcal{O}(1)$, meaning that finding all candidates for the next collision now requires only $\mathcal{O}(N)$ operations. Since no other part of the event-driven algorithm takes more than $\mathcal{O}(N)$ operations per event, bucket lists increase the overall performance of the program such that only $\mathcal{O}(N^2)$ instead of $\mathcal{O}(N^3)$ operations are needed to advance a system by a fixed interval of simulated time. In a final word, it is worth noting that using the bucket-list technique, it is well possible that no future collision is found for a given particle i . Although this means that there might exist system configurations in which *no* particle is known to have future collisions (this is particularly likely if M is large), this does not represent a problem since there is always a next bucket-change event in the system.

2.2.2.3 Reuse of previously-calculated events

The reuse of previously-calculated events has been implemented as early as in the original work that introduced event-driven MD [6]. The idea is based on noticing that if the quantifiers that describe a hard-sphere system are suitably chosen, then any event in the system leaves most of these quantifiers unchanged.

More specifically, suppose that we have already performed $l-1$ events according to the cycle in Sec. 2.2.2.2, and that for event l we have already prepared the list of candidate events $\{E_j\}_l$. Each candidate E_j in this list is determined from only a small set of information $\{I_k\}_j$, where each piece of information I_k is taken from the pool of quantifiers that characterize the system. In consequence of putting event l to effect, then, only a few of the quantifiers I_k change. If, resultingly, some set $\{I_k\}_j$ remains unchanged, then the associated event E_j will remain valid after event l and will therefore also be an element of the list of candidate events $\{E_j\}_{l+1}$ for event $l+1$.

Obviously, any such E_j can simply be reused instead of recomputing it, which can be exploited to construct $\{E_j\}_{l+1}$ according to the following three-step procedure:

- (i) remove from $\{E_j\}_l$ all those E_j for which $\{I_k\}_j$ has changed as a result of performing event l ,
- (ii) set $\{E_j\}_{l+1} = \{E_j\}_l$, and
- (iii) add to $\{E_j\}_{l+1}$ those candidate events that arise from the system's changed quantifier pool.

Step (ii) in practice does not involve any action since at no time $\{E_j\}_{l+1}$ and $\{E_j\}_l$ need to be known simultaneously. In the following, we shall therefore focus on the details of steps (i) and (iii).

In step (i), the first obvious observation is that the events l and $l+1$ cannot be the same event, which immediately entails that event l has to be removed from $\{E_j\}_l$. The remaining invalid candidate events in $\{E_j\}_l$ can be identified by suitably defining the set of information $\{I_k\}_j$ that pertains to a candidate event E_j . For this, consider the assumptions under which a particular E_j is evaluated. In the case of collision events, the principal information underlying Eq. (2.16) is that from the time of the identification of the event up to the time of actual collision the participating particles undergo a free flight. This means that if event l is a collision between the particles $P_{l,1}$ and $P_{l,2}$, then a candidate collision E_j is rendered invalid if it involves at least one of these particles. This is because obviously the collision between $P_{l,1}$ and $P_{l,2}$ disrupts the free flight of these particles. Since the assumption of free flight also constitutes the basis of Eq. (2.29), the same conclusion is effective if E_j is a bucket change of either $P_{l,1}$ or $P_{l,2}$.

If event l is a bucket-change event (say of particle P_l), then only implications on collisions have to be considered since there does not exist any other E_j that is a bucket change of P_l . The effect of a bucket-change event is rather subtle as it leaves the state of free flight of P_l unchanged. Instead, a different parameter is altered: although during the event the location of P_l is not modified, by definition after the event the particle is associated with a different bucket^{2,24}. This bucket, in turn, determines the set of buckets $\{B_{\{k_\alpha\}}\}_{P_l}$ that is used for seeking candidate collisions that particle P_l

^{2,24}Recalling the procedure in Sec. 2.2.2.2, this is not quite trivial. Numerically, evaluating Eq. (2.28) always yields a definite answer, which is a problem since a bucket-change event does not include modifications of the particle's location. Even worse, since at the time of the event, the particle is supposed to be located precisely at the interface of two buckets, unavoidable numerical rounding renders the result of Eq. (2.28) unpredictable. Therefore, a different strategy is needed to ensure that a particle is consistently associated with the correct bucket. Instead of always calculating the bucket index from Eq. (2.28), this equation is evaluated only once for each particle, namely at the beginning of the simulation, with the resulting bucket indices m_α being stored in memory fields pertaining to the particles. Subsequently, upon a bucket change of a particle, its respective memory field is adjusted, i.e., its m_α are incremented or decremented according to the bucket-change event (taking into account periodic boundary conditions). In computations like Eq. (2.29) then these stored m_α are used instead of the result of Eq. (2.28).

participates in (see Sec. 2.2.2.2). Let $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{before}}$ and $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{after}}$ be the sets of buckets before and after event l , respectively. Since in principle any bucket may be added to such a set of buckets, there is no problem in $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{after}}$ containing additional buckets stemming from $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{before}}$ —merely a small performance toll has to be paid since $\{E_j\}$ will contain more candidate collisions than necessary. Also, fortunately, there is no risk of cluttering $\{E_j\}$ with overly many candidates, as (by the mechanism described above) each extraneous collision will eventually be discarded due to other collisions. Therefore, a bucket change of P_l does not require to purge $\{E_j\}$ from candidate collisions that involve P_l . Summarizing, in step (i), elements have to be removed from the list of candidate events only if event l is a collision.

Removing invalid entries from $\{E_j\}_l$ is a computationally expensive procedure in itself if done without optimization. In a simple approach, each E_j has to be checked for validity, which requires $\mathcal{O}(N)$ operations if the bucket-list technique is used. This is because buckets allow to seek for events in a neighborhood of constant size around a particle, leading to $\mathcal{O}(N)$ elements in $\{E_j\}$ if done for each of the N particles. However, it is possible to avoid having to search all of the $\{E_j\}$ if supplementary information about the E_j is maintained: recognizing that removing events is connected to particles, it is sensible to compile information about which events are associated with a given particle. This can be achieved for instance with a so-called “linked list” [219]: each particle is provided with a memory field that contains the index j of an event, and each event is given a similar memory field. These index placeholders can be exploited to construct a chain in which each segment points to the next. The first element of this event chain is pointed to by the particle’s memory field, whereas the end of the chain is signaled by the memory field of the last event in the chain having a value outside the allowed range of j .

In fact, for the present purpose a single pointer per event is insufficient. Firstly, if an event is a collision, then it has to belong to two distinct event chains—one for each participating particle—which demands for at least two pointers per event. Secondly, in order to efficiently remove single elements from a chain, it is desirable to be able to follow the chain in “forward” and “backward” direction, which requires twice as many pointers, i.e., a total of four pointers per event. In the case of bucket changes, which are associated with only one particle, two of the pointers remain unused. Each chain originates from precisely one particle, but it turns out to be advantageous to have each particle spawn *two* chains. This is because of the following: suppose that starting at some particle i we have forward-iterated along a chain and arrived at some collision. There are then two forward pointers along which we could continue. To choose the right one, we could randomly pick one of the two pointers and check if the resulting event is associated with particle i ; if it is not, then we have to pick the other pointer. However, there is a more efficient procedure: noticing that in a collision there is always a “first particle” and a “second particle” (where the assignment is arbitrarily fixed by the collision-finding procedure), it is compelling to establish two chains per particle—one in which each element recorded the chain-spawning particle as the first particle, and one in which it consistently is the second. The iteration of these chains

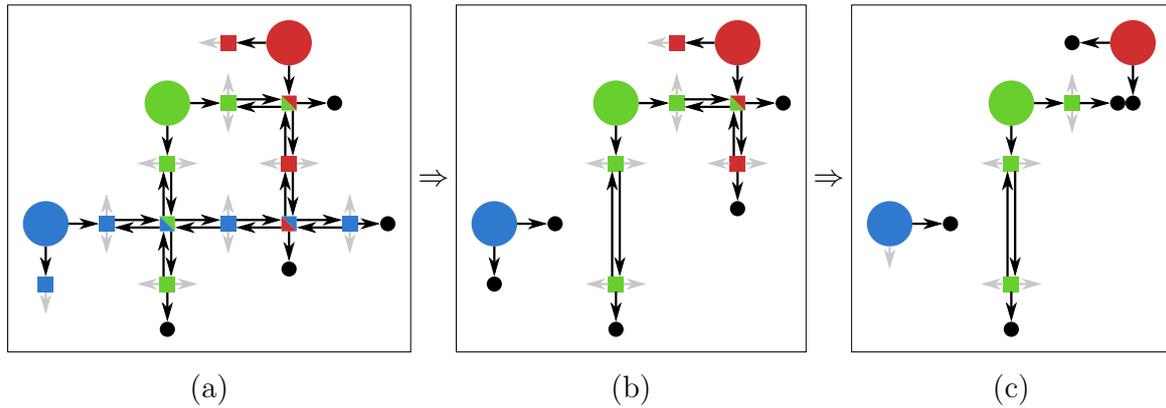


Figure 2.10: Removing elements from the list of candidate events. Schematic representation of the linked-list data structure described in detail in the text. Large colored disks: particles. Small colored squares: events. Small black disks: “nowhere”, signaling the end of a chain. Black arrows: pointers to events or to nowhere. Small light-gray arrows lead to events irrelevant to the discussion. Positions of the elements bear no physical meaning and are chosen to accommodate the pictured information. Colors highlight which elements are associated with which particle. Panel (a): initial collection of candidate events; panel (b): after removing the blue events; panel (c) after removing the red events.

can then be separated by defining the “first-particle chain” along the “first pointer pair” and the “second-particle chain” along the “second pointer pair”, where a pointer pair is constituted by one forward and one backward pointer. Bucket-change events can be incorporated in this scheme without modification by (arbitrarily) defining one of the pointer pairs to carry the relevant information.

Figure 2.10 illustrates the subtleties of removing events. As signaled by the black arrows, each event (small colored squares) points forward and backward to another event or, alternatively, to “nowhere” (small black disks), thus creating event chains of finite length. The small light-gray arrows point to events irrelevant to this discussion. Each particle (large colored disks) initiates two chains—one displayed horizontally and one vertically; the chains pertinent to a particle are displayed in the same color as the particle. Suppose now that the red and the blue particle underwent a collision event (the red/blue box). In the first step, from (a) to (b), the chains of the blue particle are eliminated. In the course of this, an element is removed from the vertical green chain, which necessitates to adjust the pointers of the preceding and the succeeding event in the green chain such that they point at each other. In the vertical red chain, the last element is removed so that in the altered chain the second element now forward-points to nowhere. In the second step, from (b) to (c), the chains of the red particle are eliminated, thereby removing the last element in the horizontal green chain. As desired, resultingly, all pointers of the red and of the blue particle now point to nowhere.

In step (iii), in direct consequence of step (i), the procedure of finding new candidate events also depends on whether event l is a collision or a bucket change. In

the collision case, finding all relevant collisions and bucket changes requires a double carrying-out of the entire search algorithm prescribed in Sec. 2.2.2.2—once with $P_{l,1}$ being the “given particle”, and once with $P_{l,2}$ in this role. In the case of event l being a bucket change, clearly first of all the next bucket change of P_l needs to be determined. For the potential collisions of P_l , the relevant effect of event l is that (as mentioned above) it alters the set of buckets $\{B_{\{k_\alpha\}}\}_{P_l}$. However, there is no need to scan all buckets in $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{after}}$ when seeking collisions according to Sec. 2.2.2.2. This is because a candidate collision is already contained in $\{E_j\}_l$ if the bucket of the collision partner is a member of both $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{before}}$ and $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{after}}$. Therefore, merely the buckets in the set difference $\{B_{\{k_\alpha\}}\}_{P_l}^{\text{after}} \setminus \{B_{\{k_\alpha\}}\}_{P_l}^{\text{before}}$ contain particles that potentially contribute new candidate collisions of P_l to $\{E_j\}$. Given a geometry as in Fig. 2.9, in three dimensions this set difference contains precisely nine buckets.

In summary, if previously-calculated events are reused and linked lists are employed in the presence of buckets^{2,25}, the computational effort of determining the set of candidates $\{E_j\}_l$ for event l is reduced from $\mathcal{O}(N)$ to $\mathcal{O}(1)$ operations. This is true for all $\{E_j\}_l$ except for the very first one since in this case there are no reusable events yet. Although this means that part of the program scales with $\mathcal{O}(N)$, the effect of the latter on the program’s overall performance is negligible since in any case $\mathcal{O}(N)$ events have to be performed to advance the simulation by a given time interval.

2.2.2.4 Particle-local times

Although conceptually simple, it can be computationally expensive to perform step (II) in the three-step procedure in Sec. 2.2.1.1. In fact, propagating all particles before each event entails $\mathcal{O}(N)$ operations per event, which is, notably, less efficient than finding the respective event if the methods outlined in Secs. 2.2.2.2 and 2.2.2.3 are implemented. In this situation, depending on the system details, advancing the particles may be the dominant consumer of computation time if N is as small as 10^3 . Fortunately, this can be circumvented—albeit by a seemingly strange method: assigning a “local time” to each particle [219]. The idea is to spatially and temporally advance a particle only if it is absolutely necessary. In the following, we shall examine which conditions constitute such a necessity.

To approach this question, first of all we note that advancing a particle i solely alters its location \mathbf{r}_i and its (local) time t_i . Therefore, failing to propagate particle i can only render an operation invalid if the latter makes explicit reference to one of the

^{2,25}If periodic boundary conditions and the event-reuse concept are to be employed in the *absence* of buckets (which is not the case in this work, but for instance in Ref. [6]), then the required precautions differ substantially from those described in this section. This is because in this case it is a priori unknown which of the infinitely-many images of a particle j around some particle i will be the next one to undergo a collision with i . A solution to this is to recompute the entire set of candidate events after the fastest particle may have traversed a quarter of the simulation box [6]. This renders the reuse of events without buckets only slightly less effective than with buckets, the reason being that a box traversing requires the same $\mathcal{O}(N)$ operations as recomputing the entire event candidate list.

named properties of this particle. The term “operations” in this context includes all computations that are necessary for the MD algorithm to proceed, namely

- (1) performing bucket changes,
- (2) performing collisions,
- (3) finding candidate bucket changes, and
- (4) finding candidate collisions.

In point (1), as laid out in Footnote 2.24, neither \mathbf{r}_i nor t_i is explicitly referenced since this information is useless for characterizing a changing bucket index.

For point (2), the relevant equations are Eq. (2.26) and Eq.(2.19), from which one can immediately see that the locations of the colliding particles, \mathbf{r}_i and \mathbf{r}_j , are referenced. Therefore, in order to correctly perform a collision event, the participating particles i and j first have to be advanced such that their local times t_i and t_j match the time of the collision event.

For point (3), the governing equation is Eq. (2.29), which explicitly involves the location \mathbf{r}_i of the particle i for which the next bucket change is sought. Recalling that this equation relates to the position of i relative to six different planes, \mathbf{r}_i has to be adjusted such that the particle is actually located in the bucket m_α . This can be achieved by advancing the particle to the time of the last event^{2.26}, which is prudent also if t_{curr} in Eq. (2.29) is to be used without adaption.

Point (4) is the most complicated among the four points; yet it is still reasonably simple. The procedure of finding a candidate collision between two given particles i and j is prescribed by Eqs. (2.11) and (2.16), which via Eq. (2.12) involve the positions \mathbf{r}_i and \mathbf{r}_j of the particles. Therefore, both i and j need to be advanced to the current system time t_{curr} . In order to find *all* candidate collisions that involve particle i , we need to also consider the potential collision partners of i , which are the particles residing in the buckets $\{k_\alpha\}_i$ as defined in Eq. (2.30). As a result, all particles located in one of the buckets $\{k_\alpha\}_i$ need to be advanced such that their local times match t_{curr} . In case of the preceding event having been a collision, the above procedure needs to be carried out for both of the participating particles being particle i .

Since in any case point (4) needs to be performed in consequence of an event, the resulting procedure reduces to the following sequence:

- (1') perform the event,

^{2.26}Here, a problem similar to that in Footnote 2.24 arises: if the last event was a bucket change, then at the time of this event the particle is located right at the interface between two buckets. In this case, one of the times t_α^\pm in Eq. (2.29) should be precisely zero and therefore be discarded. However, rounding errors may render this time slightly *larger* than zero, which would effect an unphysical repetition of the bucket-change event. This can be avoided by considering only *three* plane crossings instead of six, namely those that a particle can possibly cross, given the indices m_α and the signs of the velocity components v_α . For instance, if it is $v_x > 0$, then the change $m_x \rightarrow m_x + 1$ may be the relevant one whereas the change $m_x \rightarrow m_x - 1$ can by no means take place. Using this procedure, advancing the particle is actually superfluous if in Eq. (2.29) the reference time for t_{cross} is taken to be the particle’s local time, t_i , instead of the time of the last event in the system, t_{curr} .

- (2') advance the particles $\{i\}$ that participated in the event,
- (3') advance all particles located in the buckets $\{k_\alpha\}_i$ that surround each of the particles $\{i\}$, and
- (4') seek for candidate collisions and bucket changes as usual.

In consequence of this procedure, the computational effort of advancing particles reduces from $\mathcal{O}(N)$ to $\mathcal{O}(1)$ per event, which is owed to the fact that $\{k_\alpha\}$ is of fixed size.

There remains one caveat of particle-local times: in order to compute observables in a physically meaningful way (see Sec. 2.5), at the time of an observation the system has to be in a synchronized state in the sense that the local time of *all* particles has to be advanced to match t_{curr} . Since observations are the ultimate objective of a simulation, the algorithm would continue to scale with $\mathcal{O}(N)$ if the frequency of observations depended on the number of elapsed events. This, however, is not the case—instead observations are usually conducted at regular simulation time intervals. Recalling that $\mathcal{O}(N)$ operations are required to advance the system by a fixed interval of simulated time, propagating particles in conjunction with observations is no more costly than the ideal MD algorithm in which processing one event would take $\mathcal{O}(1)$ operations.

2.2.2.5 Efficient event list structure

As alluded to in Sec. 2.2.2.3, finding a particular element in the list of candidate events $\{E_j\}$ can be a computationally expensive procedure since at any time this list contains at least $\mathcal{O}(N)$ elements. Whereas in Sec. 2.2.2.3 the issue was to remove specific events from $\{E_j\}$, in this section the goal is to efficiently identify which of the members of $\{E_j\}$ constitutes the “next event”.

In the absence of additional physical insight, it is purely a computer-science problem to establish an economical method appropriate for this task. Following the emergence of fast computing machines, a multitude of data structures has been developed to suit various different requirements; for an overview over the vast field see, e.g., the classic work *The Art of Computer Programming* by D. Knuth, in particular volume 3 *Searching and Sorting* [153]. The specific problem of finding the next event from a list of candidates falls into the class of so-called “priority queues”. The defining property of this kind of collection is the existence of a comparison operator that allows (at least in principle) to arrange the pertinent items in an ordered sequence; consequentially, a priority queue features one element with the highest priority^{2.27}, which is then to be identified. Moreover, priority queues are typically required to allow for adding and removing elements, i.e., to dynamically grow and shrink. In the case of event-driven

^{2.27}One might also allow a priority queue to contain elements with the *same* priority; if such elements are actually present, then the queue is said to be “degenerate” and additional rules have to be introduced to establish an unambiguous ordering among the respective elements.

MD, the ordering criterion would obviously be a numeric comparison between the scheduled times of the candidate events, with the top-priority element being the event with the smallest associated time—the next event.

Priority queues constitute an integral part of a wealth of real-life applications^{2.28}, and numerous data structures have been designed to satisfy the various requirements of extracting information from such queues as well as modifying the items therein. As has been laid out before, in the case of handling the list of candidate events $\{E_j\}$, the data structure has to explicitly provide facilities for

- (i) identifying the next event,
- (ii) deleting an *arbitrary* event (not only the next one), and
- (iii) adding an event,

with the need for operation (ii) arising from the procedures described in Sec. 2.2.2.3. Two competing criteria can be invoked to select among suitable data structure candidates: on the one hand, executing the above operations should be as efficient as possible; on the other hand, the data structure should be reasonably simple in order to facilitate coding and debugging. As an example, in popular introductory implementations of event-driven MD, it is common to maintain $\{E_j\}$ as a contiguous one-dimensional array in which the event sequence does not obey any specific order. While this approach boasts a minimal implementation effort, it requires an unacceptable $\mathcal{O}(N)$ steps to identify the next event, which is owed to the fact that any member of $\{E_j\}$ could be the element in question.

Naturally, any data structure that is to outperform the unordered-list approach has to accomplish each of the operations (i)–(iii) in less than $\mathcal{O}(N)$ steps. Structures fulfilling this requirement range from sorted lists over hash tables and trees to heaps, and also include combinations of these. In this work, we shall adopt the concept of a *binary heap* [176] since it represents an excellent compromise between coding effort and program runtime. Other authors prefer a binary tree [219], which may under some circumstances outperform the heap (due to its usage of links) but requires more care on the programming side and also runs the risk of degenerating. In the following, we shall describe in detail what the structure of a standard binary heap is, how the above-listed operations (i)–(iii) translate to this structure, and how well they perform.

Figure 2.11 depicts an example binary heap. A heap is constituted by a collection of N nodes that are arranged on M levels. Each node is connected to exactly one

^{2.28}Common commercial applications of priority queues are, among others, emergency management systems, customer order services, telephone and data networks, as well as computer operating systems^{2.29}. On a different note, the concept is a core part of algorithms that identify the shortest path between any two given points in a network of nodes and connectors. Obviously, applications of this problem include computer network routing and driving directions software. Less obviously, and oddly enough, networks (and therefore priority queues) can also be used to determine solutions to Rubik's famous cube^{2.30}.

^{2.29}http://en.wikipedia.org/wiki/Priority_queue

^{2.30}http://en.wikipedia.org/wiki/Shortest_path_problem

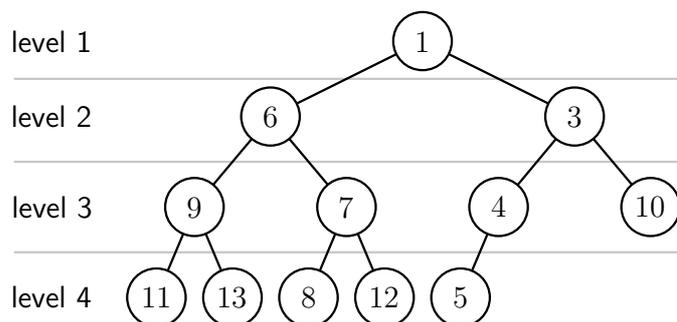


Figure 2.11: Structure of a heap. Schematic representation of a heap, in which the nodes (circles) are connected (lines) according to parent-child relationships to form the arrangement of an almost-complete binary tree. The numbers indicate the priority of each node; according to the heap property, each parent element has a higher priority than each of its children elements.

“parent” node, and can itself be parent to at most two “child” nodes. This rule leads to the structure of an (upside-down) tree in which levels with a higher index contain more nodes. Such a tree is spawned by a so-called “root” node on level 1, which is special in that it does not possess a parent. In the case of the binary heap, the tree is maintained such that all nodes on levels $1 \dots M - 2$ have exactly two children, and the nodes on level $M - 1$ are given children *starting from the left*. On level M , obviously, none of the nodes have children—otherwise there would be another level. This arrangement, which is also called an “almost-complete binary tree”, entails $M = \lceil \log_2 N \rceil$ levels; in the example shown in Fig. 2.11, the $N = 12$ nodes result in the depicted $M = \lceil 3.585 \rceil = 4$ levels.

Consider now that each of the nodes in a heap is associated with a priority. In Fig. 2.11, this priority is symbolized by the integer value inscribed in each circle; in event-driven MD of course the priority would be the associated time of the event. In either case, a lower number signals a higher priority. The so-called “heap property”, which distinguishes a heap from an otherwise unstructured tree, states that *each parent node has a higher priority than its child nodes*. It is easily verified that all nodes in Fig. 2.11 fulfill this property, i.e., that the heap in the figure is “consistent”. Since the heap property is purely local, consistency does *not* imply that all nodes on a certain level have a higher priority than the nodes on the subsequent level; this can be seen for instance from the elements with the priorities 4 and 6 in the figure. Notably, there is one paramount exception: in a consistent heap, the node at level 1 is *always* the one with the highest priority. This is fortunate (and actually by design) since it allows to identify the highest-priority element in $\mathcal{O}(1)$ steps.

Having thus accomplished operation (i) from the above list, let’s now turn to the more complicated operations of adding and removing nodes. Consider first the simpler case of adding an element to a consistent heap. The pertinent procedure is illustrated in panels (a)–(c) of Fig. 2.12, all of which depict only the right branch of the heap in Fig. 2.11. The adding procedure starts by placing the additional node (highlighted in green) in the only position that preserves the “almost-complete binary tree” arrangement, namely on level M in the leftmost unoccupied spot. Since this location is essentially random from the viewpoint of queuing, the additional element is likely to have a higher priority than its new parent—which would of course violate

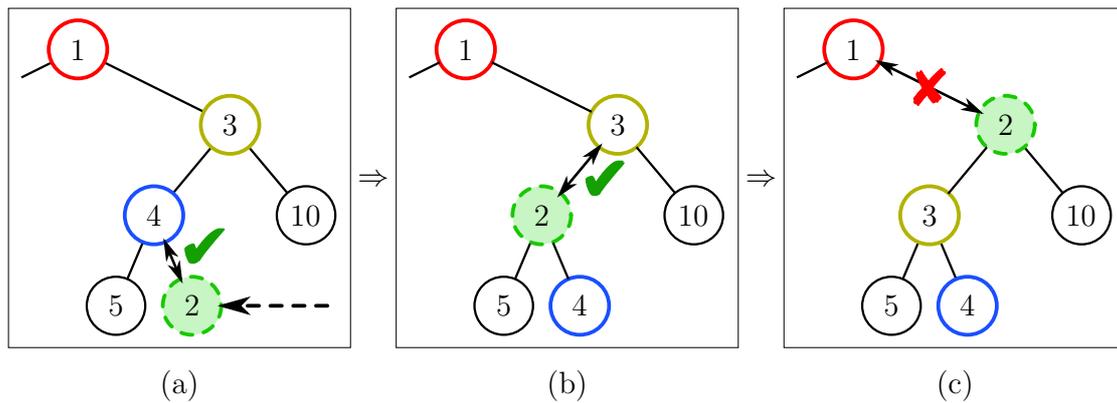


Figure 2.12: Adding a node to a heap. The open circles in panels (a)–(c) constitute the right branch of the heap in Fig. 2.11. The additional node (green disk) is first placed at the lowest level of the heap [panel (a)]. Then, it is progressively up-sifted by repeatedly comparing it to its parent [blue/yellow/red circle in panel (a)/(b)/(c)] and by swapping places if necessary because the new node’s priority is higher than that of its parent [panels (a) and (b)]. Green hooks mark required node swaps; the red cross indicates to retain the current node arrangement.

the heap property. To alleviate this heap inconsistency we introduce the concept of “sifting” a node, with the event-add case requiring a particular variation called “up-sifting”. One step in this procedure consists of comparing the additional node with its parent [the blue/yellow/red circle in panel (a)/(b)/(c)] and exchanging the two elements if these two nodes violate the heap property [as in panels (a) and (b)]. After each sifting step, the additional node attains a new parent, which prompts for another sifting step; obviously, this chain of sifting steps is terminated once the additional node has a lower priority than its new parent [panel (c)]. On average, an element with an arbitrary priority will therefore require $M/2$ sifting steps, which means that adding an event requires $\mathcal{O}(\log N)$ operations.

Removing an element from a heap is slightly more complicated than adding an element. As can be seen in Fig. 2.13(a) and (d), the first step after removing an element from a consistent heap is to take the rightmost node from level M and place it in the gap that was left by the removed node. From here, two types of actions have to be checked: up-sifting (as before) and the new procedure of “down-sifting”. Up-sifting in the node-remove case works precisely as in the node-add case. The down-sifting concept can be understood by considering panel (d), which illustrates the specific case of removing the highest-priority item. In this case, as in any other, the relocated node may have a lower priority than its new children, leading to an inconsistent heap. Just like up-sifting, down-sifting restores the heap consistency in a step-wise process. Contrary to up-sifting, finding a due element swap requires *two* comparisons: the first comparison (symbolized by the red–green arrows) determines the child with the higher priority; in the second comparison, only this child is then compared to the relocated node, and swapped if necessary. This procedure ensures that after the swapping step

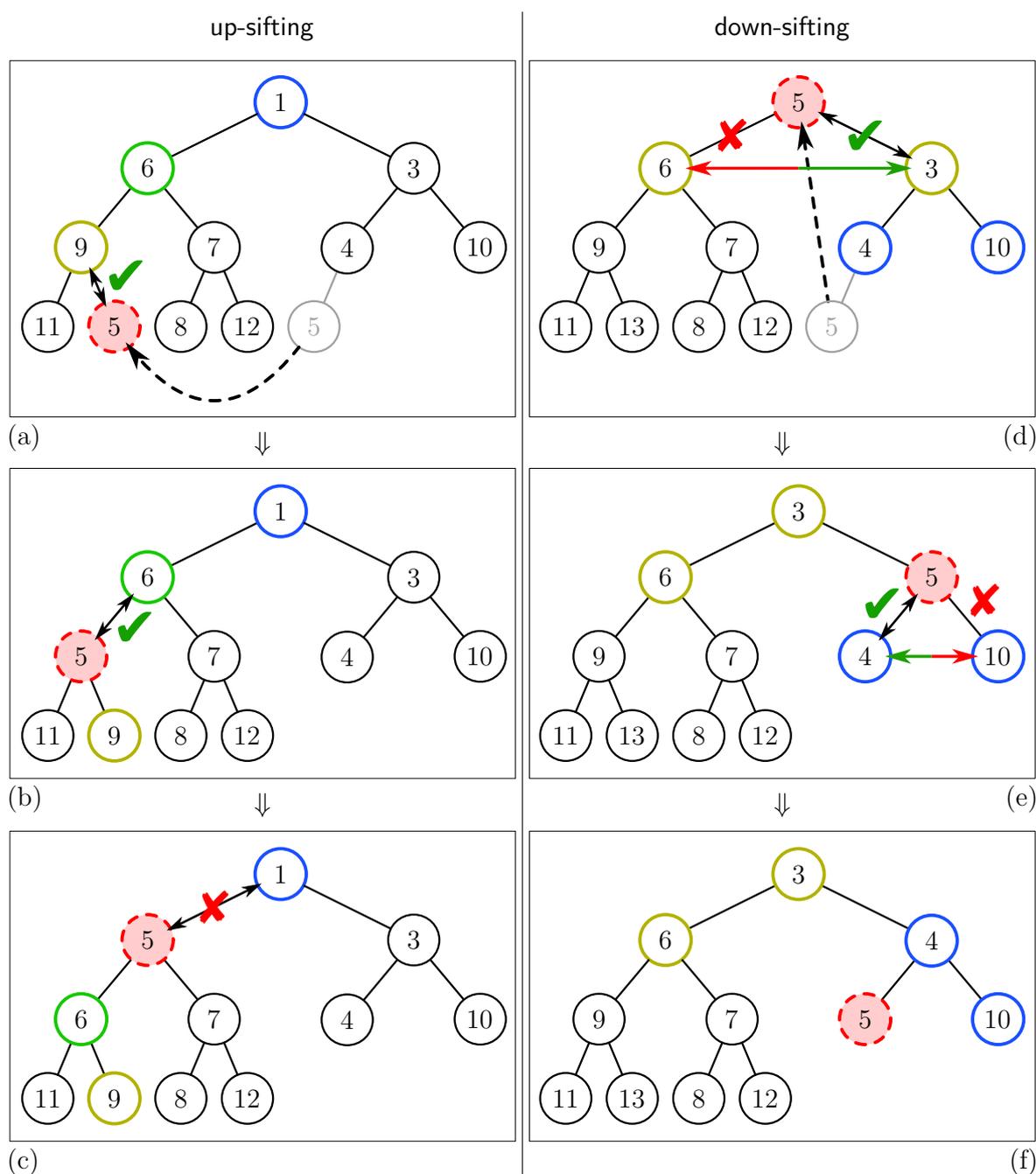
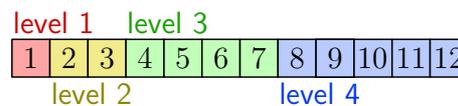


Figure 2.13: Removing a node from a heap. In the first step [panels (a) and (d)], the rightmost node from the last level (gray circle, red disk) is relocated to the position of the removed node. The left column [panels (a)–(c)] shows the succeeding up-sifting, very much like in Fig. 2.12. The right column [panels (d)–(f)] illustrates the alternatively-required down-sifting procedure. In each down-sift step, first the children of the relocated node [yellow/blue circles in panel (d)/(e)] are compared to each other (red–green arrows). Then, the relocated node is swapped with the higher-priority child (green arrow) in order to restore the heap property if it was violated [panels (d) and (e)]. Green hooks mark required node swaps; red crosses indicate to retain the current node arrangement.

Figure 2.14: Array representation of a heap. Representation of the example heap in Fig. 2.11 as a contiguous array. Colors code the heap level, numbers indicate the node index. Priorities are not depicted.



the heap property is observed. Obviously, the down-sifting steps have to be repeated until the relocated item has a higher priority than all of its children. Note that is vital to consider up-sifting *and* down-sifting whenever a node is relocated due to another one's removal. However, since in either case the number of sifting steps depends mainly on the number of levels M , removing an event requires $\mathcal{O}(\log N)$ operations—just like adding an event.

As a technical detail, one of the differences between a heap and a simple tree lies in its data handling. Due to its being an almost-complete tree, a heap can be conveniently stored as a contiguous array (see Fig. 2.14). This has the advantage that no link structure is necessary to determine the parent and the children of a node. For instance, consider the case that the elements are indexed starting from “1” as in Fig. 2.14. In this case, if a node has the index i , then its parent node is located at index $\lfloor i/2 \rfloor$ (which may be computed as an integer division) and its children have the indices $2i$ and $2i+1$. This advantage comes at the penalty of having to actually swap elements in the array; however, the associated cost is moderate since an event is constituted by only about half a dozen numbers (most of which are the links introduced in Sec. 2.2.2.3).

In summary, since operations (ii) and (iii) perform worse than operation (i), the former control the overall performance in handling the event list. This statement still holds when considering the effort for establishing the heap in the first place: since a one-element collection already constitutes a consistent heap, this “additional operation” requires merely $\mathcal{O}(1)$ steps. As a result, $\mathcal{O}(\log N)$ list operations have to be carried out subsequent to performing a single event; this is by no means great since all other parts of the program operate at $\mathcal{O}(1)$. However, unfortunately there exists no known technique to remedy this deficiency in a single-threaded program. Merely parallelly-coded programs are able to overcome this behavior, albeit at a greatly increased programming cost [195].

2.3 Initial states of quenched-annealed systems

2.3.1 Overview

As alluded to in Sec. 2.1, the initial setup of a hard-sphere (HS) quenched-annealed (QA) system represents a formidable challenge. In this section, we will point out algorithms known to succeed in setting up disordered one-component HS systems at high density, comment on problems in extending those algorithms to the QA protocol, and report on our custom solution to the problem.

The first task is to generate configurations of obstacles. This is easily accomplished, since in none of the considered cases, the packing fraction of the matrix exceeds 35% (cf. Chapter 3). Such low values reside well in the fluid regime of a monodisperse one-component HS fluid, as shown first by Alder and Wainwright [5] and Wood and Jacobson [286] and more recently by Odriozola [201]. This allows to use even the simplest of the procedures discussed in Sec. 2.3.2.1 to insert the N_m matrix particles into the system. Subsequently, the event-driven molecular-dynamics (MD) algorithm presented in Sec. 2.2 can be employed to equilibrate the obstacles-to-be.

The actual challenge is the following: Given then the positions of the N_m obstacles, permissible positions need to be found for the N_f particles that during the MD run will be allowed to move. Unfortunately, to this respect, multiple complications arise in HS-QA systems, the three most important of which are:

- (i) an overlap between any two particles is strictly forbidden,
- (ii) in the resulting configuration, the matrix particles have to be located at precisely-specified positions, and
- (iii) the entire volume not occupied by matrix particles has to be considered as locations for the fluid particles.

Note that under point (iii), it is explicitly required that the locations of the fluid particles *not* disregard “traps”, i.e., spatial regions that are entirely surrounded by matrix particles such that a fluid particle placed therein cannot escape (cf. Sec. 2.4). This is necessary in order to restore spatial isotropy and homogeneity of the fluid particle distribution upon statistically-mechanically averaging over matrix realizations (see Sec. 2.5).

2.3.2 Established packing methods

2.3.2.1 Simple methods

One of the simplest methods to arrange HS objects in a volume is trial-and-error insertion. For this, a fluid particle is placed at a random location, and is permanently assigned this position if it does not overlap a matrix particle or another fluid particle; otherwise, it is removed. This trial step is repeated until all particles have been assigned a location. Unfortunately, this procedure is inefficient even for setting up *bulk* high-density systems of hard spheres since it fails to reproduce the short-range correlations required to densely pack particles.

A more useful, straightforward method is to compress a low-density system. Compression can be carried out for instance by coordinate rescaling combined with random particle displacements, in the spirit of constant-pressure Monte Carlo (MC) [9]. In the context of QA systems, however, methods of this kind severely suffer from requirement (ii). Alternatively, sedimentation could be emulated in (temporarily) bounded systems, for instance by a simple MC algorithm (cf. Sec. 2.1.4) with displacements

biased in one direction, or MD with a constant unidirectional force combined with dissipation [76]. Unfortunately, in simple algorithmic realizations, neither compression nor sedimentation fulfill requirement (iii).

Another method commonly used to insert many particles into a system is to initially use an ordered (crystalline) configuration and to subsequently melt this arrangement. In our case, this method is inappropriate for two reasons: Firstly—concerning the porous medium—, prior to the melting, fluid particles would have to be removed from the crystal’s lattice sites in a way that no overlaps with the matrix particles occur. This would result in unacceptably low fluid densities for values of ϕ_m even as low as a few percent. Secondly, and more fundamentally, we are interested in *metastable* fluid configurations, i.e., systems under external conditions that thermodynamically favor the crystal phase. This means that melting the crystalline configuration would first require an excursion into the region of fluid stability, which clearly foils any possible advantage of an initial crystal arrangement.

2.3.2.2 Elaborate methods

Over the years, a number of elaborate algorithms have been devised to achieve very high densities in bulk disordered HS systems. Many of these methods have historically been associated with the notion of “random close packing” [12, 22, 243], which, however, has been shown to be ill-defined by Torquato et al. approximately a decade ago [261]. In a nutshell, these authors revealed that the densest-achievable ball packing depends on how much local order is allowed on average, with more order permitting packing fractions closer to that of the face-centered cubic (fcc) arrangement^{2.31}, $\phi_{\text{fcc}} = \pi/(3\sqrt{2}) \simeq 0.74$. For an overview over the vast range of phenomena and methods in random packing and jamming as well as its connection to glassy systems, we refer the interested reader to the excellent review by Torquato and Stillinger [260]. In the following, we shall highlight selected packing algorithms and briefly describe their shortcomings regarding the present context.

Historically, the first efficient algorithms for the amorphous packing of hard spheres were developed in the early 1970s by Bennett [21] and by Visscher and Bolsterli [268]. Although differing in technical details, both methods use the concept of *serial deposition*, where large amorphous aggregates are obtained by adding single particles one by one to an existing collection, placing them in “ideal” positions according to some “gravity”-like criterion. While both methods yield the desired dense amorphous packings for bulk systems, they are unfortunately not suited for setting up QA systems:

^{2.31}As an interesting side note, it has been conjectured more than four centuries ago by Johannes Kepler that the fcc arrangement represents the densest-possible arrangement of spheres in three dimensions. Despite the radiant appeal of his suggestion, until very recently the conjecture had escaped the enormous efforts to prove it. However, finally, methods of computer algebra seem to have paved the way; we recommend Footnote 2.32 as a starting point into this exciting topic, and Ref. [114] to the more technically inclined reader.

^{2.32}http://en.wikipedia.org/wiki/Kepler_conjecture

firstly, the essentially-arbitrary locations of the obstacles would lead to disproportionately large gaps between the deposited particles; secondly, the algorithms provide no means to adhere to requirement (iii).

The late 1980s and early 1990s saw the publication of the methods that to the present day are the most efficient ones for generating amorphous packings. In 1985, Jodrey and Tory introduced their “overlap-elimination” algorithm [130] in which particles are initially placed at random locations and thus overlap. Subsequently, in each step of the algorithm, the largest overlap is removed by spreading apart the corresponding particles, which clearly may create new overlaps. However, by simultaneously reducing the diameter of the particles, eventually an overlap-free state of very high density can be achieved. In the context of QA systems, while it might be possible to somehow adapt this method to requirement (ii), requirement (iii) is clearly not fulfilled, which rules out the use of this method.

In 1990, Lubachevsky and Stillinger presented their growth algorithm [177], which is not only capable of producing densely-packed *disordered* hard spheres but also of generating *partially-ordered* (“polycrystalline”) samples. It is this very feature that eventually allowed Torquato et al. to investigate the role of order in the packing of spheres [261]. The algorithm itself is essentially an event-driven MD algorithm [176], much like the one presented in Sec. 2.2, with the slight adaption that the diameter of the spheres is increased over time. While this may seem like a significant complication, a linear dependence of the diameter upon the simulation time is easily included in Eq. (2.16) since it leaves the structure of that equation unchanged [176, 177]. However, despite all its merits, the algorithm cannot be used to generate initial QA configurations since it cannot account for requirement (iii).

2.3.3 Custom method

Since in this work, we are interested in QA systems with a slowly-evolving fluid, an algorithm for generating systems with densely-packed fluid particles is indispensable. Considering the complications described in Sec. 2.3.2, we decided to devise a custom method for this task. By coincidence, the algorithm resulting from our efforts bears similarities to that employed by Chang et al. [45]; however, our method was designed independently and features more thorough optimization.

Our algorithm consists of the following three steps:

- (1) Prior to the insertion of a fluid particle, its “real” diameter (the diameter to be used during MD simulations) is reduced to some minute value (“deflated”). This greatly increases the probability of trial-and-error insertion, and allows all particles to be simultaneously present in the system throughout the search of a valid QA configuration.

- (2) In order to find such a valid configuration, a simple Metropolis-MC algorithm [190] is employed using the HS potential (see below). Obviously, using this algorithm with the minute diameters is meaningless; therefore, at the beginning of each sweep—one displacement attempt for each fluid particle in serial order—the diameter of each deflated particle is increased (“inflated”) to the maximum value possible without overlapping another particle. If for a particle this maximum value is greater than its real diameter, then inflating is done for this particular particle and it is assigned its real diameter (“fully inflated”).
- (3) A disconnected void (cf. Sec. 2.4) may be filled with (almost) any number of deflated particles; however, the particular void may be too small to accommodate all of these particles if they were fully inflated. To remedy this problem, before certain sweeps, all particles that are not yet fully inflated are removed from the simulation box and then redeflated and reinserted according to step (1). The number of sweeps separating two such removal-and-reinsertion procedures is gradually increased so as to allow for any number of sweeps to fully inflate a particle while still quickly “draining” crammed voids.

Obviously, as soon as all particles are fully inflated, the setup routine is complete and a configuration fulfilling the QA requirements has been found. Subsequently, the N_f fluid particles are assigned random velocities \mathbf{v}_i , which are then normalized to yield $k_B T = 2/(3N_f) \sum_{i=1}^{N_f} m_i |\mathbf{v}_i|^2 / 2 = 1$ (compare Footnote 1.11). In conjunction with the numerical value $m_i = 1$, this procedure results in the \mathbf{v}_i being close to unity, which is computationally advantageous. Note, however, that the actual value of the temperature T is insignificant if the reduced time unit τ , as defined in Sec. 2.5.1.6, is used to express numerical time values. In an ensuing MD run, the velocities quickly approach a Maxwellian velocity distribution even when initially being assigned at random [9].

The MC procedure in step (2) is extremely simple. The particles serially undergo individual trial moves to a random position in a small cubic neighborhood around their current location. Dealing with hard spheres, the acceptance criterion for a trial move is particularly simple: it is rejected if (and only if) it results in an overlap of two particles. Over the course of the MC run, the trial neighborhood is resized such that the average acceptance rate stabilizes around 50%. Notably, we chose to perform the inflation process in step (2) *serially* along with the MC trials. This leads to the distribution of fluid particle diameters covering a wide range at all times during the iteration, except when close to the approach of an allowed configuration. Since this introduces a “fluctuating polydispersity”, we found the generation of ordered states at high densities to be strongly suppressed even for bulk monodisperse systems.

In Appendix A.3.3, we present a brief systematic investigation of the custom initial-state algorithm presented in this section, with particular focus on its performance and

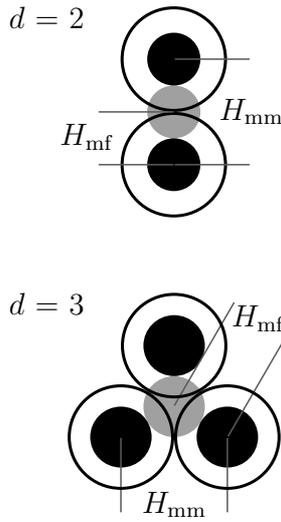


Figure 2.15: Hard-core range required for trapping. Gray disks: fluid particles. Black circles and disks: matrix particles. A concentric circle and disk represent the excluded spaces for different interactions of one matrix particle. Black circles: excluded space of the matrix–matrix interaction. Black and gray disks: excluded space of the matrix–fluid interaction. For $d = 2$, the fluid particle is located halfway between the two matrix particles, which immediately entails $H_{mf} = H_{mm}/2$ if the gray disk is to touch the black disks. For $d = 3$, the fluid particle rests at the center of the circum- and incircle of the equilateral triangle prescribed by the centers of the matrix particles, which means that $H_{mf} = (2/3) \times \cos(30^\circ) \times H_{mm} = H_{mm}/\sqrt{3}$ if touching is desired.

on its applicability resulting from the latter.

2.4 Geometry of the accessible volume

In this section, we shall be concerned with the accessible volume in a quenched-annealed (QA) mixture, i.e., with the volume that the matrix particles leave for the fluid particles to move in. Comprehensive information about the accessible volume allows to separate the fluid component of a QA system into two classes: “free” particles that can move infinitely far away from their initial position, and “trapped” particles that cannot. The superordinate idea is to evaluate the characteristics of the fluid separately for these two classes so as to extract more information about the features of the QA system. This section will describe the origin of the distinction in free and trapped particles, and elaborate on a geometric method to pinpoint and quantify this phenomenon. A detailed proof for the mathematical validity of this method is provided in Appendix A.1.

2.4.1 Traps and the percolating void

Central to the aim of identifying free and trapped particles is the notion of “traps” formed by the matrix particles, i.e., spatial regions that a fluid particle is rigorously confined to if placed therein. Regardless of the system’s dimensionality d , three conditions suffice for traps to be present in an infinitely-large QA system:

- (i) $H_{mf} > 0$, where H_{mf} is the hard-core range of the matrix–fluid interaction as in Sec. 1.1.3,
- (ii) $H_{mf} > x_d H_{mm}$, where x_d is a dimensionality-dependent scalar and H_{mm} is the hard-core range of the matrix–matrix interaction as in Sec. 1.1.3, and

(iii) $\phi_m > 0$, where ϕ_m is the density of the quenched particles.

Conditions (i) and (ii) ensure that matrix particles *can* form extended contiguous regions in which a fluid particle has infinite potential energy. In this, condition (ii) warrants that d matrix particles can be arranged close enough together that no fluid particle can cross “between” them. As can be seen from Fig. 2.15, in two and three dimensions, it is $x_2 = 1/2$ and $x_3 = 1/\sqrt{3}$, respectively; the choice $H_{mf} = H_{mm}$ used throughout this work (cf. Sec. 2.2.1) therefore clearly fulfills condition (ii). Condition (iii) ensures that some matrix particles actually *do* arrange according to Fig. 2.15; this is achieved through the essentially-random distribution of the matrix particles, which in combination with the precondition of an infinitely-large system entails that any allowed local arrangement of matrix particles is actually realized.

If conditions (i)–(iii) hold, then there exist traps in a QA system. A specific trap is defined by a set of at least $d+1$ matrix particles that form a closed $(d-1)$ -dimensional hypersurface (without holes) at which a fluid particle’s potential energy diverges. Figures 2.16 and 2.17 illustrate the nature of such traps in QA systems of d -dimensional hard hyperspheres. For simplicity, in these figures and the pertinent description, we specialize to $d = 2$ and the corresponding terminology; the description, however, is easily generalized to three dimensions as used in this work’s simulations (cf. Sec. 2.2)^{2.33}. Figures 2.16 and 2.17 show the same sample QA system in which $N_m = 50$ quenched particles are indicated by black disks. The depicted system is subject to periodic boundary conditions (PBCs), with the unique part being outlined by the dark gray square and the elements outside this square representing regularly displaced copies of elements inside the square (cf. Sec. 2.2.2.1).

Figure 2.16 shows $N_f = 63$ fluid particles (gray and blue disks) inserted into the matrix particle configuration. As in all parts of this work, all particles are taken to be additive hard spheres, i.e., each particle i is associated with merely a single radius R_i (see Sec. 1.1.3). Moreover, all particles are chosen to be identical, meaning that we can write $R_i = R$. In Fig. 2.16, the quantity R serves as the radius of the disks, which entails an area fraction of the matrix and of the fluid particles of $\phi_m \simeq 0.25$ and $\phi_f \simeq 0.32$, respectively. Notably, ϕ_m is almost an order of magnitude smaller than the area fraction of disks at hexagonal packing^{2.34}, $\phi_{hp} = \pi/\sqrt{12} \simeq 0.91$. Nevertheless, it is evident that numerous pairs of matrix particles in Fig. 2.15 are arranged such that no fluid particle can cross between them.

It is in principle no problem to identify *all* noncrossable pairs of matrix particles in a two-dimensional QA system; this can be accomplished by simply checking each possible combination of two matrix particles for whether it can be passed by a fluid particle. (Strictly speaking, this brute-force method requires to check $N_m(N_m-1)/2$ pairs; however, when proceeding visually—as we do here—, most pairs are easily dismissed as crossable.) In Fig. 2.15, noncrossable pairs are indicated by red lines that

^{2.33}For example, areas in two dimensions correspond to volumes in three dimensions, and the counterpart of line segments in $d = 2$ is triangles in $d = 3$.

^{2.34}http://en.wikipedia.org/wiki/Circle_packing

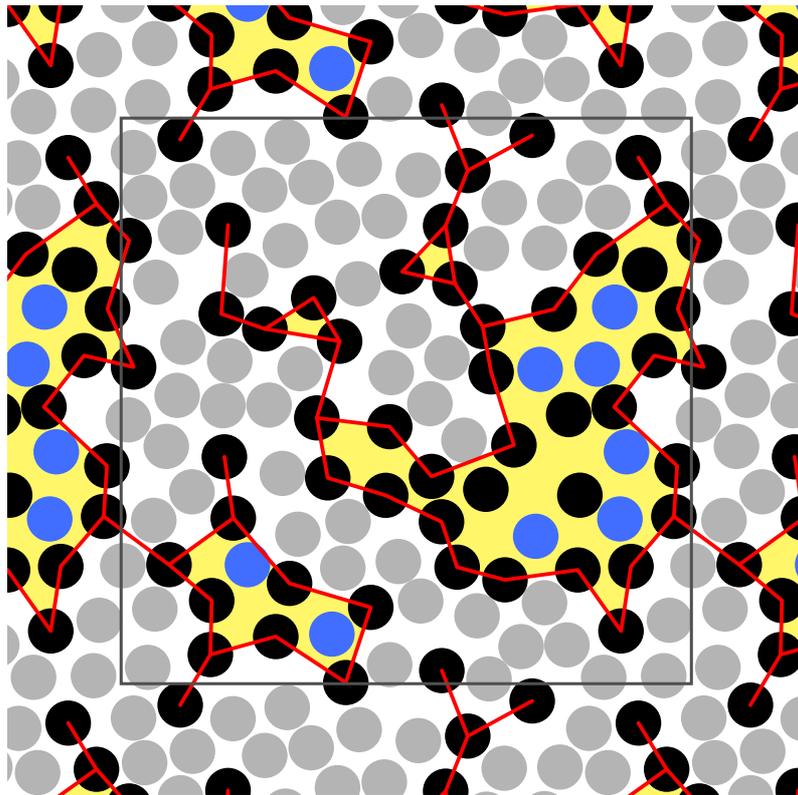


Figure 2.16: Trapped and free particles in quenched-annealed systems. Snapshot of a two-dimensional QA system of equal hard disks under PBCs. The 50 matrix and 63 fluid particles fill a combined area fraction $\phi = 0.57$. Black disks: matrix particles with radius R ; gray [blue] disks: free [trapped] fluid particles with radius R . Red lines connect matrix particle pairs that cannot be crossed by fluid particles; red lines that would run across yellow regions are suppressed. White: single, infinitely-large area; yellow: areas of finite size.

connect the respective matrix particles. The figure shows a peculiar selection of such lines, namely those that separate the white area from the yellow ones. The distinction between the differently-colored areas is the following: while numerous red lines are omitted that run across the yellow areas, *no* such lines can be found within the white area. Therefore, the white area is infinitely large (remembering the PBCs) and the yellow areas are of finite size (they are entirely bounded by red lines). This means that the latter must contain all traps of the system, which in turn allows to separate the particles according to the area type that they reside in: the gray fluid particles, which populate the white area, are “free” in that they may move infinitely far away from their initial position; the blue fluid particles, on the other hand, reside in the finite-sized yellow areas and are therefore “trapped”.

Unfortunately, the yellow and white areas give a somewhat misleading impression of the space available for configurations of the fluid particles. Consider the following: suppose we are given an arrangement of matrix particles and we want to place a single

fluid particle somewhere inside this arrangement. In this case, a useful definition of the “accessible volume” would be to describe the probability that a randomly-chosen position of the fluid particle does not entail an overlap with a matrix particle. We can adjust the definition of the accessible volume to suit this requirement by exploiting one of the basic properties of the hard-sphere (HS) potential (described in Sec. 1.1.3.2), namely that we may assign more than one radius to a particle. For instance, we may associate each matrix particle with a matrix–matrix and a matrix–fluid radius, $R_m^{(m)}$ and $R_m^{(f)}$, and each fluid particle with a fluid–matrix and a fluid–fluid radius, $R_f^{(m)}$ and $R_f^{(f)}$. For inserting a single fluid particle into a matrix configuration, only the fluid–matrix interaction with HS distance $H_{fm} = R_m^{(f)} + R_f^{(m)}$ is relevant. Since the fluid and the matrix particles effectively represent “distinct species”, we may for instance define

$$\begin{aligned} \tilde{R}_m^{(f)} &= R_m^{(f)} + R_f^{(m)} = H_{fm} = 2R & \text{and} \\ \tilde{R}_f^{(m)} &= R_f^{(m)} - R_f^{(m)} = 0 & , \end{aligned} \quad (2.32)$$

and replace $R_m^{(f)} \rightarrow \tilde{R}_m^{(f)}$ and $R_f^{(m)} \rightarrow \tilde{R}_f^{(m)}$ as described in Sec. 1.1.3.2. This is equivalent to considering the fluid particles as point particles in the fluid–matrix interaction, and to accordingly double the fluid–matrix radius of the matrix particles. As explained in Sec. 1.1.3.2, this redefinition does *not* alter the physics of the system. In the context of quenched disorder, this property of the HS interaction has first been exploited by Kim and Torquato in the context of their “cherry pit” model [146].

Figure 2.17 shows the same system as Fig. 2.16, with the radii changed according to Eq. (2.32), i.e., the radius of matrix particles is now twice as large as in Fig. 2.16 and the fluid particles are point-sized (and therefore not shown). The blue lines highlight that in the light-green area there exist paths along which the point fluid particles may move through the entire system. The connectivity of the accessible regions is now easily verified, and it is precisely the areas depicted in Fig. 2.17 that we shall henceforth refer to as “traps” (light red) and the “percolating void” (light green). In addition, by the term “void” we shall denote either one of the latter, i.e., a single trap or the percolating void. In the naming of the light-green area, the term “percolating” emphasizes the system-spanning nature; this important issue and its peculiarities in the presence of PBCs^{2.35} are elaborated on in Secs. 1.3.3.3 and 2.4.5.

From Fig. 2.17, one can see that the voids are significantly smaller than the yellow and white regions in Fig. 2.16. Specifically, if we designate the areas of the traps and of the percolating void in Fig. 2.17 and the yellow and the white areas in Fig. 2.16 as A_{trap} , A_{percol} , A_{yellow} , and A_{white} , then we see that

$$P_{\text{trap}} = \frac{A_{\text{trap}}}{A_{\text{trap}} + A_{\text{percol}}} < \frac{A_{\text{yellow}}}{A_{\text{yellow}} + A_{\text{white}}} = P_{\text{yellow}} \quad , \quad (2.33)$$

^{2.35}Notably, in the presence of PBCs, it is in principle possible that *multiple* percolating voids exist; this would be the case for instance if the matrix particles formed an infinitely-extended, impermeable flat sheet that (together with the PBCs) divided the system in an infinite number of distinct, percolating slabs.

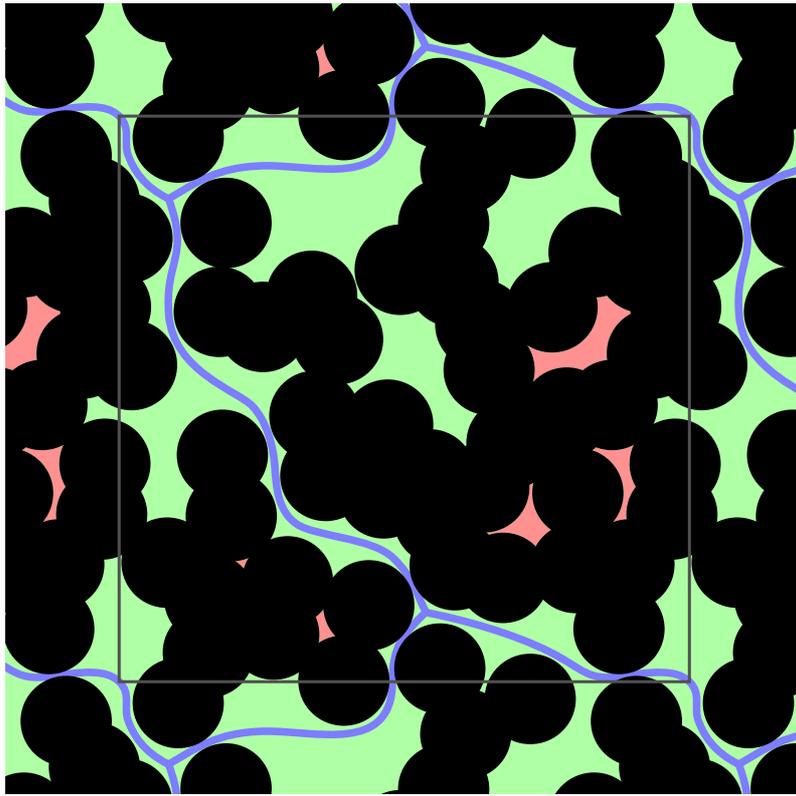


Figure 2.17: Accessible volume in quenched-annealed systems. Same system as in Fig. 2.16, with the matrix particles drawn larger and the fluid particles being of point size and therefore invisible. Black overlapping disks: matrix particles with diameter $2R$. Blue lines: example paths along which point-sized fluid particles can move infinitely far. Nonblack area: accessible volume; light green: percolating void; light red: traps.

where P_{trap} is the probability that a randomly-selected fluid particle in a QA mixture is trapped (cf. Sec. 2.5.4.2) whereas P_{yellow} bears no particular significance. For quantitative investigations, it is therefore more meaningful to consider the areas in Fig. 2.17 than those in Fig. 2.16. However, despite this difference in quantification, there is no fundamental difference between the areas in Figs. 2.16 and 2.17 from the point of view of distinguishing free and trapped fluid particles. In fact, it is even possible to partition the yellow area into distinct regions that correspond to the individual red areas in Fig. 2.17; this can be accomplished by reconsidering a suitable set of red lines that were neglected in Fig. 2.16.

Unfortunately, the methods used so far for identifying the different types of areas (or volumes) in Figs. 2.16 and 2.17 are not appropriate for direct application in computer programs. From Fig. 2.17, information was in fact extracted purely visually, with the picture serving a mere illustrative purpose. In Fig. 2.16, at least a clear protocol was used to identify noncrossable pairs of matrix particles, which at any rate constitutes a starting point. A weakness of this protocol is the brute-force method

involved; in d dimensions, such a method requires inspecting $\prod_{j=0}^{d-1} (N_m - j) / d!$ tuples of d particles, which may be a large number (for instance, $N_m = 10^4$ and $d = 3$ result in more than 10^{11} relevant tuples). This, however, is in principle a mere workload issue. The more fundamental deficiency of the protocol is its failure to associate *volumes* with noncrossable tuples of particles; such an association is of paramount importance to classify fluid particles on the basis of their location. Fortunately, there exists a geometric construction that provides not only an out-of-the-box mechanism to identify all relevant matrix-particle tuples (and thereby solves the workload problem) but that also naturally connects each of these tuples with a region in space. The mighty construction that we are referring to here—and that we will be concerned with in the following section and in Appendix A.1—is the “Delaunay tessellation”.

2.4.2 Definition of the Delaunay tessellation

In this section, we will introduce the geometric definition of the Delaunay tessellation. The first thing to notice about the name “Delaunay tessellation” probably is the term “tessellation”. This term describes a procedure to divide a given d -dimensional space into distinct, nonoverlapping regions such that these regions entirely fill (“tile”) the given space. If a space contains infinitely-many distinct points, then there obviously exist infinitely-many tessellations for this space; this is the case for essentially every continuous space. The Delaunay tessellation [64], specifically, was formally introduced by its namesake Boris Delaunay in 1934 in an essay honoring Georgy Voronoi^{2,36}. This particular tessellation partitions a d -dimensional space based on a set of distinct points that are arbitrarily distributed in that space. These points will be referred to as “Delaunay vertices” in this work, and the space-tiling regions resulting from these points shall be called “Delaunay cells”. Since in what is to follow, we will frequently make use of terms like “Delaunay x ” (where x is some object like a vertex, a cell, or the tessellation as a whole), for brevity we will mainly use the one-letter prefix “D-” in place of the honored scientist’s name (e.g., “D-cell” instead of “Delaunay cell”).

The geometric definition of the D-tessellation goes as follows. The tessellation groups the given D-vertices into several sets of $d+1$ D-vertices each, with every D-vertex belonging to at least d sets. The D-vertices in each set form the vertices of a d -dimensional hyperpolyhedron, and every such hyperpolyhedron constitutes one D-cell. Obviously, the D-vertex sets have to be chosen such that the resulting D-cells tile the given space and do not overlap. The fixed number of $d+1$ vertices entails that the

^{2,36}For two of reasons, introducing his tessellation in an article in the honor of Voronoi is a noble and notable act by Delaunay (Russian: Борис Николаевич Делонé). Firstly, it underlines the tight relationship between the Delaunay and Voronoi tessellations, as laid out in Appendix A.1. Secondly, it spotlights the strong *personal* bonds between Delaunay and Voronoi, the latter being the doctorate supervisor of the former. On a different and no less interesting note, Delaunay was not only an outstanding mathematician but also a renowned *mountaineer*—to a degree that a one of the highest peaks in the Altai mountain range is named in his honor^{2,37}.

^{2,37}<http://www.numbertheory.org/obituaries/RMS/delone/page3.html>

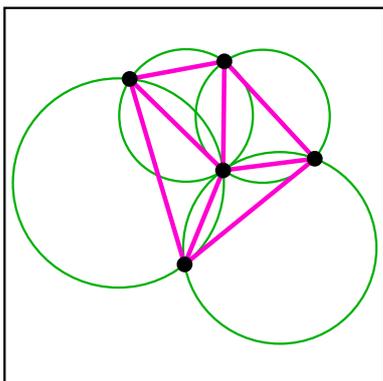


Figure 2.18: Sphères vides. Two-dimensional example for the empty-sphere property of a D-tessellation (without PBCs). Small black disks: given D-vertices. Thin green circles: “sphères vides”, defined by having no D-vertices on the inside and three D-vertices on the rim. Thick pink lines: faces of the D-cells that result from the empty circles. The configuration of D-vertices is the same as in Figs. A.1(d) and A.4.

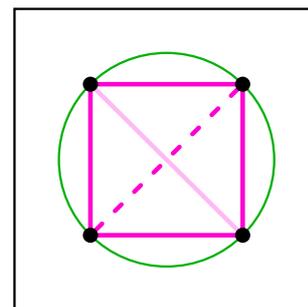
topology of each D-cell is the one of a “ d -simplex”, the simplest-possible d -hyperpolyhedron^{2.38}. In two dimensions, for instance, the tessellation consists solely of *triangles*, whereas for $d = 3$, all simplices are of *tetrahedral* topology.

Clearly, in the general case, no unique tessellation results from the mere requirement that all D-cells take the form of a d -simplex. This ambiguity can be alleviated by imposing additional constraints, for which there exist many different possibilities. The D-tessellation, specifically, adds a condition which makes reference to the fact that the $d+1$ D-vertices of a d -simplex always define a d -dimensional hypersphere. The “Delaunay property”—the hallmark of the D-tessellation—then dictates that a particular d -simplex constitutes a D-cell if (and only if) its associated d -hypersphere is *devoid of other D-vertices* [64, 212]. Writing in French and being concerned with the particular case of three dimensions, Delaunay used the name “sphères vides”—empty spheres—in his original work to refer to this property. Figure 2.18 provides a two-dimensional illustration of this peculiar property (in the absence of PBCs); in the figure the sphères vides—which in $d = 2$ reduce to circles—are outlined by thin green lines, and the corresponding D-cells are bordered by thick pink lines. It can easily be shown from the Voronoi tessellation (a geometric construction closely related to the Delaunay tessellation) that irrespective of the particular distribution of D-vertices, there always exists a tessellation in which all cells fulfill the Delaunay property (see Refs. [64, 212] and Appendix A.1).

However, the sphère-vide criterion leaves some narrow room for ambiguity. Specifically, the surface of a particular sphère vide may host $c > d+1$ D-vertices; this phenomenon is illustrated in Figure 2.19. Since by definition, a D-cell is associated with precisely $d+1$ D-vertices, the c D-vertices then have to be divided in subsets of $d+1$ D-vertices (with each D-vertex belonging to multiple subsets) to represent D-cells. In the absence of additional criteria, this subdivision is arbitrary—the only constraint being that the D-cells do not overlap and entirely cover the d -hyperpolyhedron defined

^{2.38}We found the majority of authors to use the term “simplices” when referring to the objects that we call “Delaunay cells” [70, 229, 260]. However, for symmetry with the term “Voronoi cell” (cf. Appendix A.1), we decided to retain the name “Delaunay cell”.

Figure 2.19: Degenerate Delaunay tessellations. Example of a degenerate two-dimensional D-tessellation. Small black disks: given D-vertices. Green circle: sphère vide, hosting more than three D-vertices on its surface. Dark pink lines: incomplete D-tessellation. Faint solid and dark dashed pink lines: equivalent completions of the D-tessellation.



by the c D-vertices. This ambiguity renders the tessellation “degenerate”^{2.39}.

In the context of molecular simulations, it is important to note that the D-tessellation—in any dimensionality—is well defined also in the presence of PBCs (cf. Sec. 2.2.2.1). The reason for this is that each D-cell is defined by a “local neighborhood” of D-vertices. Problems, however, may arise when using the nearest-image convention (see Sec. 2.2.2.1 again) in conjunction with very small systems; this is due to the fact that in this case, the hypersphere defined by a set of $d+1$ D-vertices may occasionally be larger than the system, meaning that multiple images of the same particle may have to be considered when verifying the empty-sphere property. In a final note, as will be detailed in Sec. 2.4.4, the sphère-vide property is central not only to the definition of the D-tessellation but also to the efficient computation of this tessellation. In fact, the local nature of this property renders these computations so efficient that the aforementioned Voronoi tessellation is typically constructed not directly but instead indirectly from its Delaunay counterpart [15].

2.4.3 Characterization of the accessible volume

We will now elaborate on the issue of how the D-tessellation can be used to characterize the accessible volume in QA systems. In this, the section at hand will focus solely on the aspect of the *procedures* that our method encompasses; the *validity* of the method will be discussed separately in Appendix A.1. The method is inspired by a number of authors, whose various works—some of which are as old as two decades [39, 48, 107,

^{2.39}Strictly speaking, in probabilistic distributions of D-vertices in continuous spaces, there do not exist sphères vides with more than $d+1$ D-vertices on their surface. For this effect to occur in such distributions, the space must consist of only a *finite* number of distinct spatial points (i.e., it must be a lattice space). Notably, this is effectively the case for every space represented in a computer’s memory. However, apart from the distribution of D-vertices, the probability for degeneracies to occur also depends on the number of distinct spatial points. Since the 64-bit arithmetics used in this work provides at least $(2^{53})^3 = \mathcal{O}(10^{48})$ distinct “lattice points” (53 being the number of bits used to represent the significand of a 64-bit floating-point variable^{2.40}), we can for all practical purposes rule out degenerate D-tessellations in this work.

^{2.40}http://en.wikipedia.org/wiki/Floating_point

229, 246, 251, 252]—aim at explaining phenomena related to those investigated in this work^{2.41}.

Our method starts off by mapping the structure of the accessible volume onto a *network* of “sites” and “bonds” (cf. Sec. 1.3.3.3 and Appendix A.1). From the D-tessellation of a given d -dimensional arrangement of matrix particles, this network is obtained by the following two steps:

- (1) Each D-cell in the tessellation represents one site in the network; a site is “occupied” only if the corresponding D-cell is “accessible”. A D-cell is considered accessible if the center of its circumhypersphere is left accessible by the matrix particles that form the D-cell.
- (2) Each D-face in the tessellation represents one bond in the network; a bond is “connecting” only if the corresponding D-face is “crossable”. A D-face is considered crossable if the center of its circumhypersphere is left accessible by the matrix particles that form the D-face.

The network resulting from these steps shall henceforth be referred to as the “Delaunay network”. The similarity in the wording of steps (1) and (2) is intentional and serves to emphasize that the definition of an occupied element is the same for the sites and the bonds aspects. The difference between sites and bonds lies merely in the number of matrix particles that define a D-cell and a D-face, which is $d+1$ and d , respectively. Therefore, although all elements are part of the same d -dimensional tessellation, the circumhypersphere of a D-cell is d -dimensional whereas its D-face counterpart is only $(d-1)$ -dimensional. For instance, in $d = 3$ (the case most important to this work) the former and the latter are represented by spheres and circles, respectively. As alluded to in Sec. 2.4.2, circumhyperspheres of D-cells and D-faces arise naturally when constructing D-tessellations; therefore, details on how to actually compute these circumspheres will be supplied in Sec. 2.4.4.2. Without going into detail here, we note that the Delaunay network is much akin to the Voronoi network introduced in Appendix A.1.1.2 and depicted in Fig. A.3.

Subsequent to establishing the D-network, we perform a *cluster analysis* of the network’s sites, i.e., we identify groups of sites for which every member site is connected by a bond to at least one of the other member sites. The algorithm used to perform this grouping is described in detail in Sec. 2.4.5. In the resultant cluster picture, each site group corresponds to precisely one of the voids present in the system. In view of the fact that there may exist an infinitely-large site group (one that spans the entire space), this procedure is tantamount to a *percolation analysis* (cf. Sec. 1.3.3.3). The percolation aspect will receive particular attention in Sec. 2.4.5 since the presence of PBCs has significant repercussions on this issue.

^{2.41}Notably, the authors of Refs. [39, 48, 107, 229, 246, 251, 252] consider a range of different physical systems and circumstances, not all of which feature the notion of “fluid” and “matrix” particles. Instead, these authors are interested for instance in whether a mobile test particle can assume a position in a particular configuration of other mobile particles.

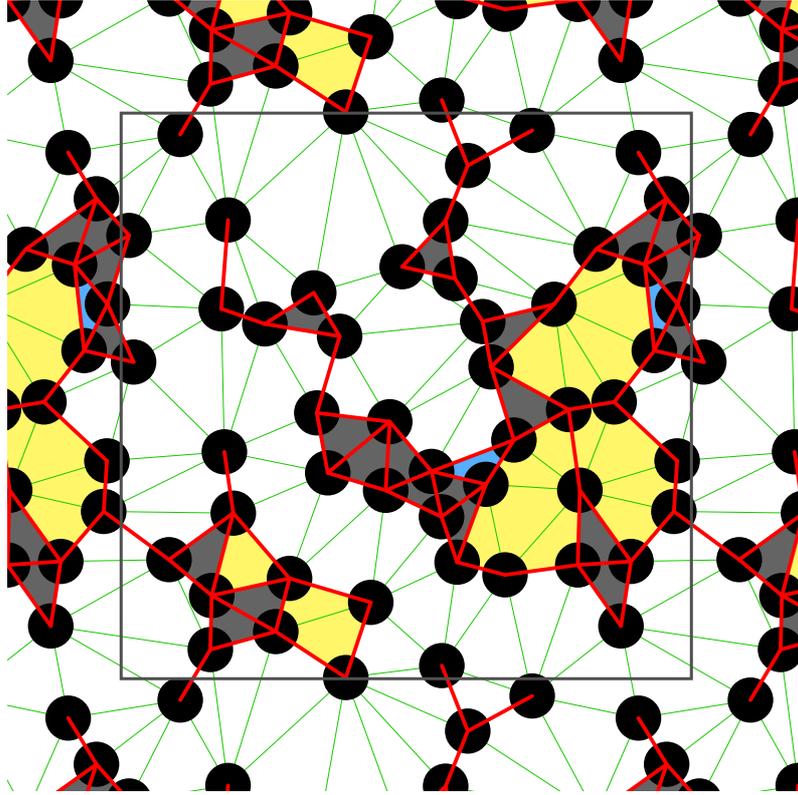


Figure 2.20: Delaunay network. D-network representation of the same system as in Figs. 2.16 and 2.17. Black disks: matrix particles of radius R . Green [red] lines: [non]crossable D-faces. White areas: D-cells that belong to the percolating void. Yellow areas: D-cells that belong to a trap. Gray areas: inaccessible D-cells. Blue areas: D-cells that do not contain accessible volume but are deemed “accessible”. The accessibility of D-cells and the crossability of D-faces are determined according to steps (1) and (2) specified in the beginning of Sec. 2.4.3.

Finally, we need to determine within which void a random accessible point is located. The procedure to achieve this consists of two straightforward steps:

- (I) Given some accessible point P_i , determine the D-cell DC_j that P_i is located in.
- (II) Having identified DC_j , determine the site group and thus the void that DC_j is part of.

This procedure serves two important purposes: (a) identifying whether a given fluid particle is free or trapped, and (b) estimating the amount of accessible volume pertaining to a void. Since the latter involves a fair share of additional methodology, the computational details of how to associate a location with a void will be described in Sec. 2.4.6 [the section elaborating on issue (b)].

As an illustration of the method presented in this section, Fig. 2.20 depicts the D-tessellation of the same arrangement of matrix particles as Figs. 2.16, 2.17, and A.3.

The figure highlights features of the void characterization resulting from the tessellation, and it is not a coincidence that Figs. 2.16 and 2.20 look quite alike. In both figures, the fluid and matrix particles are of the same radius R , the only difference between the figures being that Fig. 2.20 shows exclusively the matrix particles (black disks). Also in both figures the thick red lines indicate the relevant ones among the noncrossable pairs of matrix particles. Taking into account that Fig. 2.16 suppresses red lines that cross yellow areas, there obviously is the desired one-to-one correspondence of red lines between the figures. Further, both figures highlight the areas that are entirely enclosed by red lines. However, whereas Fig. 2.16 uses a uniform yellow color for this, Fig. 2.20 employs different colors to encode different types of D-cells: yellow D-cells are accessible [according to step (1) in the beginning of this section] and actually *do* contain accessible space, blue D-cells are accessible but do *not* contain accessible space, and gray D-cells are inaccessible. Clearly, taken together, the nonwhite D-cells in Fig. 2.20 precisely amount to the yellow area in Fig. 2.16. Further, Fig. 2.20 confirms the notion established in Figs. 2.17 and A.3 that the system contains five traps. Note, however, that the distinction between yellow and blue D-cells serves only an illustrative purpose—the D-tessellation provides no direct means to identify whether a particular D-cell does actually contain accessible volume (see Sec. 2.4.6 and Ref. [229]). Without this distinction, the D-network equates the yellow and blue cells and therefore accounts for seven traps instead of five, two of which then are of course devoid of accessible space. As a matter of fact, this difference reflects precisely the disparity between the Voronoi and the Delaunay networks (see Appendix A.1.2.2). The green lines, finally, complete the D-network in that they symbolize the crossable D-faces.

In summary, the D-tessellation provides a handle to describing all geometric features of the accessible volume in HS-QA systems that are of interest to this work. In particular, the tessellation provides an efficient means to identify the relevant noncrossable tuples of matrix particles—without even making reference to this issue—, and to assemble these tuples to noncrossable surfaces (cf. Sec. 2.4.1). Likewise, the issue of associating the tuples with volumes is naturally taken care of by the D-tessellation and its associated D-network.

Subsequent to this section, we will turn to the computational algorithms used in this work to perform the geometric analysis outlined in the above. First, in Sec. 2.4.4, we will take on the construction of D-tessellations; after that, we will turn to the aforementioned cluster and percolation analysis (Sec. 2.4.5). Finally, in Sec. 2.4.6, we shall introduce the method alluded to further above for estimating the amount of accessible volume contained in a void, with the aim of subsequently carrying out a quantitative analysis of these volumes.

2.4.4 Construction of Delaunay tessellations

In this section, we will be concerned with the issue of constructing a D-tessellation from a given set of D-vertices. For this, we first conduct a survey of packages and

algorithms that might be useful in this context. In conclusion of this survey we devised a custom method for constructing D-tessellations, which we will then elaborate on after introducing two auxiliary procedures that are vital to this custom method.

2.4.4.1 Available packages and known algorithms

The task of constructing a D-tessellation falls into the field of *computational geometry*; for an introduction to the many topics that this field encompasses see for instance the excellent book by Preparata and Shamos [212]. Given that the computation of tessellations is typically nontrivial and is for the most part not among the core skills of a physicist, the usual approach would be to make use of some publicly-available computational-geometry package. Therefore, as a first step, we conducted an extensive search among the multitude of available packages, with the goal of finding at least one that would fulfill all items from our limited set of requirements (see below). Unfortunately, as we report in the following, the search resulted in a *null result*.

- The most basic one of our requirements was that the package be able to compute *three*-dimensional D-tessellations. This fundamental prerequisite is failed by a great many packages, which instead provide merely the facility to compute *two*-dimensional D-tessellations. This is the case in particular for the following major packages:
 - GTS^{2.42}
 - Triangle^{2.43}
 - VRONI^{2.44}
- For two reasons, we clearly preferred a library solution over a stand-alone program: a library would (i) avoid having to externally convert large amounts of data and to temporarily store these data on a disk, and (ii) provide much better facilities for workflow integration. This requirement thwarted the use of the following prominent programs:
 - Qhull^{2.45}
 - Detri^{2.46}
- Any library coming into question has to lend itself to usage with the FORTRAN 90/95/2003 programming language since the latter is used extensively by the author for writing programs that require fast execution speed. Unfortunately, a number of available packages are coded in C++ and its object-oriented programming paradigm, which significantly differs from the procedural paradigm

^{2.42}<http://gts.sourceforge.net>

^{2.43}<http://www.cs.cmu.edu/~quake/triangle.html>

^{2.44}<http://www.cosy.sbg.ac.at/~held/projects/vroni/vroni.html>

^{2.45}<http://www.qhull.org>

^{2.46}<http://www.geom.uiuc.edu/software/cglist/GeomDir>

of FORTRAN. This presents a major obstacle to calling routines of these packages from within a FORTRAN program. Other packages are written in programming languages uncommon in the natural sciences such as PASCAL, which effects the availability of only very little documentation on how to integrate such libraries into FORTRAN programs. Shortcomings of these kinds forbade in particular the use of the following, highly-elaborate and acknowledged packages:

- CGAL^{2.47}
- FastGEO^{2.48}
- The package-to-be-used has to provide a reliable code basis. This in particular means that upon compilation, no warnings (much less errors) should be issued. Proper documentation of both the interfaces and the internal algorithms is of equal importance so as to avoid improper usage. Unfortunately, these criteria were not met to a satisfactory degree by the last one of the promising candidates, namely
 - GEOMPACK^{2.49}

Packages other than the named ones either failed more than one of our requirements, or failed the last one very badly (for instance by lacking documentation altogether).

In consequence, we found it preferential to implement a program for computing D-tessellations ourselves. Obviously, for this it is prudent to first survey the concepts that underlie existing solutions to the task at hand. Therefore, in the following we will briefly introduce the most prominent algorithms to construct D-tessellations, and motivate our choice among these. The interested reader may follow up on the presented methods in the excellent reviews by Aurenhammer^{2.50} [15] and by Su and Drysdale [250].

- ▶ **Flip** algorithms make use of a property inherent to pairs of D-cells that share a D-face: for such D-cells, the empty-sphere property (cf. Sec. 2.4.2) dictates that the sum of the internal angles at the two nonshared vertices be smaller than some upper bound. A flip algorithm uses this property to construct a D-tessellation as follows: First, some tessellation is created in which the cells are arbitrary nonoverlapping d -simplices whose vertices coincide with the given

^{2.47}<http://www.cgal.org>

^{2.48}<http://www.partow.net/projects/fastgeo/index.html>

^{2.49}http://people.sc.fsu.edu/~jburkardt/f_src/geompack/geompack.html

^{2.50}As a matter of fact, in his review, Aurenhammer focuses predominantly on the construction of *Voronoi* tessellations. However, in Appendix A.1.2.1, we will see that Voronoi and Delaunay tessellations actually encode the same set of information. It is therefore not surprising that a number of efficient methods have been established to obtain either one of the two tessellations from the respective other one. This effectively renders the Voronoi and Delaunay tessellations interchangeable from the point of view of their construction [15, 250].

D-vertices. Then, each pair of these cells is checked for whether the internal-angle sum adheres to the bound criterion, and if this is not the case, then the face that joins the cell pair is “flipped” such that the criterion holds. The latter step is repeated until all cells in the tessellation have been transformed to D-cells. Unfortunately, without major optimization, such algorithms may need as many as $\mathcal{O}(N^2)$ steps to generate a D-tessellation [15, 169, 250].

- ▶ **Divide-and-conquer** (D&C) algorithms are based on the fact that constructing a D-tessellation is much less expensive for small sets of D-vertices than it is for large ones. D&C algorithms exploit this fact by recursively dividing the set of vertices until each subset contains “few enough” vertices, with the subsets being grouped according to spatial regions that typically are slice-like but can in principle take any form. After constructing each subset’s D-tessellation, these tessellations are merged (“conquered”), which is the computationally expensive part of D&C algorithms and is nontrivial to optimize even for a simple slice geometry. In two dimensions and with heavy optimization, this type of algorithm has been shown to outperform all other D-tessellation methods (in absolute terms) and to require $\mathcal{O}(N \log \log N)$ operations (see Refs. [15, 69, 250] as well as Footnote 2.51).
- ▶ **Sweepline** algorithms effectively translate the construction of a d -dimensional D-tessellation to a $(d-1)$ -dimensional “dynamic” problem. Algorithms of this kind are reminiscent of event-driven molecular dynamics (cf. Sec. 2.2) in that a $(d-1)$ -dimensional hyperplane is “swept” through space and thereby generates “events” that are sequentially processed. The two relevant types of event are the encounter of a D-vertex, and the exiting of one of the spheres vides defined by each tuple of $d+1$ D-vertices for which it is yet unknown whether it defines a D-cell. Whereas sweepline algorithms in two dimensions have been shown to require $\mathcal{O}(N \log N)$ steps and are considered an efficient technique, their complexity (in both implementation and runtime) heavily increases with dimensionality, which usually constrains their use to two dimensions [15, 82, 250].
- ▶ **Insertion** algorithms incrementally construct a D-tessellation by first computing the tessellation of an (arbitrary) small subset of the given D-vertices—which is inexpensive—and by then “inserting” the remaining D-vertices one by one and updating the D-tessellation upon each insertion. The insertion is usually carried out by first creating new (non-Delaunay) cells that involve the inserted D-vertex, and by then transforming these new cells into D-cells (e.g., via flipping as described above). The transformation on average requires merely $\mathcal{O}(1)$ operations [250], which renders the creation of the new D-cells the bottleneck since for this, the D-cell in which the inserted D-vertex resides needs to be located. Without optimization, this requires $\mathcal{O}(N)$ operations per D-vertex, which, however, can be heavily optimized using bucketing techniques (like

^{2.51}<http://www.cs.cmu.edu/~quake/tripaper/triangle2.html>

for instance the one introduced in Sec. 2.2.2.2) to yield algorithms that overall scale as $\mathcal{O}(N)$ —at least if the D-vertices are randomly distributed [15, 250].

- **Addition** algorithms are similar to insertion algorithms in that they proceed incrementally; however, they require at most one invalid D-cell to be kept track of simultaneously. Addition algorithms involve two principal steps: (i) establishing one initial D-face, and (ii) finding new D-cells that are adjacent to already-known D-faces. Step (ii) can easily be accomplished by checking whether the circumhypersphere defined by some candidate D-vertex and the D-vertices of a known D-face is empty (cf. Sec. 2.4.2). Empirical data show that step (i) can be performed in $\mathcal{O}(1)$ steps, which leaves step (ii) as the critical factor: while the absence of optimization entails an overall $\mathcal{O}(N^2)$ behavior, bucketing techniques—like for the insertion algorithm case—reduce the effort to $\mathcal{O}(N)$ steps for randomly-distributed D-vertices [15, 70, 183, 250, 256].

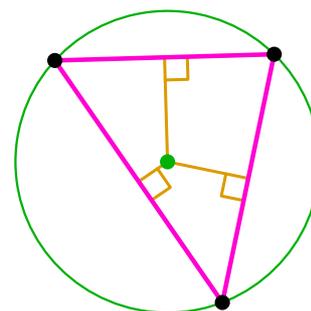
Although D&C algorithms were found to excel in execution speed, we deemed their principle to be too complex for implementation in this work. Similar reasons suggested to refrain from sweepline algorithms; since these algorithms moreover seem to be in practical use in two dimensions only, unexpected methodical problems could have occurred upon attempting an extension to three-dimensions. Flip algorithms suffer from the dimensionality problem, too, and additionally display poor performance, which actually leaves historical completeness as the only reason for continued interest in this method. In effect, this left the choice between the insertion and the addition algorithms, of which we opted for the latter due to its (arguably) slightly simpler concept.

Before we elaborate on this work’s implementation of the addition algorithm, we will discuss two auxiliary procedures that on the one hand form integral parts of the tessellation algorithm and on the other hand constitute solutions to problems interesting of their own right. The first one of these procedures, which we shall introduce in Sec. 2.4.4.2, is concerned with the construction of circumspheres; the second, detailed in Sec. 2.4.4.3, provides an efficient method for determining which particle is closest to a given point in space.

2.4.4.2 Auxiliary: Circumspheres and -circles

As laid out in Sec. 2.4.2, the D-tessellation is fundamentally based on the fact that each D-cell is associated with a circumsphere that is “empty”. Therefore, it is not surprising that most every D-tessellation algorithm invokes this property in one way or another. To phrase this condition more formally, suppose that there exists a distribution of points in d dimensions, and that $\{DX_k\}_j$ is a subset of $d+1$ of these points. Then, the Delaunay property states that the points $\{DX_k\}_j$ form a D-cell if (and only if) they reside on the surface of a d -dimensional hypersphere CS_j that is otherwise devoid of points from the distribution. In fact, checking the set $\{DX_k\}_j$ for the Delaunay property consists of two steps: (i) constructing the pertinent CS_j , and (ii) determining

Figure 2.21: Circumcircle construction. Simple construction of the circumcircle of a given triangle. Black disks: vertices of the triangle; pink lines: edges of the triangle. Orange lines: perpendicular bisectors of the triangle edges; empty orange squares: indicators for right angles. Green circle: circumcircle; green disk: center of the circumcircle.



whether CS_j is empty. Of these steps, in this section we will focus exclusively on the former. The procedure pertinent to the latter step will be the subject of Sec. 2.4.4.3.

Concerning the construction of circumhyperspheres, consider first the simplest case: two dimensions. In this case, $\{DX_k\}_j$ contains three points that form a triangle, and CS_j is the *circumcircle* of that triangle. A simple reasoning to obtain the latter is the following. Since by definition, the circumcircle center is equidistant from all triangle vertices, it must be located on the *perpendicular bisectors* of the triangle. As shown in Fig. 2.21, there are three such perpendicular bisectors—one for each of the triangle’s edges. Since every triangle has a circumcircle (a circle can in fact be defined by three points), the center of the circumcircle must be the point in which all of these bisectors intersect. This scheme is easily extended to d dimensions, namely by finding the $(d-1)$ -dimensional hyperplanes that perpendicularly bisect the $d+1$ edges of a d -dimensional simplex, and by locating the point in which all of these hyperplanes intersect. However, there exists a more elegant and computationally more efficient method due to Pedoe [209], which we will describe in the following.

For the **two-dimensional** case, we will recite the derivation of Pedoe’s method as presented in Footnote 2.52. There, the starting point is that the points of a circumcircle can be shown to conform to the equation

$$\begin{vmatrix} x^2 + y^2 & x & y & 1 \\ x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ x_3^2 + y_3^2 & x_3 & y_3 & 1 \end{vmatrix} = 0 \quad (2.34)$$

where x and y represent the components of the circumcircle’s points, and x_k and y_k (with $k \in \{1, 2, 3\}$) are the components of a point DX_k from the given vertex subset $\{DX_k\}_j$. Using Laplace expansion^{2.53}, the determinant on the left-hand side in Eq. (2.34) can then be expressed as

$$a(x^2 + y^2) + b_x x + b_y y + c = 0 \quad (2.35)$$

^{2.52}<http://mathworld.wolfram.com/Circumcircle.html>

^{2.53}http://en.wikipedia.org/wiki/Laplace_expansion

where

$$\begin{aligned}
 a &= + \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} & b_x &= - \begin{vmatrix} x_1^2 + y_1^2 & y_1 & 1 \\ x_2^2 + y_2^2 & y_2 & 1 \\ x_3^2 + y_3^2 & y_3 & 1 \end{vmatrix} \\
 c &= - \begin{vmatrix} x_1^2 + y_1^2 & x_1 & y_1 \\ x_2^2 + y_2^2 & x_2 & y_2 \\ x_3^2 + y_3^2 & x_3 & y_3 \end{vmatrix} & b_y &= + \begin{vmatrix} x_1^2 + y_1^2 & x_1 & 1 \\ x_2^2 + y_2^2 & x_2 & 1 \\ x_3^2 + y_3^2 & x_3 & 1 \end{vmatrix} .
 \end{aligned} \tag{2.36}$$

“Completing the squares” transforms Eq. (2.35) to read

$$a \left(x + \frac{b_x}{2a} \right)^2 + a \left(y + \frac{b_y}{2a} \right)^2 - \frac{b_x^2}{4a} - \frac{b_y^2}{4a} + c = 0 \tag{2.37}$$

which is of the form

$$(x - x_0)^2 + (y - y_0)^2 = r^2 , \tag{2.38}$$

meaning that Eq. (2.37) describes a circle for which the center and the radius are

$$(x_0, y_0) = -\frac{1}{2a} (b_x, b_y) \quad \text{and} \quad r = \frac{1}{2|a|} \sqrt{b_x^2 + b_y^2 - 4ac} . \tag{2.39}$$

In **three dimensions**, the procedure is entirely analogous (see Footnote 2.54). In this case, a circumsphere is described by the equation

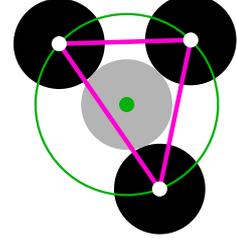
$$\begin{vmatrix} x^2 + y^2 & x & y & z & 1 \\ x_1^2 + y_1^2 & x_1 & y_1 & z_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & z_2 & 1 \\ x_3^2 + y_3^2 & x_3 & y_3 & z_3 & 1 \\ x_4^2 + y_4^2 & x_4 & y_4 & z_4 & 1 \end{vmatrix} = 0 \tag{2.40}$$

with the symbols as in Eq. (2.34), except that now each vector obviously has a z component and that we need four vertices to define a circumsphere. Like for Eq. (2.34), the determinant in Eq. (2.40) can be expanded^{2.53}, reading then

$$a(x^2 + y^2 + z^2) + b_x x + b_y y + b_z z + c = 0 \tag{2.41}$$

^{2.54}<http://mathworld.wolfram.com/Circumsphere.html>

Figure 2.22: Crossing criterion. Criterion for a fluid particle to cross a D-face. Black disks: matrix particles; gray disk: fluid particle. Pink lines: edges of the D-face. Green circle: circumcircle of the D-face; green dot: center of the circumcircle and of the fluid particle. White disks: matrix particle centers.



in which, straightforwardly, the subdeterminants are defined by

$$\begin{aligned}
 a &= + \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix} & b_x &= - \begin{vmatrix} x_1^2 + y_1^2 & y_1 & z_1 & 1 \\ x_2^2 + y_2^2 & y_2 & z_2 & 1 \\ x_3^2 + y_3^2 & y_3 & z_3 & 1 \\ x_4^2 + y_4^2 & y_4 & z_4 & 1 \end{vmatrix} \\
 & & b_y &= + \begin{vmatrix} x_1^2 + y_1^2 & x_1 & z_1 & 1 \\ x_2^2 + y_2^2 & x_2 & z_2 & 1 \\ x_3^2 + y_3^2 & x_3 & z_3 & 1 \\ x_4^2 + y_4^2 & x_4 & z_4 & 1 \end{vmatrix} & (2.42) \\
 c &= + \begin{vmatrix} x_1^2 + y_1^2 & x_1 & y_1 & z_1 \\ x_2^2 + y_2^2 & x_2 & y_2 & z_2 \\ x_3^2 + y_3^2 & x_3 & y_3 & z_3 \\ x_4^2 + y_4^2 & x_4 & y_4 & z_4 \end{vmatrix} & b_z &= - \begin{vmatrix} x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ x_3^2 + y_3^2 & x_3 & y_3 & 1 \\ x_4^2 + y_4^2 & x_4 & y_4 & 1 \end{vmatrix} .
 \end{aligned}$$

Then, like before, Eq. (2.41) can be rearranged to read

$$a \left(x + \frac{b_x}{2a} \right)^2 + a \left(y + \frac{b_y}{2a} \right)^2 - a \left(z + \frac{b_z}{2a} \right)^2 - \frac{b_x^2}{4a} - \frac{b_y^2}{4a} + \frac{b_z^2}{4a} + c = 0 \quad (2.43)$$

which in analogy to Eq. (2.38) describes a sphere for which the center and the radius read

$$(x_0, y_0, z_0) = -\frac{1}{2a} (b_x, b_y, b_z) \quad \text{and} \quad r = \frac{1}{2|a|} \sqrt{b_x^2 + b_y^2 + b_z^2 - 4ac} \quad . \quad (2.44)$$

Therefore, in two [three] dimensions, the relevant information about CS_j can be obtained by simply evaluating the determinants in Eq. (2.36) [Eq. (2.42)] and by inserting the resulting scalars into Eq. (2.39) [Eq. (2.44)]. Considering the trivial modifications from two to three dimensions, it is easy to see that the same scheme could be applied in any dimensionality.

Although seemingly only a toy version of the three-dimensional case, computing circumcircles is more than just an academic exercise in the present context: as illustrated in Fig. 2.22, it is in fact the *radius of the circumcircle* that determines whether

a D-face is crossable or not. Specifically, some D-face is crossable if (and only if)^{2.55,2.56} its circumcircle radius is larger than the hard-core range of the matrix–fluid interaction (cf. Fig. 2.15). This is the criterion referred to in point (2) in the beginning of Appendix A.1.2.2.

Circumspheres, however, are no less important in that their radius provides the criterion for their associated D-cell to be accessible. This is of course determined in the same way as for D-faces, i.e., if a circumsphere’s radius is larger than the matrix–fluid interaction range, then the corresponding D-cell is accessible. This criterion is referred to in point (1) in the beginning of Appendix A.1.2.2.

2.4.4.3 Auxiliary: Nearest-neighbor search

Suppose, as before, that we are given some distribution $\{DX_i\}$ of N points in a d -dimensional space, and that we are moreover given some arbitrary reference point P . Then, it is a challenge interesting of its own to determine which of the $\{DX_i\}$ is the one closest to P —the “nearest neighbor”. One straightforward application of such knowledge is to subsequently measure the distance between P and its nearest neighbor, and to compare that distance with some given length. This clearly constitutes one method for determining whether a circumhypersphere around P is empty, and it even constitutes an efficient one if the circumhypersphere contains many points from $\{DX_i\}$ and/or many buckets (cf. Secs. 2.4.4.2 and 2.2.2.2).

An obvious approach to finding the nearest neighbor of P would be to simply determine the distance between P and *all* of the points $\{DX_i\}$. While this approach, unfortunately, is quite inefficient if N is large, fortunately a plethora of more efficient solutions to this problem has been devised. This is due in part to Knuth, who popularized this computational-geometry task under the label “post-office problem” [153, 212]. Essentially, the underlying challenge is that of a suitable *partitioning of space*, which

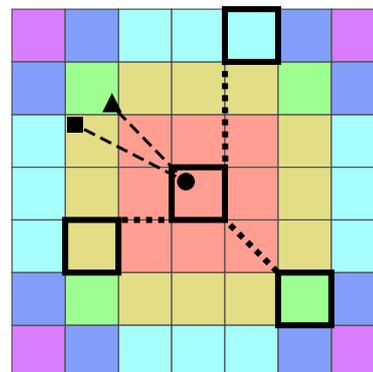
^{2.55}Notably, the circumcircle-evaluation scheme presented in this section operates on a triangle in *two-dimensional* space. In three-dimensional D-tessellations, however, the triangular D-faces are part of a *three-dimensional* space. Fortunately, it is possible to retain the two-dimensional evaluation scheme by introducing two additional steps: First, the D-face in question is *rotated* such that its normal vector is parallel to one of the coordinate axes. This coordinate axis is then neglected, and the circumcircle center of the D-face is obtained in a space spanned by the other two coordinate axes. Finally, the circumcircle center is reinterpreted as part of the three-dimensional space and subjected to the inverse of the previous rotation. The rotations involved in this procedure can be carried out for instance using Euler angles^{2.57}.

^{2.56}In fact, the circumcircle radius provides the criterion for a D-face to be crossable only in the case of a *single* matrix–fluid HS interaction. This approach would *not* be valid if instead each matrix particle interacted via a *distinct* HS range with the fluid particles. Instead, the criterion for crossability would then be given by one of the solutions to “Apollonius’s problem”, i.e., to the problem of finding the circle that is tangential to three other circles^{2.58}. In this work we did not pursue this case of a “polydisperse matrix” since it entails a host of other difficulties, starting from the MD algorithm and ending with the proper analysis of the inferred observables.

^{2.57}<http://mathworld.wolfram.com/EulerAngles.html>

^{2.58}<http://mathworld.wolfram.com/ApolloniusProblem.html>

Figure 2.23: Relative minimum-possible distances between buckets. Scheme for constructing a mask of relative minimum-possible distances (MPDs) between buckets. Buckets are constructed according to Sec. 2.2.2.2 and Fig. 2.8 therein. Squares of the same color denote MPD groups, i.e., groups of buckets that share the same relative MPD to the central bucket. Squares bordered by thick black lines highlight selected buckets; thick dotted black lines indicate the MPDs of these buckets. Black disk: example for the site P for which the nearest neighbor is to be found; filled black square and triangle: examples for points from the given point distribution. The thin black dashed lines illustrate that the disk–square distance is larger than the disk–triangle distance, despite the MPD of the yellow buckets being smaller than the MPD of the green buckets.



in many approaches is accomplished by virtue of one of the various space-partitioning tree structures known to the information sciences such as octrees^{2.59}, k -d-trees^{2.60}, and R-trees^{2.61}. Instead of such data trees, we will here exploit a structure already established for the optimization of this work’s event-driven MD algorithm: *bucket lists* (cf. Sec. 2.2.2.2).

While bucket lists—which in the context of space partitioning are also known as “locality-sensitive hashes”^{2.62}—are an obvious candidate for solving the nearest-neighbor problem, their use for this purpose is unfortunately not quite as simple as it might seem. To understand the pertinent difficulties, suppose that we have already searched a number of buckets in the vicinity of P without having found any point from $\{DX_i\}$, and that subsequently in bucket B we have found the point $DX_l \in \{DX_i\}$ at some distance $d_{l,P}$ away from P . Then, DX_l constitutes P ’s nearest neighbor only if we have previously *ruled out* that some other $DX_m \in \{DX_i\}$ exists for which $d_{m,P} < d_{l,P}$. This ruling out can only be guaranteed if prior to bucket B , we have searched *every* bucket containing a spatial point that is closer to P than is DX_l . One systematic way to achieve this would be to search the buckets in ascending order of the minimum-possible distance (MPD) between the anchor point P and any point within each bucket. However, upon altering the position of P , such an ordering would change—even if P remained within the same bucket. This means that the MPD ordering would have to be re-evaluated for every position of P , which would entirely overturn the advantage of bucket lists.

It is, however, well feasible to sort the buckets once and for all according to a similar order, namely in ascending order of the MPD between *any point in P ’s bucket*

^{2.59}<http://en.wikipedia.org/wiki/Octree>

^{2.60}<http://en.wikipedia.org/wiki/Kd-tree>

^{2.61}<http://en.wikipedia.org/wiki/R-tree>

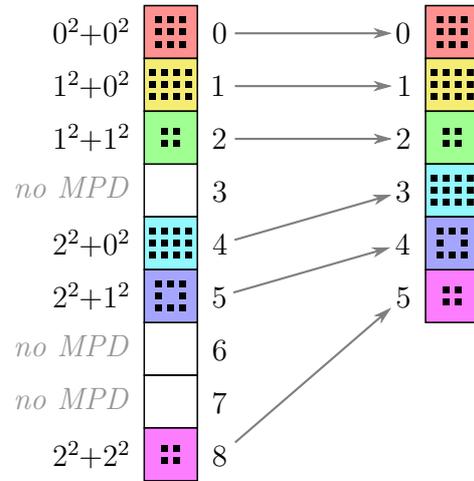
^{2.62}http://en.wikipedia.org/wiki/Locality_sensitive_hashing

and any point in each other bucket. Such an ordering can be established by once evaluating a “mask” of the *relative* MPDs between one “central” bucket and all other buckets (up to the maximum-possible distance), and by then every time translating this mask such that the central bucket is the one containing P . The two-dimensional version of such a mask is visualized in Fig. 2.23, wherein the central bucket is indicated by the central red square with thick black borders. For three selected buckets (yellow, green, and turquoise squares bordered by thick black lines) the MPD to the central bucket is highlighted by the thick black dotted lines; for those buckets, the respective MPDs are 1, $\sqrt{2}$, and 2 in units of bucket side lengths. As indicated by the colors of the squares, for each MPD there exist several buckets that share that distance. Therefore, if at some stage *any* bucket of such an “MPD group” is searched, then *all* buckets of the same MPD group have to be searched before considering to terminate the nearest-neighbor search. For instance, if the mask in Fig. 2.23 is to be used, then first all red buckets have to be checked, then (if no nearest neighbor has been found) all yellow ones, and then—one group after another—the green, turquoise, blue, and pink ones.

The criterion for terminating the nearest-neighbor search results from the following consideration. Suppose that some ongoing nearest-neighbor search has so far failed to discover a point from $\{DX_i\}$, and that the search has just finished processing some MPD group (say for instance the red one in Fig. 2.23). Then, in consequence of the above, the search has to continue by checking the next MPD group (here the yellow one) in its entirety. If in this MPD group the search encounters multiple points of $\{DX_i\}$, then obviously merely the one closest to P needs to be kept track of. Suppose that this closest point is the filled square in Fig. 2.23. Then, although the MPD groups are ordered by minimum-possible distance, there still may exist points from $\{DX_i\}$ that are *closer* to P than the square and are not yet checked by the search. This is the case for instance for the point indicated by the filled triangle in Fig. 2.23. This circumstance is possible due to the fact that most points in an MPD group are *farther* from P than the MPD of that MPD group, and that some points are even farther from P than the MPD of the next MPD group in the queue (here the green one). To accommodate for this circumstance, the criterion for terminating the nearest-neighbor finally reads as follows:

- (1) If the current MPD group does *not* contain points from $\{DX_i\}$, then proceed to the next MPD group.
- (2) If the current MPD group *does* contain points from $\{DX_i\}$, then find the point DX_c among these points that is closest to P . This spawns two sub-cases:
 - (2a) If DX_c is *farther* from P than the MPD of the next MPD group, then proceed to the next MPD group.
 - (2b) If DX_c is *closer* to P than the MPD of the next MPD group, then identify DX_c with DX_P (the nearest neighbor of P) and

Figure 2.24: Bin sorting. Schematic representation of the bin-sorting algorithm. A hash function computes an integer (a bin index) from each given value in order to assign that value to a bin. The hash function for the relative minimum-possible distances (MPDs) between buckets (cf. Fig. 2.23) is spelled out at the very left; the results of the hash function are the numbers 1 through 8. Large squares represent the bins of a pertinent array; each colored bin corresponds to a possible MPD, whereas the white bins have no corresponding MPD. Small filled black squares indicate the occupation of the bins resulting from Fig. 2.23, i.e., the number of buckets with the same relative MPD. Since some bin indices do not correspond to relative MPDs, the array is subsequently compressed.



terminate the search.

Of course, in case (2a), it is mandatory to retain DX_c as a candidate for DX_P when proceeding to the next MPD group. Notably, the additional checking of DX_c in cases (2a) and (2b) alleviates the problem that the MPD groups are constructed such that the buckets therein do not necessarily have the same minimum-possible distance to P .

Having established the order in which the buckets have to be searched, we now address the task of *arranging* the buckets in that order. Fortunately, for accomplishing this task, it is possible to use the exceptionally efficient and equally simple *bin-sorting*^{2.63} routine. This possibility is owed to two properties of the possible values of the MPDs:

- (i) the range of these values is subject to a definite lower and upper bound, and
- (ii) the values assume only a finite number of different values within that range.

Properties (i) and (ii) allow to set up an array in which all possible values are represented by a sequence of “bins” that are sorted according to a known rule. Each of the N values that are to be sorted (which may be much more than the number of *distinct* values) is then “dropped” into its corresponding bin; this requires merely $\mathcal{O}(N)$ operations if direct access to the bins is provided. The sorting is then established

^{2.63}The term “bin sorting” is probably less widespread than the equivalent term “bucket sorting”^{2.64}; however, since we reserved the buzzword “bucket” for the space partitioning in Sec. 2.2.2.2, we will use the former term to refer to the sorting method.

^{2.64}http://en.wikipedia.org/wiki/Bucket_sort

by subsequently visiting the bins in the constructed sorting order (usually a sequence from start to end) and “collecting” the values therefrom.

Since all relevant programming languages (such as `FORTRAN` or `C`) use consecutive integer numbers to indicate the elements of an array, a function has to be devised that computes such an integer from each of the values-to-be-sorted. It is moreover compulsory that the integers resulting from such a “hash function” obey the same ordering as the input values. In case of the MPDs, a sensible choice for the bin indices is the squared relative MPD in units of bucket edge lengths. In a d -dimensional system that harbors M buckets along one coordinate axis, this scheme yields 0 and $d[(M-1)/2]^2$ as the minimum and maximum bin indices, with the minimum nonzero difference being 1. Therefore, a pertinent array has to provide $d[(M-1)/2]^2+1$ bins. As an example for this hash function, Fig. 2.24 illustrates the integer values that result from the buckets in Fig. 2.23. On the upside, the integers clearly follow the same sequence as the MPDs, and each MPD is assigned a unique integer—just as required. On the downside, not every bin in the array corresponds to a possible MPD, which, however, is a common feature of hash functions and does not present a fundamental problem^{2.65}. It does, yet, suggest a subsequent step of purging the array from nonexistent MPD values, as is symbolized by the arrows in Fig. 2.24. Finally, it should be noted that each bin has to provide for the facility to accommodate *multiple* items, which then in our case form the MPD groups (cf. Fig. 2.23).

2.4.4.4 Custom construction algorithm

Having thus set the scene, we now turn to the actual algorithm used in this work for constructing a D-tessellation of a given set of points $\{DX_i\}$ in d dimensions. Inspired by the two-dimensional addition algorithm devised by Maus [183], our procedure consists of two fundamental steps:

- (1) establish at least one D-face from the final D-tessellation, and
- (2) iteratively find all D-faces that are part of less than two known D-cells, and construct the D-cells adjacent to those D-faces.

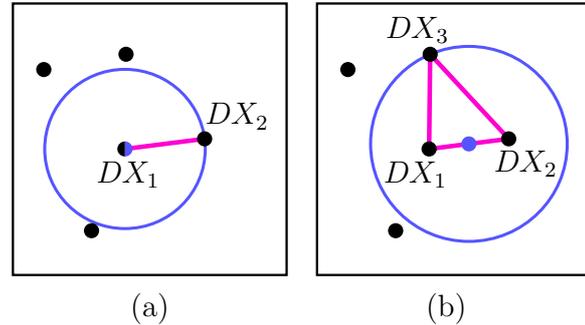
One common approach for accomplishing step (1) is to construct the *convex hull*^{2.66} of the $\{DX_i\}$, that is, to find the hyperpolyhedron with vertices from $\{DX_i\}$ that entirely encloses the $\{DX_i\}$ and is convex everywhere. As can be shown, the convex hull is a subset of the D-tessellation in the sense that each face of the convex hull is a D-face [15, 169, 183, 250]. While this approach is certainly valid, devising an efficient algorithm for finding the convex hull is a challenge of its own right, and is in fact of similar complexity (in both runtime and programming effort) as the superordinate task of constructing a D-tessellation. Fortunately, however, in this work, each distribution $\{DX_i\}$ is derived from a physical system—a QA system—, which leads to an

^{2.65}http://en.wikipedia.org/wiki/Hash_function

^{2.66}http://en.wikipedia.org/wiki/Convex_hull

Figure 2.25: The initial Delaunay cell.

Two-dimensional illustration of the procedure for finding a good candidate for the first D-cell. Black disks: D-vertices as in Figs. 2.18 and A.1(d); blue disks: barycenters, i.e., averaged locations of subsets of D-vertices. In panel (a), the barycenter coincides with DX_1 ; in panel (b), it is located halfway between DX_1 and DX_2 . Blue circles indicate the distance to each barycenter's next neighbor, disregarding D-vertices that define the barycenter. Light-pink lines: edges of the resulting candidate cell.



essentially-random distribution. Specifically, this means that the $\{DX_i\}$ do not assume some pathological regular pattern or concentrate in a small spatial region. This randomness facilitates not only the construction of convex hulls, but also opens the door to an even simpler approach that we shall detail in the following.

Suppose that we want to determine whether a D-cell is formed by some arbitrarily-chosen tuple of $d+1$ points from $\{DX_i\}$. Recalling the empty-sphere property (see Sec. 2.4.2), it is clear that doing so requires knowledge merely about the tuple in question and about the positions of the $\{DX_i\}$. Using the nearest neighbor-search technique (cf. Sec. 2.4.4.3), the empty-sphere check can be carried out within $\mathcal{O}(1)$ operations. Therefore, it is in principle possible to identify an initial D-face by simply inspecting random $(d+1)$ -tuples until a D-cell is encountered (which then consists of $d+1$ of the desired faces). However, a D-tessellation contains merely $\mathcal{O}(N)$ D-cells, while there are $\prod_{j=0}^d (N-j) = \mathcal{O}(N^{d+1})$ ways to randomly select $(d+1)$ -tuples from the $\{DX_i\}$. This means that by this method, on average a D-cell is discovered only after $\mathcal{O}(N^d)$ random trials, which renders “brute force” quite infeasible (for $N = 10^4$ and $d = 3$, for instance, $\mathcal{O}(10^{12})$ tuples would have to be inspected). Fortunately, in the case of the $\{DX_i\}$ being randomly distributed, this cost can be substantially reduced by a slight tweak to the method of choosing the members of the $(d+1)$ -tuples. Specifically, instead of naming the tuples’ members at random, consider the following procedure (illustrated in Fig. 2.25):

- (I) pick some point $DX_1 \in \{DX_i\}$ at random,
- (II) add the point $DX_2 \in \{DX_i\}$ closest to DX_1 ,
- (III) add the point $DX_3 \in \{DX_i\}$ closest to the barycenter of $\{DX_1, DX_2\}$,
- (IV) add the point $DX_4 \in \{DX_i\}$ closest to the barycenter of $\{DX_1, DX_2, DX_3\}$,

and so forth, depending on the dimensionality. Here, the “barycenter” of a set of points is simply the averaged location of these points (akin to the center of mass for equal point masses), and the “closest point” can be identified for instance using the nearest neighbor-search technique described in Sec. 2.4.4.3. Obviously, all of the points $\{DX_1, DX_2, DX_3, \dots\}$ have to be distinct, i.e., the points that define a barycenter have to be excluded from the associated nearest-neighbor search. Empirically, a $(d+1)$ -tuple

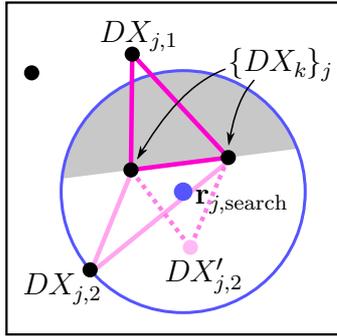


Figure 2.26: Adding Delaunay cells. Scheme inspired by Maus [183] for adding D-cells to the list of known D-cells. Black disks: D-vertices as in Figs. 2.18, 2.25, and A.1(d). Dark solid pink lines: known D-cell with vertices $DX_{j,1}$ and $\{DX_k\}_j$. Light-pink disk: hypothetical D-vertex $DX'_{j,2}$ that forms an equilateral triangle (dotted light-pink lines) with the vertices $\{DX_k\}_j$. Blue disk: circumcircle center of that equilateral triangle, used as center $\mathbf{r}_{j,\text{search}}$ for a nearest-neighbor search. Blue circle: indicator for the distance of the nearest neighbor $DX_{j,2}$ from $\mathbf{r}_{j,\text{search}}$. The vertices $DX_{j,1}$ and $\{DX_k\}_j$ are disregarded in this search, and likewise are D-vertices found in the gray area. Solid light-pink lines: resulting candidate cell.

selected by this procedure has been found to form a D-cell in one out of $\mathcal{O}(1)$ cases; in QA systems close to the percolation transition, for instance, only about two candidate tuples need to be inspected on average. Essentially, this procedure’s efficiency derives from the fact that the vertices of a D-cell are generally in spatial proximity of each other if the $\{DX_i\}$ are randomly distributed.

Despite its remarkable rate of success, the tuple-selection method outlined above is *not* suitable for constructing an *entire* D-tessellation. This is because there may exist D-cells that cannot be described by a sequence of nearest neighbors of barycenters. Completely random trial tuples *would* eventually yield the entire D-tessellation, but—as discussed above—come at too high a computational cost. For this reason, we introduce step (2) as stated in the beginning of this section, with the aim of incrementally “nucleating” the D-tessellation around the initial D-cell.

More precisely, the idea of step (2) is the following. Among the D-faces that are already known to be part of the final D-tessellation, a face is identified that is part of only *one* known D-cell. Let us call this D-face DF_j , its constituent D-vertices $\{DX_k\}_j$, and the known D-cell $DC_{j,1}$. Since in any dimension, each D-face is part of *two* D-cells^{2.67}, there exists another D-cell $DC_{j,2} \neq DC_{j,1}$ that DF_j is part of. Both $DC_{j,1}$ and $DC_{j,2}$ contain one D-vertex in addition to $\{DX_k\}_j$; these vertices we shall refer to as $DX_{j,1}$ and $DX_{j,2}$. Since by construction, $DX_{j,1}$ is already known, only $DX_{j,2}$ is of interest here. A simple approach to find $DX_{j,2}$ is to form a trial $(d+1)$ -tuple for each point that is in $\{DX_i\}$ but not in $\{DX_{j,1}, \{DX_k\}_j\}$, and to determine whether the corresponding circumhypersphere is empty. This procedure would result in an average of $\mathcal{O}(N)$ operations per addition of a D-cell, or $\mathcal{O}(N^2)$ operations for completing the D-tessellation.

Although this performance is already quite acceptable, Maus [183] showed that relatively-simple modifications can take this method even further. Specifically, due

^{2.67}If only D-cells of finite size are considered, and if the system contains a finite number of vertices, then the D-tessellation possesses a “surface” at which D-faces are part of merely *one* D-cell. In this work, however, we exclusively consider systems with PBCs (Sec. 2.2.2.1), which are effectively infinitely large and for which such a surface does not exist.

again to the local nature of the D-cells, $DX_{j,2}$ is likely to be located in the neighborhood of DF_j . This obviously calls for another nearest-neighbor search. As for the initial D-cell, this search can for instance be conducted around the barycenter of the $\{DX_k\}_j$. Working in two dimensions, Maus [183] suggested to instead start the search at the circumcircle center of a equilateral triangle formed by a hypothetical $DX'_{j,2}$ and the $\{DX_k\}_j$. This procedure is pictured in Fig. 2.26. We generalized Maus’s idea to d dimensions by considering as the search pivot the point

$$\mathbf{r}_{j,\text{search}} = \mathbf{r}_{j,\text{circ}} \pm \hat{\mathbf{r}}_{j,\text{perp}} \sqrt{d} \quad (2.45)$$

where $\mathbf{r}_{j,\text{circ}}$ is the center of the $(d-1)$ -dimensional circumhypersphere around $\{DX_k\}_j$, and $\hat{\mathbf{r}}_{j,\text{perp}}$ is the unit-length normal vector of DF_j . Whether the “+” or the “−” sign has to be chosen in Eq. (2.45) depends upon the definition of $\hat{\mathbf{r}}_{j,\text{perp}}$ and upon the location of $DX_{j,1}$ relative to the hyperplane defined by DF_j (see Sec. 2.4.6). Furthermore, it is worth noting that DF_j divides the given space into two half-spaces, each of which hosts precisely one of the D-cells that DF_j is part of. This means that candidates for $DX_{j,2}$ resulting from the nearest-neighbor search can be disregarded if they are located in the same half-space as $DX_{j,1}$ (gray area in Fig. 2.26). Since validating the latter requirement is considerably faster than checking for empty circumhyperspheres, this further accelerates the D-tessellation algorithm. Once a candidate for $DC_{j,2}$ has been identified, the corresponding D-cell candidate is checked for whether it conforms to the empty-sphere criterion. If this is not the case, then the current candidate for $DC_{j,2}$ is put on a “black list”, and the next candidate for $DC_{j,2}$ is retrieved by repeating the nearest-neighbor search around $DX'_{j,2}$ and disregarding therein all D-vertices on the black list.

As mentioned before, all systems investigated in this study are subject to PBCs (see Sec. 2.2.2.1). Therefore, this condition also applies to the corresponding D-tessellation, which effectively renders it infinitely large and surface-less (cf. Footnote 2.67). Unfortunately, however, in particular in small systems, the nearest-image convention used in association with these PBCs (cf. Sec. 2.2.2.1) may occasionally lead to a failure of a systematic search for $DX_{j,2}$. The reason for this lies in the ambiguity of the nearest-image pivot for extended objects such as a D-face or a D-cell (cf. Sec. 2.4.2). To alleviate the cases of failure, our algorithm provides a fall-back mechanism that selects *random locations* for the pivot of the nearest-neighbor search for $DX_{j,2}$, as opposed to the schematic selection described further above.

Upon finally discovering $DC_{j,2}$, it is determined whether $DC_{j,2}$ is already among the known D-cells so as to prevent double-counting. (D-cells are identical if defined by the same set of D-vertices.) In order to avoid checking $DC_{j,2}$ against all $\mathcal{O}(N)$ known D-cells, after its discovery, a D-cell is “associated” with each of its $d+1$ D-vertices. This allows to seek for possibly-existing instances of a D-cell within $\mathcal{O}(1)$ operations by polling the D-vertices of the D-cell in question for associated D-cells.

Summarizing the above measures, the algorithm presented in this section requires merely $\mathcal{O}(N)$ operations to generate the D-tessellation of a given distribution of D-vertices—provided that this distribution is random. In Appendix A.3.5, we underpin

this assertion in the form of a brief compilation of performance measurements. Methods for verifying that the algorithm introduced in this section indeed yields correct D-tessellations are discussed in Appendix A.3.4.

2.4.5 Percolation algorithm

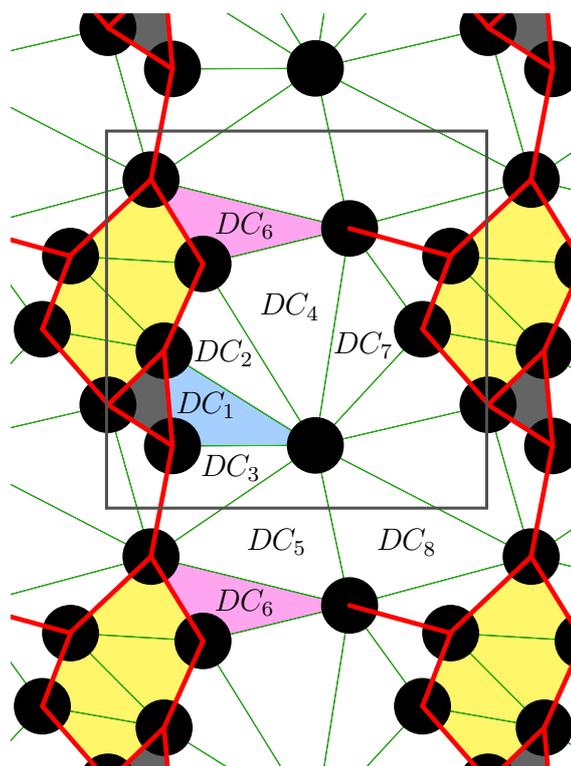
In this section, we take on the issue of identifying D-cell clusters, i.e., groups of D-cells in which every D-cell is connected to another D-cell by a crossable D-face (cf. Sec. 2.4.3). The information required to carry out this identification is encoded in its entirety in the D-network (Sec. 2.4.3). Provided that the D-cell clusters are identified correctly, each cluster contains either none or precisely one of the system's voids within the space covered by its D-cells (see Appendix A.1.2). If a D-cell cluster consists of a finite number of sites and bonds, then assembling the cluster is a relatively-minor challenge. If on the other hand PBCs (Sec. 2.2.2.1) are present and a cluster spans across the simulation box, then it is vital to avoid a multiple counting of D-cell images while still being able to identify whether the void is percolating. To accomplish this task, we devised a novel algorithm that we will describe in the following. Notably, that algorithm is generic in that it can be applied to any network of sites and bonds that is subject to PBCs.

The starting point of our algorithm is marked by choosing at random a D-cell for which it is not yet known what D-cell cluster it belongs to. This random D-cell—which at this point we call the “current” D-cell—is noted as the initial element of a list that ultimately is to contain all D-cells of one of the D-cell clusters. The current D-cell is stored in this list together with a representative location, the latter being for instance the center of the D-cell's sphere vide.

Following this, all D-cells are identified that border the current D-cell via a connecting D-face; these D-cells we will call “pending”. For each pending D-cell, an absolute location is defined (like for the initial D-cell), from which then a location *relative to the initial* (not the current) D-cell is computed. This relative location is obtained as follows: first, the difference vector between the absolute locations of the current and the pending D-cells is determined—using the nearest-image convention if applicable (cf. Sec. 2.2.2.1)—, and then, this difference vector is added to the (known) relative location of the current D-cell. Subsequently, each pending D-cell is checked for whether it is already an element of the list. Obviously, to this end there are two possibilities:

- (1) The pending D-cell is *not* on the list; in this case, the respective D-cell is simply appended to the list together with its relative location.
- (2) The pending D-cell *is* already contained in the list. This is where the use of *relative* locations becomes relevant since in this case the *stored* relative location of the pending D-cell is compared with the one just *computed*. This gives rise to two subordinate cases:

Figure 2.27: Percolation algorithm. Two-dimensional illustration of the percolation algorithm for an example D-tessellation with PBCs. Black disks: matrix particles of radius R . Green [red] lines: [non]crossable D-faces. Yellow areas: D-cells that belong to a trap; gray areas: inaccessible D-cells. Light-blue area: initial D-cell in the percolation algorithm. Light-pink areas: first D-cell for which two different periodic images are encountered by the algorithm.



- (2a) The relative locations *match*, meaning that the pending D-cell has already been visited and nothing needs to be done.
- (2b) The relative locations do *not* match. In this case the pending D-cell is a *periodic image* of a D-cell already contained in the list, which immediately leads to the conclusion that the list is associated with a *percolating void*. The latter piece of information is stored as a flag within the list; other than that, nothing needs to be done.

After completing the above procedure with all pending D-cells, the current D-cell is marked “processed”. Then, the unprocessed D-cell with the lowest list index is retrieved from the list and the same procedure is repeated. This scheme is iterated until the list contains solely processed D-cells, at which point the list encompasses all D-cells in the cluster. Notably, since in subcase (2b), the pending D-cell is *not* added to the list, even in the case of a percolating void, the list contains only a finite number of D-cells—which are then to be periodically replicated to obtain the infinitely-large void. Conversely, if the “percolating” flag is not set, then no periodicity is assumed and the pertinent void—if present—is immediately known to constitute a trap.

The algorithm described in the above is illustrated in Fig. 2.27. The figure depicts a D-tessellation in which the elements are highlighted in the same style as in Fig. 2.20, i.e., the matrix particles are of the same radius R as the fluid particles, but the latter are omitted. To achieve a reasonable number of iteration steps of the algorithm, the depicted system is substantially smaller than that in Fig. 2.20; however, it does feature

PBCs in order to illustrate the algorithm in that particular case. The algorithm then proceeds as follows. Suppose that the initial D-cell in the list is DC_1 (marked in light blue). The iteration steps resulting from this choice are:

- In **loop #1**, DC_1 is current and DC_2 and DC_3 are pending. Both DC_2 and DC_3 are added to the list.
- In **loop #2**, DC_2 is current and DC_1 and DC_4 are pending. DC_1 is encountered at the same relative location as previously saved; therefore, it is discarded and only DC_4 is added to the list.
- In **loop #3**, DC_3 is current and DC_1 and DC_5 are pending. DC_1 is discarded whereas DC_5 is added to the list.
- In **loop #4**, DC_4 is current and DC_2 , DC_6 , and DC_7 are pending. Only DC_6 and DC_7 are added to the list.
- In **loop #5**, DC_5 is current and DC_4 , DC_6 , and DC_8 are pending. DC_4 is discarded and DC_8 is ordinarily added to the list. DC_6 is encountered at a location *differing* from the encounter in loop #4, meaning that the group is percolating.

Subsequently, the procedure is continued at DC_6 in order to find the remaining D-cells of the cluster.

Before concluding this section, it is prudent to ask whether it is sensible at all to investigate percolating voids in systems with PBCs. From a technical point of view, the answer to this question is simple: it does make sense since the analysis merely operates on the information encoded in the provided systems. From a physical standpoint, however, matters are more complicated due to conflicting aspects of PBCs and QA systems: while PBCs are designed to mimic an infinitely-large system by introducing periodicity, a defining feature of the matrix in QA systems is its *nonperiodic* nature. Therefore, whenever the spatial extent of some feature of the system is larger than the system that is actually simulated—and notably percolation involves infinitely-large length scales—, effects connected to periodicities will arise that would not exist in an infinitely-large system. This problem is in fact merely the known issue of finite-size effects, which is scrutinized in the present context in Appendix A.2.

It is, however, valid to ask whether percolation in a finite system can serve at least as an *approximation* for percolation in an infinitely-large system with the same external parameters. In fact, this issue has been investigated by a number of authors [222, 276, 291] who have concluded that in finite systems under PBCs, percolation is generally more likely to occur in smaller systems. Therefore, finite systems can be used to establish either an upper or a lower bound to the “percolation threshold” (the value of the control variable at which percolation occurs in the system of interest). If the influence of the system’s size on the percolation threshold is known quantitatively, then it may even be possible to correct other observables for finite-size effects. Notably, the

authors cited above have devised methods similar to the algorithm presented in this section in order to identify percolation in systems with PBCs; however, our algorithm is more elaborate than those methods since the latter consider a cluster to be percolating simply if it touches a given number of opposing sides of the simulation box (see in particular Ref. [276]).

2.4.6 Rastering algorithm

In this section, we will first describe the geometric method used in this work for determining whether a given, random spatial point is located within a given D-cell. Following that, we will introduce a rastering algorithm for estimating the amount of accessible volume contained within a given D-cell. The latter information can subsequently be used to determine the size of all of the voids in the system in terms of their associated accessible volume, which opens the possibility for a quantitative statistical analysis of the sizes of the voids (see Secs. 2.5.4 and 3.3).

2.4.6.1 Given point within given Delaunay cell

Determining whether a given spatial point P_i is located within a given D-cell DC_j is fairly simple. The method that we will describe in this section uses the fact that the $d+1$ D-faces that bound a D-cell are in fact segments of $(d-1)$ -dimensional hyperplanes, each of which divides the given d -dimensional space into two half-spaces. The hyperplanes can therefore be thought to partition the given space into 2^{d+1} regions, precisely one of which constitutes the “inside” of DC_j . Determining whether P_i resides within DC_j therefore merely requires to check $d+1$ hyperplanes for whether P_i is located in the “correct side” in each respective case.

The first step of our method is to determine the normal vector $\mathbf{n}_{k,j}$ of a given hyperplane $H_{k,j}$. One way to obtain $\mathbf{n}_{k,j}$ in arbitrary dimension d is to evaluate the “generalized cross product”^{2.68}, which can formally be expressed as the determinant of a matrix as

$$\bigwedge(\mathbf{h}_1, \dots, \mathbf{h}_{d-1}) = \begin{vmatrix} h_1^{(1)} & \cdots & h_1^{(d)} \\ \vdots & \ddots & \vdots \\ h_{d-1}^{(1)} & \cdots & h_{d-1}^{(d)} \\ \mathbf{e}^{(1)} & \cdots & \mathbf{e}^{(d)} \end{vmatrix} = \mathbf{n}_{k,j} \quad . \quad (2.46)$$

Here, the vectors $\mathbf{h}_1, \dots, \mathbf{h}_{d-1}$ are linearly independent and parallel to $H_{k,j}$, and the vectors $\mathbf{e}_1, \dots, \mathbf{e}_d$ define the orthonormal basis in which the components of the $\mathbf{h}_1, \dots, \mathbf{h}_{d-1}$ are expressed in. The determinant in Eq. (2.46) can be evaluated via the usual determinant algebra such as Laplace expansion (see Footnote 2.53). In three dimensions—the

^{2.68}http://en.wikipedia.org/wiki/Cross_product

case relevant to this work—, Eq. (2.46) reduces to the common cross product^{2.68}:

$$\bigwedge(\mathbf{h}_1, \mathbf{h}_2) = \begin{vmatrix} h_1^{(x)} & h_1^{(y)} & h_1^{(z)} \\ h_2^{(x)} & h_2^{(y)} & h_2^{(z)} \\ \hat{\mathbf{x}} & \hat{\mathbf{y}} & \hat{\mathbf{z}} \end{vmatrix} = \hat{\mathbf{x}} \begin{vmatrix} h_1^{(y)} & h_1^{(z)} \\ h_2^{(y)} & h_2^{(z)} \end{vmatrix} + \hat{\mathbf{y}} \begin{vmatrix} h_1^{(z)} & h_1^{(x)} \\ h_2^{(z)} & h_2^{(x)} \end{vmatrix} + \hat{\mathbf{z}} \begin{vmatrix} h_1^{(x)} & h_1^{(y)} \\ h_2^{(x)} & h_2^{(y)} \end{vmatrix} = \mathbf{h}_1 \times \mathbf{h}_2 . \quad (2.47)$$

The $d-1$ linearly independent vectors in the hyperplane $H_{k,j}$, required for Eq. (2.46), can be established for instance from the d D-vertices that define $H_{k,j}$. If the locations of these D-vertices are $\mathbf{r}_1, \dots, \mathbf{r}_d$, then selected difference vectors between these locations can represent the desired vectors $\mathbf{h}_1, \dots, \mathbf{h}_{d-1}$, with one possible selection being

$$\mathbf{h}_\alpha = \mathbf{r}_d - \mathbf{r}_\alpha \quad \text{with } \alpha \in \{1, \dots, d-1\} . \quad (2.48)$$

In the second step of our method, we construct two auxiliary vectors: one vector, $\mathbf{p}_{k,j}$, runs from a point $O_{k,j} \in H_{k,j}$ towards the given point P_i , and the other vector, $\mathbf{q}_{k,j}$, points from the same $O_{k,j}$ towards a point $Q_{k,j}$ that is known to reside on the correct side of $H_{k,j}$. The many equivalent choices for the point $O_{k,j}$ include the D-vertices of the D-face $DF_{k,j}$ that defines $H_{k,j}$, the circumhypersphere center of $DF_{k,j}$, and the barycenter of $DF_{k,j}$. For the point $Q_{k,j}$ there are equally many options; however, if $Q_{k,j}$ is chosen to reside inside DC_j , then it even resides on the correct side relative to *all* hyperplanes in question. One point that certainly resides within DC_j is the barycenter of the D-cell, i.e., the averaged location of the D-vertices that define DC_j (cf. Sec. 2.4.4.4).

Having defined $\mathbf{p}_{k,j}$ and $\mathbf{q}_{k,j}$, we project these vectors onto the normal vector of $H_{k,j}$ by simply evaluating the scalar products

$$S_P = \mathbf{n}_{k,j} \cdot \mathbf{p}_{k,j} \quad \text{and} \quad S_Q = \mathbf{n}_{k,j} \cdot \mathbf{q}_{k,j} . \quad (2.49)$$

In the result of this operation, it is the *signs* of S_P and S_Q that are relevant, with the points P_i and $Q_{k,j}$ residing on the same side of $H_{k,j}$ only if those signs are equal (i.e., if both scalar products are positive or both are negative). Notably, for this comparison of signs, it is not necessary for $\mathbf{n}_{k,j}$ to be of unit length.

One direct application of the method presented in this section is to determine whether a given fluid particle in a QA mixture is trapped or free. If for each D-cell in the system, it is known whether it is associated with a trap or a percolating void (cf. Sec. 2.4.5), this can be accomplished by simply checking each of the $\mathcal{O}(N)$ D-cells for whether it contains the given fluid particle. While an acceleration of this application—e.g., using some bucketing technique (cf. Sec. 2.2.2.2)—is certainly possible, we found such optimization expendable since identifying *each* fluid particle's D-cell required merely seconds for the systems investigated in this work. In this context, it is notable that the performance of the method presented in this section compares favorably with

other methods to accomplish the same task^{2.69,2.70}.

2.4.6.2 Accessible volume within Delaunay cell

We now turn to the problem of estimating the amount of accessible volume contained within a given D-cell. Knowledge about this property for *all* D-cells immediately yields the sizes of the voids—in terms of accessible volume—in the given QA system if previously a percolation analysis has been performed (Sec. 2.4.5). Information about all void sizes in turn allows to conduct pertinent statistical analyses, which then may give way to explaining and interpreting features of the fluid component of the given QA system. Further, knowing the void sizes allows to accurately determine the matrix density at which the percolation transition of the accessible volume takes place (see Secs. 1.3.3.3 and 2.5.4 as well as Refs. [248, 252]).

Predating our work by a decade, Sastry et al. devised a method for the exact calculation of the accessible volume within a three-dimensional D-cell [229]. Unfortunately, due to the lengthy mathematical expressions involved, this method is nontrivial to implement and debug. Since accessible volumes are not in the principal focus of this work, we instead chose to determine void sizes only approximately. Our method of choice for this is to divide the given space into minute distinct, space-filling regions (which we will call “grains”), each of which is—by some criterion—considered either accessible or inaccessible as a whole. The approximate amount of accessible volume of a given D-cell DC_i can then be obtained by combining the volumes of all grains that are deemed to belong to DC_i , with each grain obviously belonging to precisely one D-cell. Clearly, the accuracy of this approximation depends upon the size and shape of the grains.

We chose, for simplicity, to divide the given d -dimensional space into d -hypercubic grains, which we arrange on a hypercubic “raster” similar to that of the buckets used in the event-driven MD algorithm (see Sec. 2.2.2.2 and Fig. 2.8 therein). In a first

^{2.69}A short performance assessment for this section’s method goes as follows. In d dimensions, the method encompasses the evaluation of $d+1$ generalized cross products, with each $d \times d$ determinant involved requiring $\sum_{i=1}^{d-1} d!/i!$ multiplication operations. Additionally, $2(d+1)$ scalar products need to be evaluated, with each prompting for d multiplications. Therefore, in $d = 3$, the method involves $4 \times 9 + 8 \times 3 = 60$ multiplications (plus some summations, but no operation of higher complexity such as square roots). If additionally the normal vector of each D-cell is stored, then only the 24 scalar product multiplications are required. However, depending on the computer architecture and the simulated system, storing additional data may even slow down evaluations due to finite caching capabilities (cf. Footnote A.9).

^{2.70}A popular alternative method for determining whether the given point P_i is located within the given D-cell DC_j is to express P_i in the “barycentric coordinates”^{2.71} pertaining to DC_j . There, if any component of P_i ’s barycentric coordinates is negative, then P_i resides outside DC_i . Unfortunately, computing barycentric coordinates involves the evaluation of $d+2$ determinants with $d+1$ rows and columns each, which results in 200 multiplications if $d = 3$. This is substantially more than the 60 multiplications that the method presented in this section requires if no normal vectors are stored, with our method moreover scaling favorably with dimensionality^{2.69}.

^{2.71}[http://en.wikipedia.org/wiki/Barycentric_coordinate_system_\(mathematics\)](http://en.wikipedia.org/wiki/Barycentric_coordinate_system_(mathematics))

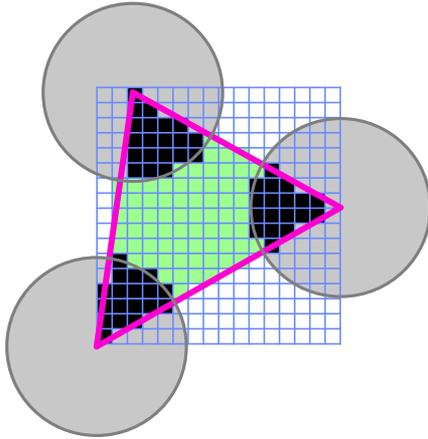


Figure 2.28: Rastering method. Two-dimensional representation of the method used to estimate the accessible volume within a D-cell. Each square represents one rastering grain. Pink triangle: D-cell. Gray disks: matrix particles with radius $2R$. Squares with blue outline: candidate grains; black-filled squares: inaccessible grains inside the D-cell; green-filled squares: accessible grains inside the D-cell.

step, each grain is classified as “accessible” or “inaccessible” depending on whether its *center*—which is well defined for a hypercube—resides within the accessible volume. Subsequently, it is determined which grains are associated with the given D-cell DC_i ; the criterion for this is again based on the center of a grain, which has to reside within DC_i for each respective grain. Since DC_i contains only a (possibly tiny) fraction of the grains in the system, we preselect candidate grains as follows. From the $d+1$ D-vertices that form DC_i , we determine the minimum and the maximum values of each coordinate, which we use to establish a d -hypercuboid that entirely encloses DC_i and has edges parallel to the coordinate axes. Only grain centers that reside inside this “circum-hypercuboid” can possibly also reside within DC_i . Due to the regular geometry of the grain raster, it is possible to compute the minimum and maximum indices of the grains that have to be considered. The accessible volume inside DC_i , finally, is then approximately given by the number of accessible grains inside DC_i times the volume of a grain, the latter being simply the raster spacing to the d^{th} power.

The rastering procedure is illustrated in Fig. 2.28, which shows a two-dimensional D-cell (pink triangle) and the corresponding matrix particles with radius $2R$ as in Fig. 2.17 (gray disks). The figure depicts candidate grains (those with centers inside the circum-hypercuboid) as squares with blue outline, whereas grains inside the D-cell are squares filled in black if inaccessible and filled in green if accessible. Notably, in our statistical analysis of void sizes (see Secs. 2.5.4 and 3.3), the grains are considerably smaller than in Fig. 2.28, in which the grains’ edges are as long as $R/6$. However, the achievable resolution is limited by both memory and computing time constraints.

2.5 Observables

In this section, we will be concerned with the task of extracting information from condensed-matter systems in general and from quenched-annealed (QA) systems in particular. For this, we will first introduce the concept of observables, and then motivate and define all such quantities—static and dynamic—that are of importance to this

work. Subsequently, we will discuss the tools for a statistical analysis of the accessible volume, before finally turning to the issue of error estimation.

It is due to emphasize in this prominent place that the actual evaluation of all observables—using input data from the molecular-dynamics (MD) algorithm of Sec. 2.2—was carried out using the `ATool` (ATomistic Object-Oriented Libraries) package developed by Daniele Coslovich [53].

2.5.1 Observables in computer simulations

In this section, we will introduce the concept of observables, and discuss various issues of this concept in general and in the context of QA systems and of computer simulations.

2.5.1.1 Phase-space density vs. observables

As described in Sec. 2.1.2, a condensed-matter system is fully characterized by the phase-space density $f_{\mathbf{Q}}(\mathbf{\Gamma})$, that is, by the likelihood that at a given time, the system assumes a particular combination of its internal degrees of freedom [108, 115, 238, 258]. The degrees of freedom are notationally condensed in the phase-space vector $\mathbf{\Gamma}$, which typically contains the particles' loci \mathbf{r}^N and momenta \mathbf{p}^N in a system realization with N particles. The symbol \mathbf{Q} denotes the set of fixed external conditions that the system is subjected to, and (as alluded to before) we assume that the system under investigation has reached thermal equilibrium, i.e., that $f_{\mathbf{Q}}(\mathbf{\Gamma})$ is independent of time. (The sensitive issue of *establishing* thermal equilibrium will be discussed in Sec. 2.5.5.)

Unfortunately, $f_{\mathbf{Q}}(\mathbf{\Gamma})$ cannot be determined directly for the overwhelming majority of many-body systems. This is owed to Eq. (2.4), via which the phase-space density involves the partition function $\Omega_{\mathbf{Q}}$, which in turn due to Eq. (2.2) requires information about *all* of the system's phase-space points. However, for most systems, the phase-space density itself is not particularly informative anyway (even if known entirely) due to the amount of information it encompasses. Therefore, typically a different approach is pursued to characterize condensed-matter systems: observables. Despite the inevitable fact that any observable contains less information than $f_{\mathbf{Q}}(\mathbf{\Gamma})$, a well-chosen observable may provide much more insight into a system than the raw phase-space density. This is immediately evident when considering the macroscopic world, where experimental measurements are well capable of providing a meaningful description of a material—even if only macroscopic features are accessible. While this relieves the investigator from explicitly determining $f_{\mathbf{Q}}(\mathbf{\Gamma})$, in exchange it presents him or her with the task of selecting observables that are appropriate for characterizing the system of interest.

Alas, recalling Eq. (2.5), we see that $f_{\mathbf{Q}}(\mathbf{\Gamma})$ also plays a pivotal role in computing the average value $\langle A \rangle_{\mathbf{Q}}$ of an observable A under the conditions \mathbf{Q} . However, as discussed in Sec. 2.1.3, if we consider an *ergodic* system, then we can replace Eq. (2.5) by Eq. (2.6), i.e., the phase-space average $\langle A \rangle_{\mathbf{Q}}^{\text{ps}}$ by the time average $\langle A \rangle_{\mathbf{Q}}^{\text{time}}$. Since

Eq. (2.6) does not contain any reference to $f_{\mathbf{Q}}(\mathbf{\Gamma})$, this strategy relieves the investigator entirely of having to compute the phase-space density explicitly. On the down side, this procedure introduces the additional concept of a *trajectory* in phase space, along which a sequence of $\mathbf{\Gamma}$'s is visited. Despite the seeming disadvantage, such a time evolution is precisely what takes place in real systems, and of course many real materials can be described to possess well-defined properties.

2.5.1.2 Time-dependent observables

In computer investigations, it is common to evaluate averages of the type $\langle A \rangle_{\mathbf{Q}}^{\text{time}}$ from data generated in MD simulations (cf. Sec. 2.1.3 and Refs. [6, 9]). Among others, the event-driven MD algorithm presented in Sec. 2.2 is suited for this task; in this specific case the external conditions read $\mathbf{Q} = NVE$, where N is the number of particles of the system, V its volume, and E its total internal energy. The reason for using MD instead of Monte-Carlo (MC) simulations is that the former technique allows naturally for computing observables of the form $A(t)$, i.e., for time-dependent quantities. Although the MC scheme also generates phase-space trajectories, these trajectories generally do *not* represent time evolutions^{2.72}. Since time-dependent observables are of paramount importance for characterizing glass-forming systems [1, 86, 115, 138, 154, 158, 214, 254, 269, 284] and, more generally, for quantifying any dynamic feature of a material, MD is typically clearly preferable to MC in these contexts.

MD simulations provide complete information about the phase state point $\mathbf{\Gamma}(t)$ occupied at a certain simulation time t since knowing all degrees of freedom [in the present case the locations $\mathbf{r}^N(t)$ and the momenta $\mathbf{p}^N(t)$] is essential to MD algorithms. Therefore, any observable $A(\mathbf{\Gamma}(t))$ computed from this information is formally exact (except for numerical errors). However, for two reasons, computer simulations are inherently unable to take *all* phase-space points into account. Firstly, any simulation and therefore any pertinent phase-space trajectory is finite. Secondly, even along a finite phase-space trajectory, there are infinitely-many $\mathbf{\Gamma}$'s, out of which a computer program can consider only a subset. As a result, any average $\langle A \rangle_{\mathbf{Q}}^{\text{time}}$ obtained from MD simulations must necessarily be an *approximation*. If, for instance, over the course of a simulation, we conduct M measurements at equidistant time intervals Δt , such an approximation can be expressed by

$$\langle A \rangle_{\mathbf{Q}}^{\text{time}} \stackrel{\text{Eq. (2.6)}}{=} \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^{\tau} A(\mathbf{\Gamma}(t)) dt \approx \frac{1}{M} \sum_{m=1}^M A(\mathbf{\Gamma}(m \Delta t)) . \quad (2.50)$$

^{2.72}In fact, there do exist MC methods like Kinetic MC^{2.73} that produce time-like sequences of phase-space points. However, it is important to realize that none of these schemes is capable of determining *time scales* from first principles. On the contrary, data obtained from MC simulations have to be either rescaled to match a known time scale [28, 55], or the microscopic MC moves have to be associated with time scales known from other methods^{2.73}.

^{2.73}http://en.wikipedia.org/wiki/Kinetic_Monte_Carlo

While it is essential to determine the conditions under which this approximation is meaningful and justified, a discussion of this issue requires more detailed knowledge about the observables that are relevant to the investigated system. Therefore, for the time being we will regard Eq. (2.50) as valid and defer a detailed assessment to Sec. 2.5.5.

The three most important types of dynamic quantifiers are connected by “Einstein relations” [74, 274]

$$\gamma = \lim_{t \rightarrow \infty} \frac{1}{2t} \langle |A(t) - A(0)|^2 \rangle \quad (2.51)$$

and “Green-Kubo relations” [106, 166]

$$\gamma = \int_0^\infty \langle \dot{A}(t) \dot{A}(0) \rangle dt \quad . \quad (2.52)$$

Here, we have notationally suppressed the dependence of the averages upon \mathbf{Q} , as we shall do the following unless needed for clarity. The symbol γ on the left-hand side of both equations denotes a generic scalar “transport coefficient” [9, 115]; quantities of this type are only of marginal interest in this work since they are mainly suited for *quantitative* comparison whereas our foremost goal is to investigate the *qualitative* behavior of QA systems. The expression on the right-hand side of Eq. (2.51) provides the template for one of the most important dynamic observables, the mean squared displacement, which will be introduced and discussed in Secs. 2.5.3.1 and 3.2.4. The expression on the right-hand side of Eq. (2.52) contains an important class of observables—“correlators” [9, 115]—which we will discuss in more detail in the following.

2.5.1.3 Correlators

Static correlators are fairly simple objects. Their general form is obtained by replacing the argument of the integral in Eq. (2.6) by the product of two observables A and B evaluated at the same time t :

$$\langle AB \rangle = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau A(t) B(t) dt \quad . \quad (2.53)$$

Here, we have used the compact notation $A(t) = A(\Gamma(t))$, and likewise for B . In the simplest version of such a correlator, A and B are scalar functions of their argument (and thus accordingly is $\langle AB \rangle$); however, A and B also could be vectors or even tensors. *Dynamic* correlators are slightly more complicated objects in which A and B are evaluated at *different* times:

$$\langle A(t') B(t'') \rangle = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau A(s + t') B(s + t'') ds \quad . \quad (2.54)$$

In thermal equilibrium, the correlator is a function only of the time difference t between the arguments of A and B . Therefore, under this condition, an equivalent expression is

$$\langle A(t) B \rangle = \lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau A(s + t) B(s) ds \quad . \quad (2.55)$$

Here, the expression on the left-hand side is notationally condensed from $\langle A(t)B(0) \rangle$; this is common practice since by construction only the time difference t is relevant [115].

A special role is played by dynamic correlators for which $B = A$, with the resulting functions of the form $\langle A(t)A \rangle$ being called “auto-correlation functions”. These functions feature the general property of being maximal for $t = 0$, which offers the distinct possibility for normalization without having to consider an extended time range [115]. Except for the time derivatives, the argument of the integral on the right-hand side of Eq. (2.51) is of this type. Analyzing various auto-correlators constitutes the bulk of this investigation’s results part (Chapter 3).

2.5.1.4 Logarithmic time scales

For two reasons, dynamic observables in this work are considered on *logarithmic* time scales: Firstly, systems with slow dynamics typically exhibit multiple time scales that may differ by several orders of magnitude [154, 241]. Secondly, and more importantly, in glassy systems, the structural relaxation time is extremely sensitive to changes in the external control parameters (cf. Sec. 1.2), which makes it impossible to illustrate pertinent trends in a single linear-time plot.

Unfortunately, in MD simulations, it is methodologically more demanding to evaluate correlators on logarithmic than on linear time scales. The challenge lies in the fact that MD simulations typically generate system configurations at regular simulation time intervals. Obviously, to resolve the system’s shortest time scale of interest, τ_{\min} , these intervals must be chosen shorter than τ_{\min} . On the other hand, we also want to obtain information about the longest relevant time scale, τ_{\max} , which in the equal-interval scheme means that at least τ_{\max}/τ_{\min} configurations have to be generated. This results in a problematically-large number of configurations if $\tau_{\max} \gg \tau_{\min}$, and precisely this is the case in glass-forming systems.

There are two principal methods at the investigator’s dispense to overcome this difficulty. The first, probably more obvious one is to generate configurations at logarithmically spaced times, which comes at the expense of an increased programming effort as well as substantially-worse statistics when evaluating observables for small times. The second approach is to first conduct an MD run longer than τ_{\max} , and to then perform a number of runs that start at randomly-chosen configurations of the long run and last for a shorter time τ_{short} . These short runs then serve to provide observable averages at a higher time resolution for times up to $t = \tau_{\text{short}}$. Unfortunately, using this method, there will appear to be small “jumps” in an observable’s average $\underline{A}(t)$. This is because for $t \leq \tau_{\text{short}}$, we chose to calculate $\underline{A}(t)$ from the short runs only, and for $t > \tau_{\text{short}}$ from the long runs only. Since different simulation runs always yield slightly different values of $\underline{A}(t)$ (see Sec. 2.5.5), this leads to the aforementioned jumps right at the times τ_{short} . Despite this visual drawback, we opted for the second method—a choice largely made to fit the capabilities of the `ATool` library [53]. Notably, we refrained from introducing ad-hoc methods to render $\underline{A}(t)$ visually smoother since the jumps merely reflect the inherent simulational uncertainties.

2.5.1.5 Quenched disorder

In the case of QA systems as introduced in Sec. 1.3.4.2, another complication is added to the picture [97, 99, 100, 159–162, 174, 180, 181, 189, 266]. Imagine that we have used the QA protocol to create one particular realization of the quenched disorder. Then, by definition, there are spatial regions that fluid particles are more likely to occupy than others. For hard spheres, there are even zones that are entirely inaccessible—regardless of how long we wait, and no matter at which allowed positions we initially place our fluid particles (cf. Sec. 2.4). Moreover, the pinned particles locally entail directional dependencies in the behavior of the fluid particles. In consequence of these effects, the quenched disorder *breaks the spatial homogeneity and isotropy* of the thermodynamically-averaged fluid. While these symmetries are broken also by well-characterized confinement geometries like parallel-plate slits, cylindrical channels, and spherical pores, in disordered confinement this constitutes an undesired effect since there is no uniquely-defined reference geometry. Instead, disordered matrices can be realized in infinitely-many arrangements which are equivalent but *not* the same.

However, by construction, the matrix in QA systems follows statistical rules (see Sec. 1.3.4.2) that render *the matrix itself* spatially homogeneous and isotropic if it is averaged over all possible realizations. This means that the homogeneity and isotropy of the annealed component in a QA mixture can be *restored* if we consider not only a thermal average but additionally, in a second step, a statistical-mechanical average over the quenched disorder. Notationally, we shall express this procedure by the symbol $\overline{\cdots}^{\mathbf{Q}'}$; for instance, the prescription in Eq. (2.6) including this average reads

$$\langle A \rangle_{\mathbf{Q}}^{\text{time } \mathbf{Q}'} = \overline{\lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau A(t) dt}^{\mathbf{Q}'} . \quad (2.56)$$

Like \mathbf{Q} , the index \mathbf{Q}' denotes external conditions, only this time those that the matrix configurations were obtained from. Obviously, in the context of the present work, it is $\mathbf{Q}' = NVE$ just as for \mathbf{Q} ; note, however, that save V , the quantities contained in \mathbf{Q} and \mathbf{Q}' are independent of each other. In the following, like \mathbf{Q} we shall notationally suppress \mathbf{Q}' , except for selected explicative reasons.

Since in many ways, a QA system resembles a mixture of two “species” of particles, it would in principle be possible to define each observable separately for the matrix (“m”) and the fluid (“f”) particles. This concept could also be extended to correlators, i.e., both arguments of the integral could be treated to pertain to one of the species, which would result in four different varieties for each correlator—“mm”, “mf”, “ff”, and “fm” (the fourth combination being equivalent to “mf” in auto-correlators). However, in the context of the present work, we are interested *exclusively* in the properties of the fluid particles, i.e., in “f”- and “ff”-type quantities. Therefore, these indices will not be explicitly typeset unless necessary for clarity. The motivation behind focusing on the fluid is that the array of obstacles can be viewed as a mere external potential that the fluid particles are subjected to, i.e., that the particulate nature of the confinement is actually irrelevant.

2.5.1.6 Reduced units

In order to lend meaning to the numbers that represent a system, they must be associated with units. For the purpose of simplicity and computational efficiency, it is advantageous to choose these units in a way that many stored and calculated properties assume values either equal to or close to one. Units fulfilling this requirement are commonly called “reduced units”. In this work, all quantities that are associated with a unit are given in the following system of reduced units:

<i>dimension</i>	<i>unit symbol</i>	<i>defined by</i>
length	σ	particle diameter
mass	μ	particle mass
time	τ	$\tau = \sqrt{\frac{k_B T}{\mu \sigma^2}}$

Here, T is some temperature, and k_B is Boltzmann’s constant. In a system consisting of hard spheres, the choice of T does not impact the dynamics except for an overall time factor (see Sec. 1.1.3.2). This is because in this kind of system, there is no potential energy, and according to the equipartition theorem (cf. Footnote 1.11), it is $(3/2)Nk_B T = \sum_{i=1}^N m_i |\mathbf{v}_i|^2 / 2$, where N is the number of particles in the system and m_i , and \mathbf{v}_i are the mass and the velocity of particle i . The time unit τ is designed to follow the same functional dependence upon T as do the velocities \mathbf{v}_i so that, employing τ as the unit of time, altering the temperature does not change the numerical values of dynamic observables. Also, note that the definitions of σ and μ are unambiguous since in every system considered in this work, all particles have the same diameter and the same mass.

2.5.2 Static observables

Static observables provide a time-independent characterization of a material. In the present work, the only static observables considered were structural correlators, which were mainly used to examine whether a particular QA system is of fluid or crystalline character (cf. Sec. 3.2.3).

2.5.2.1 Radial distribution function

Consider a correlator as in Eq. (2.53), with the functions A and B being delta functions of the positions \mathbf{r}_i and \mathbf{r}_j of two arbitrary particles i and j . If we sum all such correlators for all pairs of distinct particles in the system, we obtain the “two-particle density” [115]

$$\rho^{(2)}(\mathbf{r}, \mathbf{r}', t, t') = \sum_{i \neq j}^{N_f} \delta(\mathbf{r} - \mathbf{r}_i(t)) \delta(\mathbf{r}' - \mathbf{r}_j(t')), \quad (2.57)$$

where N_f is the number of fluid particles in the system, and we do not avoid double-counting pairs. The time dependencies have been introduced for future reference and

can be disregarded for the time being. In a homogeneous system, we can express the position \mathbf{r} relative to \mathbf{r}' , and integrate the result over \mathbf{r}' without losing any information. We can normalize this expression such that another integral over \mathbf{r} simply yields the volume V of the system; this requires a factor V/N_f^2 since the integration is performed over $N_f(N_f - 1) \simeq N_f^2$ delta functions. Actually carrying out this integral then yields the conventional “pair correlation function” [115]

$$\begin{aligned} g^{(2)}(\mathbf{r}) &= \int d\mathbf{r}' \frac{V}{N_f^2} \overline{\langle \rho^{(2)}(\mathbf{r} + \mathbf{r}', \mathbf{r}') \rangle} \\ &= \int d\mathbf{r}' \frac{V}{N_f^2} \overline{\left\langle \sum_{i \neq j}^{N_f} \delta(\mathbf{r} + \mathbf{r}' - \mathbf{r}_i) \delta(\mathbf{r}' - \mathbf{r}_j) \right\rangle} \\ &= \frac{V}{N_f^2} \overline{\left\langle \sum_{i \neq j}^{N_f} \delta(\mathbf{r} - [\mathbf{r}_i - \mathbf{r}_j]) \right\rangle} . \end{aligned} \quad (2.58)$$

In an isotropic system, this expression can be further simplified by removing the dependence upon angular variables. Using the notation $r_{ij} = |\mathbf{r}_j - \mathbf{r}_i|$ this results in

$$g^{(2)}(r) = \frac{V}{N_f^2} \overline{\left\langle \sum_{i \neq j}^{N_f} \delta(r - r_{ij}) \right\rangle} , \quad (2.59)$$

which, as is easily verified, still yields V upon simple integration over r . However, conventionally it is required that the “radial distribution function” $g(r) = c g^{(2)}(r)$ fulfill

$$\int 4\pi r^2 g(r) dr = V , \quad (2.60)$$

which immediately gives $c = 1/(4\pi r^2)$ and therefore

$$g(r) = \frac{1}{4\pi r^2} \frac{V}{N_f^2} \overline{\left\langle \sum_{i \neq j}^{N_f} \delta(r - r_{ij}) \right\rangle} . \quad (2.61)$$

The significance of the radial distribution function is twofold [9, 115, 219]: Firstly, it is experimentally accessible via the static structure factor $S(k)$ (see Sec. 2.5.2.2), which is essentially the Fourier transform of $g(r)$. Secondly, $g(r)$ contains much information about the local structure of a material. In simple fluids, for instance, up to a few particle diameters, $g(r)$ typically exhibits a wave-like pattern of peaks and troughs (seen, e.g., in Fig. 3.4), which indicates the existence of “shells” of particles neighboring each particle. Also, in the fluid case it is $g(r \rightarrow \infty) = 1$ and typically $g(0) = 0$, which reflects the lack of correlation between the positions of fluid particles over large distances and their steric repulsion at short separations, respectively.

2.5.2.2 Static structure factor

As alluded to in Sec. 2.5.2.1, the static structure factor $S(k)$ is essentially the Fourier transform of the radial distribution function $g(r)$. Like $g(r)$, it can be described as a

correlator, only this time in terms of the Fourier-transformed one-particle density

$$\rho_{\mathbf{k}}(t) = \int d\mathbf{r} e^{-i\mathbf{k}\cdot\mathbf{r}} \sum_{i=1}^{N_f} \delta(\mathbf{r} - \mathbf{r}_i(t)) = \sum_{i=1}^{N_f} e^{-i\mathbf{k}\cdot\mathbf{r}_i(t)} \quad (2.62)$$

where again N_f is the number of fluid particles, and the time dependency will be of significance only later. In terms of this density, the structure factor is then defined as [115]

$$S(\mathbf{k}) = \frac{1}{N_f} \overline{\langle \rho_{\mathbf{k}} \rho_{-\mathbf{k}} \rangle} , \quad (2.63)$$

where the negative sign in the second index ensures that $S(\mathbf{k}) \in \mathbb{R}$. The connection to $g^{(2)}(\mathbf{r})$ is established via [115]

$$\begin{aligned} S(\mathbf{k}) &= \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} \sum_{j=1}^{N_f} e^{-i\mathbf{k}\cdot\mathbf{r}_i} e^{i\mathbf{k}\cdot\mathbf{r}_j} \right\rangle} \\ &= 1 + \frac{1}{N_f} \overline{\left\langle \sum_{i \neq j}^{N_f} e^{-i\mathbf{k}\cdot(\mathbf{r}_i - \mathbf{r}_j)} \right\rangle} \\ &= 1 + \frac{1}{N_f} \overline{\left\langle \sum_{i \neq j}^{N_f} \iint d\mathbf{r} d\mathbf{r}' e^{-i\mathbf{k}\cdot(\mathbf{r} - \mathbf{r}')} \delta(\mathbf{r} - \mathbf{r}_i) \delta(\mathbf{r} - \mathbf{r}_j) \right\rangle} \\ &= 1 + \frac{1}{N_f} \iint d\mathbf{r} d\mathbf{r}' e^{-i\mathbf{k}\cdot(\mathbf{r} - \mathbf{r}')} \rho^{(2)}(\mathbf{r}, \mathbf{r}') \\ &= 1 + \frac{N_f}{V} \int d\mathbf{r} e^{-i\mathbf{k}\cdot(\mathbf{r} - \mathbf{r}')} g^{(2)}(\mathbf{r}) . \end{aligned} \quad (2.64)$$

In a homogeneous and isotropic fluid, the dependence of the structure factor on the wave vector \mathbf{k} is simplified to a dependence upon its modulus k , with the relation between $S(k)$ and $g(r)$ [cf. Eq. (2.61)] in this situation reading [115]

$$S(k) = 1 + \frac{N_f}{V} 4\pi \int dr r^2 g(r) \frac{\sin kr}{kr} . \quad (2.65)$$

The static structure factor is an important observable because it is directly accessible to experiments such as light scattering or neutron scattering [3, 23, 36, 95, 104, 115, 185–187, 193, 214]. As evidenced by Eqs. (2.64) and (2.65), it contains essentially the same information as $g(r)$. Despite the more abstract nature of Fourier space, $S(k)$ more readily allows access to two pieces of information:

- (i) $S(0)$ is directly related to the compressibility of the corresponding material [115], and
- (ii) the height of the first peak of $S(k)$ (cf. Sec. 3.5) provides a simple criterion for distinguishing a crystalline from a fluid-like phase [116].

The method alluded to under item (ii) is commonly known as the ‘‘Hansen-Verlet criterion’’, which essentially states that any substance in which the first peak of the structure factor exceeds a value of ~ 3 is most probably a crystal. The relevance of this criterion in the context of this work will be discussed in Sec. 3.2.3.

2.5.3 Dynamic observables

Dynamic observables provide the key information to characterize systems in which dynamic arrest occurs. In consequence, analyzing this kind of observables consumes the lion’s share of this work’s results part (Chapter 3).

2.5.3.1 Mean squared displacement

The mean squared displacement (MSD) is a relatively-simple observable that quantifies the average distance traveled by one of the N_f fluid particles after some time t . It is one of the most frequently analyzed quantities in computer simulations concerned with the dynamics of a system [9, 16, 18, 75, 113, 115, 148, 154, 156, 172, 197, 198, 219, 242, 246, 252, 273]. The definition of the MSD derives from Eq. (2.51), and it is reminiscent of an auto-correlator, being defined by

$$\begin{aligned} \delta r^2(t) &= \overline{\lim_{\tau \rightarrow \infty} \frac{1}{\tau} \int_0^\tau ds \frac{1}{N_f} \sum_{i=1}^{N_f} |\mathbf{r}_i(t+s) - \mathbf{r}_i(s)|^2} \\ &= \frac{1}{N_f} \left\langle \sum_{i=1}^{N_f} |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \right\rangle, \end{aligned} \quad (2.66)$$

where $\mathbf{r}_i(t)$ is the location of particle i at time t . Notably, $\delta r^2(t)$ consists of N_f independent terms, each of which involves only a single one of the N_f fluid particles. Observables of this kind are commonly called “self-averaging”; they are particularly suited for numerical computations since the additional average reduces the associated statistical error by a factor $\sqrt{N_f}$ (cf. Sec. 2.5.5).

Like other dynamic observables, $\delta r^2(t)$ in glassy systems may exhibit features at very different time scales. A helpful means to identify and quantify these features in the MSD is its “logarithmic derivative”

$$z(t) = \frac{d[\log \delta r^2(t)]}{d[\log t]} \quad (2.67)$$

which in an assumed functional dependence

$$\delta r^2(t) \propto t^z \quad (2.68)$$

represents the momentary value of the exponent z . The value of $z(t)$ in a particular time regime is an indicator for the physical processes that take place in this regime. For instance, an uninterrupted ballistic motion by all particles entails $z = 2$, whereas diffusive motion due to stochastic events obeys $z = 1$ (see, e.g., [113, 191]). The presence of other values of z in QA systems (along with possible explanations) will be discussed in Sec. 3.2.4.

Although not of further relevance to this work, for completion it shall not remain unmentioned that the transport coefficient associated with $\delta r^2(t)$ is—up to a constant—

the diffusion coefficient D . The pertinent relation was found by Einstein [74] and, independently^{2.74}, by Smoluchowski [274] to be

$$D = \lim_{t \rightarrow \infty} \frac{1}{6t} \overline{|A(t) - A(0)|^2}, \quad (2.69)$$

i.e., comparing with Eq. (2.51), it is $\gamma = 3D$ in the case of the mean squared displacement. The reason for which we did not conduct a general investigation on the diffusion coefficient is that Eq. (2.69) is valid only if $\lim_{t \rightarrow \infty} z(t) = 1$, which, as will be shown in Sec. 3.2.4, is *not* always the case in QA systems.

2.5.3.2 Collective intermediate scattering function

In this work, we consider multiple different variants of intermediate scattering functions (ISFs). The motivation for this procedure stems partly from theoretical predictions [159–162], and in other parts from the aim of analyzing as many aspects of QA systems as possible. ISFs are probably the most popular quantity to observe in systems with dynamic arrest [36, 62, 87, 88, 90, 93, 104, 115, 147, 148, 154, 155, 197, 198, 241, 265, 272], the reason for which will become clear shortly. Since the functional features of ISFs in the context of glass-forming systems are detailed in Sec. 1.2, in this section we shall instead focus on general features of ISFs and on technical aspects. The pertinent discussion (subsequent to the definition of this section’s ISF) is then to be regarded common to all ISFs considered in this work.

The collective ISF, $F(k, t)$, which we are going to consider first, is a quantity closely related to the static structure factor (see Sec. 2.5.2.2). Like $S(k)$, it is defined as a correlator of k -space densities [Eqs. (2.62) and (2.63)], with the additional aspect of considering these densities at different times:

$$F(k, t) = \frac{\overline{\langle \rho_{\mathbf{k}}(t) \rho_{-\mathbf{k}}(0) \rangle}}{\overline{\langle \rho_{\mathbf{k}}(0) \rho_{-\mathbf{k}}(0) \rangle}} = \frac{\overline{\langle \rho_{\mathbf{k}}(t) \rho_{-\mathbf{k}} \rangle}}{N_f S(k)}. \quad (2.70)$$

The collective ISF encodes how the density correlations represented by $S(k)$ diminish as time elapses, and thereby quantifies the *structural relaxation* of the material. Since structural relaxation is the slowest of all processes in glass-forming systems [93, 104], examining ISFs is the method of choice for quantifying the macroscopic behavior of a system undergoing dynamic arrest.

Among the technical issues alluded to is the wave vector k . When visually examining ISFs, k is commonly treated as a parameter while the time t is considered as the continuous variable. There are two widespread methods of analyzing series of ISFs: In the first one, $F(k, t)$ is evaluated for a single set of external parameters

^{2.74}Equation (2.69) was first published by Einstein in one of his famous 1905 papers. However, Einstein’s work was in fact *preceded* by Smoluchowski, who, however, hesitated to publish his findings due his feeling of lacking experimental evidence. Smoluchowski eventually published his derivation of Eq. (2.69) only in direct response to Einstein’s work.

(temperature, pressure, etc.) but for multiple wave vectors. This allows to identify dynamical processes that take place at different length scales (cf. Sec. 2.5.2.2). The second method is to choose a fixed wave vector \tilde{k} and to instead vary the external parameters. Due to the inevitable uncertainty of observable averages calculated from computer simulations, it is reasonable from purely computational grounds to focus on the wave vector $\tilde{k} = k_{\max}$ for which $F(k_{\max}, 0) = S(k_{\max}) = \max(S(k))$ since for this wave vector the relative statistical error is typically the smallest. In our systems (as in most liquids), k_{\max} corresponds to the first peak of the static structure factor (see Sec. 3.2.3 and Fig. 3.5 therein), which renders the choice of k_{\max} physically sensible as well since it corresponds to structures comparable in size to the system's particles.

Unfortunately, in systems with quenched disorder, the collective ISF can never fully relax to zero whenever the matrix packing fraction $\phi_m > 0$. The reason for this is that in these systems, the one-particle density as defined in Eq. (2.62) is inhomogeneous even if thermodynamically averaged, i.e., $\langle \rho_{\mathbf{k}} \rangle \neq \text{const}$ (see Sec. 2.5.1.5). These frozen-in density fluctuations, imprinted by the matrix particles, entail that one-particle densities at different times *always* retain some degree of correlation, i.e., $\lim_{t \rightarrow \infty} \langle \rho_{\mathbf{k}}(t) \rho_{-\mathbf{k}} \rangle > 0$. An additional disorder average does *not* change this inequality since a positive correlation results from *every* matrix realization; this is simply due to the fact that $\rho_{\mathbf{k}}(t)$ is always positive (and so is a product of these densities). This finding is a major problem since normally a nonrelaxing dynamic correlator is a hallmark of an arrested phase. Therefore, unfortunately, $F(k, t)$ is useless for distinguishing nonarrested and arrested (glassy) phases of QA systems.

2.5.3.3 Connected intermediate scattering function

The failure of the collective ISF to differentiate between arrested and nonarrested phases in QA systems can be remedied by considering a slightly altered correlator. Recalling the problem, $F(k, t)$ in the presence of quenched disorder does not relax due to frozen-in density fluctuations. A straightforward alternative is to instead consider densities from which we *subtract* those imprinted fluctuations, and to construct correlators of these altered densities. More formally, we can define the fluctuations of $\rho_{\mathbf{k}}(t)$ relative to its thermodynamic average as

$$\delta \rho_{\mathbf{k}}(t) = \rho_{\mathbf{k}}(t) - \langle \rho_{\mathbf{k}} \rangle . \quad (2.71)$$

Using this object, we can first define the static correlator

$$S_c(k) = \frac{1}{N_f} \overline{\langle \delta \rho_{\mathbf{k}} \delta \rho_{-\mathbf{k}} \rangle} , \quad (2.72)$$

which is commonly referred to as the ‘‘connected structure factor’’. Accordingly, the auto-correlator involving these densities

$$F_c(k, t) = \frac{\overline{\langle \delta \rho_{\mathbf{k}}(t) \delta \rho_{-\mathbf{k}}(0) \rangle}}{\overline{\langle \delta \rho_{\mathbf{k}}(0) \delta \rho_{-\mathbf{k}}(0) \rangle}} = \frac{\overline{\langle \delta \rho_{\mathbf{k}}(t) \delta \rho_{-\mathbf{k}} \rangle}}{N_f S_c(k)} \quad (2.73)$$

is called the “connected ISF”. Unlike $F(k, t)$, this quantity *does* decay to zero for $t \rightarrow \infty$, provided that the system is ergodic—just as desired.

Unfortunately, the capability of $F_c(k, t)$ to characterize *arrested* systems is questionable at best (cf. Sec. 2.5.5). The reason for this is that neither $F_c(k, t)$ nor $S_c(k)$ is well defined in nonergodic systems; this is owed to Eq. (2.71), via which $\delta\rho_{\mathbf{k}}$ involves the ergodic average $\langle\rho_{\mathbf{k}}\rangle$. Consequences are that $F_c(k, t)$ does not follow any particular relaxation pattern in the arrested case; instead, it may for instance assume (meaningless) values below zero due to the subtraction in $\delta\rho_{\mathbf{k}}$, or even relax to zero due to some canceling of errors. The latter, importantly, implies that arrest does not always lead to $F_c(k, t \rightarrow \infty) \neq 0$. In turn, $F_c(k, t \rightarrow \infty) \neq 0$ does not necessarily result from arrest; such deviations from zero can instead also be caused by numeric and (other) systematic errors.

Another perspective on the issue of the decay of $F_c(k, t)$ in nonergodic systems is added by explicitly inserting the density fluctuations [Eq. (2.71)] into the definition of the connected ISF [Eq. (2.73)]. This procedure gives rise to the following relationship between the connected and the collective ISF:

$$\begin{aligned} N_f S_c(k) F_c(k, t) &= \overline{\langle \delta\rho_{\mathbf{k}}(t) \delta\rho_{-\mathbf{k}}(0) \rangle} \\ &= \overline{\langle [\rho_{\mathbf{k}}(t) - \langle\rho_{\mathbf{k}}\rangle] [\rho_{-\mathbf{k}}(0) - \langle\rho_{-\mathbf{k}}\rangle] \rangle} \\ &= \overline{\langle \rho_{\mathbf{k}}(t) \rho_{-\mathbf{k}}(0) \rangle} - \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} - \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} + \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} \\ &= N_f S(k) F(k, t) - \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} . \end{aligned} \quad (2.74)$$

In the third step, we have used that $\langle\rho_{\mathbf{k}}(0)\rangle = \langle\rho_{\mathbf{k}}\rangle$ due to the arbitrary time origin upon averaging, and in the fourth step, the definition of $F(k, t)$ was exploited [Eq. (2.70)]. The relationship in Eq. (2.74) reveals that $F(k, t)$ and $F_c(k, t)$ are related by a mere offset and a time-independent factor and therefore inherently feature the same time scales for dynamic phenomena.

In the following, we will derive two relationships that allow to evaluate the offset $\overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle}$. The first one is obtained by expanding the connected structure factor [Eq. (2.72)] in a way similar to Eq. (2.74), yielding

$$\begin{aligned} N_f S_c(k) &= \overline{\langle \delta\rho_{\mathbf{k}} \delta\rho_{-\mathbf{k}} \rangle} \\ &= \overline{\langle [\rho_{\mathbf{k}} - \langle\rho_{\mathbf{k}}\rangle] [\rho_{-\mathbf{k}} - \langle\rho_{-\mathbf{k}}\rangle] \rangle} \\ &= \overline{\langle \rho_{\mathbf{k}} \rho_{-\mathbf{k}} \rangle} - \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} - \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} + \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} \\ &= N_f S(k) - \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} . \end{aligned} \quad (2.75)$$

In the last line, we have used the definition of the (full) static structure factor from Eq. (2.63). Using the notion of the “blocked structure factor” $S_b(k)$, this can be re-expressed to yield the first relationship as

$$S(k) - S_c(k) = \frac{1}{N_f} \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} = S_b(k) . \quad (2.76)$$

The second way to evaluate $\overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle}$ makes use of the general fact that any two quantities in an ergodic system are uncorrelated if considered at infinite temporal separation [115, 161, 162], i.e., that

$$\lim_{t \rightarrow \infty} \langle A(t)B \rangle = \langle A \rangle \langle B \rangle . \quad (2.77)$$

Applying this relation to the collective ISF, we obtain the second relationship as

$$\lim_{t \rightarrow \infty} S(k) F(k, t) = \lim_{t \rightarrow \infty} \frac{1}{N_f} \overline{\langle \rho_{\mathbf{k}}(t) \rho_{-\mathbf{k}} \rangle} = \frac{1}{N_f} \overline{\langle \rho_{\mathbf{k}} \rangle \langle \rho_{-\mathbf{k}} \rangle} = S_b(k) . \quad (2.78)$$

Equations (2.76) and (2.78) represent two independent methods to evaluate $S_b(k)$. In combination with

$$S(k) F(k, t) = S_c(k) F_c(k, t) + S_b , \quad (2.79)$$

which is a simple re-expression of Eq. (2.74), this provides a more stringent means to assess the assumption of ergodicity than simple observations of $F_c(k, t)$'s decay pattern. Nevertheless, by the discussion further above, it is clear that characterizations of nonergodic systems via $F_c(k, t)$ are in any case to be taken with great care. This issue has to be born in mind particularly in the context of comparing $F_c(k, t)$ with theoretical predictions by mode-coupling theory (cf. Sec. 1.4.3).

2.5.3.4 Single-particle intermediate scattering function

The single-particle—or “self”—ISF represents a subset of the information contained in the collective ISF. Like the latter, it is a density–density correlator, the only difference being that it correlates *single-particle* densities:

$$\rho_{\mathbf{k}}^{(i)}(t) = \int d\mathbf{r} e^{i\mathbf{k} \cdot \mathbf{r}} \delta(\mathbf{r} - \mathbf{r}_i(t)) = e^{-i\mathbf{k} \cdot \mathbf{r}_i(t)} . \quad (2.80)$$

Using this expression, the single-particle ISF is defined as

$$F_s(k, t) = \frac{1}{N_f} \sum_{i=1}^{N_f} \overline{\langle \rho_{\mathbf{k}}^{(i)}(t) \rho_{-\mathbf{k}}^{(i)}(0) \rangle} . \quad (2.81)$$

Like the mean squared displacement, $F_s(k, t)$ is a self-averaging quantity (cf. Sec. 2.5.3.1), which typically renders this quantity much more precise (in terms of statistical errors) than the collective ISF.

Nijboer and Rahman [115, 200] showed that it is possible to expand the self ISF in terms of the particle displacement as

$$F_s(k, t) = \exp \left\{ -\frac{1}{6} k^2 \delta r^2(t) \left[1 + \mathcal{O}(k^4) \right] \right\} . \quad (2.82)$$

In this expression, the terms involving k^4 or higher powers of k quantify the degree to which the particle motion is “non-Gaussian”. In the “Gaussian approximation”, in which these terms vanish, it is therefore

$$\lim_{k \rightarrow 0} F_s(k, t) = \lim_{k \rightarrow 0} \exp \left\{ -\frac{1}{6} k^2 \delta r^2(t) \right\} = 1 - \frac{1}{6} k^2 \delta r^2(t) , \quad (2.83)$$

where the last equality results from a simple Taylor expansion. The message conveyed by Eq. (2.83) is that in the limit of infinitely-long wave vectors, the self ISF encodes the same information as the mean squared displacement. It is, however, important to note that one of the hallmarks of complex liquids systems is the *breakdown* of the Gaussian approximation [53, 62, 75, 90, 93, 104, 113, 155, 241, 246], meaning that the observed dynamic properties of such systems may differ depending on whether they are identified via $F_s(k, t)$ or $\delta r^2(t)$.

2.5.3.5 Decay quantification and long-time value subtraction

In the context of glass-forming systems, an important feature of any ISF $F_x(k, t)$ is the time $\tau_x^{(\alpha)}$ for it to decay to its long-time value $f_x(k) = \lim_{t \rightarrow \infty} F_x(k, t)$. In simple glass-forming systems, it has been found that usually the final (“ α ”) relaxation can be fitted excellently by a “stretched exponential” in which the desired quantities $\tau_x^{(\alpha)}$ and $f_x(k)$ are parameters (cf. Sec. 1.2 and Refs. [3, 36, 62, 87, 90, 93, 104, 154, 155, 197, 198, 220, 241, 265, 272]). In QA systems, however, such a fitting is thwarted by the considerable variation in the decay patterns of ISFs (see Secs. 3.2.5 and 3.2.7). Therefore, to infer values for $\tau_x^{(\alpha)}$ and $f_x(k)$, simpler definitions of these quantities need to be employed, our choice of which shall be detailed in the following.

The decay time $\tau_x^{(\alpha)}$ we defined in this work as the time required for the corresponding $F_x(k, t)$ to relax below some value c . Specifically, we decided to always use the value $c = 0.1$, which—while in fact arbitrary—is reasonable in most of the investigated systems even without further treatment (see below). This is because this value is (i) small enough to indeed quantify the final relaxation of any $F_x(k, t)$, and (ii) large enough to allow most $F_x(k, t)$ ’s to actually decay below c . Concerning the latter issue, it is important to note that it is *impossible* to infer $\tau_x^{(\alpha)}$ if $f_x(k) > c$, which unfortunately constitutes a major problem since the possible values of $f_x(k)$ in QA systems cover almost the entire interval $[0, 1]$ (see Secs. 3.2.5 and 3.2.7).

Fortunately, the conflicting requirements to the value of c can be resolved by first determining the long-time value $f_x(k)$, and by then rescaling the correlator of interest $F_x(k, t)$ such that its values cover the entire interval $[0, 1]$ instead of just the interval $[f_x(k), 1]$. The main difficulty here is to reliably identify $f_x(k)$ when data are available only for a finite range of time $[0, t_{\max}]$. To accomplish the recasting under this condition, we developed a heuristic algorithm consisting of the following steps:

- (1) Given some wave vector \tilde{k} , determine the global minimum of the ISF of interest $F_x(k, t)$:

$$M_{\tilde{k}} = \min \left(F_x(\tilde{k}, t) \mid 0 \leq t \leq t_{\max} \right) . \quad (2.84)$$

- (2) Adopt the following (standard) rule to compute the standard deviation of $F_x(\tilde{k}, t)$ in the time range $[t_0, t_{\max}]$:

$$S_{\tilde{k}}(t_0) = \left(\left[\frac{1}{\Delta t} \int_{t_0}^{t_{\max}} dt t^2 F_x(\tilde{k}, t) \right] - \left[\frac{1}{\Delta t} \int_{t_0}^{t_{\max}} dt F_x(\tilde{k}, t) \right]^2 \right)^{1/2} . \quad (2.85)$$

Here, the shorthand $\Delta t = t_{\max} - t_0$ has been used, and the meaning of t_0 will become clear below. The first bracketed term in Eq. (2.85) is the ISF's second moment in the considered range, and the second bracketed term is its average value.

- (3) Compute $S_{\tilde{k}}(t_0)$ for a series of progressively-increasing values of t_0 , starting at zero. Exit this series as soon as $S_{\tilde{k}}(t_0^*)/(1 - M_{\tilde{k}}) < a$, where a is some pre-set tolerance threshold, and t_0^* is the value of t_0 upon exiting the series. If the above inequality is fulfilled, then $F_x(\tilde{k}, t)$ is considered to be constant in the time range $[t_0^*, t_{\max}]$.
- (4) If $t_{\max}/t_0^* > b$, where b is another pre-set threshold, then identify $f_x(\tilde{k}) = M_{\tilde{k}}$ and define

$$\hat{F}_x(\tilde{k}, t) = \frac{F_x(\tilde{k}, t) - f_x(\tilde{k})}{1 - f_x(\tilde{k})}. \quad (2.86)$$

Otherwise, $f_x(\tilde{k}) < M_{\tilde{k}}$, that is, $F_x(k, t)$ has not reached its long-time value within the time range considered.

The algorithm features two parameters, a and b , to adjust the accuracy of the identification $f_x(\tilde{k}) = M_{\tilde{k}}$. In this work, we chose the values $a = 0.008$ and $b = 10$, thereby requiring $F_x(\tilde{k}, t)$ to vary (on average) less than one percent over the final available time decade to pronounce it “relaxed”.

By design, any ISF $\hat{F}_x(k, t)$ resulting from the above algorithm fully relaxes to zero. Since none of the parameters used in Eq. (2.86) depend upon time, the temporal scales of the dynamic features in $\hat{F}_x(k, t)$ and $F_x(k, t)$ are precisely the same. This finally allows to apply the simple decay-below- c criterion (outlined in the beginning of this section) to any $F_x(k, t)$, regardless of the value of $f_x(k)$ —as long as $f_x(k)$ can be pinpointed. We will put this method to use when attempting to compare the relaxation time scales of the self ISF to those of the connected ISF (see Sec. 3.2.8).

2.5.3.6 Van Hove correlation function

The Van Hove correlation function was introduced by Léon Van Hove to describe “Correlations in Space and Time” (the actual title of his 1957 milestone paper [264]). Van Hove's function complements the picture of static and dynamic correlations functions in that it is both the Fourier transform of the collective ISF (see Sec. 2.5.3.2) and the time-dependent equivalent of the pair correlation function (cf. Sec. 2.5.2.1). Its definition is similar to Eq. (2.58), with a significant difference being that it is defined in terms of the real-space one-particle density

$$\rho(\mathbf{r}, t) = \sum_{i=1}^{N_f} \delta(\mathbf{r} - \mathbf{r}_i(t)), \quad (2.87)$$

a quantity reminiscent of $\rho^{(2)}(\mathbf{r}, \mathbf{r}', t, t')$ and closely related to $\rho_{\mathbf{k}}(t)$ [see Eqs. (2.57) and (2.62), respectively]. The Van Hove function (VHF) is then a full dynamic correlator of two such densities [115]:

$$\begin{aligned}
 G(\mathbf{r}, t) &= \frac{1}{N_f} \overline{\langle \rho(\mathbf{r}, t) \rho(0, 0) \rangle} \\
 &= \frac{1}{N_f} \overline{\langle \rho(\mathbf{r} + \mathbf{r}', t) \rho(\mathbf{r}', 0) \rangle} \\
 &= \int d\mathbf{r}' \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} \sum_{j=1}^{N_f} \delta(\mathbf{r} + \mathbf{r}' - \mathbf{r}_i(t)) \delta(\mathbf{r}' - \mathbf{r}_j(0)) \right\rangle} \quad (2.88) \\
 &= \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} \sum_{j=1}^{N_f} \delta(\mathbf{r} - [\mathbf{r}_i(t) - \mathbf{r}_j(0)]) \right\rangle} .
 \end{aligned}$$

The VHF is normalized such that $\int d\mathbf{r} G(\mathbf{r}, t) = N_f$. As for the radial distribution function, in a homogeneous and isotropic system, no information is lost when reducing the dependence upon \mathbf{r} to a dependence upon r . Like for $g(r)$, in this case, typically an additional factor $1/(4\pi r^2)$ is added to the normalization, leaving us with

$$G(r, t) = \frac{1}{4\pi r^2} \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} \sum_{j=1}^{N_f} \delta(r - |\mathbf{r}_i(t) - \mathbf{r}_j(0)|) \right\rangle} . \quad (2.89)$$

Being a sum, the Van Hove function can easily be split into two contributions [115, 264],

$$G(r, t) = G_s(r, t) + G_d(r, t) , \quad (2.90)$$

where $G_s(r, t)$ is the “self” and $G_d(r, t)$ the “distinct” part of the VHF, defined by

$$G_s(r, t) = \frac{1}{4\pi r^2} \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} \delta(r - [\mathbf{r}_i(t) - \mathbf{r}_i(0)]) \right\rangle} , \quad \text{and} \quad (2.91)$$

$$G_d(r, t) = \frac{1}{4\pi r^2} \frac{1}{N_f} \overline{\left\langle \sum_{i \neq j}^{N_f} \delta(r - [\mathbf{r}_i(t) - \mathbf{r}_j(0)]) \right\rangle} . \quad (2.92)$$

Comparing Eq. (2.92) with Eq. (2.58), we notice that it is actually the *distinct* part of the Van Hove function—not the full $G(r, t)$ —that (up to a constant factor) constitutes the time-dependent version of the radial distribution function.

In this work, we shall be interested exclusively in the *self* part of the VHF. Focusing on $G_s(r, t)$ is common in investigations of glass-forming systems [53, 75, 87, 89, 111, 113, 126, 154, 156, 246, 273] since the self part more readily provides information about the average local particle motion than does the distinct part. This capability of $G_s(r, t)$ plays an important role in identifying and quantifying the “cage picture” and the associated notion of “dynamic heterogeneities”, as discussed in Secs. 1.2, 3.2.9, and 3.4.3.

Further, it is easy to show that the self part of the Van Hove function is the spatial Fourier transform of the single-particle ISF [Eq. (2.81)]. This entails that $G_s(r, t)$ and $F_s(k, t)$ encode the same information. While considering $G_s(r, t)$ might therefore seem redundant, for one its being a real-space quantity makes interpretation more intuitive. Moreover, when plotted, the VHF is commonly considered with the distance r as the continuous variable and the time t as a parameter, thereby complementing the views delivered by the self ISF. A consequence of using time as the parameter is that the evolution of $G_s(r, t)$ has to be encoded in a series of curves. Since commonly merely one such series is collected in a single plot, investigations of the Van Hove function are typically limited to considering only one combination of the system's external parameters at a time.

Like for the single-particle ISF, a connection can be established between the self part of the VHF and the mean squared displacement. In fact, in the case of $G_s(r, t)$ the connection is even more natural, with $\delta r^2(t)$ being simply the *second moment* of the Van Hove function's self part [104, 113, 154, 246]:

$$\begin{aligned} \delta r^2(t) &= \int dr r^2 G_s(r, t) \\ &= \int dr r^2 \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} \delta(r - [\mathbf{r}_i(t) - \mathbf{r}_i(0)]) \right\rangle} \\ &= \frac{1}{N_f} \overline{\left\langle \sum_{i=1}^{N_f} |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \right\rangle} \end{aligned} \quad (2.93)$$

Here, the normalization of $G_s(r, t)$ is the same as that of the full Van Hove function in Eq. (2.88). Obviously, taking the second moment of the VHF's self part is equivalent to the combined Gaussian approximation and $k \rightarrow 0$ limit in the single-particle ISF [cf. Eq. (2.83)]. However, it is worth noting that the Gaussian approximation was originally formulated in terms of the VHF, where it equivalently states that $G_s(r, t)$ is Gaussian at all times, i.e., $G_s(r, t) = A \exp\{-B r^2\}$, where A and B are two positive constants [115]. As is clear from Secs. 1.2 and 3.2.9, a hallmark of glassy systems is the non-Gaussianity of the Van Hove function, which underlines the caution that the equivalence of the self ISF and the MSD [Eq. (2.83)] has to be taken with.

2.5.4 Accessible volume quantifiers

We now shift our attention to quantities that characterize the accessible space in hard-sphere (HS) QA systems. Bearing in mind that the primary aim of this work is still to investigate the *fluid* component of a QA mixture—not the matrix—, we limit our study to quantities significant to that aim. Specifically, we decided to consider the overall amount of accessible volume, the presence of percolation, the fraction of accessible volume located in traps, and (in some more detail) the distribution of trap sizes. The evaluation of either of these quantities is dependent, in one way or another, upon data generated by the algorithms in Sec. 2.4. Further, similar to the procedure described in

Sec. 2.5.1.5, all of these quantities are subject to a compulsory average over different matrix realizations; as in the case of observables, this average (where applicable) will be indicated by the symbol $\overline{\dots}$.

2.5.4.1 Accessibility probability: grains, D-faces, D-cells

Possibly the simplest quantitative characteristic of the accessible volume in an HS-QA system is its overall amount. To measure the latter, no knowledge is required about the extent and connectivity of the voids in a system; the necessary information can instead be obtained for instance from the first step in the rastering method (Sec. 2.4.6.2), where it is identified which grains in a system are accessible. For simplicity, in place of the actual amount of accessible volume—which depends upon the system size—we will consider here the fraction of the system’s overall volume that is constituted by accessible volume. By design of the grains, this quantity is approximated by the probability $P_{\text{grain}}(\phi_m)$ that a random grain is accessible in an HS-QA system with matrix packing fraction ϕ_m .

The significance of $P_{\text{grain}}(\phi_m)$ lies in its description of how much space is available to a single tracer fluid particle for propagation in the matrix of an HS-QA system. In fact, Mittal et al. [196] suggested that $P_{\text{grain}}(\phi_m)$ is the sole parameter determinative of the dynamics of such a tracer—provided that all particles in the QA system interact hard sphere-like and that the matrix packing fraction is well below ϕ_m^* , the percolation threshold of the accessible volume (cf. Sec. 2.5.4.2). This is particularly interesting in view of the fact that it is also possible to obtain a highly-accurate *analytic* expression for $P_{\text{grain}}(\phi_m)$. The derivation of this expression is based on the fact that $P_{\text{grain}}(\phi_m)$ is related to the excess chemical potential μ , which quantifies the cost of inserting an additional particle into a system. The derivation starts with the equation of state for a monocomponent, monodisperse HS fluid; for this we use the Carnahan-Starling (CS) approximation,

$$\frac{\beta_m P_m^{\text{CS}}}{\rho_m} = \frac{1 + \phi_m + \phi_m^2 - \phi_m^3}{(1 - \phi_m)^3}, \quad (2.94)$$

which was introduced by Carnahan and Starling in Eq. (10) of Ref. [44] and which constitutes a superb approximation of the real equation of state of HS fluids [115]. Since the fluid described by Eq. (2.94) is supposed to form the matrix of an HS-QA system, all quantities in the equation pertain to the fluid from which that matrix was quenched. This is indicated by the subscript “m”, where specifically P_m denotes the pressure, $\beta_m = 1/(k_B T_m)$ the inverse temperature, ρ_m the number density, and ϕ_m the packing fraction of that particular HS fluid. From Eq. (2.94), the expression

$$\beta_m \mu_m^{\text{CS}}(\phi_m) = \frac{8\phi_m - 9\phi_m^2 + 3\phi_m^3}{(1 - \phi_m)^3} \quad (2.95)$$

for the excess chemical potential was obtained by Lee in Eq. (2.7) of Ref. [170]. Via the equation

$$P_{\text{grain}}^{\text{CS}}(\phi_m) = \exp \left\{ -\beta_m \mu_m^{\text{CS}}(\phi_m) \right\}, \quad (2.96)$$

which Reiss et al. derived in Eq. (2.11) of Ref. [221], the excess chemical potential in turn determines the amount of accessible volume in the HS fluid, with the latter therefore being a function solely of ϕ_m . Notably, in Eq. (2.96), the quantity $P_{\text{grain}}^{\text{CS}}$ is ignorant of the concept of grains; the subscript merely serves to indicate that $P_{\text{grain}}^{\text{CS}}$ and P_{grain} describe the same physical quantity by different means.

From the individual elements of a Delaunay tessellation (Sec. 2.4.3), it is possible to compute a number of quantities that provide information relatively similar to that encoded in P_{grain} . Of these, we will consider here the following:

- (I) the probability $P_{\text{cell}}(\phi_m)$ that a random Delaunay cell is accessible, and
- (II) the probability $P_{\text{face}}(\phi_m)$ that a random Delaunay face is crossable.

Here, the criteria for accessibility and crossability are those defined in steps (1) and (2) in the beginning of Sec. 2.4.3. The similarity of P_{cell} and P_{face} to P_{grain} is a consequence of D-cells, D-faces, and grains being in fact merely slightly different local objects, all of which are considered accessible according to a single representative point. Qualitatively, the three probabilities can therefore be expected to show a similar dependence upon ϕ_m , which allows to cross-check conclusions drawn from either of these quantities. Notably, for the determination of neither P_{cell} nor P_{face} it is necessary to perform a percolation analysis.

2.5.4.2 Probabilities for percolation and trapping

Suppose now that data have been established about the distinct voids in an HS-QA system (Sec. 2.4.5), and that for each of these voids, their size in terms of accessible volume has been determined (Sec. 2.4.6). Before considering quantifiers that operate on this information, it is important to realize that analyzing void sizes inherently involves the concept of percolation, which—as mentioned in Sec. 1.3.3.3—is well defined only for an infinitely-large system. It is therefore of significance that all systems investigated in this work are subject to periodic boundary conditions (PBCs, cf. Secs. 2.2.2.1 and 2.4.5), the impact of these conditions being that the occurrence of percolation depends not only upon the matrix packing fraction, ϕ_m , but also upon the system size (i.e., the size of the periodically-replicated system images). However, if a quantifier is well designed, then its relevant features depend *monotonically* upon the size of the system, which then allows meaningful extrapolations to infinite systems. We will indicate dependencies on the system size by introducing the number of matrix particles, N_m , as a parametric argument to any quantity affected by this effect.

Perhaps the simplest quantity to obtain from the void information is the probability $P_{\text{percol}}(\phi_m; N_m)$ that an HS-QA system contains a percolating void. P_{percol} in an infinitely-large system is a step function [248] centered about ϕ_m^* , the value of ϕ_m at which the percolation transition of the accessible volume takes place. P_{percol} therefore provides a natural means to estimate ϕ_m^* ; this holds even if the step function is smeared out due to finite system sizes. Notably, in the latter case it is of the utmost importance

that multiple system realizations be sampled since P_{percol} for a single realization can only be either zero or unity.

Another straightforward quantity to investigate is $P_{\text{trap}}(\phi_m; N_m)$, the fraction of accessible volume belonging to traps (as opposed to the percolating void). In an infinitely-large system with a sharp percolation transition, P_{trap} is unity for $\phi_m > \phi_m^*$ since for these ϕ_m values, no percolating void exists. However, P_{trap} does *not* take the shape of a step function even in an infinite system since below ϕ_m^* there always exist traps—irrespective of the system size. Given that P_{trap} (like P_{percol}) is smeared out for finite systems, this non-step shape is unfortunate since it complicates obtaining the infinite-size value of P_{trap} for $\phi_m < \phi_m^*$ from a finite-size extrapolation. Therefore, it is less practicable to extract ϕ_m^* from P_{trap} than from P_{percol} . However, the significance of P_{trap} is a different one anyway: as a consequence of the random insertion of fluid particles into the matrix structure (cf. Sec. 2.3), P_{trap} relatively accurately^{2.75} reflects the average fraction of fluid particles that are located in a trap (the “trapping fraction”). This allows in principle to compute an observable for the full fluid from a weighted average of the same observable for the trapped and the free particles, with the weight of the trapped particles being an appropriate function of P_{trap} .

2.5.4.3 Trap-size distributions

We now turn to characterizing the actual sizes of the voids in an HS-QA system. The first step in doing so is to realize that the volume of the percolating void(s) in a system with matrix packing fraction ϕ_m and matrix particle number N_m can be expressed as

$$V_{\text{percol}}(\phi_m; N_m) = \left[\frac{\pi}{6} N_m \phi_m^{-1} \right] P_{\text{grain}}(\phi_m) [1 - P_{\text{trap}}(\phi_m; N_m)] , \quad (2.97)$$

where $(\pi/6)N_m\phi_m^{-1}$ is the system’s volume, P_{grain} the accessible fraction of the system’s volume, and $1 - P_{\text{trap}}$ the fraction of accessible volume located in the percolating void. Assuming that there exists only one percolating void (cf. Footnote 2.35), this means that the size of this void can be calculated solely from quantities already discussed. Therefore, we will in the following consider *only traps*. The quantity central to the subsequent discussion is the probability $P(V; \phi_m, N_m)$ —shorthand “ $P(V)$ ”—that a given trap contains the accessible volume V . Being a probability distribution, $P(V)$ can numerically be approximated by the scheme

$$P(V; \phi_m; N_m) \simeq \overline{N_{\text{trap}}^{-1} \sum_{V < V_i < V + \Delta V} 1} , \quad (2.98)$$

^{2.75}In fact, P_{trap} slightly—but systematically—overestimates the fraction of trapped particles if the fluid packing fraction of a system is close to the maximum-possible value. The reason for this is that the particles in a trap will still be able to move (sometimes more, sometimes less) even if no further fluid particle can be inserted into that trap. Since this effect is negligible in the percolating void, it is the accessible volume representative of the movement in traps that, added up, accounts for the overestimation.

where N_{trap} is the number of traps in the system, V_i is the volume of trap i , and the volume ΔV is small enough to yield a suitable number of intervals but large enough to render the approximation reasonable for all V values of interest. In order to emphasize features of $P(V)$ for large V , we also consider the quantity

$$P(V; \phi_m; N_m) V \simeq \overline{N_{\text{trap}}^{-1} \sum_{V < V_i < V + \Delta V} V_i} . \quad (2.99)$$

Although similar in definition, the quantities in Eqs. (2.98) and (2.99) bear different physical significance: whereas $P(V)$ focuses on traps in that it quantifies the probability for a random trap to be of size V , its counterpart $P(V)V$ assesses the probability that a random trapped particle is located in a trap of size V . The latter interpretation suggests to compare $P(V)V$ for instance with the infinite-time limit of the Van Hove function for the trapped particles in an HS-QA system (cf. Sec. 2.5.3.6).

Remembering that $P(V)$ and $P(V)V$ depend not only upon V but also upon ϕ_m and N_m , it is reasonable to condense the details of the dependence upon V into a single representative number if dependencies upon ϕ_m and N_m are to be investigated. Two meaningful choices for such single numbers are the *first and second moments*^{2,76} of $P(V)$

$$\hat{V}_1(\phi_m; N_m) = \overline{\left[\int_0^\infty P(V)V dV \right]} / \overline{\left[\int_0^\infty P(V) dV \right]} \simeq \overline{\left[\sum_{V_i} V_i \right]} / \overline{\left[\sum_{V_i} 1 \right]} \quad \text{and} \quad (2.100)$$

$$\hat{V}_2(\phi_m; N_m) = \overline{\left[\int_0^\infty P(V)V^2 dV \right]} / \overline{\left[\int_0^\infty P(V)V dV \right]} \simeq \overline{\left[\sum_{V_i} V_i^2 \right]} / \overline{\left[\sum_{V_i} V_i \right]} , \quad (2.101)$$

where we normalized the first and the second moments with the zeroth and first moments, respectively, in order to render both \hat{V}_1 and \hat{V}_2 of dimension ‘‘volume’’. Although both expressions represent some sort of average trap size, their interpretation differs significantly: whereas \hat{V}_1 describes the average size of a randomly-chosen trap, \hat{V}_2 quantifies the average size of the trap that a random trapped particle resides in. Therefore, \hat{V}_2 is better suited to pinpoint the effects of confinement on the fluid component of an HS-QA system.

The most important characteristic of \hat{V}_2 , however, is its capability to highly accurately estimate ϕ_m^* from finite systems (ϕ_m^* , again, being the value of ϕ_m at percolation). This capability of \hat{V}_2 comes about as follows. Suppose that some trap T_j in the system is similar in size to all of the system’s other traps combined. Then, from the above interpretation and from the definition in Eq. (2.101), it is clear that the value of \hat{V}_2 will be comparable to the size of T_j . Since T_j , while obviously finite in a finite system, is infinitely large itself in an infinitely-large system, \hat{V}_2 diverges if a trap like T_j exists in a system with $N_m \rightarrow \infty$.

^{2,76}<http://mathworld.wolfram.com/RawMoment.html>

As is easy to rationalize, in an infinitely-large system, a trap like T_j can be present only for $\phi_m = \phi_m^*$ (cf. Sec. 1.3.3.3). To see this, consider first a system in which the largest void is just barely percolating, i.e., a system with $\phi_m = \phi_m^* - d\phi_m$, where $d\phi_m$ is an infinitesimal amount of matrix packing fraction. Then, the addition of a single matrix particle (i.e., of $d\phi_m$) may cut this percolating void into two traps, with these traps then obviously being very large. For $\phi_m > \phi_m^*$, no infinitely-large trap can exist since adding particles in excess of $\phi_m = \phi_m^*$ only reduces the sizes of traps. For $\phi_m < \phi_m^*$, traps are generally finite as well since it is impossible for an infinitely-large trap to have no connection with the percolating void (the latter, by definition, always exists for $\phi_m < \phi_m^*$). Therefore, only for $\phi_m = \phi_m^*$ an infinitely-large trap can be present.

In a finite system with PBCs, breaking the percolating void at $\phi_m = \phi_m^* - d\phi_m$ by adding $d\phi_m$ has slightly different consequences: instead of being almost infinitely large, the resulting traps in this case will be comparable in size to the unique part of the system. Since—by a similar reasoning as for the $N_m \rightarrow \infty$ case—traps of such size can exist only in some vicinity around the percolation threshold, \hat{V}_2 [see Eq. (2.101)] will exhibit a pronounced maximum for $\phi_m = \phi_m^*$ in finite systems. While identifying ϕ_m^* via this maximum is second in accuracy only to finite size–scaling techniques [248], it is still advisable to consider large systems, the reason being that the percolation threshold in periodically-replicated systems actually depends upon the size of the system, with ϕ_m^* increasing monotonically with that size (cf. Sec. 1.3.3.3). Relying on large systems is preferential also due to the positive scaling with the system size of the maximum’s height (see above), thereby allowing for a more clear-cut identification of ϕ_m^* .

Although \hat{V}_1 may also exhibit maxima and minima as ϕ_m is varied, these features unfortunately do not provide information about ϕ_m^* . In fact, since \hat{V}_1 is the quotient of V_{trap} (the combined size of all traps in the system) and N_{trap} (the number of those traps), and since V_{trap} is simply the product of the known quantities P_{trap} , P_{grain} , and the system’s volume [cf. Eq. (2.97)], the only original information encoded in \hat{V}_1 is N_{trap} . Although this information could be quantified more succinctly by simply counting traps, we decided to nevertheless consider \hat{V}_1 due to its mathematical similarity to \hat{V}_2 , the ready availability of $P(V)$, and the average trap size being an interesting quantity per se.

2.5.5 Equilibration and error estimation

In Sec. 2.5.1.2, we deferred the discussion about computing an observable average $\langle A \rangle_{\mathbf{Q}}$ from MD simulation data. Recalling Eq. (2.50) and suppressing therein the dependence upon the external conditions \mathbf{Q} , an observable average^{2.77} can be approximated by $\langle A \rangle \simeq \underline{A} = \frac{1}{M} \sum_{m=1}^M A(\Gamma(m \Delta t))$ if an MD simulation delivers M measurements at

^{2.77}The conventional notation of a simulational average is the symbol $\overline{\dots}$. However, since we reserved the overstrike for the average over matrix realizations (see Sec. 2.5.1.5), we instead indicate simulation averages by an underline, $\underline{\dots}$.

times equally spaced apart by Δt . For \underline{A} to reliably reproduce the true observable average $\langle A \rangle$, the probability of a certain phase-space point $\mathbf{\Gamma}$ to occur in the definition of \underline{A} has to be close to the phase-space density $f(\mathbf{\Gamma})$. This is nontrivial to ensure. By construction, the limiting factor is the duration of the simulation; therefore, we need a means to assess which span of simulated time τ_{sim} is “long enough”. The usual approach [9, 85, 219] is to examine the inherent time scales $\{\tau_i\}$ of the simulated system, and to choose $\tau_{\text{sim}} > \max\{\tau_i\}$. This condition ensures that the system is (in principle) capable of reaching any arbitrary phase-space point $\mathbf{\Gamma}$ within the allotted simulation time^{2.78}, which is an intuitively sensible condition and reflects the fact that averages of the form $\langle A \rangle$ are meaningful only in an ergodic system (see Footnote 2.13).

Before we discuss how to obtain the time scales, we note that (incidentally) being able to arrive at any $\mathbf{\Gamma}$ within some simulated time τ_{sim} is not only required for rendering \underline{A} reliable—it is also a necessary condition for *equilibrating* a system. As mentioned before (see Sec. 2.1.2), for the statistical methods employed in this work to yield meaningful results, any examined system needs to be in a state of thermal equilibrium. Therefore, a two-step procedure is imperative for the investigations in this work: first, the system of interest has to be brought to equilibrium (“equilibration run”), and only then measurements can be conducted on this system (“production run”). Remembering that the computation of reliable averages and the establishment of thermal equilibrium make the same demands on τ_{sim} , it is natural to use that span of simulation time—once determined—for fixing the duration of both the equilibration and the production run.

Coming back to the time scales, obviously the difficulty lies in *identifying* those scales; since each τ_i is associated with an observable A_i , this problem is equivalent to finding the “right” observables^{2.79}. In the context of dynamic arrest, a plethora of investigations have shown that the slowest time scale is set by the *structural relaxation* of the material [36, 53, 62, 86, 87, 89, 90, 93, 104, 154, 214, 241, 254, 269, 272, 281]. The structural relaxation—or at least aspects of it—is quantified by density–density correlators, i.e., by the ISFs and the Van Hove functions (cf. Sec. 2.5.3). The lower bound for the simulated time τ_{sim} is therefore the relaxation time of one of these observables. As we shall see in Sec. 3.2.5, in HS-QA systems, the slowest time scale is set by *single-particle* density correlators, i.e., by $F_s(k, t)$ and $G_s(r, t)$. Unfortunately, while we already discussed how to distill relaxation times from correlators (Sec. 2.5.3.5), it is not easy to implement an automated procedure that determines τ_{sim} concurrently with an MD simulation.

^{2.78}The phase space of a system—particularly that of an HS-QA system—may consist of disconnected domains separated by regions of zero occupation probability [$f(\mathbf{\Gamma}) = 0$]. In this case, the system can of course only reach the phase-space points that belong to the same domain as the initial phase-space point.

^{2.79}Although an observable may exhibit features on multiple time scales, it is only the largest of these that is relevant in the context of the simulation length. Also, it is worth noting that some of the τ_i may have the same physical origin and therefore be identical.

An obvious, yet inefficient solution would be to assess the decay of the correlators a posteriori. However, fortunately, it is also possible to extract the same time scale information from a related simpler quantity: the mean squared displacement. Extensive comparison between $\delta r^2(t)$ and $F_s(k, t)$ reveals that the MSD's logarithmic derivative $z(t)$ (see Sec. 2.5.3.1) generally remains unaltered for times beyond the relaxation of the self ISF to its long-time value $f_s(k)$. Using this rule, τ_{sim} can be set by the time required for $z(t)$ to reach its long-time value $z(\infty) = \lim_{t \rightarrow \infty} z(t)$. Unfortunately, this criterion is not easily tracked during a simulation either. However, a distinct advantage of the MSD is that in addition to time-scale features, it naturally relates to real-space length scales. This can be exploited by redefining τ_{sim} as the time that $\delta r^2(t)$ requires to arrive at some length l_{cutoff} , i.e.,

$$\delta r^2(\tau_{\text{sim}}) = l_{\text{cutoff}}^2, \quad (2.102)$$

where τ_{sim} is of the order of τ_{max} if l_{cutoff} is equal to the system's largest inherent length scale l_{max} . Although strictly speaking this merely shifts the problem from times to lengths, in the absence of a-priori knowledge about the system's dynamic features, length scales are more intuitively estimated than time scales.

In a simple monatomic system deep within the fluid regime, the only inherent length scale is set by the range of the interaction potential [115]. However, other (diverging) length scales emerge as a phase transition is approached [108, 238]. While the exact nature of the glass transition is still a matter of debate, many works provide evidence that this transition is accompanied by a diverging length scale of some kind (cf. Sec. 1.2 and Refs. [19, 25–27, 34, 43, 53, 62, 71, 78, 79, 104, 135, 137, 139, 154, 167, 241, 277, 282]). The type of disorder introduced in the QA protocol entails additional length scales that likewise may diverge (see Sec. 1.3.4.2 and Refs. [45, 111, 113, 148, 150, 159–162, 196, 252]). Since any diverging length scale can be translated to a diverging time scale—for instance via the MSD—it is up to the investigator to choose the maximum length l_{cutoff} to accommodate to, and to not examine phenomena beyond l_{cutoff} and its associated time scale. In this work we chose to define

$$l_{\text{cutoff}} = 10 \quad (2.103)$$

in reduced units (cf. Sec. 2.5.1.6), which is much larger than the diameter of the particles and allows to investigate at least the onset of a possibly-present diverging length scale and its related dynamic phenomena.

In fact, in a system close to dynamic arrest, the simulation time τ_{sim} obtained from Eq. (2.102) may be very long. Therefore, we additionally imposed the condition

$$\tau_{\text{sim}} \leq \tau_{\text{cutoff}} \quad (2.104)$$

where τ_{cutoff} is some simulation time span chosen to ensure that an MD simulation be truncated after at most a few weeks of real time. Obviously, such a truncation results in a nonequilibrated system, the further treatment of which then requires special caution (see below). Notably, a combination of Eqs. (2.104) and (2.102) is suitable for

defining *dynamic arrest* by selecting an appropriate simulation time span τ_{arrest} and pronouncing a system arrested if

$$\delta r^2(\tau_{\text{arrest}}) \leq l_{\text{cutoff}}^2 . \quad (2.105)$$

For convenience, we chose the length l_{cutoff} in Eq. (2.105) to be the same as in Eq. (2.103). Equation (2.105) is employed in the systematic investigations presented in Sec. 3.2.1, where for this purpose we defined

$$\tau_{\text{arrest}} = 30\,000 \quad (2.106)$$

in reduced units (cf. Sec. 2.5.1.6). While this value of τ_{arrest} was chosen essentially arbitrarily, in combination with Eq. (2.105) it happened to reasonably reproduce the well-established glass-transition density $\phi_f \simeq 0.59$ of *bulk* HS systems (see Sec. 3.2.1 and Refs. [36, 78, 79, 151, 182, 214, 241, 265]).

The requirement of thermal equilibrium, as mentioned before, dictates that a production run (PR) be preceded by an equilibration run (ER). The details of our pertinent procedure read as follows: during the ER, the MSD is monitored, and the run is terminated as soon as Eq. (2.102) is fulfilled or Eq. (2.104) is violated. The final configuration of the ER then constitutes the initial configuration of the PR, with the length of the PR (in terms of simulated time) being defined as that of the ER. Subsequent to the PR, from the self ISF's relaxation, it is determined whether the chosen l_{cutoff} sufficed for equilibration.

Owing to the time-cutoff procedure [Eq. (2.104)], a number of runs performed in this work did not fulfill the condition that $F_s(k, t_{\text{sim}})$ is approximately equal to the long-time value $f_s(k)$. It is, however, in the very nature of QA systems that some systems are harder to equilibrate than others; in fact, in some regions of the parameter space, equilibration is even outright impossible. This is—by definition—the case for dynamically-arrested states, and also for systems in which no percolating void exists. In such parameter-space regions, we cautiously dared to also interpret *nonequilibrated* runs, albeit only under the condition that the errors margins of the relevant observables be “small” (see below). In any case, any such interpretation is clearly documented in the respective sections (see in particular Secs. 3.2.4, 3.2.5, and 3.2.7).

The latter consideration turns the spotlight to the important issue of error estimation. As laid out in the beginning of this section, any observable average \underline{A} obtained from MD data is necessarily an approximation of the true thermodynamic average $\langle A \rangle$ and therefore associated with an error. Fortunately, it is possible to assess the reliability of such an imperfect average. The tool at call is basic statistics, with the relevant quantity being the conventional “standard error of the mean”^{2.80}

$$s_{\underline{A}} = \frac{1}{\sqrt{K}} \sigma_{\underline{A}} = \frac{1}{\sqrt{K}} \sqrt{\frac{1}{K} \sum_{k=1}^K (\underline{A}_k - \mu_{\underline{A}})^2} . \quad (2.107)$$

^{2.80}http://en.wikipedia.org/wiki/Standard_error

Here, the \underline{A}_k 's represent K different sample averages of the observable A , and $\mu_{\underline{A}} = \frac{1}{K} \sum_{k=1}^K \underline{A}_k$ denotes the mean of those averages. Before we discuss *how* to obtain the different sample averages, it is important to realize that Eq. (2.107) is valid only if all of the \underline{A}_k 's are *statistically independent* of each other^{2,80}.

One straightforward method to obtain multiple sample averages is to break the trajectory generated by an MD run into nonoverlapping sub-trajectories, each of which is then used to evaluate an \underline{A}_k . In the case of this method, statistical independence demands that the initial and final system configuration of each sub-trajectory be uncorrelated in terms of some ISF or VHF. While this condition might be difficult to ensure, it is precisely this method that is used in real-life experiments. Computer simulations, however, provide the opportunity for a more elegant procedure to obtain different sample averages: conducting multiple simulations with different (random) initial configurations. In this scheme, the only constraint applies to the initial configurations, all of which have to represent the same external conditions. In this work, for two reasons we pursued the latter method: its simplicity, and (more importantly) the fact that in QA systems, multiple system realizations have to be generated and averaged anyway to restore homogeneity and isotropy (cf. Sec. 2.5.1.5). Notably, it is precisely this matrix-averaging procedure that constitutes the *only*—and at the same time the natural—way to obtain different sample averages for quantifiers of the matrix structure (cf. Sec. 2.5.4).

It is important to be aware of the fact that the standard error captures only *statistical* deviations of $\underline{A}_{\mathbf{Q}}$ from the true average $\langle A \rangle_{\mathbf{Q}}$. Equation (2.107) does *not* account for *systematic* deviations built into the data acquisition. Such systematic errors arise for instance from the practice to measure \underline{A} under the external conditions $\underline{\mathbf{Q}}$ while actually trying to determine $\langle A \rangle$ with the external conditions $\langle \mathbf{Q} \rangle$. Although trying to achieve $\underline{A}_{\mathbf{Q}} \simeq \langle A \rangle_{\langle \mathbf{Q} \rangle}$ seems futile if $\underline{\mathbf{Q}} \neq \langle \mathbf{Q} \rangle$ from the outset, this is precisely what is widely attempted in computer simulations. The reason for this is that in many investigations, the system of interest is *infinitely large*—a condition that by its very nature escapes the capabilities of computer simulations. If such an infinitely-large system is the object of interest, a common procedure is to instead consider a “large” system, in the hope of not missing any essential effects. Fortunately, there exist more stringent tests than “hope” as to whether a system of finite size approximates an infinitely-large system to a satisfactory extent. Recalling that both dynamic arrest and porous media may involve large length scales (cf. Secs. 1.2 and 1.3), the issue of finite-size systems is of considerable concern to this work. Therefore, in Appendix A.2 we shall discuss and apply methods to assess and quantify the effects of a limited system size in QA systems.

Chapter 3

Results

3.1 General remarks

In this chapter, we present and discuss the results of the computer simulations conducted by the author for the purpose of the work at hand. As laid out at length in the previous chapters, the primary subject of this work is the dynamic arrest of the fluid component of hard-sphere quenched-annealed (HS-QA) systems. Since the author has published most of the findings discussed in this chapter in scientific journals (cf. the [Preface](#)), it seems fitting to present the findings in an according fashion—i.e., roughly chronologically—so as to convey the evolution of the fact-finding process. In this spirit, we will first discuss the structure and the dynamics of the full fluid component of HS-QA systems (without regard to the details of the QA matrix), then elaborate on the structure and the connectivity of the pores of the matrix, and finally consider separately the properties of those QA fluid particles that are trapped and those that are not.

In the majority of the HS-QA systems simulated in this work, the numbers of the matrix and of the fluid particles add up to $N_m + N_f = N = 1000$. Therefore, N is stated explicitly only if for some system $N \neq 1000$. The relatively-small figure $N = 1000$ was chosen due to the fact that the investigation of dynamic arrest by its very nature requires to consider systems with long relaxation times, and equilibrating such systems is more readily accomplishable if N is small. The drawback of using systems with few particles is the potential presence of finite-size effects, which is investigated in detail in [Appendix A.2](#). Further, as discussed in [Secs. 1.3.4.2](#) and [2.5.1.5](#), when computing observable averages in QA systems, it is imperative that averages be taken over multiple independent realizations of systems with the same external parameters. Unless stated otherwise, we accommodated for this fact—wherever required—by considering ten such realizations. As a welcome side effect, as detailed in [Sec. 2.5.5](#), multiple realizations allow to infer margins of statistical uncertainty. Wherever appropriate, these margins are represented in the diagrams in the form of error bars, each of which indicates the standard error of the mean [see [Eq. \(2.107\)](#)] for the presented datum.

The computing runs of this work—i.e., the molecular-dynamics simulations and the subsequent post processing—were conducted largely on two high-performance computing clusters: (i) the Soft Matter Theory (SMT) group’s own local cluster, which in the years of the simulations (2008–2011) was composed of 68 top-notch CPU cores, and (ii) the Vienna University of Technology’s Phoenix and Phoenix 2 clusters, which at that time consisted of 130 medium-range and 256 high-end CPU cores, respectively. The calculations of this work consumed approximately 100 000 hours of CPU time, equating to exclusive uninterrupted calculations on all cores of the SMT cluster in excess of nine weeks.

Before commencing the discussion of the results, it is appropriate to explicitly mention a number of publications in which closely-related simulational investigations were carried out and which served as major inspirations of this work. Grouping them by principal author as well as by relevance, these works include those by Kim et al. [147–150], Yethiraj et al. [45, 50, 251, 252], Franosch, Höfling et al. [18, 83, 111, 113, 246, 247], Voigtmann, Horbach et al. [121, 273], Fenz et al. [75], Moreno and Colmenero [197, 198], Gallo, Rovere et al. [87–91], Gimel, Nicolai et al. [16, 96], Berthier and Kob [29], Mittal, Errington, and Truskett [196], Karmakar et al. [135, 136], and Jardat et al. [129].

3.2 Full fluid

As laid out in Secs. 1.3.4.2 and 1.4.3, the primary objects of interest in hard-sphere quenched-annealed (HS-QA) systems are the structure and the dynamics of their *fluid* component, as opposed to those of their matrix. In the absence of reasons to do otherwise, the first logical step is to consider all fluid particles to be equivalent. This “full-fluid” approach we shall pursue in this section.

3.2.1 Mean squared displacement: Kinetic diagram

Recalling the intriguing predictions of the replica mode-coupling theory (RMCT, see Sec. 1.4.3), one of the foremost goals of this work is to use computer simulation data to establish a kinetic diagram (KD) of HS-QA systems. This requires to investigate, for the relevant external parameters, whether or not pertinent systems are dynamically arrested. Since KDs provide a succinct overview over the HS-QA parameter space, we shall commence Sec. 3.2 with a KD based on a quantity that is easily computable even without dedicated post-processing: the mean squared displacement (MSD).

As described in Sec. 2.5.5, one definition of dynamic arrest involving the MSD is provided by Eq. (2.105), the pertinent numerical values being $\tau_{\text{arrest}} = 3 \times 10^3$ for the threshold time and $l_{\text{cutoff}}^2 = 10^2$ for the squared threshold length. In Fig. 3.1—as in all KD renditions to follow—the HS-QA parameter space is represented by the matrix

^{3.1}http://www.zid.tuwien.ac.at/hpc/phoenix_linux_cluster

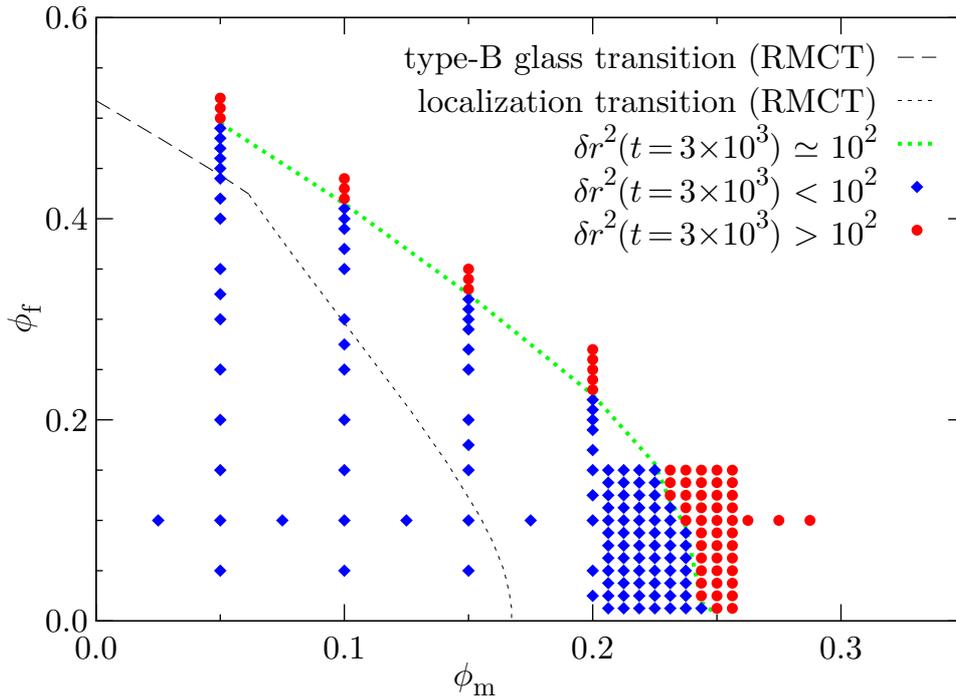


Figure 3.1: HS-QA kinetic diagram: mean squared displacement. Symbols: state points (combinations of the matrix and the fluid packing fractions, ϕ_m and ϕ_f) at which HS-QA systems were simulated. Blue diamonds: nonarrested systems; red disks: arrested systems. The arrest criterion is Eq. (2.105), which involves the mean squared displacement, $\delta r^2(t)$. Thick dotted green line: interpolation through adjacent nonarrested and arrested state points. Thin black lines: transition lines predicted by RMCT for single-particle properties (cf. Fig. 1.19).

packing fraction, ϕ_m , on the abscissa and the fluid packing fraction, ϕ_f , on the ordinate. The figure indicates for several dozen state points (combinations of these parameters) whether the corresponding systems are arrested (red disks) or not (blue diamonds). To arrive at this distinction, of the independent system realizations simulated at each indicated state point (cf. Sec. 3.1), we required at least 50% to be arrested according to Eq. (2.105) to mark the state point “arrested”.

The thick dashed green line interpolates between adjacent nonarrested and arrested state points. An extrapolation of this line towards $\phi_m = 0$ (not shown) suggests that dynamic arrest in the case of bulk hard spheres occurs for $\phi_f \approx 0.58$, which is in excellent agreement with literature values [104, 204, 214]. Included for comparison are the RMCT single-particle transition lines (thin black lines) introduced in Fig. 1.19. While the shapes of the MSD and the RMCT arrest lines are evidently quite similar, their locations in the KD differ considerably. This is only moderately surprising since it has been shown for a number of systems that the temperature and/or the density of dynamic arrest are not uncommon to be underestimated by MCT by 20% or even more [93, 104, 154–156].

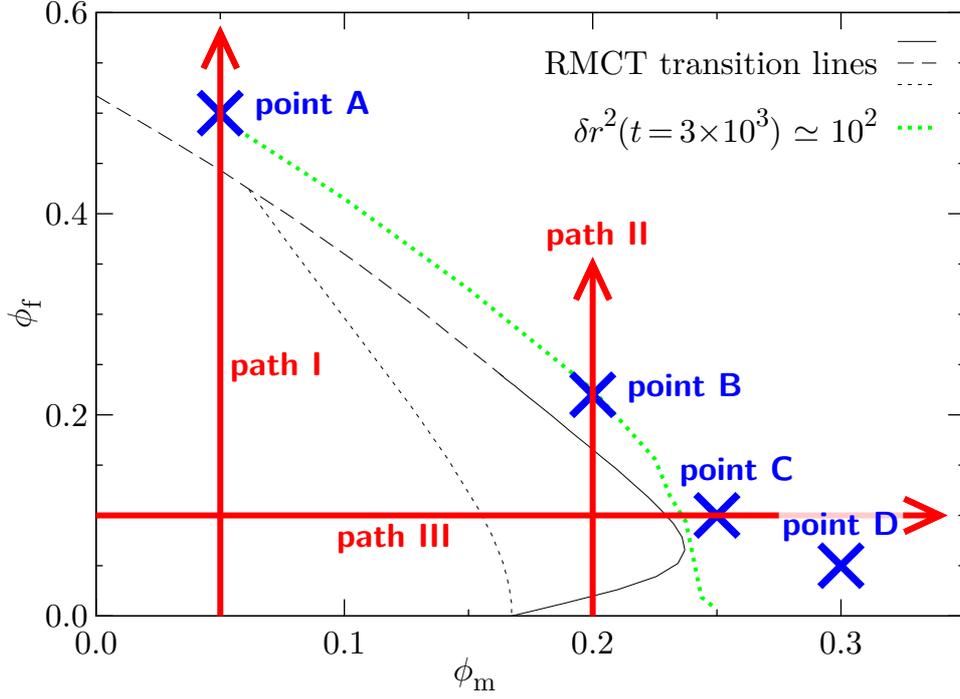


Figure 3.2: Paths and points in the HS-QA parameter space. Locations of the points (blue crosses) and the paths (red arrows) at and along which various observables are presented in detail in this work. Included for reference are the RMCT transition lines from Fig. 1.19 as well as the MSD arrest line from Fig. 3.1.

3.2.2 Paths and points in the parameter space

In Sec. 3.2.1, we established the parameter-space regions in which HS-QA systems are arrested according to Eq. (2.105). The next straightforward step is to investigate selected quantities in more detail for individual state points. For this task, which we will commence in the following section, we will repeatedly focus on specific paths and points in the parameter space. For future reference, these paths and points are summarized in Fig. 3.2, the pertinent data reading

- path I: constant $\phi_m = 0.05$
- path II: constant $\phi_m = 0.20$
- path III: constant $\phi_f = 0.10$
- point A: $\phi_m = 0.05, \phi_f = 0.50$
- point B: $\phi_m = 0.20, \phi_f = 0.22$
- point C: $\phi_m = 0.25, \phi_f = 0.10$
- point D: $\phi_m = 0.30, \phi_f = 0.05$

A primary objective of this work, as mentioned in Sec. 1.4.3, is to compare the results of our computer simulations with the predictions of the replica mode-coupling theory (RMCT; see Sec. 1.4.3). How can such comparisons be accomplished in a meaningful way using the above paths and the paths in Fig. 1.19? According the latter figure, RMCT predicts that the localization transition for $\phi_f \rightarrow 0$ occurs at $\phi_m^* \simeq 0.17$. As stated above, our simulations for $\phi_f = 0.1$ rather suggest $\phi_m^* \gtrsim 0.25$, which implies

that for $\phi_f \rightarrow 0$, the value of ϕ_m^* is even larger^{3.2}. According to Fig. 1.19, RMCT further predicts that the dynamic arrest for $\phi_m = 0$ takes place at $\phi_f = 0.52$, whilst our simulations suggest that in this case $\phi_f = 0.58$ (see Sec. 3.2.1). It is therefore reasonable to treat the RMCT parameters, ϕ_m^{th} and ϕ_f^{th} , as different from the simulation parameters, ϕ_m^{sim} and ϕ_f^{sim} , and as proportional to each other by the relations

$$\begin{aligned}\phi_m^{\text{sim}} &\simeq \phi_m^{\text{th}} \times 1.5 & \text{and} \\ \phi_f^{\text{sim}} &\simeq \phi_f^{\text{th}} \times 1.1 & .\end{aligned}\tag{3.1}$$

Keeping in mind that paths 1, 2, and 3 are used in the investigation of RMCT (Sec. 1.4.3) whereas paths I, II, and III refer to our simulations, this suggests the following correspondences:

- For path 1, the ideal counterpart would be at $\phi_m^{\text{sim}} = 0.15$. This value is most closely approximated by path II. However, given the different natures of the theoretical and the simulational investigations, it is prudent to also consider path I when comparing data for path 1 with simulational data.
- For path 2, the optimum counterpart would be at $\phi_m^{\text{sim}} = 0.3$. This value is situated deep within the parameter-space region that was defined as “arrested” in Fig. 3.1. Since only limited simulations are possible in this region, no path in the simulation parameter space directly corresponds to path 2.
- For path 3, the best-case counterpart would be at $\phi_f^{\text{sim}} = 0.17$. This value is approximated by path III, albeit not particularly closely. Considering the features of Fig. 1.19, however, it is reasonable to assume that observables at constant $\phi_f^{\text{sim}} = 0.17$ and at constant $\phi_f^{\text{sim}} = 0.10$ (path III) differ only quantitatively.

3.2.3 Static structure

In the context of the static structure of HS-QA systems, the most important concern is the issue of whether in all simulated systems, the particles of the QA fluid do indeed represent a fluid (as unspokenly assumed in Sec. 3.2.1) and not a crystal. A simple approach to this question is visual inspection. To this end consider Fig. 3.3, which depicts snapshots of representative HS-QA system realizations at points A and C, with the white and the dark gray spheres denoting fluid and matrix particles, respectively. In panel (b), representing point C, the matrix particles clearly prevent the fluid particles from arranging in an ordered pattern^{3.3}. In panel (a)—showing point A—, this is

^{3.2}In Sec. 3.3.4, the much more precise value $\phi_m^* \simeq 0.2512$ is obtained for the localization transition at $\phi_f \rightarrow 0$.

^{3.3}Since the matrix particles represent an equilibrium configuration when considered by themselves (Sec. 1.3.4.2), and since for the ϕ_m values relevant to HS-QA systems, the matrix particles are

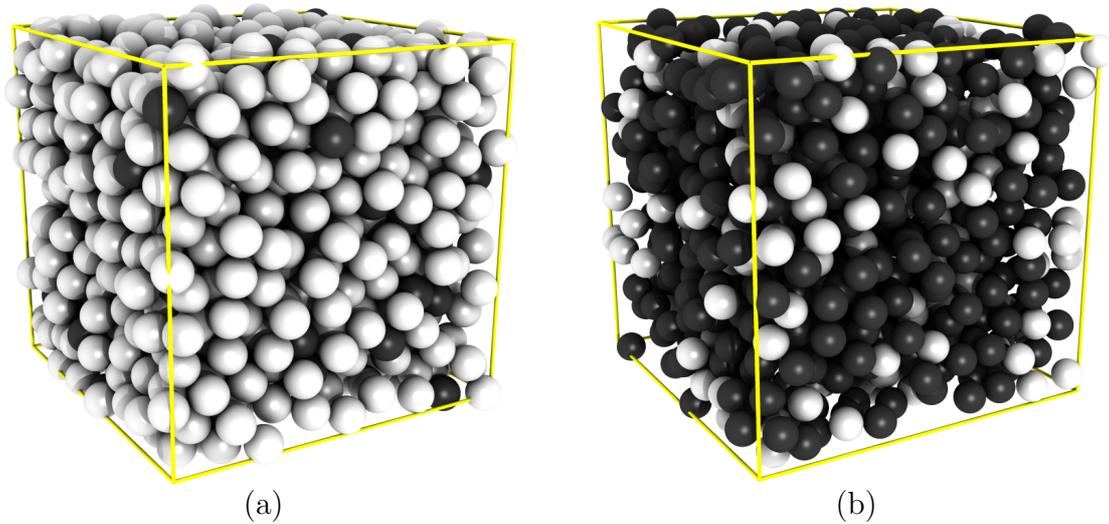


Figure 3.3: HS-QA system snapshots: points A and C. Computer images of representative HS-QA systems. Panel (a): system at point A where $(\phi_m, \phi_f) = (0.05, 0.50)$; panel (b): system at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$. Both systems contain $N_m + N_f = 1000$ particles. White spheres: fluid particles; dark gray spheres: matrix particles. Yellow lines: simulation cell replicated by periodic boundary conditions (cf. Sec. 2.2.2.1).

likely the case, too, albeit to a much lower degree of certainty. This finding is not unexpected since, as discussed in Sec. 1.3.3.2, disordered boundary conditions introduce frustration only when influencing a sufficient fraction of the particles of interest.

A classical means to examine a substance's static structure in more detail is to determine its radial distribution function, $g(r)$, and/or its static structure factor, $S(k)$, as defined in Eqs. (2.61) and (2.65), respectively. Those two quantities are depicted in Figs. 3.4 and 3.5 for series of state points along paths I, II, and III (cf. Fig. 3.2). As the figures show, the degree of local order in HS-QA systems increases with both ϕ_f and ϕ_m . Also, the figures evidence that all features of both $g(r)$ and $S(k)$ change continuously as ϕ_f and ϕ_m are varied, which strongly suggests that no thermodynamic phase transition takes place.

A comparison of the $g(r)$ data in Figs. 3.4 and 1.9 allows to conclude that the considered QA fluids are indeed amorphous. Firstly, the overall shapes of the $g(r)$'s in Fig. 3.4 are typical of HS fluids. Secondly, in Fig. 3.4(a), the maximum at $r \simeq 2$ develops a split-peak shape as ϕ_f is increased, which, as mentioned in Footnote 1.52, is a hallmark of HS glasses and of random close packing structures. Thirdly, any crystalline order—local or global—would probably be of the face-centered cubic (fcc) type, as in

arranged in fluid configurations, the positions of the matrix particles in the investigated systems are guaranteed to be disordered.

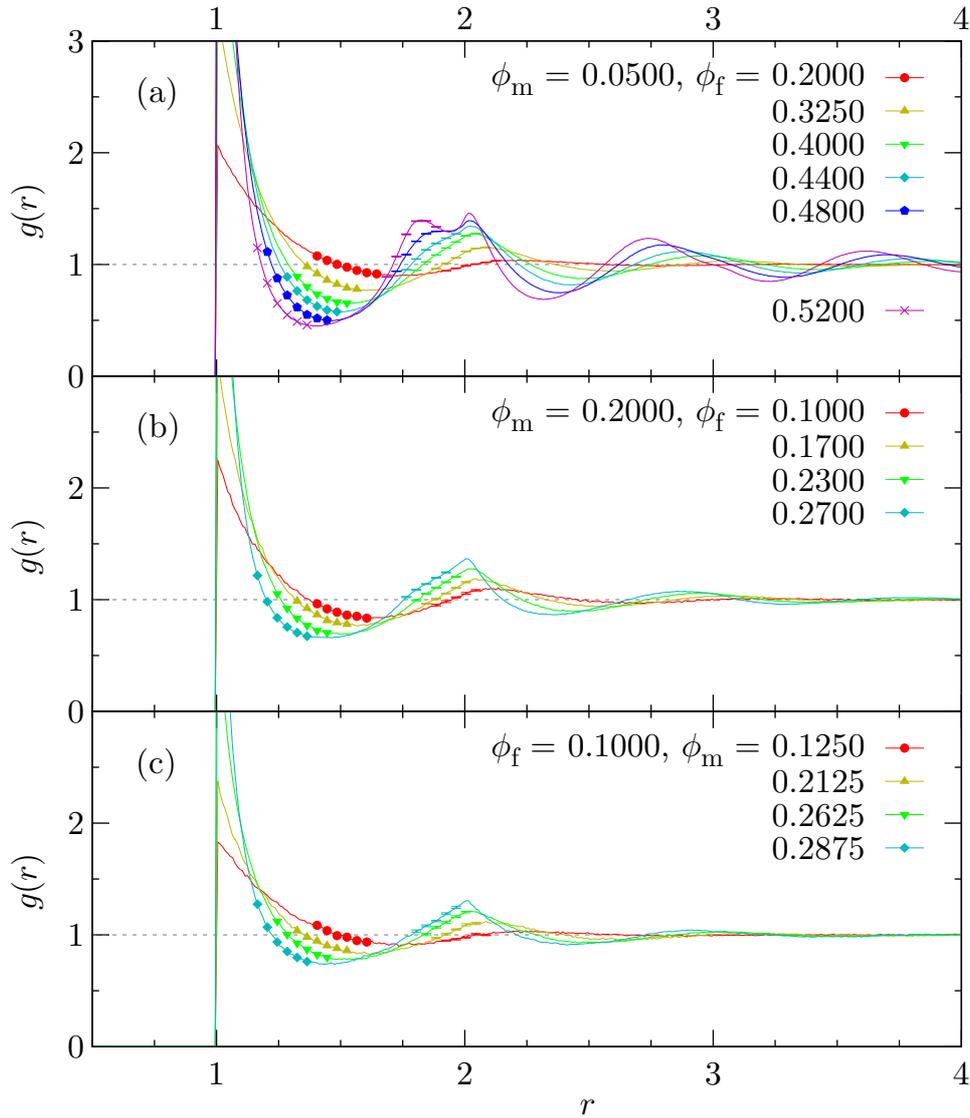


Figure 3.4: HS-QA radial distribution function. Radial distribution function, $g(r)$, for state points along paths I, II, and III (cf. Fig. 3.2). Error bars: see Sec. 3.1.

the case of bulk hard spheres^{3,4}. Such order would entail a local $g(r)$ maximum for $r_{\text{fcc}} = \Phi\sqrt{2}$, where $\Phi = \{(\phi_m + \phi_f)/\phi_{\text{max}}\}^{1/3}$ with $\phi_{\text{max}} = \pi/(3\sqrt{2})$ being the volume fraction of close-packed monodisperse hard spheres^{3,4}. For the state point $(\phi_m, \phi_f) = (0.05, 0.50)$, this would imply a peak at $r_{\text{fcc}} \simeq 1.56$, which is clearly absent.

Considering that $S(k)$ and $g(r)$ essentially contain the same information (cf. Sec. 2.5.2.2), it is not surprising that the $S(k)$'s in Fig. 3.5 are typical of HS fluids, too. However, Fig. 3.5(a) also suggests that some degree of crystallinity may be present for $(\phi_m, \phi_f) = (0.05, 0.52)$, the reason being that the first maximum of the pertinent $S(k)$

^{3,4}http://en.wikipedia.org/wiki/Sphere_packing

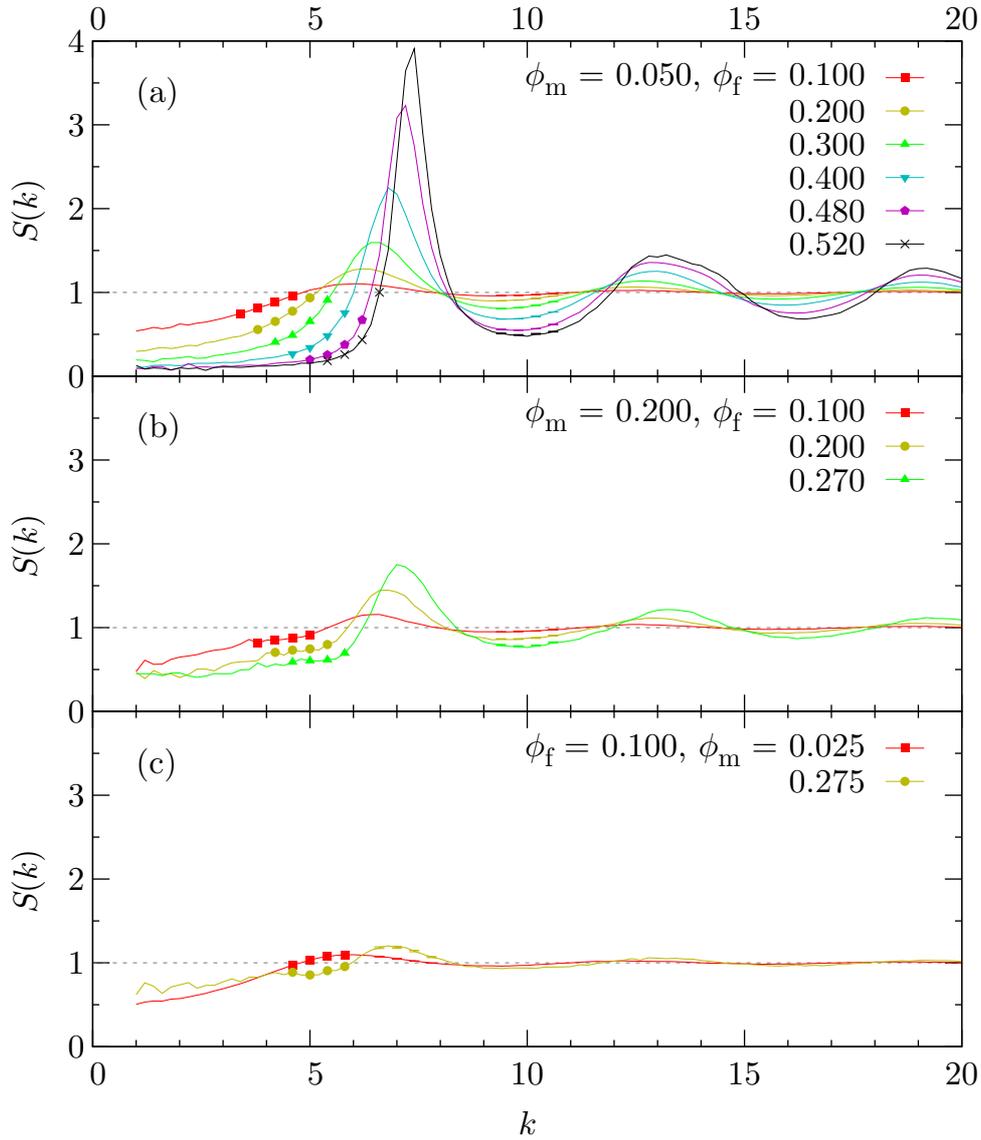


Figure 3.5: HS-QA static structure factor. Static structure factor, $S(k)$, for state points along paths I, II, and III (cf. Fig. 3.2). Error bars: see Sec. 3.1.

curve exceeds a value of 4, which is well above the Hansen-Verlet threshold of ~ 3 (see Sec. 2.5.2.2). Taking into account the previous reasoning in this section, this may indicate the presence of transient crystallites, which is not uncommon in substances approaching dynamic arrest, and which has recently been suggested to even be the very cause of various phenomena associated with the glass transition [231, 255].

In conclusion, it can be stated with certainty that even the most dilute QA matrix considered in this work ($\phi_m = 0.05$) is sufficient to suppress long-range ordering within QA fluids. Since the work at hand focuses primarily on the effects of QA matrices upon dynamic arrest, no systematic attempts were made to pinpoint the threshold value of ϕ_m below which long-range ordering of QA fluids is possible.

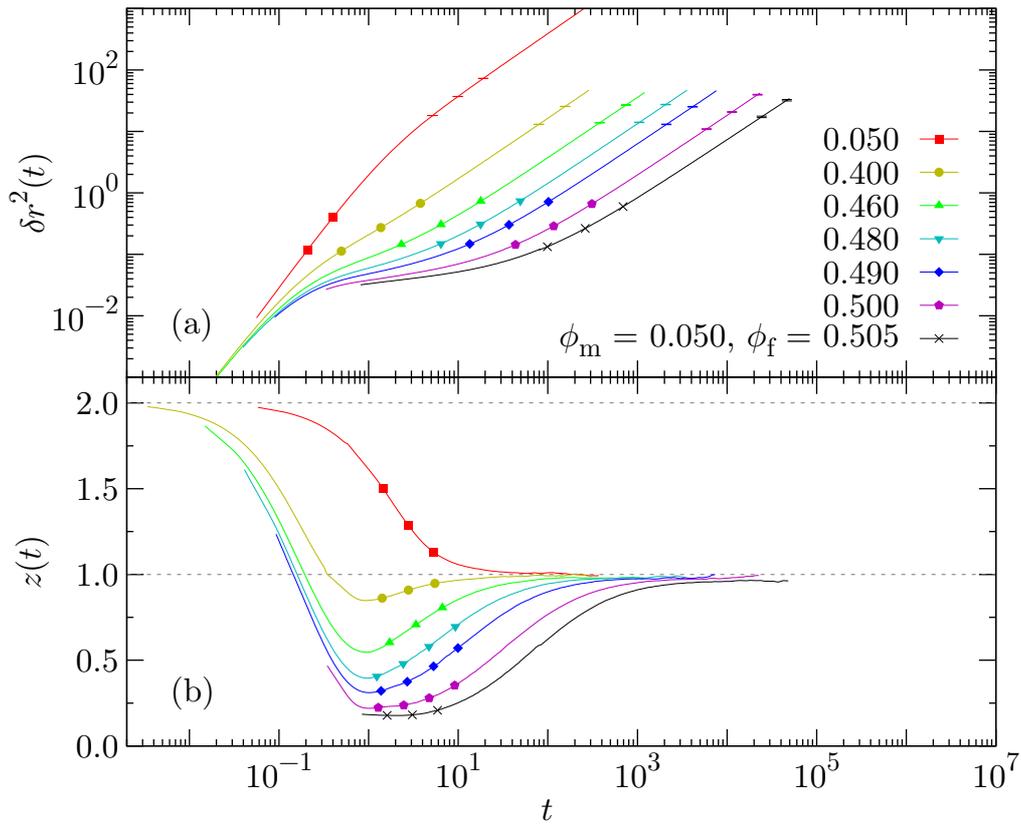


Figure 3.6: HS-QA mean squared displacement: path I. Mean squared displacement, $\delta r^2(t)$, and its logarithmic derivative, $z(t)$, for state points along path I (cf. Fig. 3.2). Error bars: see Sec. 3.1.

3.2.4 Mean squared displacement: Paths

In Sec. 3.2.1, we determined the value of the MSD at the time elapsed within a simulation in order to define dynamic arrest and to construct a KD. In this section, we shall examine the MSD, $\delta r^2(t)$, in more detail and complementarily consider its logarithmic derivative, $z(t)$, so as to emphasize the MSD’s various features. The definition of these quantities can be found in Eqs. (2.66) and (2.67), respectively. In the spirit of Sec. 3.2.3 and of Figs. 1.19 and 3.2, we will present $\delta r^2(t)$ and $z(t)$ for state points along paths I, II, and III, with the pertinent figures considering the time t on logarithmic scales (cf. Secs. 2.5.1.4 and 2.5.3.1) so that phenomena at disparate time scales can be identified.

Figure 3.6 portrays $\delta r^2(t)$ and $z(t)$ along path I, that is, for state points with $\phi_m = 0.05$ and with various values of ϕ_f . As the figure shows, at short times, the MSD grows quadratically ($z = 2$) for all state points, which reflects the fact that a particle’s velocity in event-driven MD remains unaltered until an obstacle is encountered (cf. Sec. 2.2). This time regime is thus called the “ballistic” or “inertial” regime. For the lowest ϕ_f value considered, the particles subsequently cross over to a diffusive regime ($z = 1$) after propagating some few particle diameters. In systems with larger ϕ_f , in contrast, the ballistic regime is followed by an intermediate regime in which z drops

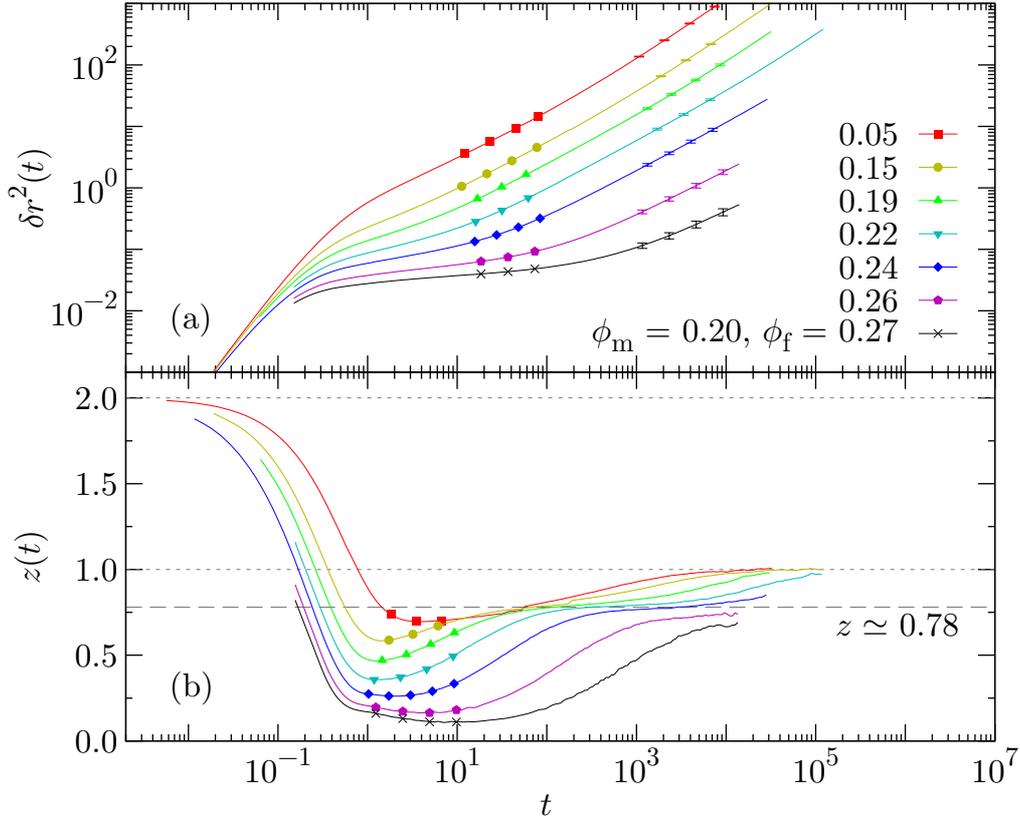


Figure 3.7: HS-QA mean squared displacement: path II. Mean squared displacement, $\delta r^2(t)$, and its logarithmic derivative, $z(t)$, for state points along path II (cf. Fig. 3.2). Error bars: see Sec. 3.1.

below unity and as low as $z \simeq 0.15$ for $\phi_f = 0.505$. This feature originates in the cage effect and is typical of many glass-forming systems [88, 89, 154, 156, 211]. As asserted in Sec. 1.2.7, the duration of the caging indeed depends strongly upon ϕ_f . Finally, in the limit of large times, diffusive behavior is recovered for each considered state point.

Figure 3.7 illustrates $\delta r^2(t)$ and $z(t)$ for state points along path II, i.e., for $\phi_m = 0.2$ and for varying ϕ_f . Although not immediately obvious, the MSD differs significantly between the state points along path II and the state points along path I. Similar features are the initial ballistic regime and the subsequent decrease of z below unity. Overall, however, for path II the decrease is sharper than for path I, and a decrease is also observed for the lowest- ϕ_f state point on path II. This suggests that in this time regime, not only caging but also confinement effects are relevant in the case of path II. The subsequent acceleration differs from that observed for path I in two respects. Firstly, the recovery of diffusion requires considerably-longer times; secondly—and more importantly—the functional form is different, namely in that for an extended intermediate time, z remains at a roughly constant sub-unity value. Notably, this value ($z \simeq 0.78$) is essentially independent of the ϕ_f value^{3.5}. In the remainder of

^{3.5}In Fig. 3.7, it is $z < 0.78$ in the subdiffusive regime for state points along path II with large ϕ_f .

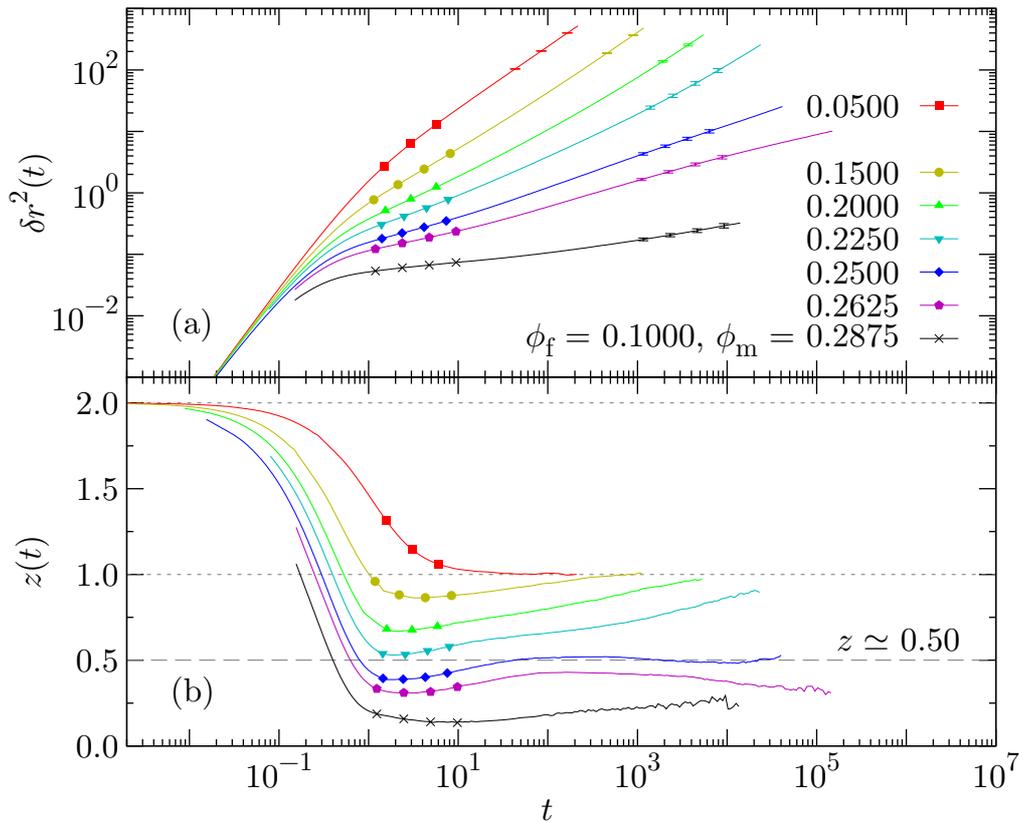


Figure 3.8: HS-QA mean squared displacement: path III. Mean squared displacement, $\delta r^2(t)$, and its logarithmic derivative, $z(t)$, for state points along path III (cf. Fig. 3.2). Error bars: see Sec. 3.1.

this work we shall refer to this regime as “*the* subdiffusive regime”, although in a broader sense caging likewise implies subdiffusion. This serves to remind of the effects of confinement and of the fact that those effects are foreign to simple glass formers.

Figure 3.8, finally, shows $\delta r^2(t)$ and $z(t)$ for state points along path III, meaning that now ϕ_f is held constant at a value of 0.1 while ϕ_m is varied. Except for the state point with the lowest ϕ_m (which coincides with the lowest- ϕ_f state point in Fig. 3.6), all state points exhibit a ballistic regime, a caging regime, and a subdiffusive regime. Concerning the latter, panel (b) of the figure evidences that both the extent of the subdiffusive regime and the average value of z therein are highly dependent upon ϕ_m . This implies that the value $z = 0.78$ observed in the subdiffusion regime of path II does not bear any particular significance. Most strikingly, for $\phi_m \geq 0.25$, ordinary diffusion is *not* recovered in the long-time limit. While for $\phi_m > 0.25$, the value of z instead tends towards zero^{3,6}, for $\phi_m = 0.25$, it *remains* at a value of approximately 0.5

This may be due to the fact that those systems are not well equilibrated with respect to the criterion defined in Sec. 2.5.5.

^{3,6}Since the systems at $(\phi_m, \phi_f) = (0.2875, 0.1000)$ are not well equilibrated, the pertinent $\delta r^2(t)$ and $z(t)$ have to be interpreted with caution (cf. Sec. 2.5.5). Beyond the largest depicted time, $t = 10^4$,

for more than three decades in time before ultimately increasing beyond that value. This suggests that there exists a value $\phi_m^* \gtrsim 0.25$ for which z neither increases towards unity nor tends towards zero for an indefinite time. As we will show later in this work, such long-time subdiffusion (cf. Sec. 1.3.2) is tied intimately to a percolation transition in the accessible volume at ϕ_m^* . The fact that for $\phi_m = 0.2625$ the value of z ultimately decreases towards zero allows to conclude that $0.25 < \phi_m^* < 0.2625$. Taking into account the dependence of z upon ϕ_m in the subdiffusive regime, $\lim_{t \rightarrow \infty} z$ for ϕ_m^* is probably somewhat lower than 0.5.

Let us now compare these observations with the predictions of RMCT as summarized in Fig. 1.22. For this, we need to recall that in our computer simulations, we considered only equilibrated systems (cf. Sec. 1.2.4), meaning that comparisons are sensible only for the nonarrested (green) curves in Fig. 1.22. For these cases, in both depicted paths (path 1 at constant $\phi_m = 0.10$, and path 3 at constant $\phi_f = 0.15$), RMCT predicts that there are three distinct time regimes of the MSD:

- (i) an initial diffusive regime,
- (ii) a subdiffusive regime with $z = 1/2$, and
- (iii) a final diffusive regime.

In regime (iii), RMCT predicts the MSD to be diffusive in all nonarrested systems except for those at ϕ_m^* . This is confirmed by our simulations. In regime (i), the MSD is ballistic ($z = 2$) in the simulations and diffusive ($z = 1$) in RMCT; however, as mentioned in Footnote 1.68, the short-time dynamics is irrelevant for the predictions of RMCT and therefore for comparisons with simulation data. In regime (ii), finally, the RMCT predictions genuinely differ from our observations: Firstly, for neither path 1 nor path 3, a distinct caging regime is found in RMCT. Secondly, according to RMCT, the value of z in the subdiffusive regime invariably is $1/2$, which is at odds with our finding that it depends upon ϕ_m and attains the value 0.5 only for ϕ_m^* . However, for the specific case of ϕ_m^* , the RMCT prediction is in superb agreement with our observation.

In summary, the MSD's most interesting feature in simulations of HS-QA systems is the subdiffusive regime, which for ϕ_m^* prevails indefinitely. Unfortunately, the insights attained in this section are insufficient to pinpoint the cause(s) of this feature. One may speculate that it is due in part to caging, in part to trapping, in part to an (emerging) fractal nature of the accessible volume, and in part to dimensional reduction, perhaps resembling single-file diffusion.

3.2.5 Intermediate scattering functions: Paths

Beginning with this section, we direct our focus to the various intermediate scattering functions (ISFs) of HS-QA systems. In the present section, we will investigate the

the depicted $z(t)$ will probably attain a maximum and subsequently level off to zero.

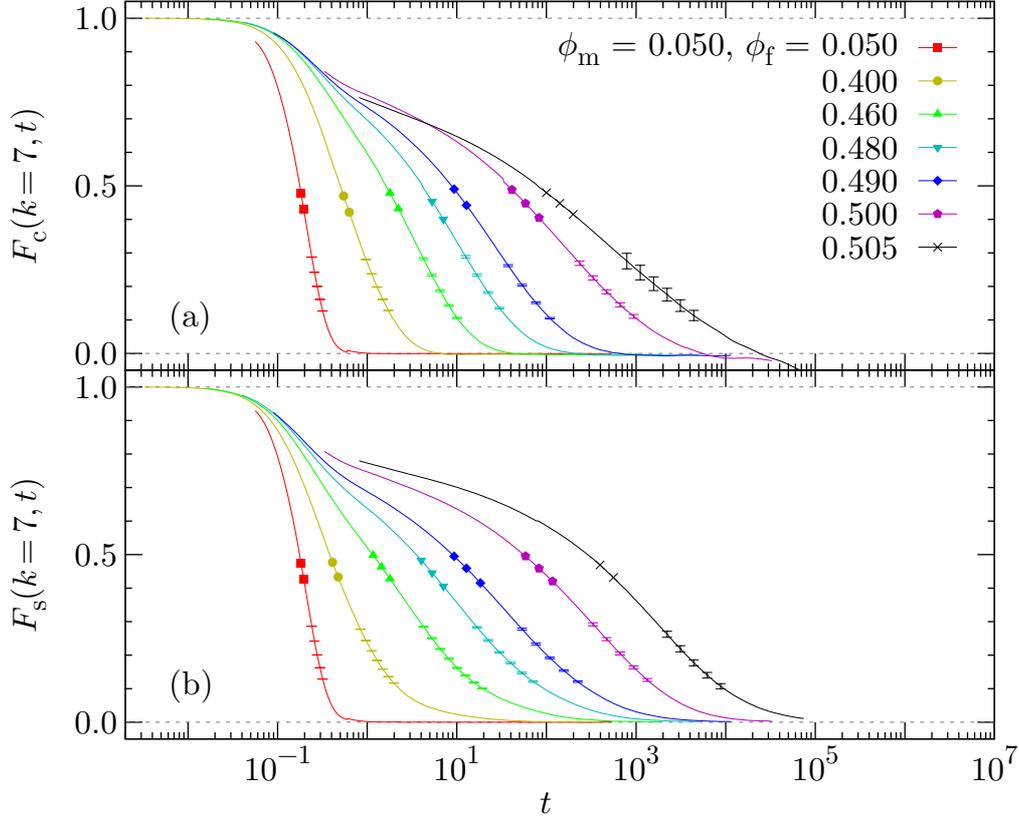


Figure 3.9: HS-QA intermediate scattering functions: path I. Connected intermediate scattering function, $F_c(k, t)$, and single-particle intermediate scattering function, $F_s(k, t)$, for state points along path I (cf. Fig. 3.2) and for the wave vector $k = 7$. Error bars: see Sec. 3.1.

connected ISF and the single-particle (“self”) ISF along the same parameter-space paths as for the MSD in Sec. 3.2.4, i.e., along paths I–III. The definitions of the ISFs can be found in Eqs. (2.73) and (2.81), respectively. The knowledge gathered in this section will subsequently be used in Sec. 3.2.6 to construct KDs on the basis of these ISFs.

Figure 3.9 shows the connected ISF, $F_c(k, t)$, and the self ISF, $F_s(k, t)$, for state points along path I (constant $\phi_m = 0.05$). The wave vector $k = 7$ used in both panels is close to the first maximum of $S(k)$ (cf. Fig. 3.5), which leads to a favorable signal-to-noise ratio and spotlights the structural relaxation at a length scale comparable to the particles’ diameter. As is evident from the figure, the relaxation times and patterns of $F_c(k, t)$ and $F_s(k, t)$ are very similar—except perhaps for the largest- ϕ_f state point, where, however, the systems are not as-well equilibrated as for the other state points (note the unphysical decay of $F_c(k, t)$ to below zero). For the largest depicted ϕ_f values, regimes of β decay, caging, and α decay can clearly be distinguished, with the caging-regime plateau changing only in extent but not in height as ϕ_f is varied (cf. Sec. 1.2.7). Such type-B transitions are typical of simple glass formers (compare

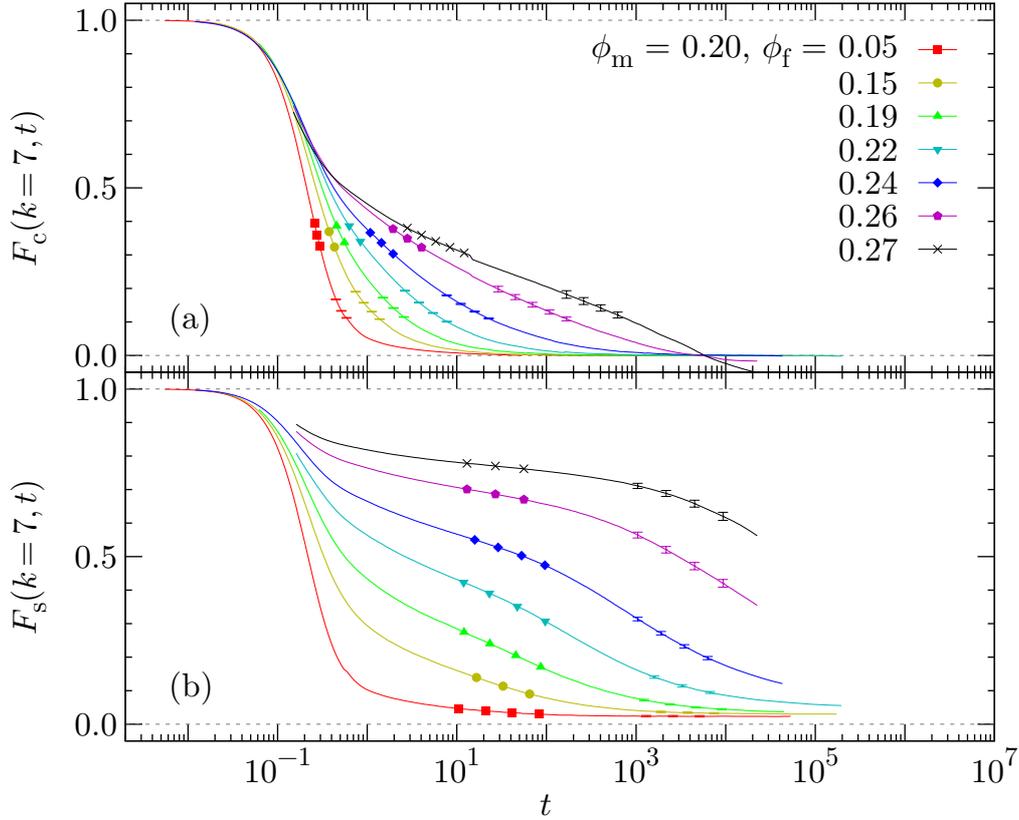


Figure 3.10: HS-QA intermediate scattering functions: path II. Connected intermediate scattering function, $F_c(k, t)$, and single-particle intermediate scattering function, $F_s(k, t)$, for state points along path II (cf. Fig. 3.2) and for the wave vector $k = 7$. Error bars: see Sec. 3.1.

Fig. 1.10), meaning that in this parameter-space region, there is no qualitative effect of the QA matrix on the structural relaxation.

Fig. 3.10 illustrates $F_c(k, t)$ and $F_s(k, t)$ for state points along path II (constant $\phi_m = 0.2$), the wave vector again being $k = 7$. Clearly, for each ISF, the relaxation pattern differs considerably from that along path I (Fig. 3.9). Even more importantly, the connected and the self ISF are quite dissimilar to each other, the most important difference being the significantly-disparate relaxation times. This can be quantified using the times t_c and t_s required for $F_c(k, t)$ and $F_s(k, t)$ to decay to a value of 0.1, respectively (cf. Sec. 2.5.3.5). For $\phi_f = 0.22$, for instance, we find $t_c = 7.9 \times 10^0$ and $t_s = 5.4 \times 10^3$, which represents a difference of almost three decades in time, with the *connected* ISF relaxing faster.

Considering $F_c(k, t)$'s substantially-prolonged relaxation times upon increasing ϕ_f , it can be assumed that the connected ISF approaches a transition for ϕ_f values slightly beyond 0.27 (the largest ϕ_f value depicted). However, for three reasons it is difficult to pinpoint whether this transition is of type A (continuous) or type B (discontinuous; cf. Sec. 1.4.3). Firstly, caution has to be exercised in the interpretation of the data at

$\phi_f \geq 0.26$, the reason being the imperfect equilibration with respect to the criterion defined in Sec. 2.5.5 (note again the unphysical decay to below zero). Secondly, the features of type-A transitions may differ between simulation and theory. Thirdly, it is possible that an intermediate-time plateau emerges for $\phi_f > 0.27$. Unfortunately, since equilibration times rise sharply beyond $\phi_f = 0.27$ (cf. Sec. 3.2.4), it is doubtful that even more sophisticated simulation techniques can resolve this question. Turning to $F_s(k, t)$, an intermediate-time plateau can be observed at large ϕ_f . As for path I, this plateau exists only beyond a certain ϕ_f value, and subsequently extends in time as ϕ_f is increased. However, unlike for path I, the height of the plateau increases with ϕ_f . Another significant difference between $F_s(k, t)$ along paths I and II is the fact that for path II, the self ISF does *not* fully relax but instead attains a nonzero long-time value (LTV). Although this LTV, $f_s(k)$, decreases weakly as ϕ_f is lowered, it remains well above zero even for the lowest depicted ϕ_f value.

It is nontrivial to rationalize these observations. The relaxation of $F_s(k, t)$ to nonzero values is likely due to “trapped” fluid particles (particles that cannot move infinitely far away from their initial position; cf. Sec. 2.4), which is consistent with the observed $F_c(k, t)$ as confinement does not leave traces in correlators of fluctuations (cf. Sec. 2.5.3.3). The plateaus of $F_c(k, t)$ and $F_s(k, t)$, in contrast, are more puzzling. In classical glass formers, plateaus result from caging, and the strong dependence upon ϕ_f of the relaxation patterns indicates that packing effects indeed play a role. However, in $F_c(k, t)$ a plateau is absent, whereas in $F_s(k, t)$ it is prominent and even varies in height. Therefore, it can be assumed that the classical caging effect, if it exists, is profoundly altered by the presence of the matrix, a possibility being for instance an impact on the cage size. Unfortunately, at this point, the geometric complexity of this problem does not allow to draw any definite conclusions.

Fig. 3.11, finally, depicts $F_c(k, t)$ and $F_s(k, t)$ for state points along path III (constant $\phi_f = 0.1$) at the wave vector $k = 7$. As is evident, both quantities differ markedly between path III and the other two paths. However, $F_c(k, t)$ and $F_s(k, t)$ for path III itself are quite similar to each other, namely in that both quantities relax in a single step. The two quantities differ merely in that $F_c(k, t)$ relaxes strictly to zero in all depicted cases while $F_s(k, t)$ attains a nonzero LTV, $f_s(k)$. The value of $f_s(k)$ strongly depends upon ϕ_m , which is consistent with the reasoning introduced in the context of path II that $f_s(k) > 0$ is caused by trapped particles. For both $F_c(k, t)$ and $F_s(k, t)$, the relaxation times towards the LTV increase with ϕ_m , but they do so only moderately for the connected ISF, suggesting that $F_c(k, t)$ is not close to a dynamic arrest transition within the considered ϕ_m range.

How do RMCT’s predictions fare against all these observations? As laid out in Refs. [159–162], for small ϕ_m , there is much similarity between RMCT and the classical mode-coupling theory in that either one predicts a type-B transition for both $F_c(k, t)$ and $F_s(k, t)$. Our simulations along path I confirm this transition type. In Sec. 1.4.3, we reported RMCT’s predictions for the connected ISF along path 2 and for the self ISF along path 3 (see Figs. 1.20 and 1.21). While path 3 corresponds to our path III, for path 2 there is no directly-corresponding path in our simulations (cf. Sec. 3.2.2).

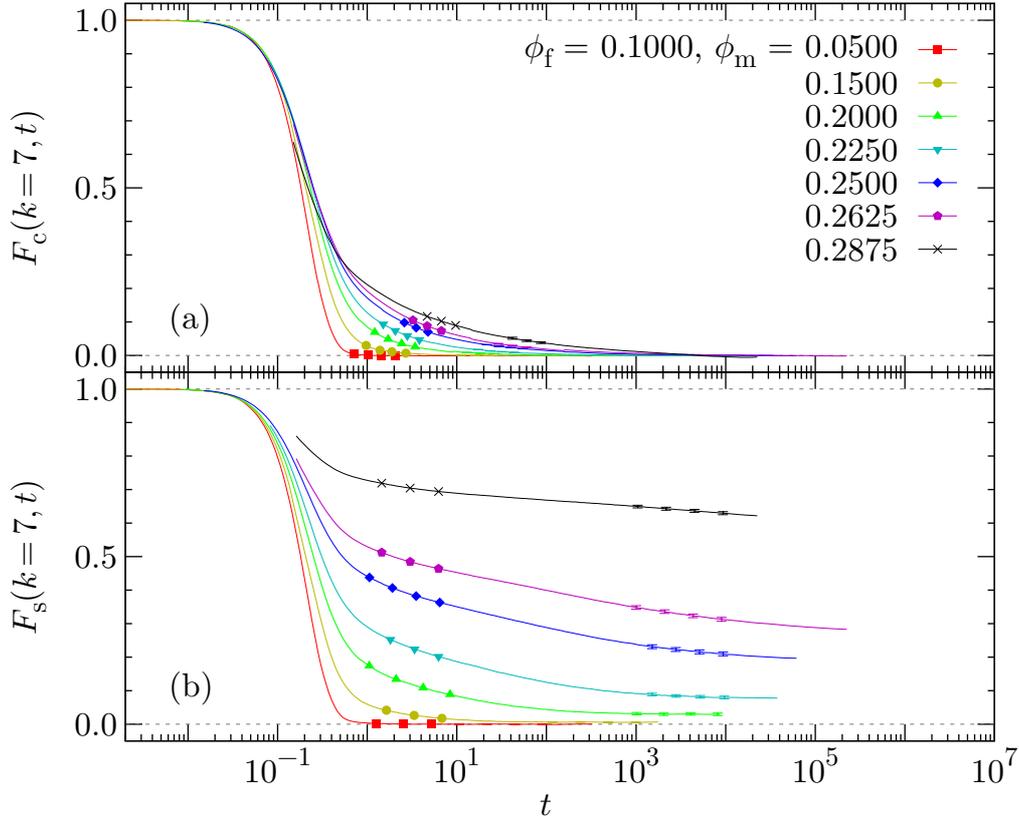


Figure 3.11: HS-QA intermediate scattering functions: path III. Connected intermediate scattering function, $F_c(k, t)$, and single-particle intermediate scattering function, $F_s(k, t)$, for state points along path III (cf. Fig. 3.2) and for the wave vector $k = 7$. Error bars: see Sec. 3.1.

However, since for $F_c(k, t)$, the arrest transition occurs roughly at the intersection of paths 2 and 3 (cf. Figs. 1.19 and 3.2), it is reasonable to compare this quantity between path 2 and path III. Considering these correspondences, RMCT's predictions are quite satisfactory for both ISFs, those for $F_s(k, t)$ arguably being somewhat better. The only clear difference—due perhaps to the simulational limitations—is the fact that we did not observe $f_c(k) = \lim_{t \rightarrow \infty} F_c(k, t) > 0$ for any state point.

The deviation between RMCT's predictions and our simulation results is more pronounced for the case of path 1. For $F_c(k, t)$, RMCT finds a type-B transition while our data along path II rather resemble a type-A transition (despite $f_c(k) = 0$ in all investigated cases). However, according to Fig. 1.19, the theory does find type-A transitions at $\phi_m \gtrsim 0.16$, which is commensurate with path II (cf. Sec. 3.2.2). The case of $F_s(k, t)$, finally, is the least satisfactory: while RMCT predicts that along path 1 there is a type-A transition, and clearly separated from that a type-B transition, according to our simulations, there may either be a rather intricate interplay of those transitions over an extended parameter-space region, or even a fundamentally different transition in that region.

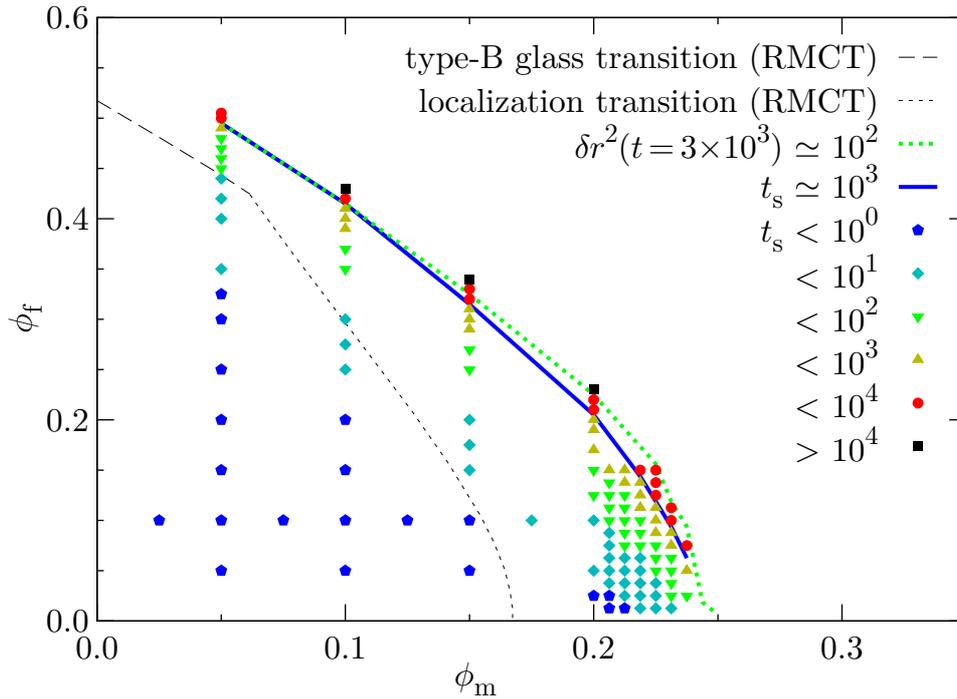


Figure 3.12: HS-QA kinetic diagram: single-particle ISF. Symbol colors and shapes: relaxation time t_s of $F_s(k=7, t)$ at the respective state point. Thick solid blue line: interpolation between state points for which $t_s \simeq 10^3$. Thick dotted green line: MSD arrest line from Fig. 3.1. Thin black lines: RMCT transition predictions for single-particle properties (cf. Fig. 1.19)

3.2.6 Intermediate scattering functions: Kinetic diagrams

Let us now construct KDs on the basis of the single-particle (“self”) and the connected ISFs. To accomplish this task, we focus on a number of distinct state points, at each of which we evaluate the full functional behavior of $F_s(k, t)$ and $F_c(k, t)$ and determine the times t_s and t_c that the respective ISF at the wave vector $k = 7$ requires to relax to the value 0.1 (cf. Sec. 3.2.5). Notably, t_s and t_c by this definition are not defined if the respective ISF exceeds 0.1 in the long-time limit—which, however, does not present major problems as we shall see below. (For a further discussion of the merits and problems of the definition of t_s and t_c see Sec. 2.5.3.5.)

Figure 3.12 illustrates the KD obtained from $F_s(k, t)$. Included as a thick solid blue line is an iso-relaxation time line, i.e., a line interpolating between state points at which t_s attains approximately the same value. This value we chose to be $t_s = 10^3$, which is reasonably close to the actual dynamic arrest and nevertheless establishable in large parts of the KD. Further, using this t_s value, the onset of the line at small ϕ_m and large ϕ_f roughly coincides with the arrest line inferred in Sec. 3.2.1, the latter being included in the figure in the same thick dotted green appearance as in Fig. 3.1. As can be seen from the figure, the self ISF and the MSD arrest lines are remarkably similar, despite small differences at large ϕ_m and small ϕ_f . This, on the one hand,

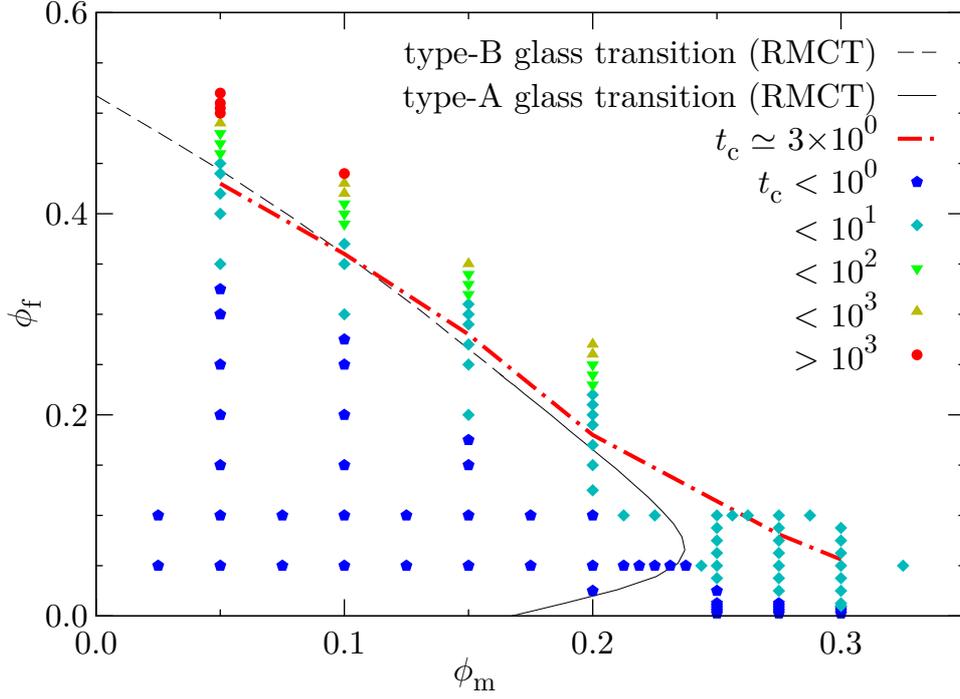


Figure 3.13: HS-QA kinetic diagram: connected ISF. Symbol colors and shapes: relaxation time t_c of $F_c(k=7, t)$ at the respective state point. Thick dash-dotted red line: interpolation between state points for which $t_c \simeq 3 \times 10^0$. Thin black lines: RMCT transition predictions for collective properties (cf. Fig. 1.19).

is reasonable to expect in the light of both $F_s(k, t)$ and $\delta r^2(t)$ being single-particle properties. On the other hand, the similarity is remarkable considering that the arrest definitions differ significantly and that for many state points, t_s is not even defined (see above). Fig. 3.12 thus lends further credence to the transition lines predicted by RMCT for single-particle properties (included in the figure for comparison as thin black lines; cf. Sec. 3.2.1).

Figure 3.13 depicts the KD based on $F_c(k, t)$. As in the case of the self-ISF KD, the figure includes an iso-relaxation time line (the thick dash-dotted red line), which in the present case interpolates between state points for which $t_c \simeq 3 \times 10^0$. The latter value is significantly smaller than in the case of $F_s(k, t)$ in order to allow for the iso- t_c line to be traced into the parameter-space region of large ϕ_m and small ϕ_f . Recalling RMCT's predictions for the collective properties (thin black lines) and the scaling between the parameters of theory and simulation (see Sec. 3.2.2), it is precisely this region that the theory finds a reentrant regime in (cf. Sec. 1.4.3).

Unfortunately, while at low ϕ_m and large ϕ_f the agreement with the theory is fine, our simulations do not confirm the existence of the reentrant pocket. There are several possible reasons for this finding. For instance, the reentrancy's location within the parameter space may differ from the predictions (even taking into account the corrections of Sec. 3.2.2), and in particular it may be located at larger values of ϕ_m and/or ϕ_f . This would be unfortunate since—as pointed out in Secs. 2.5.5, A.3.2,

and A.3.3—in that region it is difficult to impossible to establish equilibrated HS-QA systems and therefore to unveil any phenomena. Further, iso-relaxation time lines may not always be representative of RMCT transition lines since the theory defines a transition strictly as the emergence of a nonzero LTV. Finally, the culprit may be the very core of RMCT. In Ref. [161], Krakoviack explored the omission of either the linear or the quadratic vertex from the theory (cf. Sec. 1.4.3). The result of this exercise is illustrated in Fig. 4 of Ref. [161] in the form of a “hypothetic” KD for $F_c(k, t)$. Surprisingly, our iso- t_c line in Fig. 3.13 closely resembles the RMCT transition line obtained by disregarding the linear vertex. Since, according to Krakoviack, that vertex accounts for the effects of confinement, this might indicate that RMCT’s predictions for the collective properties of HS-QA systems are fundamentally flawed.

3.2.7 Total intermediate scattering function

In Secs. 3.2.5 and 3.2.6, we found that the functional behavior of $F_c(k, t)$ as well as the KD based on $F_c(k, t)$ differ significantly between RMCT’s predictions and our simulations. In order to shed additional light on the collective behavior of HS-QA systems, in this section we will examine another collective quantity, namely the conventional collective ISF, $F(k, t)$, which we will refer to here as the “total” ISF so as to distinguish it from other ISFs. As laid out in Secs. 2.5.3.2 and 2.5.3.3, $F(k, t)$ and $F_c(k, t)$ are related by Eq. (2.79), which can be rearranged to read

$$F(k, t) = \frac{S_c(k)}{S(k)} F_c(k, t) + \frac{S_b(k)}{S(k)} , \quad (3.2)$$

meaning that we can evaluate the total ISF not only from Eq. (2.70), but independently also by combining Eqs. (2.73), (2.65), and (2.72). This allows to assess systematic errors inherent to $F(k, t)$ and $F_c(k, t)$ resulting for instance from insufficient equilibration.

In Fig. 3.14, the versions of $F(k, t)$ evaluated from Eqs. (2.70) and (3.2) are shown as lines and symbols, respectively, with the depicted state points residing along path I ($\phi_m = 0.05$) and the wave vector being $k = 7$. As is evident, the two versions match near-perfectly for $\phi_f \leq 0.48$, while beyond that value deviations are apparent. This is in accordance with the notion that the equilibration of the large- ϕ_f systems along path I is worse than that of the smaller- ϕ_f ones (cf. Sec. 3.2.5). Nevertheless, according to the error bars, the two versions mostly agree within the statistical limits. The figure also confirms the expectation that $F(k, t)$ does not always relax to zero but instead typically attains a nonzero LTV, $f_t(k)$. Interestingly, $f_t(k)$ varies not only with ϕ_m (see below) but also with ϕ_f , and notably it is generally sizable.

Figures 3.15 and 3.16 illustrate the problems posed by nonzero LTVs when interpreting the total ISF. Both figures depict KDs, i.e., the shapes and colors of the symbols represent the value of some quantity at the respective state point. In Fig. 3.15, this quantity is the time t_t required by $F(k, t)$ to relax to the value of 0.1, while in Fig. 3.16, the quantity is the normalized blocked structure factor $\hat{S}_b(k) = S_b(k)/S(k)$.

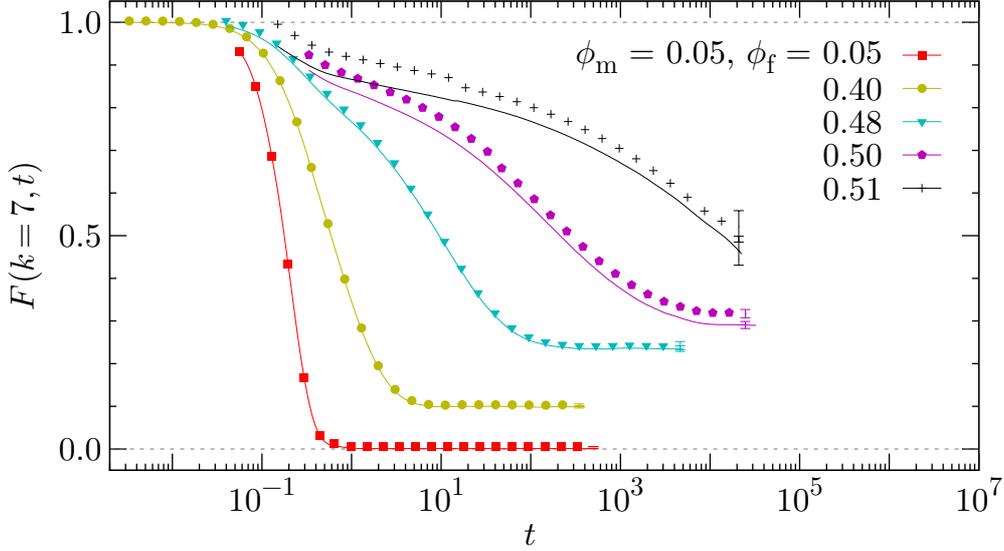


Figure 3.14: HS-QA total intermediate scattering function. Total (conventional) intermediate scattering function, $F(k, t)$, for state points along path I and for the wave vector $k = 7$. Lines: direct evaluation from Eq. (2.70); symbols: evaluation via Eq. (3.2) involving $F_c(k, t)$, $S_c(k)$, and $S(k)$. Error bars: see Sec. 3.1.

The latter, according to Eq. (3.2), is equal to $f_t(k)$ if we assume that the LTV of $F_c(k, t)$ vanishes^{3.7}. Finally, in both figures the considered wave vector is $k = 7$.

Both figures show a thick solid red line, which represents an iso- t_t line for $t_t \simeq 3 \times 10^0$. Although the latter value is relatively small, it is close to the maximum of t_t for which an iso- t_t line can be traced across large parts of the kinetic diagram. The reason for this is revealed by Fig. 3.16, which as an extra-thick dash-double-dotted gray line additionally includes an iso- $\hat{S}_b(k)$ line for $\hat{S}_b(k) = 0.1$. As is obvious, the iso- $\hat{S}_b(k)$ and the iso- t_t lines largely coincide. This suggests that the latter line encodes essentially no dynamic information (cf. Footnote 3.7), and it also explains why Fig. 3.15 depicts only few state points to the upper right of the iso- t_t line: in that region, t_t is simply not defined for most state points due to $f_t(k) > 0.1$. Therefore, in conclusion, $F(k)$ provides only marginal additional information about the dynamics of HS-QA systems.

3.2.8 Long-time value of the single-particle ISF

In the previous sections, we repeatedly referred to times required by ISFs to relax to some threshold value. These relaxation times, we noted, are undefined if the threshold

^{3.7}Since $S(k)$, $S_b(k)$, and therefore $\hat{S}_b(k)$ are static (time-independent) quantities, Fig. 3.16 is not, in the strict sense, a *kinetic* diagram. However, given that most depictions of the HS-QA parameter space in this work *do* display a dynamic quantity, we chose to abide by this name and to highlight the terminological misfit by quotes.

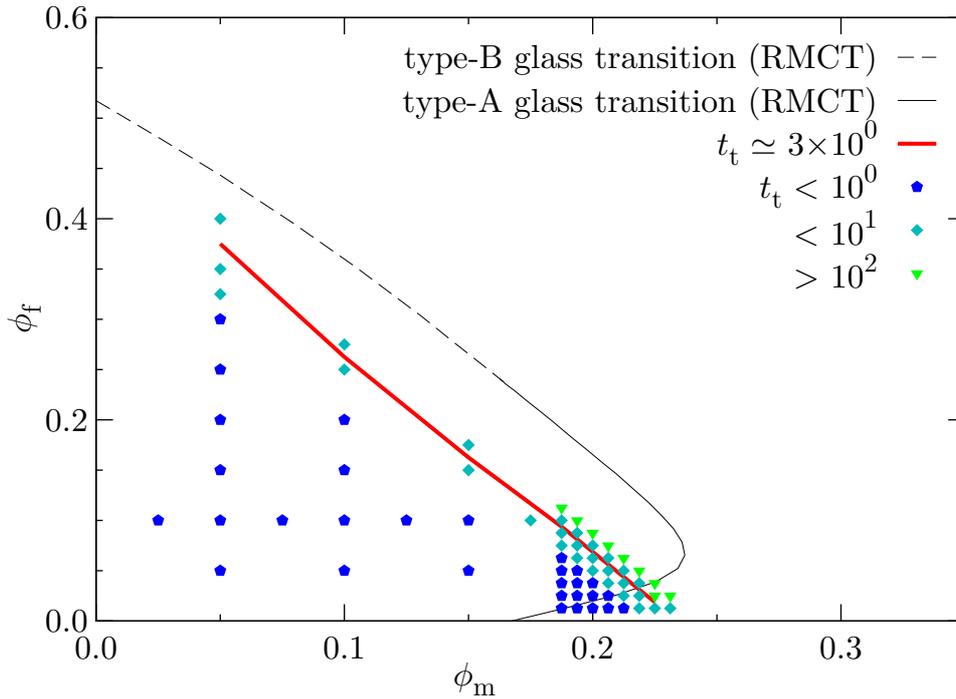


Figure 3.15: HS-QA kinetic diagram: total ISF. Symbol colors and shapes: relaxation time t_t of $F(k=7, t)$ at the respective state point. Thick solid red line: interpolation between state points for which $t_t \simeq 3 \times 10^0$. Thin black lines: RMCT transition predictions for collective properties (cf. Fig. 1.19).

value is exceeded by the LTV of the ISF in question. The example of $F(k, t)$ revealed that this may significantly impact the interpretation of KDs (cf. Sec. 3.2.7), which suggested to evaluate threshold-value relaxation times preferably for fully-relaxing quantities. This implied to consider $F_c(k, t)$ in place of $F(k, t)$ since those ISFs differ only by time-independent quantities.

As we saw in Sec. 3.2.5, another ISF that may feature a nonzero LTV is $F_s(k, t)$. Unfortunately, there exists no simple ISF that fully relaxes in HS-QA systems and differs from the self ISF merely by static quantities. However, in Sec. 3.2.5, we speculated that the LTV of the self ISF, $f_s(k)$, assumes a nonzero value only in the presence of trapped particles, meaning that $F_s(k, t)$ would relax to zero if in its evaluation exclusively the free (nontrapped) particles were considered. In the absence of a method to distinguish trapped from free particles, it seems reasonable to *emulate* the result of such a separation by purely mathematically reducing $f_s(k)$ to zero.

Our procedure of choice for this emulation is described in Sec. 2.5.3.5 and is illustrated by Fig. 3.17. Panel (a) of the figure contains the same information as Fig. 3.11(b), i.e., it depicts $F_s(k, t)$ for state points along path III ($\phi_f = 0.1$) and for $k = 7$. Additionally, as thin dotted lines it includes $f_s(k)$ as determined by the procedure in Sec. 2.5.3.5. Panel (b) of the figure shows $\hat{F}_s(k, t)$, the result of that procedure, which in a nutshell involves the subtraction of $f_s(k)$ and subsequently a renormalization so that $\hat{F}_s(k, t)$ ranges from zero to unity. Depicted as dark solid lines

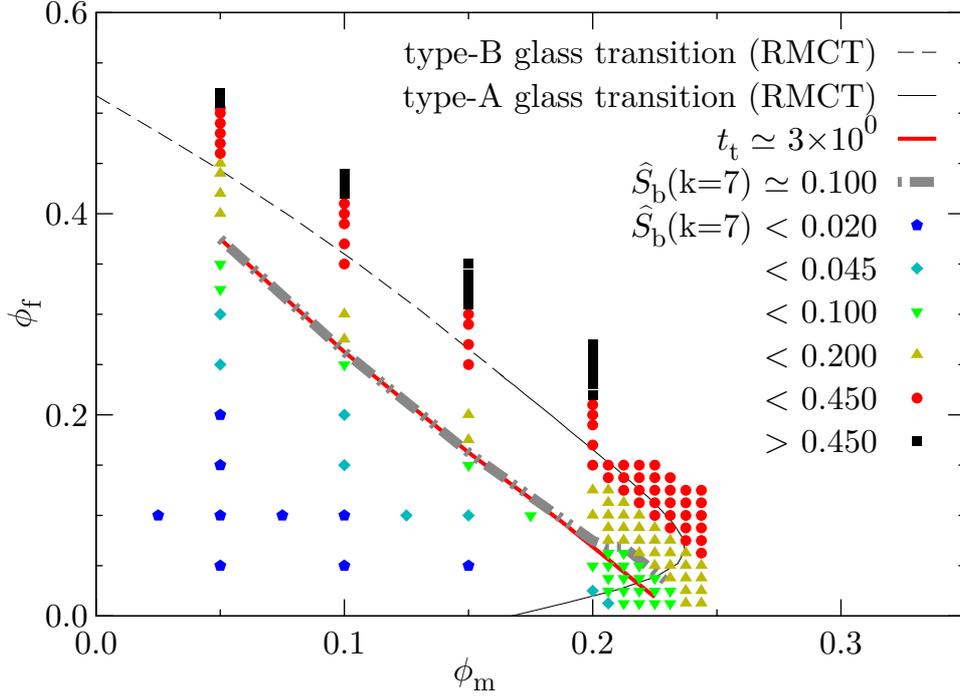


Figure 3.16: HS-QA “kinetic diagram”^{3.7}: normalized blocked structure factor. Symbol colors and shapes: normalized blocked structure factor, $\hat{S}_b(k)$, for $k = 7$ at the respective state point. Extra-thick dash-double-dotted gray line: interpolation between state points for which $\hat{S}_b(k) \simeq 0.1$. Thick solid red line: total-ISF arrest line from Fig. 3.15. Thin black lines: RMCT transition predictions for collective properties (cf. Fig. 1.19).

and as faint dashed lines are the state points for which the procedure can and cannot be applied, respectively, the distinguishing criterion being whether or not $f_s(k)$ can be determined at the state point in question.

In Fig. 3.18, the symbol colors and shapes encode the time \hat{t}_s required by $\hat{F}_s(k, t)$ to relax to the value 0.1 for $k = 7$. The thick dashed blue line represents an iso- \hat{t}_s line for $\hat{t}_s = 10^0$. Included for comparison are the self-ISF arrest line (thick solid blue line) and the connected-ISF arrest line (thick dash-dotted red line) as inferred in Sec. 3.2.6. Clearly, there is considerable similarity between the iso- \hat{t}_s and the iso- t_c lines, whereas both of these lines differ significantly from the iso- t_s line. The principal difference is found at large ϕ_m and small ϕ_f : while a slight extrapolation suggests that the iso- \hat{t}_s line intersects the abscissa at $\phi_m \approx 0.25$, for the other two lines, such an intersection can neither be confirmed nor ruled out by the depicted data. This finding lends further credibility to the KD based on $F_c(k, t)$. Finally, the lines indicate that at intermediate and large ϕ_m , the relaxation of $F_c(k, t)$ is faster than that of $\hat{F}_s(k, t)$, which confirms the pertinent notion established in Sec. 3.2.5.

Completing the picture, in Fig. 3.19, the symbol shapes and colors represent the value of $f_s(k)$ at the respective state point. Included for comparison are the $F_s(k, t)$ and $\hat{F}_s(k, t)$ arrest lines from Figs. 3.12 and 3.18, respectively. As the figure shows,

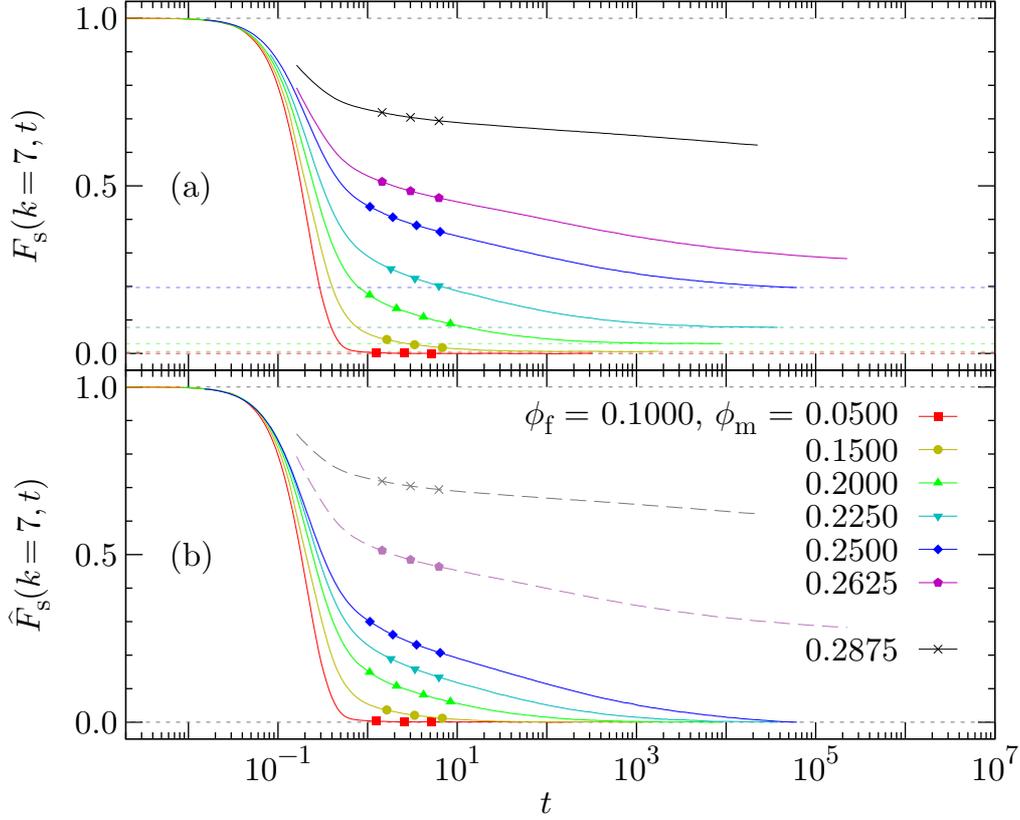


Figure 3.17: HS-QA single-particle ISF: long-time value subtraction. Single-particle ISF, $F_s(k, t)$, and modified single-particle ISF, $\hat{F}_s(k, t)$, for state points along path III at the wave vector $k = 7$. Panel (a): same data as in Fig. 3.11(b); additional thin dotted lines: long-time value $f_s(k)$ from the procedure in Sec. 2.5.3.5. Panel (b): result of applying the procedure to panel (a). Dark solid lines: procedure is applicable; faint dashed lines: procedure is not applicable. (Details: see text.)

$f_s(k)$ depends strongly upon ϕ_m but only weakly upon ϕ_f . Two features of the figure reflect that determining $f_s(k)$ from simulations is difficult: firstly, according to the symbol shapes and colors, $f_s(k)$ does not vary smoothly with ϕ_m and ϕ_f ; secondly, the figure includes only few symbols to the upper right of the $\hat{F}_s(k, t)$ arrest line. The latter, as Fig. 3.10 reveals, is due to the final relaxation of $F_s(k, t)$ towards $f_s(k)$, which in this parameter-space region is very slow and therefore frequently not within the simulated time interval.

Unfortunately, Fig. 3.18 demonstrates that $f_s(k)$ is a poor indicator of dynamic arrest, even when disregarding the uncertainties. The reason for this assessment lies in the iso- $f_s(k)$ lines, none of which even remotely coincides with any of the previously-determined arrest lines. On the other hand, the strong dependence of $f_s(k)$ upon ϕ_m supports the notion that nonzero LTVs of $F_s(k, t)$ relate to trapped particles. The additional weak dependence upon ϕ_f likely reflects the incapability of fluid particles to rearrange within traps. In this view, $f_s(k)$ amounts to a static quantity largely

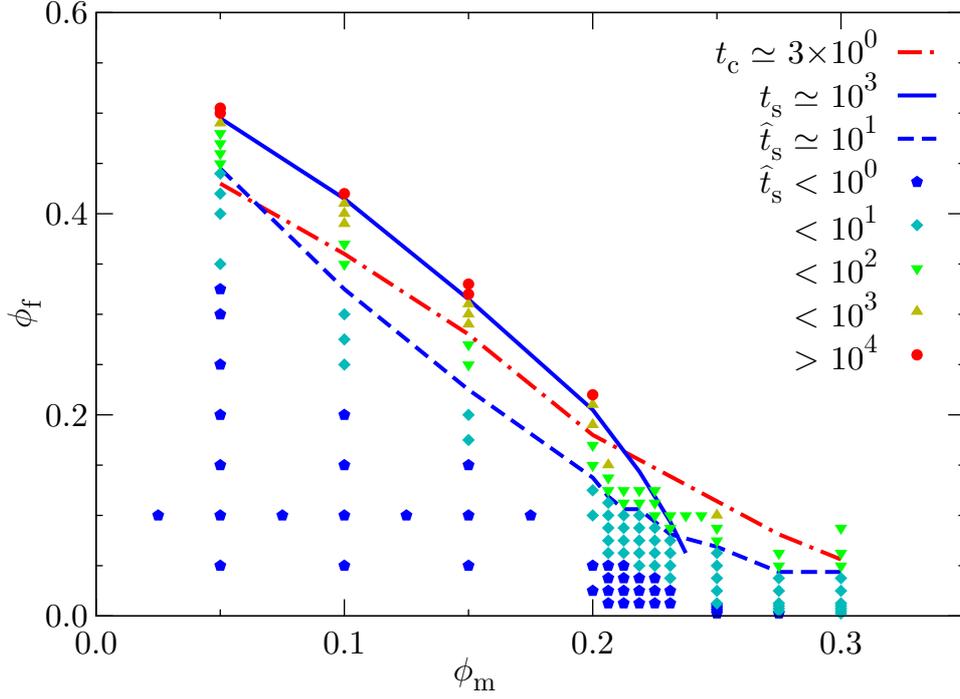


Figure 3.18: HS-QA kinetic diagram: self ISF without long-time value. Symbol colors and shapes: relaxation time \hat{t}_s of $\hat{F}_s(k=7, t)$ at the respective state point. Thick dashed blue line: interpolation between state points for which $\hat{t}_s \simeq 10^1$. Thick solid blue line: $F_s(k, t)$ arrest line from Fig. 3.12; thick dash-dotted red line: $F_c(k, t)$ arrest line from Fig. 3.13.

determined by the matrix (cf. Footnote 3.7). Alas, Fig. 3.18 also shows that $f_s(k)$ is insensitive to the most important feature of the matrix structure—the percolation threshold in the accessible volume (cf. Sec. 1.3.3.3). This is manifested in the continuous change of the LTV across the threshold at $\phi_m \simeq 0.25$ (cf. Sec. 3.2.4). The latter, however, is only moderately surprising considering that $f_s(k)$ for any specific particle is determined essentially by the distance it may travel away from its original position—and this distance does not change qualitatively across the threshold. In summary, these findings strongly suggest further investigation of the matrix structure.

3.2.9 Self part of the Van Hove function

As the final act of our full-fluid investigation, we consider the self part of the Van Hove function (the “self VHF”) as defined in Eq. (2.91). As laid out in the equation’s derivation in Sec. 2.5.3.6, the self VHF, $G_s(r, t)$, is closely related to both the MSD and the single-particle ISF and complements the information encoded in these quantities in at least two ways. Firstly, it spotlights the average local movement of the particles and thereby may elucidate mechanisms of dynamic arrest. Secondly, the self VHF reveals details of the matrix pores when considered in the infinite-time limit, the reason being that in that limit it reflects essentially only the trapped particles. The latter

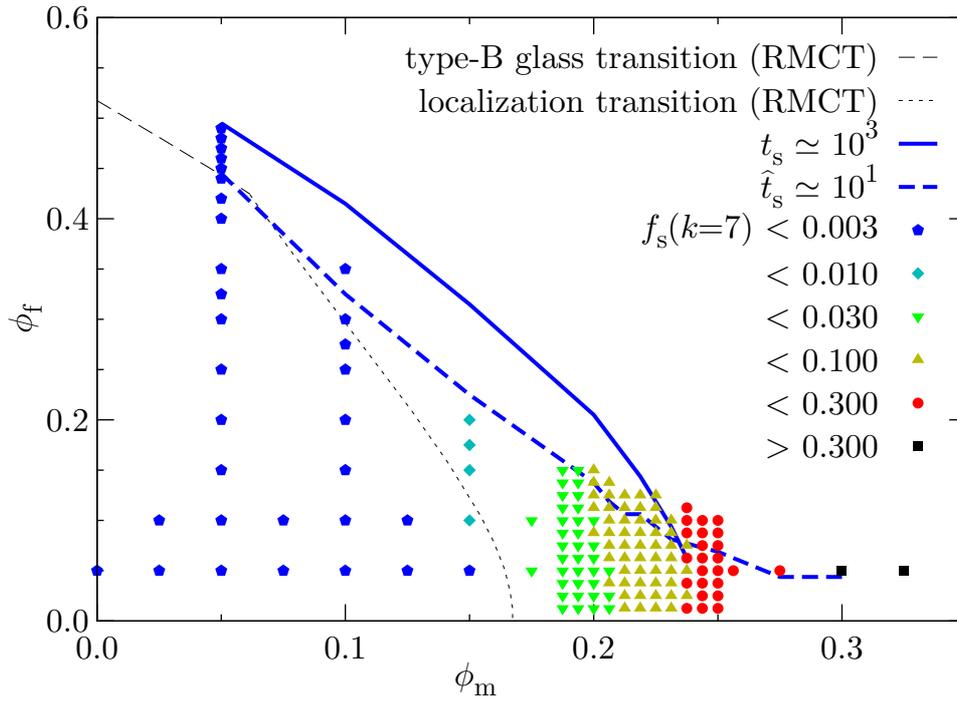


Figure 3.19: HS-QA “kinetic diagram”: self-ISF long-time value. Symbol colors and shapes: self-ISF long-time value, $f_s(k)$, for $k = 7$ at the respective state point. Thick solid blue line: self-ISF arrest line from Fig. 3.12. Thick dashed blue line: modified-self ISF arrest line from Fig. 3.18. Thin black lines: RMCT transition predictions for single-particle properties (cf. Fig. 1.19).

originates in the fact that for $t \rightarrow \infty$, the contribution of the free particles to $G_s(r, t)$ is infinitesimally small at all r . Therefore, the probability for a random fluid particle to be trapped may be estimated as

$$P_{\text{trap}} \simeq \int_0^{\tilde{r}} G_s(r, \tilde{t}) dr, \quad (3.3)$$

where the allotted time \tilde{t} and the cutoff distance \tilde{r} should be chosen large enough for an increase of either one to change P_{trap} only by a marginal amount. Unfortunately, in the simulation of some HS-QA systems, the free particles do not move significantly farther than the trapped particles within any computationally achievable time. This is the case in particular in systems containing large traps. In those cases, the achievable value of t may not suffice to select \tilde{r} and to evaluate P_{trap} in a meaningful way. This constitutes a problem in particular when attempting to determine the value $\phi_m = \phi_m^*$ beyond which $P_{\text{trap}} = 1$, that is, to pinpoint the percolation transition in the accessible volume (cf. Sec. 1.3.3.3). Relying only on the self VHF, this transition is instead more reliably determined from the functional form of $\lim_{t \rightarrow \infty} G_s(r, t)$, the reason being that in this limit, the function reflects the average size of traps. As described in Sec. 1.3.3.3, the latter is expected to be particularly large at the percolation transition, which implies that ϕ_m^* may be inferred as the value of ϕ_m at which $G_s(r, t)$ attains a maximum if r and t are held at some fixed, large value.

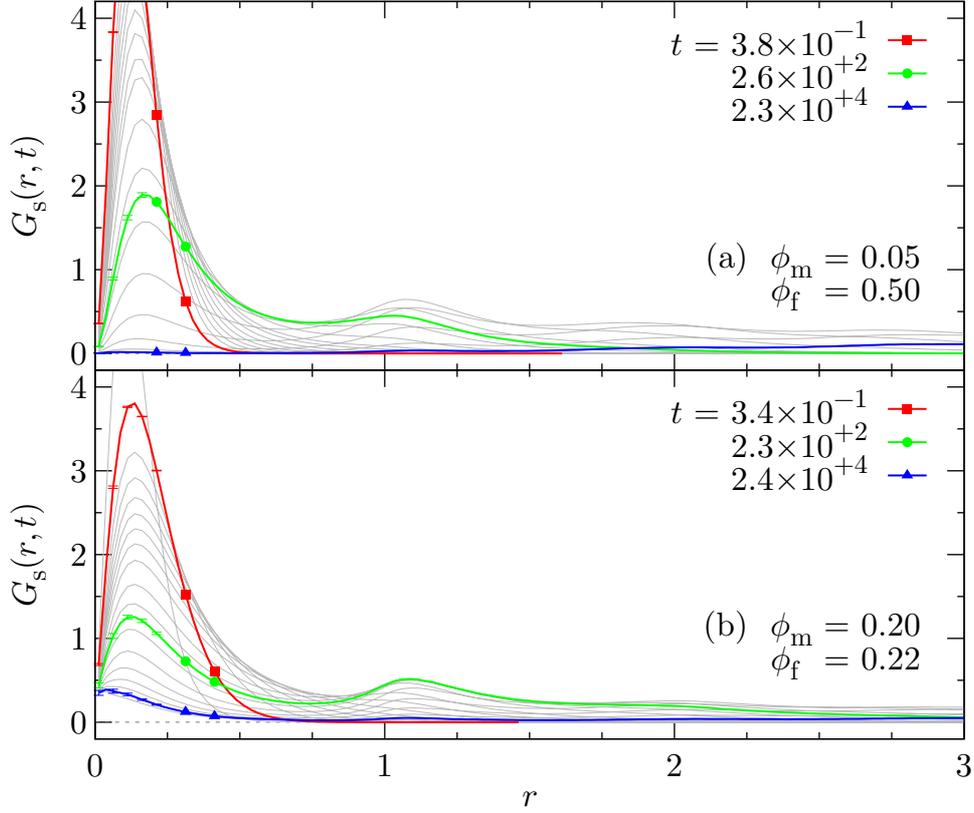


Figure 3.20: HS-QA self Van Hove function: points A and B. Lines: self Van Hove function, $G_s(r, t)$, at times differing by a factor of $\sim\sqrt[6]{10}$. Thick colored lines: selected times roughly coinciding across all panels of Figs. 3.20 and 3.21. Thin gray lines: $G_s(r, t)$ at other times. Panel (a): point A at $(\phi_m, \phi_f) = (0.05, 0.50)$; panel (b): point B at $(\phi_m, \phi_f) = (0.20, 0.22)$. Error bars: see Sec. 3.1.

Figures 3.20 and 3.21 visualize the self VHF in the conventional way, that is, the parameter r is considered as the continuous variable while the dependence upon t is represented by a series of curves. As mentioned in Sec. 2.5.3.6, this approach necessitates the focusing on a single state point at a time^{3.8}. In the figures, the lines represent the function at times spaced apart by a factor of $\sim\sqrt[6]{10}$, meaning that there are about six curves per time decade. The thick colored lines highlight the self VHF at selected times, with the same colors and symbols indicating roughly the same times across all panels of Figs. 3.20 and 3.21. Depicted as thin gray lines is $G_s(r, t)$ at all other times considered.

Figure 3.20 illustrates the self VHF for the two state points A and B residing at $(\phi_m, \phi_f) = (0.05, 0.50)$ and $(\phi_m, \phi_f) = (0.20, 0.22)$, respectively. As Fig. 3.2 shows, these points are located close to the intersection of the MSD arrest line with path I and with path II. In both cases, $G_s(r, t)$ at short times roughly resembles a Gaussian

^{3.8}In the case of the various ISFs investigated in this work, multiple state points are instead displayed in a single plot at the expense of having to consider one specific wave vector k .

distribution. As time progresses, the maximum of this distribution decreases while it remains centered about approximately the same value of r . This is consistent with the cage picture introduced in Sec. 1.2.7, where particles are restricted to small spatial regions unless collective rearrangements take place. Upon such rearrangements, the particles are expected to “hop” by approximately one particle diameter at a time. This is confirmed in both panels in that a second maximum emerges at $r \simeq 1.1$ for $t \approx 10^2$. Beyond this time, even a modest third maximum can be identified at $r \simeq 2$, until eventually $G_s(r, t)$ recovers a shallow, essentially Gaussian shape centered at $r > 3$.

Aside from these commonalities, the two panels of Fig. 3.20 also exhibit differences. For one, the distances between the depicted curves indicate that the first maximum decreases considerably faster for point A than for point B. Also, all maxima are more pronounced for point A. More significantly, at small r and large t , the self VHF differs *qualitatively* between the two panels, namely in that in panel (b), it remains nonzero at $r < \tilde{r} = 0.5$ even for the largest depicted time, $\tilde{t} = 8.5 \times 10^4$. Since in that spatial region, the self VHF changes only marginally for $t > 10^4$, it is reasonable to input \tilde{r} and \tilde{t} into Eq. (3.3), yielding the probability $P_{\text{trap}} \simeq 8\%$ for a fluid particle to be trapped in systems at point B. Further, since for $t = \tilde{t}$ the self VHF is very nearly zero at $r > \tilde{r}$, there likely exist only few traps with a spatial extent exceeding \tilde{r} .

Figure 3.21 depicts the self VHF for points C and D, i.e., for $(\phi_m, \phi_f) = (0.25, 0.10)$ and $(\phi_m, \phi_f) = (0.30, 0.05)$. Recalling Fig. 3.2 and Sec. 3.2.1, point C is situated both on path III and close to the percolation transition in the accessible volume, whereas point D resides at a ϕ_m value well past the transition. In panel (d), the curves at large times are narrowly spaced at all r , meaning that the largest time in that panel, $\tilde{t} = 7.8 \times 10^4$, excellently approximates the infinite-time limit. Therefore, for point D, it is meaningful to use \tilde{t} along with $\tilde{r} = 3$ (the largest depicted distance) to evaluate Eq. (3.3). The resulting $P_{\text{trap}} \simeq 98\%$ very closely reflects the expectation that beyond the percolation transition, all particles are trapped. In panel (c), in contrast, at the largest depicted distance and time, $\tilde{r} = 3$ and $\tilde{t} = 7.8 \times 10^4$, the self VHF significantly exceeds zero while leveling off only very slowly with r . Therefore, in that case it is uncertain whether $G_s(\tilde{r}, \tilde{t})$ reasonably represents $\lim_{t \rightarrow \infty} G_s(r, t)$ at all values of r . Evaluating Eq. (3.3) anyway, the values of \tilde{r} and \tilde{t} as above yield $P_{\text{trap}} \approx 58\%$, which is a fairly small—and therefore probably unreliable—value considering that point C is thought to be located in the vicinity of the percolation transition where $P_{\text{trap}} = 1$.

The functional features of $G_s(r, t)$ in Fig. 3.21 represent—to some degree—a continuation of the trend from Fig. 3.20(a) to Fig. 3.20(b): the larger ϕ_m is, the slower the self VHF evolves and the less prominent its maxima are. However, there also exist significant deviations from this pattern. For instance, for points C and D, the second maximum persists even for $t \rightarrow \infty$ while for points A and B it does not, suggesting that in the former cases, the average trap is sufficient in size to permit collective particle rearrangements. Also, $\lim_{t \rightarrow \infty} G_s(r, t)$ at large distances is *greater* in panel (c) than in panel (d). This implies that traps on average are larger at point C than at point D, which is consistent with the notion of the average trap size attaining a maximum at

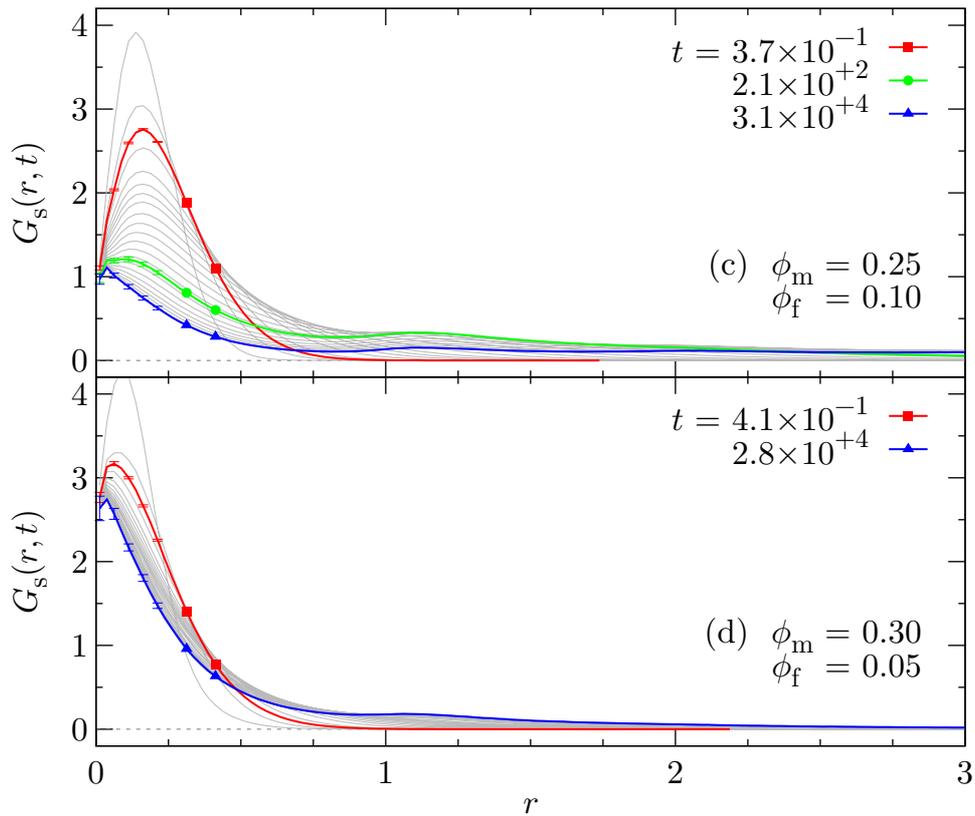


Figure 3.21: HS-QA self Van Hove function: points C and D. Lines: self Van Hove function, $G_s(r, t)$, at times differing by a factor of $\sim\sqrt[6]{10}$. Thick colored lines: selected times roughly coinciding across all panels of Figs. 3.20 and 3.21. Thin gray lines: $G_s(r, t)$ at other times. Panel (c): point C at $(\phi_m, \phi_f) = (0.25, 0.10)$; panel (d): point D at $(\phi_m, \phi_f) = (0.30, 0.05)$. Error bars: see Sec. 3.1.

the percolation transition of the accessible volume. Thus, Figs. 3.20 and 3.21 establish that $0.22 < \phi_m^* < 0.30$, which is in accordance with the previous sections (albeit significantly less precise).

3.3 Structure of the accessible volume

In this section, we take on the task of quantifying the structural features of the accessible volume in monodisperse hard-sphere quenched-annealed (HS-QA) systems. Let us briefly recall the subject matter. Under certain conditions, the fluid component of an HS-QA system may include particles that cannot move infinitely far away from their initial position. Those particles we will call “trapped”, while all other fluid particles we will refer to as “free”. The trapping of fluid particles may occur if (i) sufficiently many matrix particles are present and are appropriately arranged, and if (ii) the interactions between the fluid and the matrix particles are of a suitable type. As discussed in Sec. 2.4.1, both of these criteria hold in the systems investigated in this work. Since

moreover in our systems the particles are monodisperse, each fluid particle experiences the exact same matrix environment (cf. Sec. 1.1.3.2), which greatly simplifies the task at hand.

Consider now system volume regions instead of particles. With respect to the center of a fluid particle, the presence of the matrix particles gives rise to two types of such regions: those in which the potential energy of a fluid particle diverges, and those in which it vanishes. The latter regions we will collectively call the “accessible volume”. The accessible volume exists in the form of distinct, noninterconnected “voids”, the size of each of which may be infinite (“percolating voids”) or finite (“traps”). In Sec. 2.4 and Appendix A.1, we discussed in excruciating detail how a Delaunay (“D-”) tessellation may reveal for every void in a system whether it constitutes a trap or a percolating void.

Figure 3.22 illustrates the D-tessellation of an actual HS-QA system realization, namely of the one depicted in Fig. 3.3(b) located at point C where the matrix and fluid packing fractions are $(\phi_m, \phi_f) = (0.25, 0.10)$. Figure 3.22(a) focuses on the system’s matrix particles (dark gray spheres) while neglecting its fluid particles. In panel (b), superimposed onto those particles are all D-edges (red lines) of the pertinent D-tessellation^{3.9}. In panel (c), the matrix particles are removed as a segue to panel (d) where, finally, the presence of a percolating void in the system is exposed by the highlighting of all pertinent D-cells as solid green tetrahedra.

In Sec. 2.4.6, we specified an algorithm to estimate a void’s “size”, i.e., the amount of the accessible volume associated with it. In brief, the algorithm essentially involves the following steps:

- (i) Given some D-cell, define a volume that completely contains the D-cell.
- (ii) Within that volume, define a number of suitably distributed points. Among these points, determine those located within the given D-cell.
- (iii) Among the latter points, determine those representing allowed locations for the center of a fluid particle.

The accessible volume within the given D-cell may subsequently be calculated from the number of points resulting from steps (ii) and (iii), respectively. Once the accessible volume of each D-cell in the system is known, and once the connectivity of the D-cells has been determined, the size of any void in the system is easily evaluated.

Figure 3.23 illustrates the algorithm for the same system realization as in Fig. 3.22, i.e., for a system at point C (cf. Fig. 3.2). In both panels of the figure, the colored volumes collectively represent the entire accessible volume. Colored in green and red

^{3.9}In Fig. 3.22(b), some Delaunay-edge end points appear to not coincide with the center of a matrix particle. This is a mere artifact of the periodic boundary conditions (PBCs). As is typical in visualizations of systems with PBCs, each element is included in the figure exactly once. Since the definition of a Delaunay edge involves *two* matrix particles, it may come to pass that one of those particles is represented in the figure by one of its periodic images.

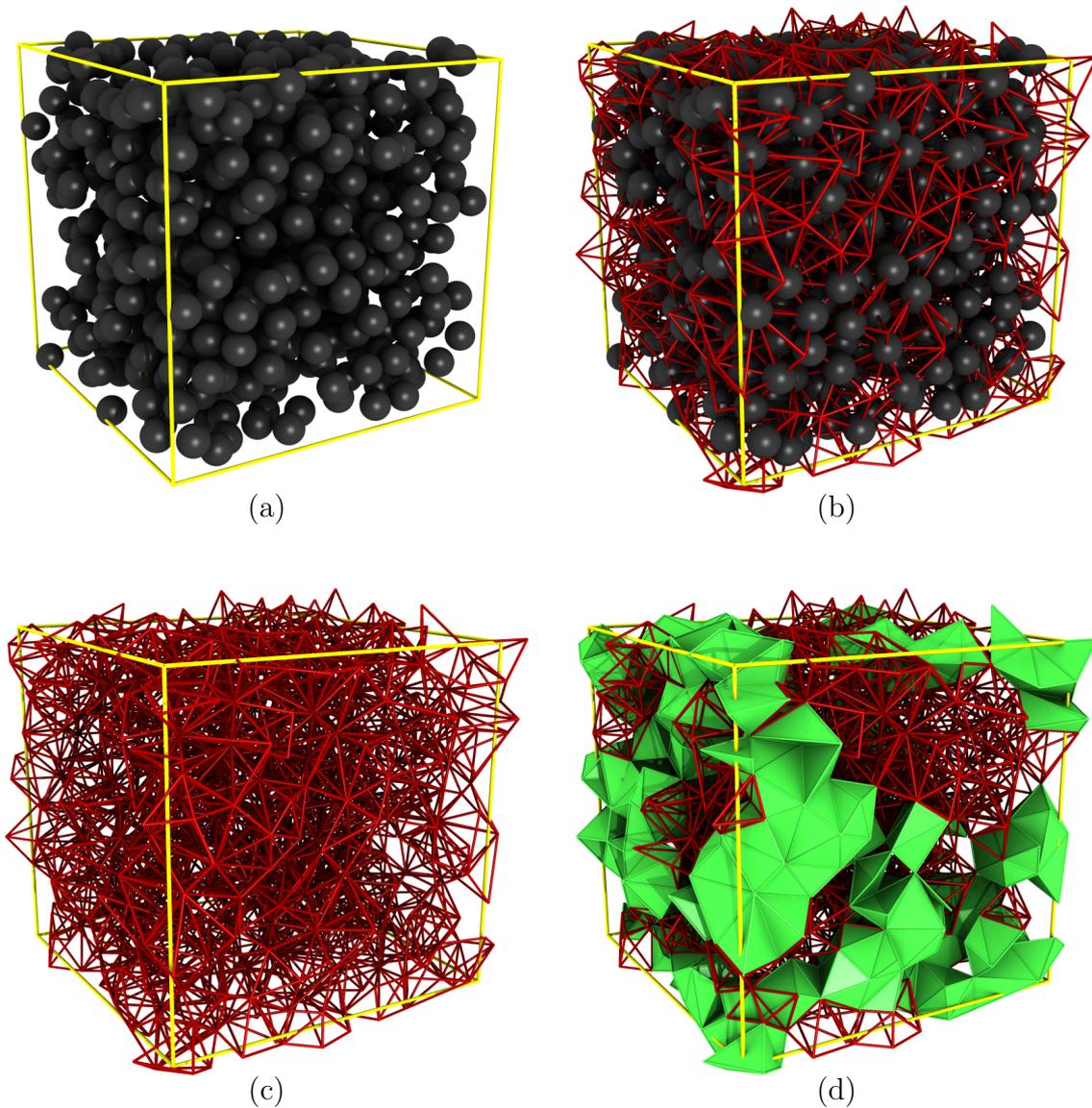


Figure 3.22: Delaunay tessellation of HS-QA systems. Illustration of the Delaunay tessellation of an actual HS-QA system. Same system realization as in Fig. 3.3(b), containing $N_m = 715$ matrix particles and residing at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$. Panel (a): only matrix particles (dark gray spheres). Panel (b): matrix particles superimposed onto all Delaunay edges (red lines)^{3,9}. Panel (c): only Delaunay edges. Panel (d): Delaunay edges of traps and Delaunay cells of percolating void (solid green tetrahedra). Yellow lines: simulation cell replicated by periodic boundary conditions (cf. Sec. 2.2.2.1).

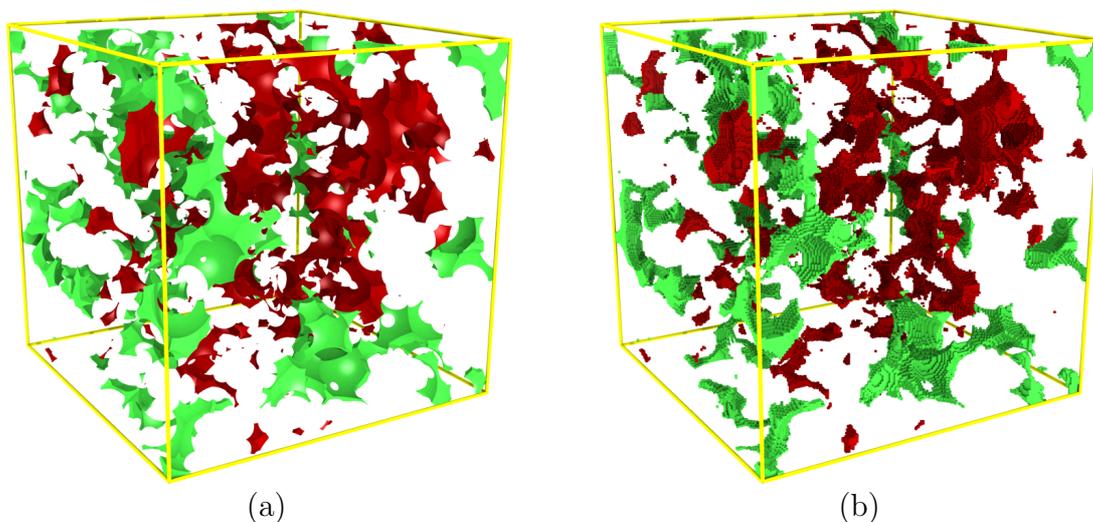


Figure 3.23: Void size determination in HS-QA systems. Illustration of the rastering algorithm for an actual HS-QA system. Same system realization as in Fig. 3.3(b) and Fig. 3.22, containing $N_m = 715$ matrix particles and residing at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$. Green volumes: percolating void; red volumes: traps. Panel (a): exact visual appearance of voids determined by imaging software^{3.11}. Panel (b): extent of voids determined by rastering algorithm; grain size: $0.065 \times 0.065 \times 0.065$. Yellow lines: simulation cell replicated by periodic boundary conditions (cf. Sec. 2.2.2.1).

are the system’s percolating void and its traps, respectively. The difference between the two panels lies in the following. In panel (a), the appearance of the voids was determined using the internal functions of the image generation software^{3.10}, resulting in a visual representation of the voids that is exact to the resolution of the image. This image therefore provides a reference for panel (b), in which the extent of the voids was determined by the algorithm described above. Close inspection of panel (b) reveals that the voids are composed of small cubic elements (“grains”), each of which represents the volume associated with one of the points defined in the algorithm’s step (ii). The cubic shape of the grains reflects our choice of arranging the points on a cubic lattice. In the figure, the lattice constant of that lattice is 0.065, equating to about fifteen grains per particle diameter or $\sim 5.5 \times 10^6$ grains in the simulation cell.

^{3.10}All three-dimensional illustrations in this work were realized using the ray-tracing software POV-Ray^{3.11}. The appearance of the voids in Fig. 3.22(b) and in several other visualizations was determined by means of the “constructive solid geometry” features of that software. Generating the images essentially involved two steps: (i) filling the volume with all cells of the Delaunay tessellation in question, each cell in appropriate color, and (ii) boring spherical holes into those cells, each sphere centered at a matrix particle and measuring 2σ in diameter (the sum of the diameters of a fluid and a matrix particle).

^{3.11}<http://www.povray.org>

Unless stated otherwise, the same lattice constant was employed in the remainder of this investigation of the accessible volume’s structure. Finally, since the arrangement and the processing of the grains in our algorithm is analogous to the rendering of raster (i.e., pixel) computer graphics, we also refer to our algorithm as a “rastering” algorithm (cf. Sec. 2.4.6).

3.3.1 Accessibility probabilities

Starting with this section, we report on our investigation of the key quantities that characterize the accessible volume in monodisperse HS-QA systems. In infinitely-large systems, these quantities would depend solely upon the matrix packing fraction, ϕ_m . However, in computer simulations—including those in this work—it is merely possible to mimic infinitely-large systems by means of periodic boundary conditions (PBCs; see Sec. 2.2.2.1). This mimicking cannot account for structures spatially exceeding the volume replicated by the PBCs, which is relevant recalling that porous media may exhibit diverging length scales (cf. Sec. 1.3.3.3). Those scales will effectively be cut off, leading to any associated quantity appearing to depend upon the system’s size. Ironically, it is this very effect that renders observed quantities particularly sensitive to diverging length scales. For this reason, we deliberately determine all possibly-affected quantities not only for varying ϕ_m but also for varying system sizes as represented by the number of matrix particles, N_m .

In the present section, we consider three different quantities for the description of the accessible volume in HS-QA systems. The first one is the probability $P_{\text{grain}}(\phi_m; N_m)$ for a random spatial point to reside within the accessible volume. This quantity is evaluated using the rastering algorithm^{3,12} (outlined further above and described in detail in Sec. 2.4.6) and is particularly simple in that it does not require knowledge about the D-tessellation of the system in question. This contrasts with the second and third considered quantities, namely the probability $P_{\text{face}}(\phi_m; N_m)$ for a random D-face to be crossable and the probability $P_{\text{cell}}(\phi_m; N_m)$ for a random D-cell to be accessible. These quantities are meaningful only if the D-tessellation is known, with the crossability and accessibility being determined by the criteria specified in Sec. 2.4.3, respectively.

Figure 3.24 illustrates P_{grain} , P_{face} , and P_{cell} for an interval of ϕ_m containing ϕ_m^* , i.e., the value of ϕ_m at which the percolation transition in the accessible volume takes place. Anticipating the analysis of Fig. 3.27(b), the most accurate value of ϕ_m^* determined in this work is indicated by the thin dashed gray line stretching vertically through all panels of Fig. 3.24. Clearly, none of the considered quantities even remotely indicates a transition at ϕ_m^* ; instead, they vary monotonically and only very moderately within the considered ϕ_m range. Therefore, as has been pointed out before [196, 252], the dynamics of HS-QA systems at $\phi_m \approx \phi_m^*$ cannot be inferred from P_{grain} , and

^{3,12}The unusual abbreviation P_{grain} serves as a reminder of the fact that the rastering algorithm by which P_{grain} is evaluated in this work involves distinct spatial units that we chose to call “grains”. However, the definition of P_{grain} is in fact ignorant of the notion of those grains (cf. Sec. 2.5.4.1).

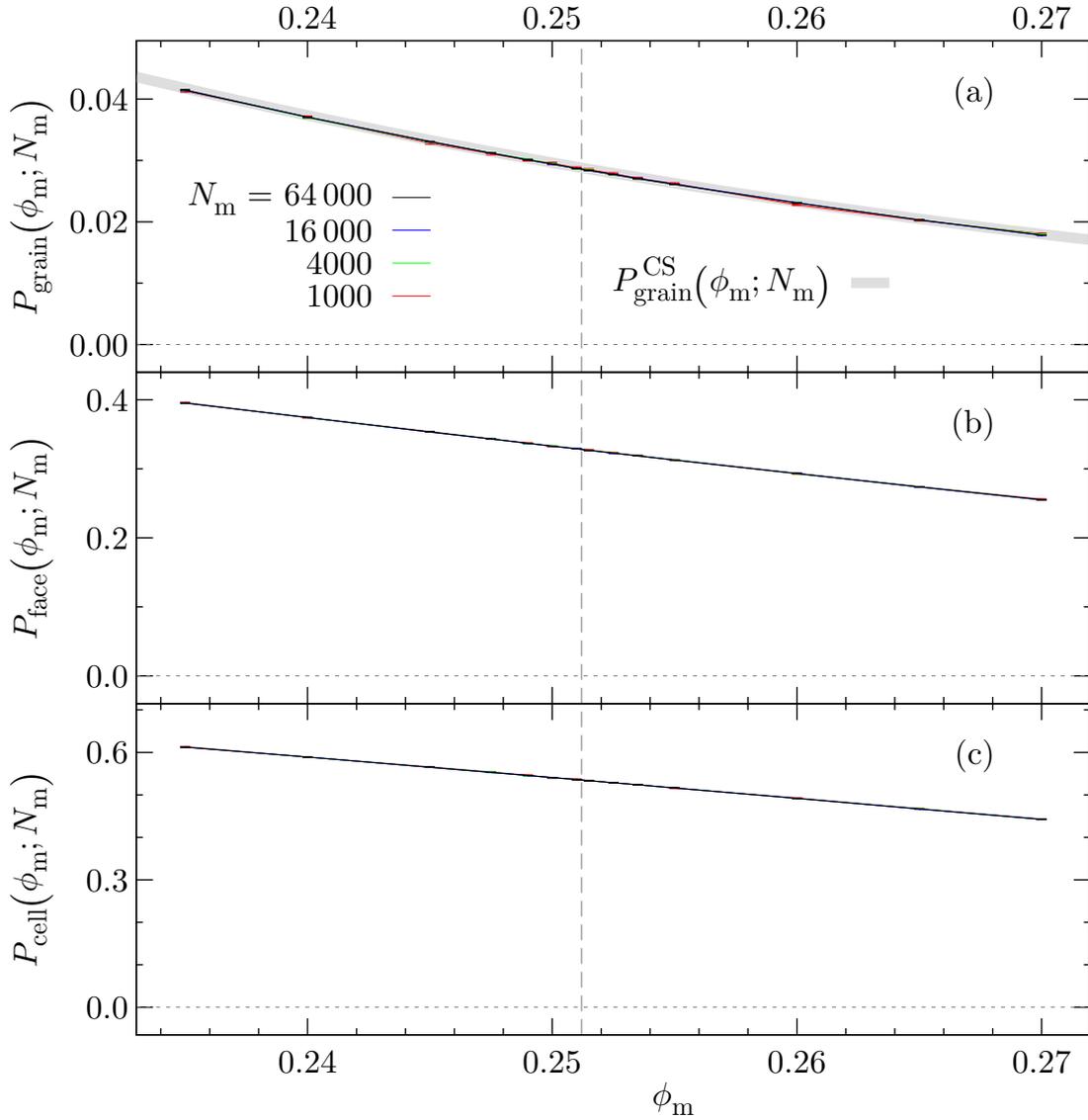


Figure 3.24: Accessibility probability: grains, D-faces, D-cells. Probabilities for accessibility as functions of the matrix packing fraction, ϕ_m . Panel (a): probability $P_{\text{grain}}(\phi_m; N_m)$ for grains to be accessible. Panel (b): probability $P_{\text{face}}(\phi_m; N_m)$ for Delaunay faces to be accessible, i.e., crossable. Panel (c): probability $P_{\text{cell}}(\phi_m; N_m)$ for Delaunay cells to be accessible. Colors of thin lines: results for different matrix particle numbers, N_m . Extra-thick light-gray line in panel (a): $P_{\text{grain}}^{\text{CS}}(\phi_m; N_m)$ as defined in Eq. (2.96). Dashed vertical line: percolation threshold from Fig. 3.27(b). Error bars: see Sec. 3.1.

neither can it be from P_{face} or P_{cell} . As is evidenced by the stupendous agreement between the lines of different color, all quantities in Fig. 3.24 are independent of the size of the system. Therefore, none of those quantities can involve a diverging length scale, which further supports the notion of P_{grain} , P_{face} , and P_{cell} being oblivious to the percolation transition. Finally, P_{grain} is in excellent agreement with the superimposed extra-thick light-gray line, the latter denoting the theoretically-inferred $P_{\text{grain}}^{\text{CS}}$ as defined in Eq. (2.96). This, together with the near-invisibility of the error bars, lends further credibility to the accuracy of our findings. In a final side note, we found the average number of D-cells per matrix particle, $n_{\text{cell}}(\phi_m)$, to monotonically decrease from $n_{\text{cell}}(0.235) \simeq 6.584$ to $n_{\text{cell}}(0.270) \simeq 6.547$, the corresponding numbers for D-faces equaling exactly the twofold of those figures in systems under PBCs (cf. Appendix A.3.4).

3.3.2 Probabilities for percolation and trapping

In Sec. 2.4.5, we described an algorithm by which D-cell clusters—D-cells interlinked by crossable D-faces—can be identified in D-tessellations of HS-QA systems. An essential step of this algorithm is to resolve whether or not a D-cell cluster encompasses infinitely-many D-cells when taking into account PBCs. Since each D-cell cluster circumscribes precisely one void, the algorithm naturally deduces whether or not the accessible volume in the system of interest is percolating. Infinitely-large systems always contain a percolating void for $\phi_m < \phi_m^*$ and never do so for $\phi_m > \phi_m^*$, meaning that a transition takes place at $\phi_m = \phi_m^*$. Realizations of finite systems in the vicinity of ϕ_m^* , in contrast, exhibit a percolating void only with some probability $P_{\text{percol}}(\phi_m; N_m)$, the latter generally assuming values *between* zero and unity (cf. Sec. 2.5.4.2).

Another application of the D-cell cluster algorithm is to determine the trapping fraction, i.e., the probability $P_{\text{trap}}(\phi_m; N_m)$ that a fluid particle is trapped after being randomly inserted into the system (see again Sec. 2.5.4.2). This quantity may be inferred as the fraction of accessible volume located within traps. Unlike P_{percol} , the trapping fraction at $\phi_m < \phi_m^*$ assumes values between zero and unity irrespective of the system size, reflecting that HS-QA systems at $\phi_m > 0$ on average always contain traps (cf. Sec. 2.4.1). For $\phi_m > \phi_m^*$, the trapping fraction is strictly unity if the considered system is infinitely large. In contrast, finite systems at $\phi_m > \phi_m^*$ contain a percolating void only with some sub-unity probability, meaning that in such systems $P_{\text{trap}} < 1$ even in that ϕ_m range.

Panel (a) of Fig. 3.25 depicts $P_{\text{percol}}(\phi_m; N_m)$ within a ϕ_m range coinciding with that of Fig. 3.24. As postulated, P_{percol} varies not only with ϕ_m but also considerably with N_m . With respect to the step function expected for infinitely-large systems, P_{percol} is significantly enhanced for $\phi_m > \phi_m^*$ while it is equally depressed for $\phi_m < \phi_m^*$. This effect is most pronounced for the smallest simulated systems, which conforms with the notion of large length scales being cut off in small PBC-replicated systems (see above as well as Secs. 2.5.4.2 and 3.3.1). The value of ϕ_m^* may be estimated from the change of P_{percol} from unity to zero for the largest considered N_m value. This

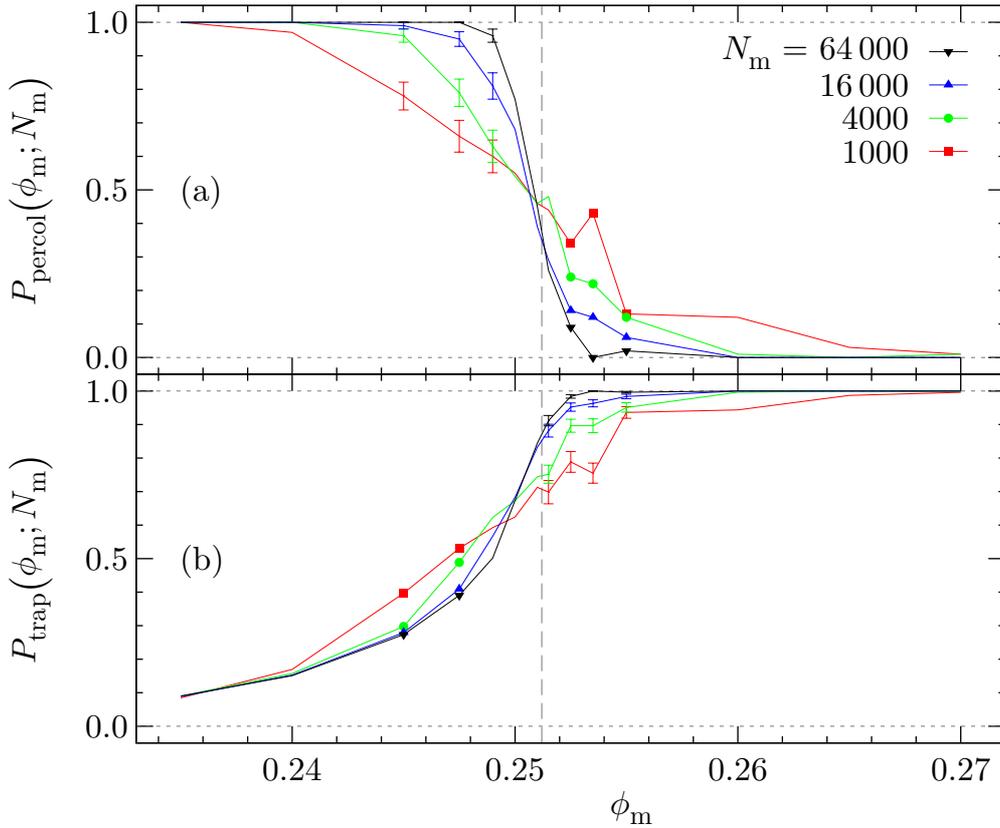


Figure 3.25: Probabilities for percolation and trapping. Probabilities as functions of the matrix packing fraction, ϕ_m . Panel (a): probability $P_{\text{percol}}(\phi_m; N_m)$ for the accessible volume to be percolating. Panel (b): probability $P_{\text{trap}}(\phi_m; N_m)$ for a fluid particle to be trapped. Line colors and symbol shapes: results for different matrix particle numbers, N_m . Dashed vertical line: percolation threshold from Fig. 3.27(b). Error bars: see Sec. 3.1.

change takes place within the interval $0.248 \lesssim \phi_m^* \lesssim 0.254$, which indeed contains the more precise ϕ_m^* value determined from Fig. 3.27(b) (thin dashed gray vertical line). Notably, it is a coincidence that all P_{percol} curves for different N_m intersect within a narrow ϕ_m range; this finding is unrelated to scaling considerations and does not provide a more precise estimate of ϕ_m^* [248].

Panel (b) of Fig. 3.25 shows $P_{\text{trap}}(\phi_m; N_m)$ within the same ϕ_m range as panel (a). As expected, P_{trap} is less suited than P_{percol} for the task of inferring ϕ_m^* . This is rooted in the fact that the value of P_{trap} is known a priori only for $\phi_m > \phi_m^*$ in infinitely-large systems. Therefore, P_{trap} in finite systems merely allows to deduce an upper limit of ϕ_m^* , namely the ϕ_m value at which P_{trap} approaches unity for the largest considered N_m value. As the figure evidences, the pertinent value $\phi_m \simeq 0.253$ is slightly smaller (and therefore more precise) than the upper limit inferred above from P_{percol} . Finally, reconsider the values of P_{trap} at $\phi_m < \phi_m^*$ which we previously disregarded. Those data illustrate that even if ϕ_m is as large as $0.95\phi_m^*$, approximately 90% of the fluid particles are *free*.

3.3.3 Trap-size distributions

Having performed the D-cell cluster analysis (cf. Secs. 2.4.5 and 3.3.2) for a number of systems, and having used the rastering algorithm (see the beginning of Sec. 3.3 as well as Sec. 2.4.6) to evaluate the size of all voids in those systems, the natural next step is to investigate the distribution of the void sizes. However, before continuing along these lines, it is vital to call to mind that the rastering algorithm is subject to a fundamental limitation: by its very principle, the algorithm yields only an approximate value V' for the size of a given void.

On average, the volume contributed to V' by a surface grain (a grain intersected by the surface of the given void) deviates from the actual accessible volume within that grain by one quarter^{3.13} of a grain volume, V_g . For two reasons, the resulting relative error of V' is largest for small voids: (i) the number of surface grains scales^{3.14} only as $N_{sg} \propto V^{2/3}$, and (ii) the contributions of the surface grains to the error tend to cancel each other out if N_{sg} is large^{3.15}. Therefore, given the system size (via ϕ_m and N_m) as well as the total number of grains in the system, N_g , one may define a void size V_{min} beyond which the calculated size of a void is “probably sufficiently accurate”. In this context, it is of importance that the achievable value of N_g is subject to computational limits. The dominant limit in our implementation of the rastering algorithm is that of memory, the reason being that we chose to store the data of all N_g grains simultaneously in the computer’s RAM^{3.16}. If in a given system, the value of V_{min} is dissatisfactory even when using the maximum-possible N_g value, one may additionally reduce the size of the system, i.e., decrease N_m . Obviously, such a decrease is advisable only if the relevant features of the system at large length scales continue to be represented adequately.

Having established these limits, let us now consider two different representations of the void size data. Firstly, we calculate the probability $P(V)$ for a randomly-chosen void to contain the amount of accessible volume V (cf. Sec. 2.5.4.3), and secondly, we compute the probability $P(V)V$ that a randomly-chosen point of the accessible volume is located within a void of size V . As alluded to by the notation, the latter quantity simply equates to the volume-weighted version of the former (see again Sec. 2.5.4.3).

^{3.13}Suppose that each surface grain were counted as fully covered by accessible volume (case A). Then, the actual accessible volume within a surface grain would on average be represented with an error of $V_g/2$. The same would hold if surface grains were always noted as devoid of accessible volume (case B). Since the rastering algorithm allows for both case A and case B, the pertinent error amounts to half of the aforementioned error.

^{3.14}With respect to some linear extent L , surfaces scale as L^2 while volumes scale as L^3 .

^{3.15}If the contributions of the surface grains to the error of V' were normally distributed—which they are not since the possible error values are bounded—, then the average error in the calculated size of a void with N_{sg} surface grains would be $\sqrt{N_{sg}}(V_g/4)$.

^{3.16}In principle, it is fairly simple to avoid the problem of limited computer memory in implementations of the rastering algorithm. However, this usually comes at the cost of considerable increases in the computing effort (the latter being substantial already in our implementation), thereby shifting the limitation of N_g to this factor.

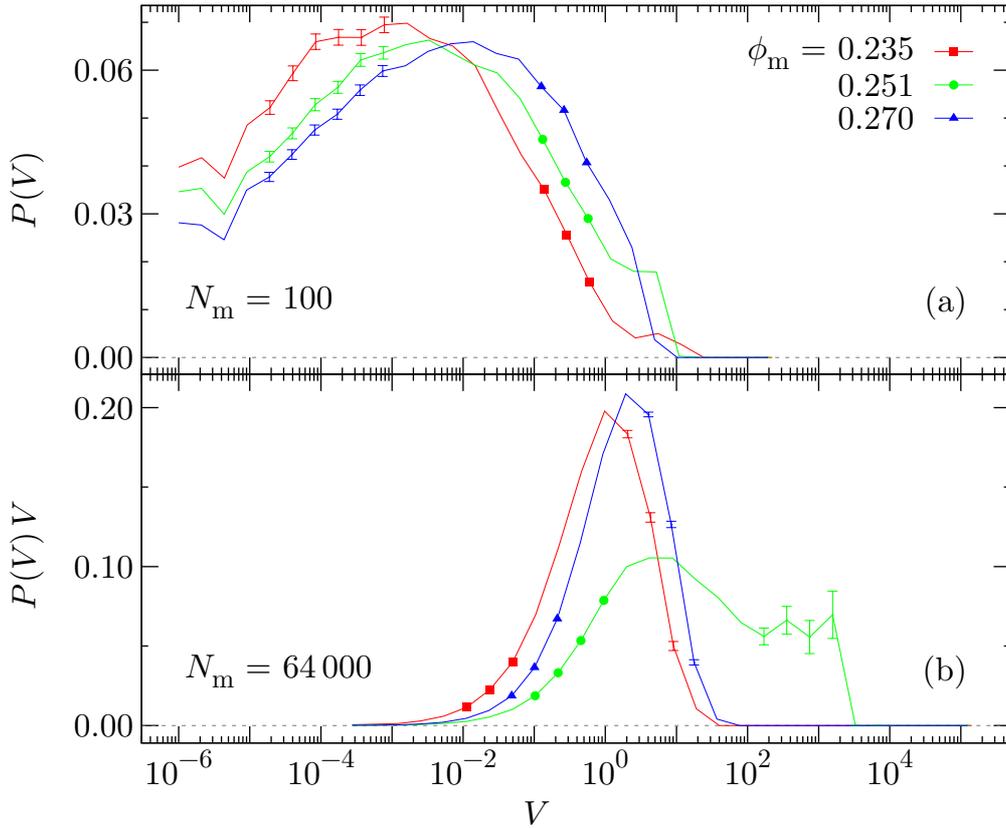


Figure 3.26: Trap-size distributions. Panel (a): probability $P(V)$ for a trap to be of size V . Panel (b): probability $P(V)V$ for a random point of the accessible volume to be located in a trap of size V . Same colors and symbols: systems of the same matrix packing fraction, ϕ_m . The matrix particle numbers are $N_m = 100$ in panel (a) and $N_m = 64\,000$ in panel (b). Error bars: see Sec. 3.1.

For two reasons, in our investigation we disregard the possibly-present percolating void: (i) its share of the accessible volume is known to be $1 - P_{\text{trap}}$ [compare Eq. (2.97)], and (ii) in the void size distribution of infinitely-large systems, a percolating void merely adds a contribution at $V = \infty$. Therefore, in more precise terms, in this section we consider the distribution of *trap* sizes.

Panel (a) of Fig. 3.26 presents the unweighted distribution $P(V)$. In the figure, V is considered as the continuous variable while the distribution's dependence upon ϕ_m (which is notationally suppressed) is represented by a series of curves. For the latter, three distinct values of ϕ_m are considered, one of which coincides with the percolation threshold $\phi_m^* = 0.2512$ [inferred from Fig. 3.27(b)] while the other two are located slightly above and below ϕ_m^* , respectively. Since the rastering algorithm was limited to $N_g \simeq 2 \times 10^8$ grains, the size of the investigated systems was adjusted to as few as $N_m = 100$ matrix particles. By this adjustment, $P(V)$ is represented acceptably for values of V as small as $V_{\text{min}} \simeq 10^{-5}$, the latter value being marked by the unphysical upward kink of $P(V)$ at small V . Nevertheless, this choice of N_m retains the possibility

for traps to attain sizes as large as $V \simeq 10^1$. The functional form of $P(V)$ overall is relatively simple, with a single broad maximum located at $V \simeq 10^{-3}$ for $\phi_m = 0.235$ and at $V \simeq 10^{-2}$ for $\phi_m = 0.270$. Interestingly, a similar monotonic shift with ϕ_m applies to the distribution as a whole. For all ϕ_m , a rapid decrease of $P(V)$ can be observed at $V > 10^1$ while an analogous decrease for $V < 10^{-5}$ seems plausible. Given that these findings also hold for $\phi_m = \phi_m^*$, it is likely that the traps in HS-QA systems at the percolation transition are not self-similar and therefore not fractal^{3.17}. In fact, $P(V)$ in general is remarkably insensitive to the transition, its only indication being the small shoulder at $V \simeq 10^1$ for $\phi_m = \phi_m^*$.

Panel (b) of Fig. 3.26 illustrates the volume-weighted distribution $P(V)V$ in the same form and for the same ϕ_m values as $P(V)$ in panel (a). As is obvious from the functional form of this quantity, the predominant technical issue in this case is the representation at large values of V . This issue required the consideration of systems with as many as $N_m = 64\,000$ matrix particles, thus providing system volumes in excess of $V \simeq 10^4$, where $P(V)V$ essentially attains zero. As a result of the volume weighting (which enhances features at large V) the presence of a percolation transition is clearly indicated by $P(V)V$. This finding—which is in stark contrast to panel (a)—is manifested in the functional form of $P(V)V$, which differs significantly between $\phi_m = \phi_m^*$ on the one hand and the other two ϕ_m values on the other hand. Specifically, for $\phi_m = \phi_m^*$, the distribution assumes the considerable value 0.05 for trap sizes as large as 2×10^3 , while $P(V)V$ for the other two ϕ_m values approaches zero already at $V < 10^2$. This clearly indicates the presence of very large traps at $\phi_m = \phi_m^*$, which is the hallmark feature of a percolation transition (cf. Sec. 1.3.3.3).

3.3.4 Moments of the trap-size distributions

In this section, we finally turn to the quantities that—disregarding finite size-scaling techniques—allow for the most accurate [248] determination of the percolation transition in the accessible volume: the moments of the trap-size distribution. Specifically, we consider the first moment, $\hat{V}_1(\phi_m; N_m)$, and the second moment, $\hat{V}_2(\phi_m; N_m)$, as defined in Eqs. (2.100) and (2.101), respectively. Investigating these quantities constitutes a natural extension of Sec. 3.3.3 in that both \hat{V}_1 and \hat{V}_2 correspond to the (normalized) arithmetic mean of one of the distributions discussed in that section.

Panel (a) of Fig. 3.27 illustrates the first moment, $\hat{V}_1(\phi_m; N_m)$, for systems of four different sizes (lines of different color). As the figure shows, \hat{V}_1 varies only weakly with N_m and exhibits a maximum at $\phi_m \simeq 0.2525$. Unfortunately, this maximum is *not* centered at the percolation transition, as is evidenced by the much more accurate value of ϕ_m^* determined in panel (b) (dashed vertical line; see below). To rationalize these findings, consider the substantial similarity between \hat{V}_1 in Fig. 3.27(a) and P_{trap} in Fig. 3.25(b). This similarity reflects the fact that both of the latter quantities involve the combined accessible volume of all traps. This volume is merely normalized

^{3.17}<http://en.wikipedia.org/wiki/Fractal>

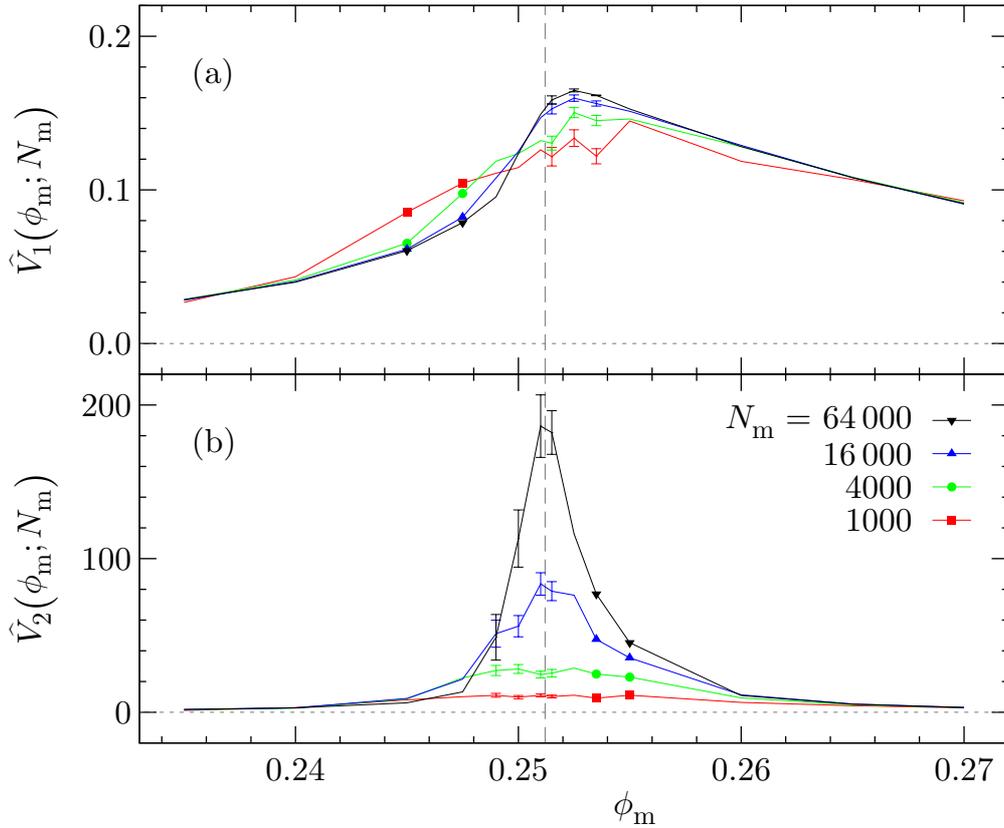


Figure 3.27: Moments of the trap-size distribution. Trap-size distribution moments as functions of the matrix packing fraction, ϕ_m . Panel (a): first moment, $\hat{V}_1(\phi_m; N_m)$; panel (b): second moment, $\hat{V}_2(\phi_m; N_m)$. Line colors and symbol shapes: results for different matrix particle numbers, N_m . Dashed vertical line: percolation threshold inferred from panel (b). Error bars: see Sec. 3.1.

differently, namely by the total accessible volume for P_{trap} as opposed to the number of traps for \hat{V}_1 . The latter implies that the number of traps increases monotonically for $\phi_m > \phi_m^*$, which gives rise to the existence of the maximum of \hat{V}_1 in the first place.

Panel (b) of Fig. 3.27, at last, presents the second moment of the trap-size distribution, $\hat{V}_2(\phi_m; N_m)$. The value of this quantity—by its very construction—is dominated by the largest trap in the system. As discussed before at numerous occasions, infinitely-large HS-QA systems are expected to contain an infinitely-large trap at $\phi_m = \phi_m^*$. In PBC-replicated systems, however, the maximum trap size is limited by the system size, meaning that at $\phi_m \simeq \phi_m^*$, one should observe a direct correspondence between the size of the largest trap and the size of the system. The data presented in Fig. 3.27(b) confirm this expectation in that the values of \hat{V}_2 at $\phi_m \lesssim \phi_m^*$ sharply rise as N_m is increased. Since in the present context, the percolating void is disregarded (cf. Sec. 3.3.3), this rise of \hat{V}_2 is succeeded by an equally abrupt decrease at $\phi_m \gtrsim \phi_m^*$. This implies that the percolation threshold is marked by the ϕ_m value at which \hat{V}_2 attains its maximum. Clearly, the latter is most reliably determined from the largest N_m value, yielding the

figure $\phi_m^* \simeq 0.2512$ for this work's most precise estimation of the percolation transition in the accessible volume of monodisperse HS-QA systems.

3.4 Free and trapped particles

As discussed before on numerous occasions, in hard-sphere quenched-annealed (HS-QA) systems, the matrix component divides the particles of the fluid component into the two classes of “free” and “trapped” particles. In Sec. 3.2, we speculated that this distinction gives rise to some of the more unusual features of the investigated quantities. Equipped with the background of Sec. 1.3, the methodology of Secs. 2.4 and 2.5, and the results of Sec. 3.3, this section thoroughly reiterates the investigation of Sec. 3.2 with respect to this aspect. Among the considered quantities are the mean squared displacement, the self intermediate scattering function, and the self part of the Van Hove function, i.e., the entire set of single-particle observables investigated in Sec. 3.2.

As in the previous sections, let us start out with a brief visual recollection of the subject matter. Figure 3.28 visualizes the geometric aspects of the issue at hand using the same HS-QA system realization as in Figs. 3.3(b), 3.22, and 3.23. Being located at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$, this system in particular features the traits of a percolation transition in the accessible volume (cf. Sec. 3.3). Panel (a) of the figure illustrates how the matrix particles (dark gray spheres) determine the shape of the voids, the latter being displayed exactly as in Fig. 3.23(a), with the green and the red volumes representing the percolating void and the traps, respectively. In panel (b), the matrix particles are removed in favor of the fluid particles (white spheres), with the co-displayed voids emphasizing the allocation of the fluid particles within the voids. Panel (c) differs from panel (b) merely in that the trapped particles are highlighted in blue color while the white guise of the free particles is retained. In panel (d), finally, the voids make room for the matrix particles, illustrating that from a visual standpoint the distinction between trapped and free particles appears to be virtually arbitrary.

3.4.1 Mean squared displacement

In this section, we extend the discussion of the mean squared displacement (MSD) as carried out previously in Sec. 3.2.4. Specifically, in line with the outset of Sec. 3.4, we investigate the MSD separately for the free and the trapped particles in HS-QA systems, referring to the respective quantities by the symbols $\delta r_{\text{free}}^2(t)$ and $\delta r_{\text{trap}}^2(t)$. As before, we highlight details of the MSD by furthermore considering its logarithmic derivative, which for the two particle classes we shall denote by $z_{\text{free}}(t)$ and $z_{\text{trap}}(t)$, respectively. As in the case of the full fluid, the definitions of these quantities are provided by Eqs. (2.66) and (2.67), the only modification over their previous use being the involved sums, which are now limited to either the free or the trapped particles. As in Sec. 3.2.4, we consider the MSD and its derivative for state points along certain

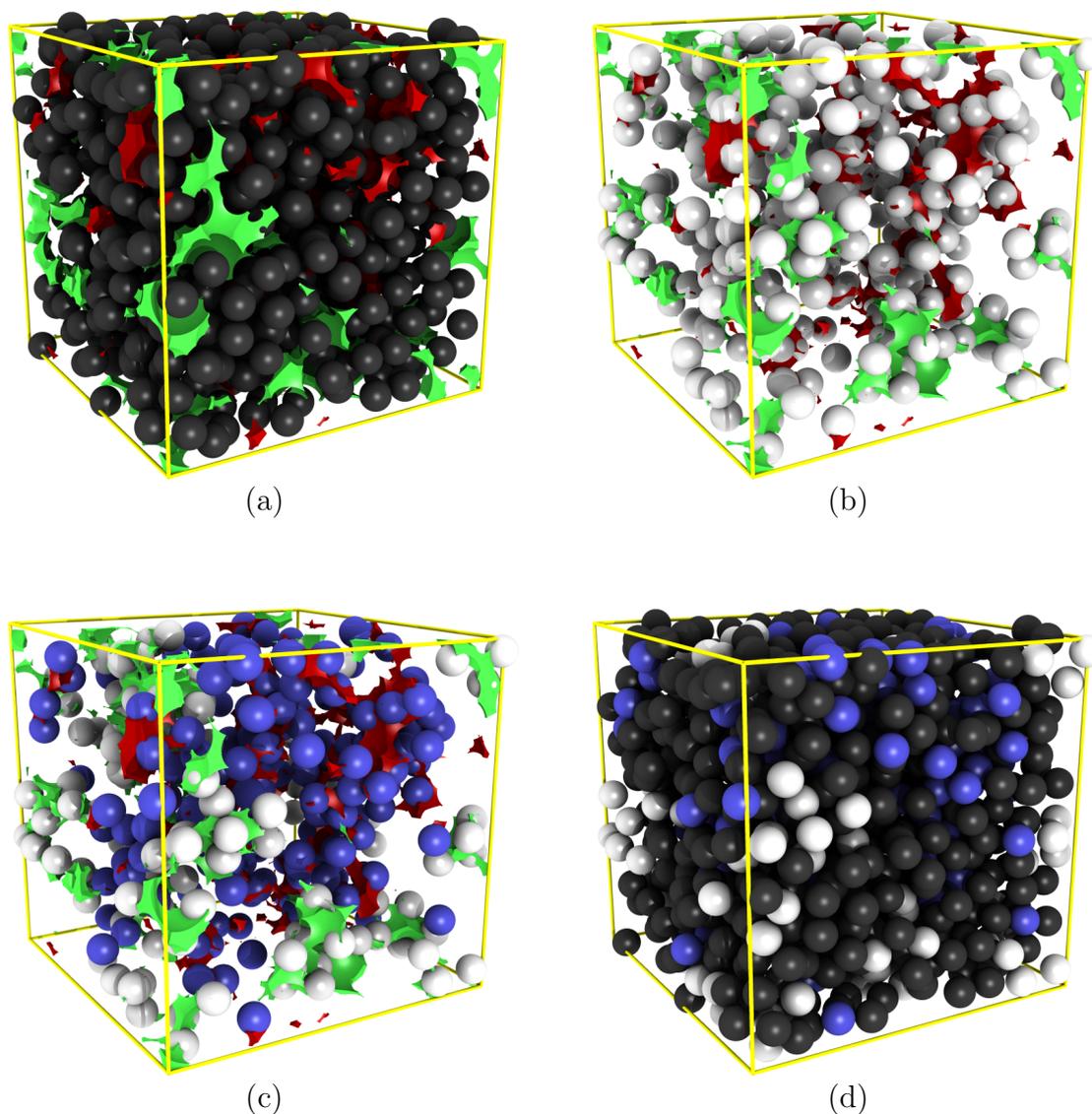


Figure 3.28: Free and trapped fluid particles in HS-QA systems. Illustration of the distinction between free and trapped particles. Same HS-QA system realization as in Figs. 3.3(b), 3.22, and 3.23, located at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$. Panel (a): percolating void (green volume) and traps (red volumes) superimposed onto matrix particles (dark gray spheres). Panel (b): fluid particles (white spheres) superimposed onto voids as in panel (a). Panel (c): same as panel (b), except for the blue color highlighting the trapped particles. Panel (d): matrix particles, free particles, and trapped particles. Yellow lines: simulation cell replicated by periodic boundary conditions (cf. Sec. 2.2.2.1).

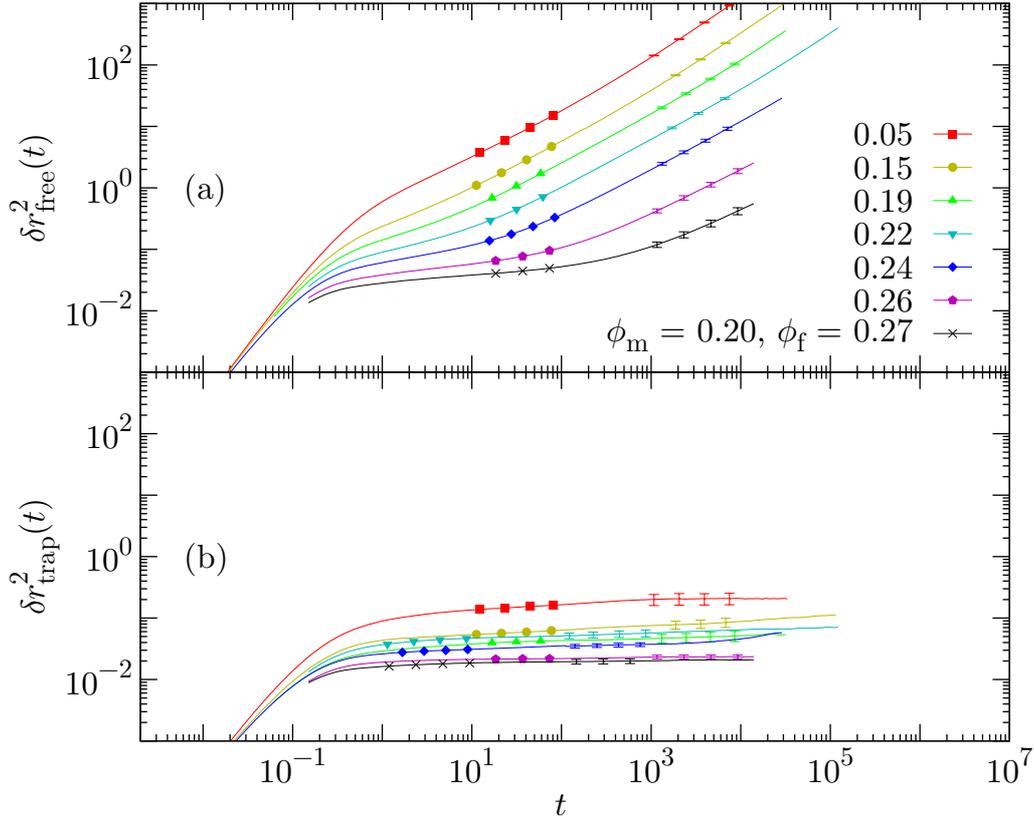


Figure 3.29: Mean squared displacement: free vs. trapped, path II. Mean squared displacement for the particles that are free, $\delta r_{\text{free}}^2(t)$, and those that are trapped, $\delta r_{\text{trap}}^2(t)$, at state points along path II (constant $\phi_m = 0.2$). Error bars: see Sec. 3.1.

paths in the parameter space (cf. Fig. 3.2), with our focus, however, being limited now to paths II and III. The reason for this choice lies in the fraction of trapped particles [cf. Fig. 3.25(b)], which exceeds marginal values only in the vicinity of the percolation transition of the accessible volume at $\phi_m^* \simeq 0.2512$, that is, well beyond the low ϕ_m value of path I.

Figures 3.29 and 3.30 show the MSD and its logarithmic derivative for the same systems as Fig. 3.7, i.e., for state points along path II where $\phi_m = 0.2$. As in the whole set of figures that remain to be discussed, the figures illustrate data pertinent to the free particles in panel (a) and to the trapped particles in panel (b). A comparison with Fig. 3.7 evidences that $\delta r_{\text{free}}^2(t)$ and $\delta r^2(t)$ —the MSD of the full fluid component—do not appreciably differ at any time scale. (Briefly recounting Sec. 3.2.4, depending on the ϕ_f value, the initial ballistic behavior may be succeeded by a caging regime and a subdiffusive regime until eventually ordinary diffusive behavior prevails in all cases.) The agreement is substantiated by $z_{\text{free}}(t)$ and $z(t)$, which likewise are in unison despite their considerable susceptibility to dynamic changes. Above all, this implies that nearly all features of the full-fluid MSD originate in the free particles, notably including the subdiffusive behavior at intermediate times where both z and z_{free} attain

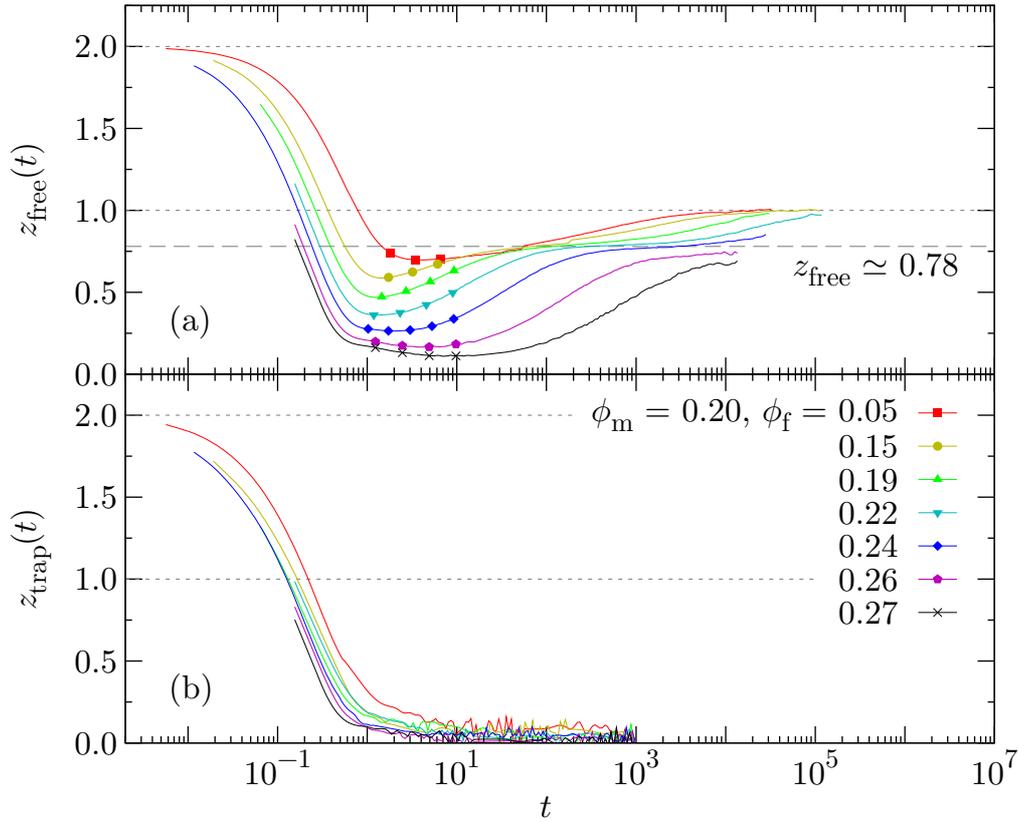


Figure 3.30: MSD's logarithmic derivative: free vs. trapped, path II. Logarithmic derivative of the mean squared displacement for the particles that are free, $z_{\text{free}}(t)$, and those that are trapped, $z_{\text{trap}}(t)$, at state points along path II (constant $\phi_m = 0.2$). Error bars: see Sec. 3.1.

a value of approximately 0.78.

The dominance of the free particles is not surprising recalling Fig. 3.25(b), where it is shown that the probability for a fluid particle to be trapped remains well below 10% at $\phi_m = 0.2$. However, of comparable importance is the fact that the propagation of the trapped particles essentially comes to a halt prior even to the caging regime of the free particles. This quick cease of motion highlights the marginal extent of a typical trap at ϕ_m values even as large as 0.2. Notably, while z_{trap} at $t = 10^0$ is close to zero irrespective of ϕ_f , the long-time value (LTV) of $\delta r_{\text{trap}}^2(t)$ does vary with the fluid packing fraction. This reflects the effect of “crowding” within traps, i.e., the fact that from the perspective of a trapped particle, the size of the surrounding trap is effectively reduced if that trap is populated by additional fluid particles and if the particles within the trap cannot rearrange.

In Figs. 3.31 and 3.32, the free- and trapped-particle MSDs and their logarithmic derivatives are illustrated for the same systems as in Fig. 3.8, i.e., for state points residing along path III where $\phi_f = 0.1$. As in the case of path II, the figures indicate a considerable similarity between the MSD of the free particles and that of the full fluid

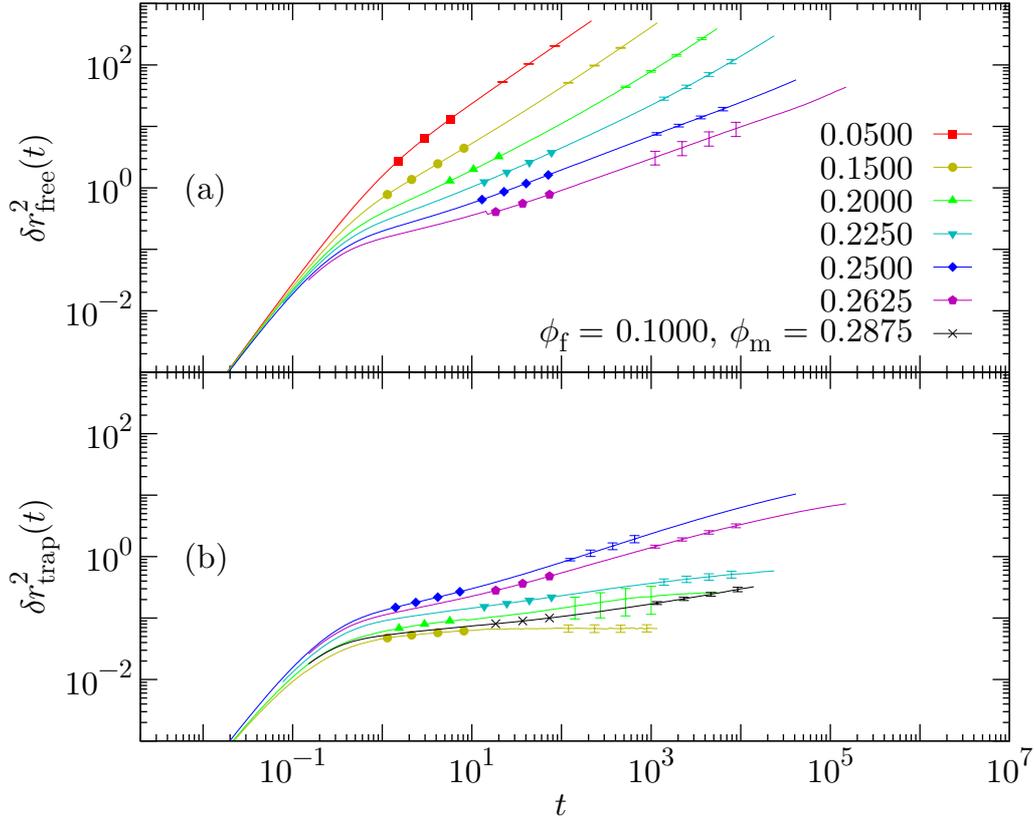


Figure 3.31: Mean squared displacement: free vs. trapped, path III. Mean squared displacement for the particles that are free, $\delta r_{\text{free}}^2(t)$, and those that are trapped, $\delta r_{\text{trap}}^2(t)$, at state points along path III (constant $\phi_f = 0.1$). Error bars: see Sec. 3.1.

component shown in Fig. 3.8. Upon closer inspection, however, the cases with a ϕ_m value of 0.25 or more^{3.18} also reveal significant differences between $\delta r_{\text{free}}^2(t)$ and $\delta r^2(t)$ and, even more prominent, between $z_{\text{free}}(t)$ and $z(t)$. Specifically, the free-particle MSD exhibits a clear tendency to recover diffusive behavior ($z_{\text{free}} = 1$) in the long-time limit for all depicted ϕ_m values, as opposed to the full-fluid MSD, where this is the case only for $\phi_m < 0.25$ (cf. Sec. 3.2.4). In the vicinity of the percolation transition in the accessible volume at $\phi_m^* \simeq 0.2512$ (cf. Sec. 3.3.4), the free particles clearly undergo subdiffusive motion, meaning that the subdiffusion observed for the full fluid component is not (or at least not only) a result of trapping. While the values of z_{free} in the respective subdiffusive regimes slightly exceed the corresponding values of z , both quantities assume a value close to 0.5 whenever ϕ_m is in the vicinity of ϕ_m^* .

^{3.18}Figures 3.31 and 3.32 include free-particle data for systems at $\phi_m > \phi_m^* \simeq 0.2512$, i.e., for systems beyond the percolation threshold in the accessible volume. This reflects the finite size of the systems, which is owed to their realization as computer simulations. The finite size entails a sub-unity probability P_{trap} for a particle to be trapped even if $\phi_m > \phi_m^*$ (see Sec. 3.3.2). However, P_{trap} rapidly approaches unity for $\phi_m > \phi_m^*$ [cf. Fig. 3.25(b)], meaning that the effort for determining the quantities of interest to satisfactory accuracy increases steeply with ϕ_m .

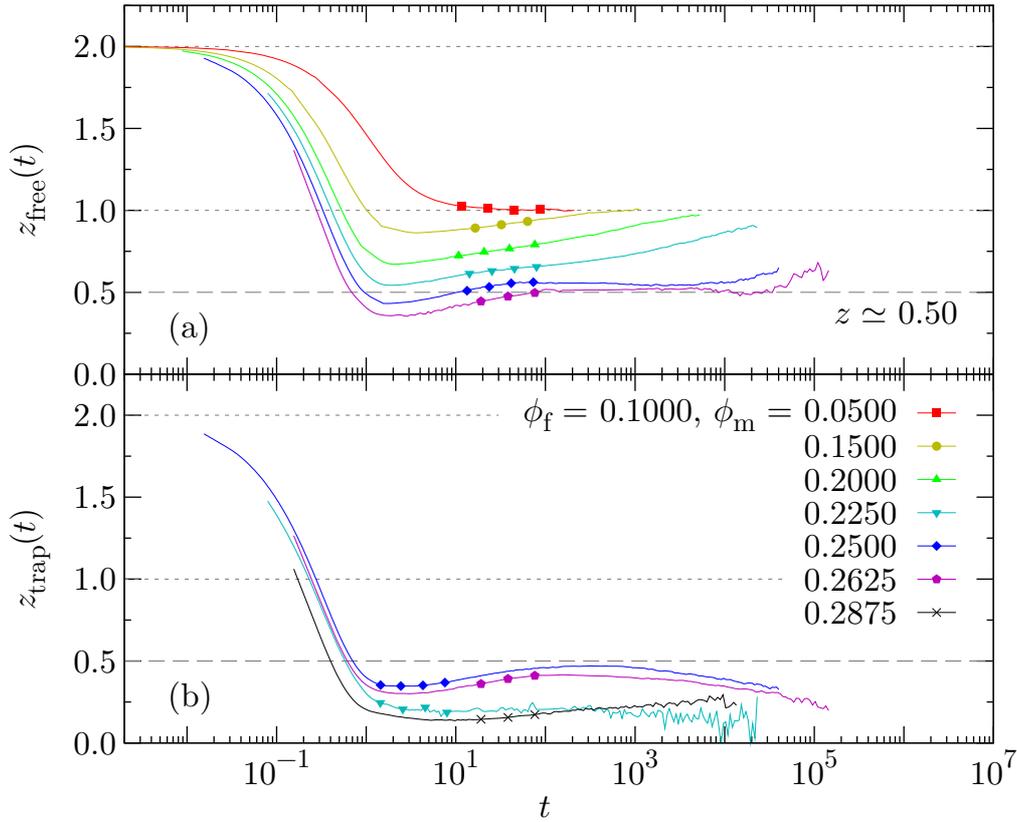


Figure 3.32: MSD's logarithmic derivative: free vs. trapped, path III. Logarithmic derivative of the mean squared displacement for the particles that are free, $z_{\text{free}}(t)$, and those that are trapped, $z_{\text{trap}}(t)$, at state points along path III (constant $\phi_f = 0.1$). Error bars: see Sec. 3.1.

These findings are substantiated by the essentially-coinciding time regimes over which subdiffusion is observed.

The differences between the free-particle and the full-fluid MSDs can be accounted for by the trapped particles. The latter are, by definition, more restricted in their movement than the free particles, which is reflected in Figs. 3.31 and 3.32 in that $\delta r_{\text{trap}}^2(t)$ always tends to saturate whereas $\delta r_{\text{free}}^2(t)$ generally increases beyond any limit. According to the figures, the full-fluid MSD (Fig. 3.8) at $\phi_m \gtrsim \phi_m^*$ is more similar to the trapped- than to the free-particle MSD. This directly results from the probability P_{trap} for a particle to be trapped, which in the immediate vicinity of ϕ_m^* well exceeds 50% even in the investigated 1000-particle systems [cf. Fig. 3.25(b)]^{3.19}. Further, consider

^{3.19}For values of ϕ_m significantly smaller than ϕ_m^* , it is challenging to determine the trapped-particle MSD to an adequate accuracy. This originates in the fact that even for ϕ_m values as large as $0.95 \phi_m^*$, the probability P_{trap} for a particle to be trapped remains as low as 10% (see Sec. 3.3.2). While this still allows to consider $\delta r_{\text{trap}}^2(t)$ for $\phi_m \geq 0.15$, the amplified fluctuations in the logarithmic derivative limit a sensible study of $z_{\text{trap}}(t)$ to $\phi_m \geq 0.225$. Notably, however, in contrast to the issue discussed in Footnote 3.18, HS-QA systems at $\phi_m < \phi_m^*$ always contain trapped particles, meaning that to the present issue, the system size is only of minor importance.

that the LTV of $\delta r_{\text{trap}}^2(t)$ is determined largely by the sizes of a system's traps. Following the curves in Fig. 3.31(b) in order of increasing ϕ_m , this means that the traps grow for $\phi_m < \phi_m^* \simeq 0.2512$ while they shrink for $\phi_m > \phi_m^*$. This is in line with Sec. 3.3.4, where by pure geometry we revealed that the second moment of the trap-size distribution, \hat{V}_2 , attains a maximum at ϕ_m^* . Finally, Figs. 3.31 and 3.32 indicate that for $\phi_m \simeq \phi_m^*$ and $t \lesssim 10^3$, the MSDs of the trapped and of the free particles are comparable. This reflects not only the fact that traps in HS-QA systems are large in the vicinity of the percolation transition (see above), but also that the traps and the percolating void in this ϕ_m regime are structurally similar (as visually apparent from Fig. 3.23). The latter in particular rationalizes the essentially-coinciding values of z_{trap} , z_{free} , and z in the regimes of ϕ_m and t in question.

3.4.2 Single-particle intermediate scattering function

In this section, we take on the task of investigating the single-particle (“self”) intermediate scattering function (ISF) separately for the free and the trapped particles in HS-QA systems. We shall refer to the respective quantities by the symbols $F_s^{\text{free}}(k, t)$ and $F_s^{\text{trap}}(k, t)$, and inspect them for state points along a subset of the parameter-space paths considered for the self ISF of the full fluid component (see Sec. 3.2.5). The subset merely includes paths II and III (cf. Fig. 3.2) since—as stated earlier in Sec. 3.4.1—at the ϕ_m value of path I, the trapped particles are vastly outnumbered by the free particles (cf. Sec. 3.3.2) and are therefore essentially irrelevant. In analogy to the MSD (cf. Sec. 3.4.1), both the free- and trapped-particle variants of the self ISF are defined by Eq. (2.81), the difference to the full-fluid correlator being the sum within the equation, which now involves only the free or only the trapped particles. Notably, we will not examine the effects of trapping onto collective ISFs such as the connected ISF or the total (conventional) ISF (see Secs. 2.5.3.2 and 2.5.3.3) since those quantities were not found to exhibit noteworthy features in the vicinity of the percolation transition of the accessible volume at $\phi_m^* \simeq 0.2512$ (cf. Secs. 3.2.5 and 3.3.4).

Figure 3.33 illustrates the self ISF for state points along path II (constant $\phi_m = 0.2$) and for the wave vector $k = 7$, the latter residing close to the first maximum of the static structure factor (see Sec. 3.2.3). Panel (a) displays $F_s^{\text{free}}(k, t)$, which exhibits essentially the same functional features as the full-fluid self ISF as shown in Fig. 3.10. This in particular includes the intermediate-time plateau for large ϕ_f , the existence of which in the full-fluid correlator can therefore be attributed to the free particles. Notwithstanding, consider the state points at which the pertinent systems are well equilibrated according to Fig. 3.33, i.e., the cases in which $F_s^{\text{free}}(k, t)$ at the largest depicted time likely resembles $\lim_{t \rightarrow \infty} F_s^{\text{free}}(k, t)$. In those cases, $F_s(k, t)$ and $F_s^{\text{free}}(k, t)$ differ noticeably in that the latter invariably relaxes to zero in the long-time limit. This finding substantiates the assumption expressed in Sec. 3.2.5 that nonzero LTVs of the self ISF are engendered by trapped particles.

The latter warrants a closer look at the trapped-particle self ISF, shown in panel (b) of Fig. 3.33 for the same state points and the same wave vector as $F_s^{\text{free}}(k, t)$ in

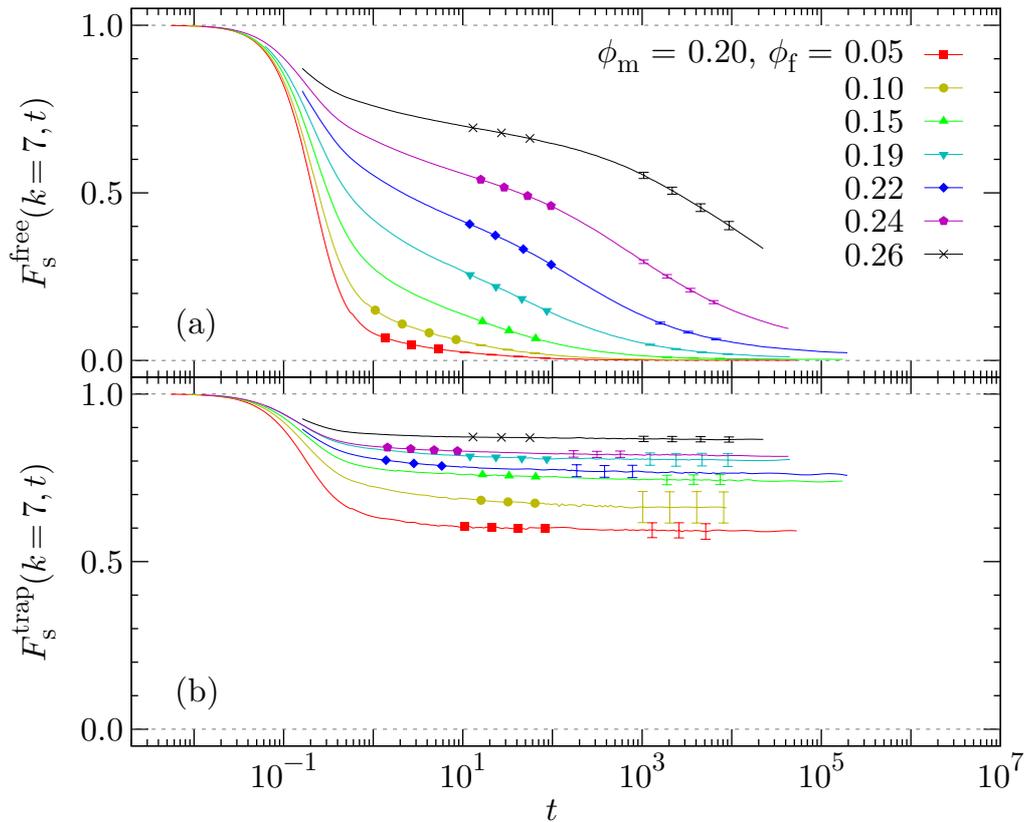


Figure 3.33: Single-particle ISF: free vs. trapped, path II. Self intermediate scattering function for the particles that are free, $F_s^{\text{free}}(k, t)$, and those that are trapped, $F_s^{\text{trap}}(k, t)$, at state points along path II (constant $\phi_m = 0.2$) and for the wave vector $k = 7$. Error bars: see Sec. 3.1.

panel (a). In the long-time limit, the trapped-particle correlator relaxes only partially in all depicted cases, which is not surprising considering the definition of trapping. Of a more intricate origin is the monotonic increase of $\lim_{t \rightarrow \infty} F_s^{\text{trap}}(k, t)$ with ϕ_f , which arises from the fact that—as mentioned in Sec. 3.4.1—particles within densely populated traps may be incapable of switching places. In combination with the increasing trapping probability P_{trap} (cf. Sec. 3.3.2), this well accounts for the LTVs of $F_s(k, t)$ as observed in Fig. 3.10. On a different note, the functional form of $F_s^{\text{trap}}(k, t)$ differs significantly from that of $F_s^{\text{free}}(k, t)$ in that the former does not exhibit signs of an intermediate-time plateau. This substantiates the interpretation that the plateau is present in $F_s(k, t)$ only due to the corresponding feature of the free-particle self ISF (see above). Therefore, the analysis at hand can contribute only marginally to the clarification of this effect.

In complement to the investigation of $F_s^{\text{free}}(k, t)$ and $F_s^{\text{trap}}(k, t)$ for path II, let us examine the same quantities for state points along path III (constant $\phi_f = 0.1$). The pertinent data are illustrated in Fig. 3.34, the considered state points coinciding with those for $F_s(k, t)$ in Fig. 3.8 and the wave vector being the usual $k = 7$ (see above). As evidenced by the two figures just mentioned, the relaxation patterns of the

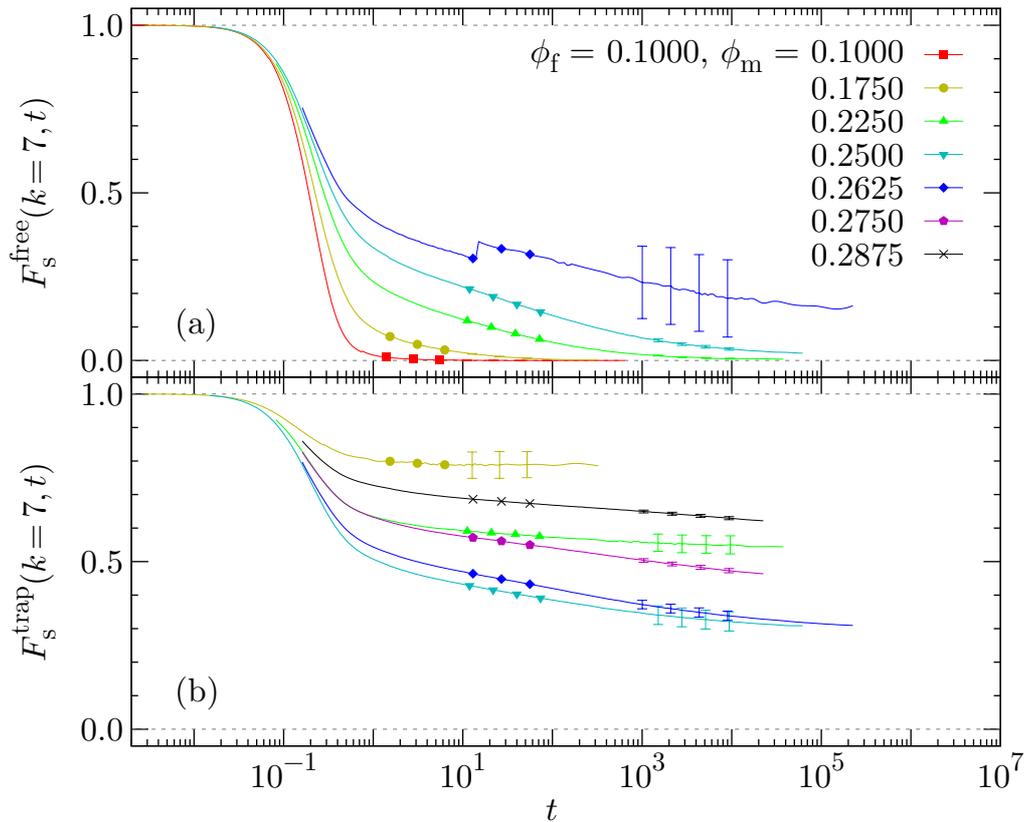


Figure 3.34: Single-particle ISF: free vs. trapped, path III. Self intermediate scattering function for the particles that are free, $F_s^{\text{free}}(k, t)$, and those that are trapped, $F_s^{\text{trap}}(k, t)$, at state points along path III (constant $\phi_f = 0.1$) and for the wave vector $k = 7$. Error bars: see Sec. 3.1.

free-particle and the full-fluid self ISFs are substantially similar. However, consider that $F_s^{\text{free}}(k, t)$ and $F_s(k, t)$ in the figures consistently approximate their LTVs at the largest depicted times^{3.20}. Of the LTVs, only those of $F_s(k, t)$ assume nonzero values while those of $F_s^{\text{free}}(k, t)$ essentially vanish. This observation is comparable to the one made for the free-particle and the full-fluid self ISF along path II (see above), where we concluded that the nonzero LTVs of $F_s(k, t)$ originate solely in the presence of trapped particles.

This again suggests a closer inspection of $F_s^{\text{trap}}(k, t)$, the latter being displayed in panel (b) of Fig. 3.34 for the same state points and the same wave vector as $F_s^{\text{free}}(k, t)$ in panel (a). As the panel illustrates, the LTV of the trapped-particle self ISF significantly exceeds zero in all depicted cases. This is reminiscent of the trapped-particle self ISF for path II (see above). Interestingly, however, an increase of ϕ_m entails a *nonmonotonic* variation of $\lim_{t \rightarrow \infty} F_s^{\text{trap}}(k, t)$ in that the latter decreases for $\phi_m \lesssim \phi_m^*$

^{3.20}As Fig. 3.34 indicates, $F_s^{\text{free}}(k, t)$ for the case of $\phi_m = 0.2625$ is associated with sizable uncertainties. This circumstance is owed to the issue discussed in Footnote 3.18. In consequence, the pertinent long-time limit of $F_s^{\text{free}}(k, t)$ —which appears to exceed zero—is likely unreliable.

while it increases for $\phi_m \gtrsim \phi_m^*$. In other words, the relaxation of the trapped-particle self ISF is most pronounced in the vicinity of the percolation transition. This feature is analogous to the nonmonotonically-varying maximum of both the trapped-particle MSD and its logarithmic derivative for path III (see Fig. 3.8), which we attributed to the fact that the second moment of the trap-size distribution, \hat{V}_2 , attains a maximum at $\phi_m = \phi_m^*$ (cf. Secs. 3.4.1 and 3.3.4). As in the latter case, and as in the case of $F_s(k, t)$ for path III, the long-time values of the corresponding trapped-particle quantities as well as the trapping probability, P_{trap} (cf. Sec. 3.3.2), soundly rationalize the observed LTVs of the full-fluid quantities.

Finally, with the free-particle self ISF available, we are now in the position to assess the hypothesis formulated in Sec. 3.2.8 that the modified self ISF, $\hat{F}_s(k, t)$, essentially equates to $F_s^{\text{free}}(k, t)$. Verifying this hypothesis is of the utmost importance to comprehending the dynamics of HS-QA systems in general and to interpreting the differences between the single-particle and connected ISFs in particular. In fact, it was this hypothesis that incited our efforts to identify the free and the trapped particles in HS-QA systems in the first place. Briefly recalling the story behind $\hat{F}_s(k, t)$, this quantity was obtained in Sec. 3.2.8 by removing the LTV from the full-fluid self ISF, $F_s(k, t)$, under the assumption that the presence of the LTV is due purely to trapped particles.

The means to elucidate this hypothesis is Fig. 3.35, in which we juxtapose the relevant ISFs for state points along path III (constant $\phi_f = 0.1$). As in all depictions of ISFs in this work, the quantities are displayed for the wave vector $k = 7$. Comparing panels (a) and (b), it is clear that $\hat{F}_s(k, t)$ indeed closely resembles $F_s^{\text{free}}(k, t)$. The residual differences originate in at least two phenomena. Firstly, the structural relaxation of the trapped particles deviates from that of the free particles, the reason lying in the different geometries of the traps and of the percolating void. This is noticeable in the functional form of $\hat{F}_s(k, t)$ whenever the probability for a particle to be trapped significantly exceeds zero. Secondly, a close inspection of panel (b) reveals that at elevated ϕ_m values, the LTV of $F_s^{\text{free}}(k, t)$ slightly exceeds zero. This is likely owed to the fluid–fluid trapping effects detailed in Sec. 3.4.4.

In the light of these observations, the interpretations of Sec. 3.2.8 are clearly vindicated. Above all, this includes the relaxation times inferred by the procedure of Sec. 2.5.3.5, on the basis of which we concluded that the kinetic diagram based on $\hat{F}_s(k, t)$ (Fig. 3.18) and the kinetic diagram compiled from $F_c(k, t)$ (Fig. 3.13) exhibit similar traits. The residual differences are limited to an overall shift of the relaxation times, which is evident when comparing $F_s^{\text{free}}(k, t)$ to the connected ISF, $F_c(k, t)$, depicted in panel (c) of Fig. 3.35.

3.4.3 Self part of the Van Hove function

As the final step of our exploration of free- and trapped-particle observable variants, we consider the self part of the Van Hove function (the “self VHF”). In continuation of our

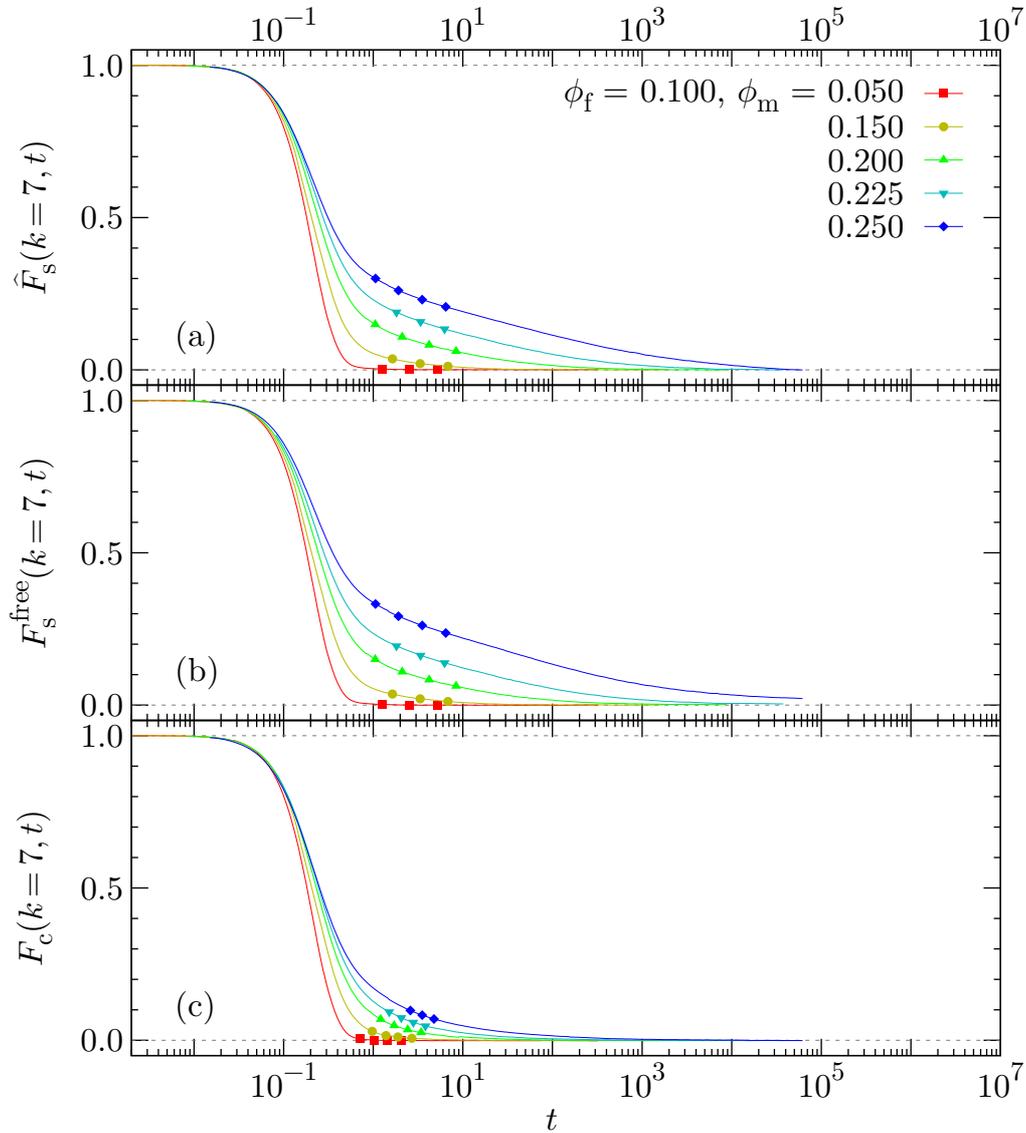


Figure 3.35: Comparison of ISFs at path III. Juxtaposition of various ISFs at path III (constant $\phi_f = 0.1$) for the wave vector $k = 7$. Panel (a): modified full-fluid self ISF, $\hat{F}_s(k, t)$, as in Fig. 3.17. Panel (b): free-particle self ISF, $F_s^{\text{free}}(k, t)$, as in Fig. 3.34. Panel (c): connected ISF, $F_c(k, t)$, as in Fig. 3.11(a). Error bars: see Sec. 3.1.

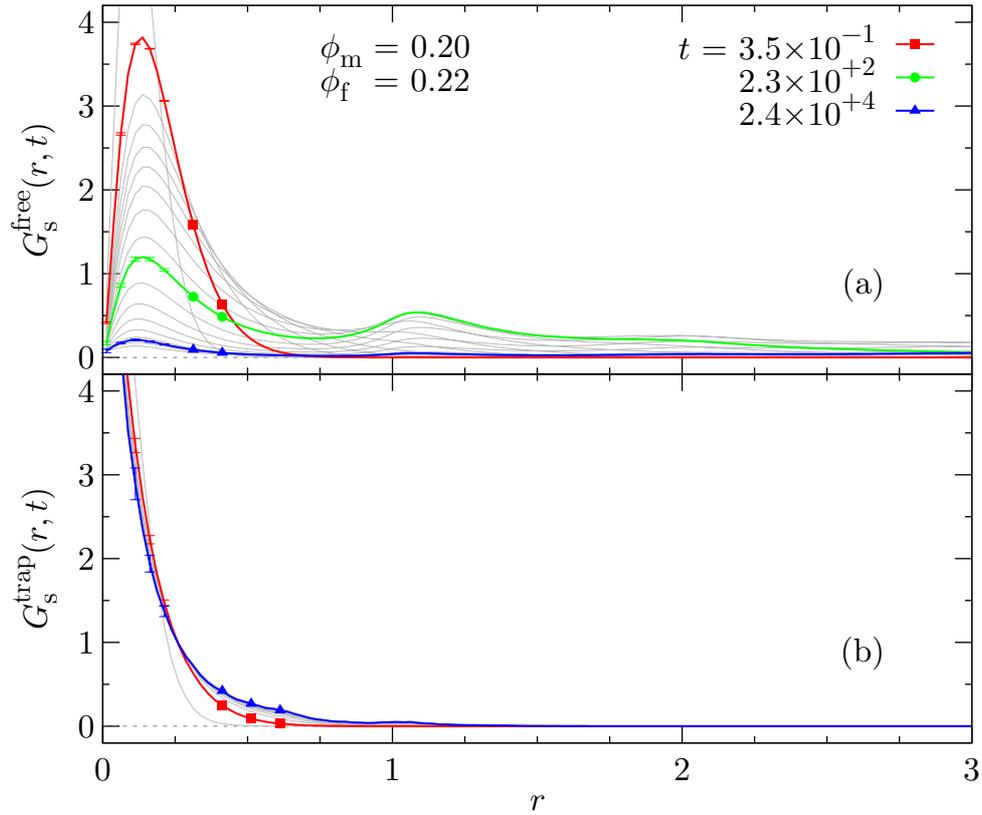


Figure 3.36: Self Van Hove function: free vs. trapped, point B. Self VHF of the free particles, $G_s^{\text{free}}(r, t)$, and of the trapped particles, $G_s^{\text{trap}}(r, t)$, at point B where $(\phi_m, \phi_f) = (0.20, 0.22)$ and for times differing by a factor of $\sim \sqrt[6]{10}$. Thick colored lines: selected times roughly coinciding across Figs. 3.36 and 3.37. Error bars: see Sec. 3.1.

previous notation, we shall refer to these quantities by the abbreviations $G_s^{\text{free}}(r, t)$ and $G_s^{\text{trap}}(r, t)$, and in analogy to the discussions in Secs. 3.4.1 and 3.4.2, both variants are defined by a single equation, Eq. (2.91), by limiting the sum therein to the respective particles. The visual representations of this work consider the self VHF as a function of the distance and account for its dependence upon time by series of curves, thereby necessitating each self-VHF diagram to focus on a single set of external parameters (compare Secs. 2.5.3.6 and 3.2.9). While in Sec. 3.2.9, we examined $G_s(r, t)$ for the four points A through D, the latter having been selected due to their proximity to the MSD and the self-ISF arrest transitions (recall Figs. 3.2 and 3.12), the current investigation will be conducted for the two points B and C only. The reason for this limitation lies in the probability P_{trap} for a particle to be trapped (see Sec. 3.3.2), which is very nearly zero at point A ($\phi_m = 0.1$) and extremely close to unity at point D ($\phi_m = 0.3$) even in the 1000-particle systems considered in our simulations (cf. Secs. 3.4.1 and 3.4.2).

Figure 3.36 illustrates the free- and trapped-particle variants of the self VHF at point B where $(\phi_m, \phi_f) = (0.20, 0.22)$. As a comparison of panel (a) with Fig. 3.20(b)

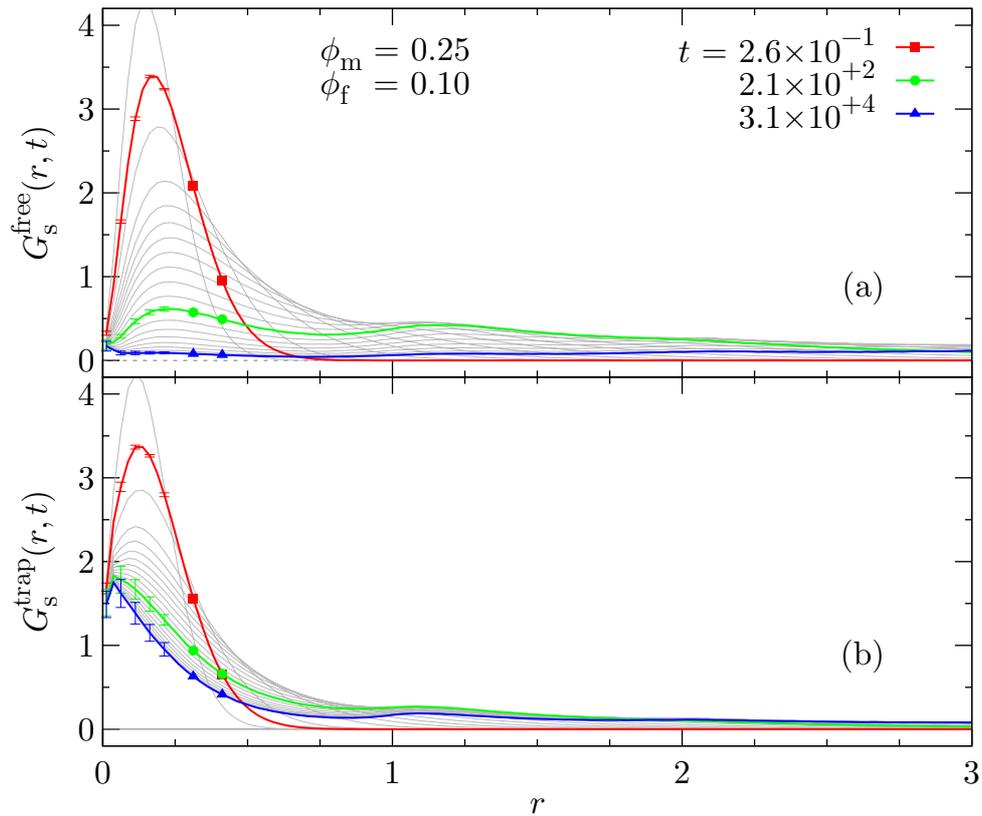


Figure 3.37: Self Van Hove function: free vs. trapped, point C. Self VHF of the free particles, $G_s^{\text{free}}(r, t)$, and of the trapped particles, $G_s^{\text{trap}}(r, t)$, at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$ and for times differing by a factor of $\sim \sqrt[6]{10}$. Thick colored lines: selected times roughly coinciding across Figs. 3.36 and 3.37. Error bars: see Sec. 3.1.

evidences, both the temporal and the spatial features of $G_s^{\text{free}}(r, t)$ and $G_s(r, t)$ are very similar. This includes the essentially-invariant position at $r \simeq 0.15$ of the first maximum and the rapid decrease of the latter, as well as the presence of a second maximum (cf. Sec. 3.2.9), all indicating that the free particles undergo caging and collective rearrangements of their own right. The approach to a Gaussian distribution at long times likewise takes place on a similar time scale. The only significant difference between free-particle and the full-fluid self VHF exists at $r < 0.25$ and at large times, where the former exhibits decidedly smaller values than the latter. This is the case in particular at $r = 0$, where $G_s^{\text{free}}(r, t)$ consistently remains in the immediate vicinity of zero. Nevertheless, even at the largest depicted times, the free-particle self VHF considerably exceeds zero at $r < 0.25$, which probably originates in one or more of the fluid–fluid trapping effects portrayed in Sec. 3.4.4.

Allowing for the respective normalizations, the difference between $G_s^{\text{free}}(r, t)$ and $G_s(r, t)$ is embodied by $G_s^{\text{trap}}(r, t)$ as displayed in panel (b) of Fig. 3.36. While the former two quantities largely coincide (see above), the full-fluid and the trapped-particle

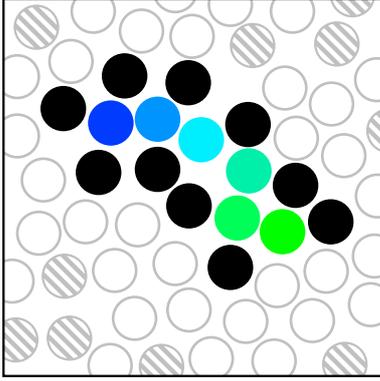


Figure 3.38: Free particles in dead end. Cutaway section of a representative hard-disk QA system, illustrating free particles located in an extended dead-end branch of the accessible space. Solid black disks: matrix particles delineating the dead end. Solid disks of blue to green hue: free particles located within the dead end. Gray-and-white striped disks: other matrix particles; gray circles: other fluid particles. The “blueness” of the highlighted free particles corresponds to their distance to the mouth of the dead end.

self VHF differ markedly in two aspects. Firstly, relative to $G_s(r, t)$, the time required by $G_s^{\text{trap}}(r, t)$ to closely resemble its long-time limit is minimal. Secondly, the functional form of the trapped-particle self VHF is substantially simpler than that of the full-fluid self VHF in that it is limited to a single, narrow maximum centered at $r = 0$. This is in accordance with the trap sizes determined in Secs. 3.3.3 and 3.3.4, with the minute magnitude of the second maximum of $G_s^{\text{trap}}(r, t)$ at $r \simeq 1$ highlighting that only extremely few traps are sufficient in size to allow fluid particles trapped therein to switch places. On a final note, recall that in Sec. 3.2.9, we estimated from the full-fluid self VHF that $P_{\text{trap}} \simeq 8\%$ of the particles are trapped in system realizations at point B. Although the ϕ_m value of point B is not represented in Fig. 3.25, moderate extrapolation suggests that the actual value of P_{trap} is somewhat lower (irrespective of the system size)^{3.21}.

Figure 3.37 portrays $G_s^{\text{free}}(r, t)$ and $G_s^{\text{trap}}(r, t)$ for point C, the latter being located at $(\phi_m, \phi_f) = (0.25, 0.10)$, i.e., in proximity of the percolation transition of the accessible volume at $\phi_m^* \simeq 0.2515$ (cf. Sec. 3.3.4). Recalling Fig. 3.21(a), the full-fluid correlator exhibits features reminiscent of both the free- and the trapped-particle self VHF depicted in panels (a) and (b) of Fig. 3.37, respectively. Specifically, both $G_s^{\text{free}}(r, t)$ and $G_s^{\text{trap}}(r, t)$ feature a pronounced and quickly-decreasing maximum at $r \simeq 0.15$, as well as a second maximum at $r \simeq 1.1$, which confirms the interpretation of Sec. 3.2.9 that the free and the trapped particles undergo caging and collective rearrangements independently. Further, both self-VHF variants at large distances remain significantly above zero at large times, and in both cases, the approach to the infinite-time limit requires considerable time. In the case of $G_s^{\text{trap}}(r, t)$, this starkly contrasts with the case of point B. All of these findings are rooted in two properties of HS-QA systems at the ϕ_m value in question: (i) in these systems, the probability P_{trap} for a particle to

^{3.21}In Sec. 3.2.9, the probability P_{trap} for a particle to be trapped was estimated to be $\int_0^{\tilde{r}} G_s(r, \tilde{t}) dr$ based on the assumption that at the time \tilde{t} most of the free particles have moved farther than the distance \tilde{r} . This gives rise to two sources of uncertainty. On the one hand, if \tilde{t} is chosen too small or \tilde{r} too large, then P_{trap} is overestimated due to free particles contributing to the integral. On the other hand, if \tilde{r} is chosen too small, then P_{trap} is underestimated as some trapped particles may not be accounted for by the integral. Given the relatively-small value of \tilde{r} used in the estimations of Sec. 3.2.9, this suggests that in the case of point B, the considered time \tilde{t} was insufficient.

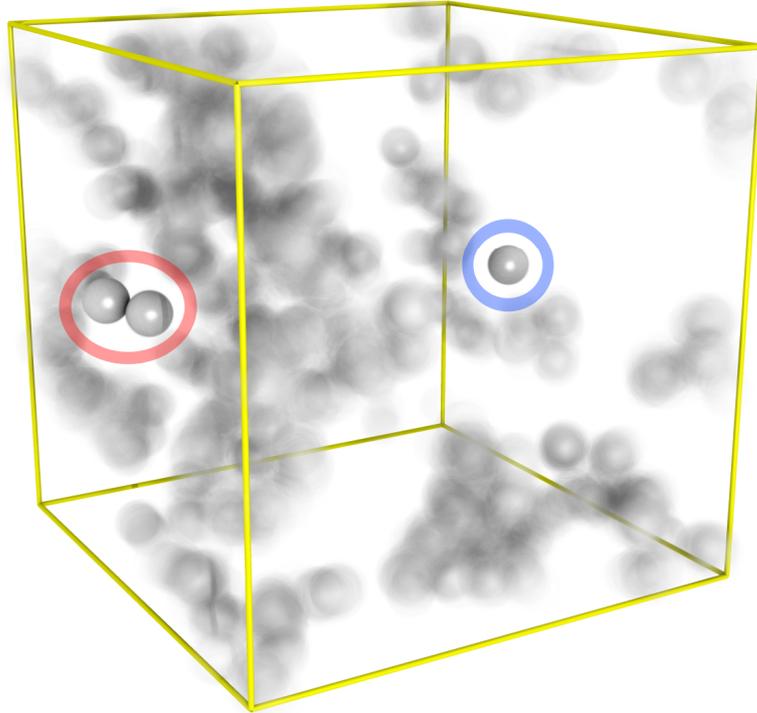


Figure 3.39: Free-particle motion. Visual appearance of the free particles in a representative HS-QA system, visually averaged over one hundred different points in time spaced evenly over 300 000 time units. Depicted is the same system as in Figs. 3.3(a), 3.22, 3.23, and 3.28 located at point C, i.e., at $(\phi_m, \phi_f) = (0.25, 0.10)$. Circumscribed by a light red ellipse are two interlocking free particles; encircled in light blue is a free particle blocked by a trapped particle.

be trapped is about 60% irrespective of the system size (see Fig. 3.25), and (ii) traps in these systems may be very large (cf. Secs. 3.3.3 and 3.3.4).

However, Fig. 3.37 also reveals considerable differences between the free- and trapped-particle self VHF at point C. These differences are most pronounced at large times, where $G_s^{\text{free}}(r, t)$ relaxes to values close to zero at all distances whereas $G_s^{\text{trap}}(r, t)$ remains well above zero in particular at $r < 0.5$. Naturally, as in the case of point B, this reflects the finite distance that trapped particles may travel within their surrounding traps. Yet, at $r > 0.5$, the LTV of the trapped-particle self VHF likewise remains noticeably larger than zero, which is a consequence of the sizes of the traps in the respective systems (see above). Notably, the LTV of $G_s^{\text{free}}(r, t)$ at $r = 0$ slightly exceeds zero, which, as for point B, probably results from the fluid–fluid trapping effects discussed in Sec. 3.4.4. Finally, consider the fraction of trapped particles in system realizations at point C, which in Sec. 3.2.9 was estimated to be 58%. Contrary to the assessment articulated in that section, and contrary to the relatively-poor P_{trap} estimation for point B (see above), this value is in fact in respectable agreement with the value $P_{\text{trap}} \simeq 60\%$ mentioned above.

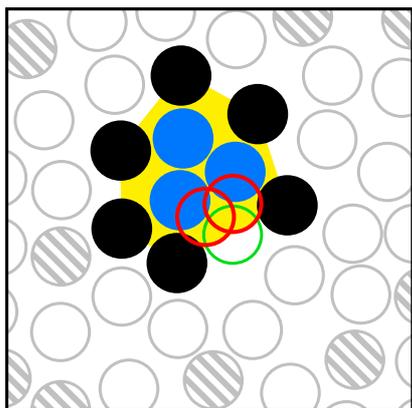


Figure 3.40: Interlocking free particles. Cutaway part of a model hard-disk QA system, illustrating free particles that mutually block their exiting from a pocket of the percolating void. Solid black disks: matrix particles delimiting the pocket. Yellow area: visual aid to identify the pocket. Solid blue disks: free particles within the pocket. Green circle: indicator that the pocket’s mouth is crossable by a fluid particle. Red circles: indicators that the free particles within the pocket block each other’s way to the pocket’s mouth. Gray-and-white striped disks: other matrix particles; gray circles: other fluid particles.

3.4.4 Fluid–fluid trapping effects

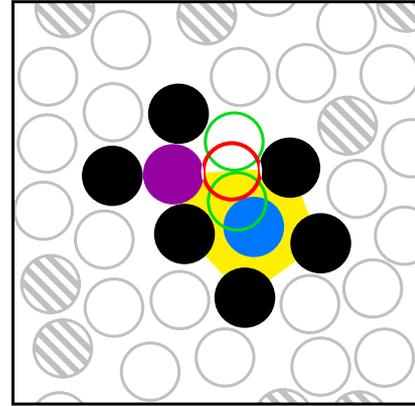
In the previous sections, we found that within the pertinent margins of uncertainty, the free-particle variants of the observables of interest largely exhibit the expected features. On few occasions, however, the respective uncertainties were too small to rationalize the discrepancies between observation and expectation. Specifically, in Sec. 3.4.2, we revealed that the LTV of the free-particle self ISF, $F_s^{\text{free}}(k, t)$, slightly exceeds zero and that it does so particularly for ϕ_m values in the vicinity of the percolation threshold of the accessible volume at $\phi_m^* \simeq 0.2515$ (cf. Sec. 3.3.4). Similarly, in Sec. 3.4.3 we noted that the LTV of the free-particle self VHF, $G_s^{\text{free}}(r, t)$, remains somewhat above zero at $r = 0$ in the regime of $\phi_m \simeq \phi_m^*$. These observations deviate from the expectation that both $F_s^{\text{free}}(k, t)$ and $G_s^{\text{free}}(k, t)$ generally relax entirely in this parameter-space region. The purpose of the present section is to outline three effects that contribute to these deviations.

The first one among the effects in question is illustrated in Fig. 3.38. The figure shows a cutaway section of a representative hard-disk QA system^{3.22} in which the matrix particles associated with the effect are represented by solid black disks and the relevant free particles are depicted as solid disks of blue to green hue. Included as gray-and-white striped disks and as gray circles are other matrix and other fluid particles, respectively. The black matrix particles form the boundary of an extended branch of accessible space that is connected to the percolating void at only one end (the “mouth”). Within this “dead end”, the colored free particles reside. For one of these particles to leave the dead end, the latter has to be exited by *all* of the free particles located closer to the mouth of the dead end. The probability for such a particle relocation to take place decreases dramatically with the number of involved particles since it requires the complete vacating of substantial parts of the accessible space. It is likely that movements of this kind are significantly less probable than string- or loop-like collective rearrangements of the same number of particles in the

^{3.22}Hard-disk QA systems constitute the two-dimensional analog of hard-sphere QA systems. Since the trapping effects discussed in this section exist in both hard-disk and hard-sphere QA systems, we illustrate these effects using the two-dimensional variant so as to increase the clarity of the arguments.

Figure 3.41: Free particle blocked by trapped particle.

Cutaway section of a representative hard-disk QA system, showing a trapped particle that blocks a free particle from exiting a pocket of the percolating void. Yellow area: visual aid to identify the pocket. Solid purple disk: trapped particle. Solid black disks: matrix particles that delineate parts of the pocket and that confine the trapped particle. Solid blue disk: free particle within the pocket. Green circles: indicators that the matrix particles by themselves do not confine the free particle. Red circles: indicator that the trapped particle together with one of the matrix particles blocks the free particle's path out of the pocket. Gray-and-white striped disks: other matrix particles; gray circles: other fluid particles.



bulk of the same fluid (cf. Fig. 1.12). However, we duly note that this effect is not qualitative but only quantitative in nature, the reason being that the probability for particles to vacate a dead end always exceeds zero.

This starkly contrasts with the two effects that remain to be discussed, both of which were discovered by the author in examinations of animated visual representations of HS-QA systems. Unfortunately, there exists no straightforward procedure to include and discuss video material in works such as this one. Therefore, we instead created a substitute visualization that displays the graphically-averaged appearance of the free particles in an HS-QA system at a large number of different points in time (one hundred) spaced regularly over a large time interval (300 000 time units; cf. Sec. 2.5.1.6). Figure 3.39 illustrates the result of this procedure for the same system realization as depicted in Figs. 3.3(a), 3.22, 3.23, and 3.28, which is located at point C where $(\phi_m, \phi_f) = (0.25, 0.10)$. In the figure, the results of the effects in question are immediately obvious in that the three affected particles appear crystal clear due to the very near cessation of their motion.

Figure 3.40 exemplifies the effect of interlocking free particles. Like Fig. 3.38, the figure shows a section of a representative hard-disk QA system wherein the black and blue solid disks mark the matrix and free particles relevant to the effect, respectively. The gray-and-white striped disks and the gray circles, as before, stand for other matrix and fluid particles. Outlined by the yellow area is a pocket of the percolating void bounded largely by the black matrix particles. As in the above dead-end case, and as indicated by the green circle, this pocket is connected to the percolating void by a “mouth” that can be passed by a free particle. Within the pocket reside the three free particles in question. As the two red circles highlight, none of these particles is able to move towards the pocket's mouth since every conceivable path is obstructed by another free and/or matrix particle. Further, the pocket geometry renders it impossible for the blue particles to rearrange collectively (i.e., circularly; see Fig. 1.12) so as to maneuver out of the yellow area. Thus, the blue particles are “interlocking” (mutually blocking) and are *effectively trapped* despite residing within the percolating void.

Figure 3.41, finally, illustrates the subtle effect of a trapped particle effectively confining a free particle. The figure again depicts a cutaway part of a hard-disk QA system in which the matrix and free particles of interest are shown as black and blue solid disks, respectively. Additionally, the dark-purple solid disk represents a trapped particle. The gray-and-white striped disks and the gray circles reprise their roles of other matrix and fluid particles. Shaded in yellow is a small pocket of the percolating void, the extent of which is determined by the following consideration. As the green circles show, the black particles alone do not confine the blue particle. However, as evidenced by the red circle, the purple particle together with one of the black particles renders it impossible for the blue particle to leave the yellow area. An effective trapping of this kind requires peculiar geometric conditions—in particular, the motion of the trapped particle needs to be severely constrained by the surrounding matrix particles. Nevertheless, our visual observations suggest that in HS-QA systems, the blocking of free particles by trapped particles occurs with a frequency comparable to that of the interlocking of free particles.

Conclusion

This concluding section serves a twofold purpose: firstly, it briefly summarizes the principal results of this work, and secondly, on that basis, it outlines various routes along which its research theme could be continued.

The paramount achievement of this work is the in-depth characterization that we established for the static and in particular for the dynamic properties of hard-sphere quenched-annealed (“HS-QA”) systems. As our key findings, we revealed that the fluid particles of these systems may be subject to discontinuous or continuous dynamic arrest transitions (Sec. 3.2.5), single-particle and collective dynamics on disparate time scales (Sec. 3.2.5), subdiffusion (Sec. 3.2.4), trapping and collective rearrangements (Sec. 3.2.9), and strong local ordering at large overall densities (Sec. 3.2.3). On a global scale, we substantiated the relevance of these phenomena by means of “kinetic diagrams”, i.e., by mapping out the quantities of interest over the entire accessible parameter space (Secs. 3.2.1, 3.2.6, 3.2.7, and 3.2.8).

To elucidate the physics underlying our findings, we geometrically determined the extent of the pores formed by the matrix in HS-QA systems. A pertinent statistical analysis allowed us, among others, to pinpoint the matrix density at which the pores undergo a percolation transition (Sec. 3.3). More importantly, however, on the basis of the geometric information, we identified those fluid particles of an HS-QA system that are located within finite-sized pores. Thus being able to separate fluid particles in free and trapped particles, we showed that only the structure of the free particles fully relaxes (Sec. 3.4.2), determined that both subdiffusion and collective rearrangements occur separately among trapped and free particles (Secs. 3.4.1 and 3.4.3), and identified mechanisms by which fluid particles may induce the trapping of free particles (Sec. 3.4.4). Yet, possibly our most significant discovery in the context of free and trapped particles was a different one, namely the fact that the single-particle and the collective dynamics of HS-QA systems are of considerable similarity if only the free particles are considered—in stark contrast to the full-fluid case (Secs. 3.2.8 and 3.4.2).

Thanks to the level of detail of our characterization, we were furthermore in the position to succinctly compare our simulation data with the various predictions of the replica mode-coupling theory (RMCT; see Sec. 1.4.3). The comparisons showed that for the single-particle properties of HS-QA systems, the predictions agree remarkably well with our data. This agreement particularly holds for the relevant aspects of kinetic diagrams (Secs. 3.2.1 and 3.2.6) and also included many functional features of the

pertinent observables (Sec. 3.2.4 and 3.2.5). In contrast, for the collective properties of HS-QA systems, our investigation revealed considerable differences between RMCT's predictions and our data. Here, the principal discrepancy involved the connected intermediate scattering function (ISF), which in our simulations always fully relaxed while it does not invariably do so according to RMCT (Sec. 3.2.5). In direct consequence, the connected-ISF kinetic diagram differs significantly between theory and simulation (Sec. 3.2.6). In the light of the similarity between the single-particle and the collective dynamics when considering only the free particles of HS-QA systems (see above), we speculated that RMCT's difficulties in this context are rooted in the trapping of fluid particles.

Going beyond this work, there are three avenues along which it would be sensible to continue the research theme of dynamic arrest in HS-QA systems: one could

- (i) refine the analysis of this work by considering larger systems, longer simulation runs, and/or more points in the parameter space,
- (ii) extend the examination by computing additional observables and/or by focusing on other physical aspects, or
- (iii) shift the investigation towards systems that are slightly different from HS-QA systems yet related to them.

In the remainder of this concluding section, we shall outline a representative selection of possible ways to continue this work in terms of the above classification.

In the context of avenue (i), consider for instance the following issue, which to some degree has already been pursued by this author. Recall the case of path III (fluid packing fraction $\phi_f = 0.1$; cf. Sec. 3.2.2), where HS-QA systems exhibit long-time subdiffusion in the vicinity of the percolation transition of the accessible volume (matrix packing fraction $\phi_m = \phi_m^* = 0.2512$; cf. Sec. 2.4). In Sec. 3.2.4, we determined that the logarithmic derivative of the mean squared displacement (MSD) in these cases attains the value $z_{\text{HS-QA}} \simeq 0.5$. Consider further the Lorentz gas (cf. Sec. 1.3.4.2), which like QA systems is defined in terms of a matrix and a fluid component, and which differs from hard-sphere QA systems only in two aspects: its matrix particles are distributed completely at random (allowing overlaps), and it contains only a single tracer particle (i.e., $\phi_f \rightarrow 0$). Like the HS-QA matrix, the Lorentz gas matrix features a percolation transition, and previous studies [111, 113] revealed that associated with this transition is long-time subdiffusive motion of the tracer particle with $z_{\text{Lorentz}} \simeq 0.32$.

In view of the significant difference between the values of z_{Lorentz} and $z_{\text{HS-QA}}$, the fluid particles in the respective systems might be subject to fundamentally distinct physical effects. While this hypothesis is appealing, a trivial difference between the systems in question is their value of ϕ_f . In order to rule out any influence of this parameter, we performed additional simulations for HS-QA systems at $\phi_m = \phi_m^*$ and

at $\phi_f \rightarrow 0$, with preliminary results suggesting that the long-time value of $z_{\text{HS-QA}}$ in this case is nearly identical to z_{Lorentz} . Motivated by this finding, we conducted further simulations of HS-QA systems at $\phi_m = \phi_m^*$ and at various values of ϕ_f between zero and 0.15. The simulations were substantially longer and the systems substantially larger than those presented in this work, yielding the following results. Firstly, the value $z_{\text{HS-QA}} = 0.5$ found in Sec. 3.2.4 likely represents only an extended transient regime, with $z_{\text{HS-QA}}$ subsequently tending towards z_{Lorentz} . Secondly, the value of $z_{\text{HS-QA}}$ in the transient regime most certainly increases monotonically with ϕ_f . Thirdly, as ϕ_f is increased towards the dynamic arrest transition, the transient regime probably persists indefinitely. Finally, as an exciting by-product, the new data at last suggest that HS-QA systems do feature a reentrant regime, namely in that the value of the MSD at very long times might be marginally *larger* for $\phi_f \gtrsim 0$ than for $\phi_f \rightarrow 0$.

While these results are tantalizing—particularly due to pertinent predictions by RMCT [162]—, they unfortunately are only preliminary and require significant substantiating before any publication is warranted^{C.1}. In continuation of these findings, future work could for instance examine whether the single-particle ISF at the above parameter-space points also indicates a reentrant behavior. A rather different imaginable undertaking would be to try to locate parameter-space regions that would correspond to higher-order (“ A_3 ”) dynamic singularities predicted by RMCT [159, 161, 162], which could be accomplished for instance by identifying logarithmic decay patterns of ISFs [159, 197, 198, 211, 241, 242].

In terms of avenue (ii), there are several conceivable ways in which the study of HS-QA systems could be extended. Possibly the simplest approach would be to reinterpret data that already exist (or that could be easily obtained by established methods) so as to extract additional information about the systems. By this approach, it would for instance be possible to investigate the “fragility” of HS-QA systems—a quantity defined via the relaxation times at different parameter-space points [3, 11, 154]—the expectation being that the presence of the matrix alters this property of the fluid. Also, one could determine the extent of the HS-QA parameter-space region in which there exists long-range order among the fluid particles, thus assessing the degree of frustration imposed onto the fluid by the matrix (cf. Sec. 1.3.3.2). This analysis could be conducted on the basis of radial distribution function and/or static structure factor data (cf. Sec. 3.2.3). A different approach in the context of avenue (ii) would be to substantiate previous conclusions by inferring equivalent information with different means, one example being the re-evaluation of ISF relaxation times by the use of alternative methods, involving for instance integrations over the ISFs in question.

However, perhaps the most productive approach to extend this work would be to consider both new observables *and* new physical aspects. As an example, in the aforementioned context of frustration and crystallization, the extent and role of possibly-present crystallites could be elucidated by the computation of spherical harmonics—

^{C.1}Notwithstanding their uncertain status, the new results were discussed by the author in an oral presentation at the 8th Liquid Matter Conference in Vienna in 2011.

based bond-order parameters such as q_6 and/or q_4 [168, 255, 260]. Even more interesting may be a more in-depth investigation of the slow dynamics of HS-QA systems. For instance, examinations of video material suggest that under suitable circumstances, the mobility of the particles of the QA fluid may be highly dependent upon the spatial region. A widely-used simple quantity to characterize such “dynamic heterogeneities” (cf. Sec. 1.2.5) is the “non-Gaussian parameter” α_2 [89, 140, 220, 282], and a more sophisticated one—which recently received tremendous attention—is the four-point dynamic susceptibility χ_4 [25–28, 36, 78, 155, 220]. Further, inspired by the notion of spatial heterogeneity in systems with slow dynamics, a number of recent works have determined various observables for small spatial regions and/or for small groups of particles [133, 171, 228, 244, 245, 255]. This concept could be adapted to many quantities inspected in this work—characterizing for instance the local mobility of fluid particles in HS-QA systems—, and might contribute to elucidating the microscopic origin of many phenomena described in this work. In a broader sense, the quantities outlined in this paragraph might ultimately even shed light onto the fundamental origin of dynamic arrest, an issue that eventually we did not address in the analysis part of this work^{C.2}.

With the latter consideration, we naturally segue to avenue (iii)—the shifting of focus to somewhat different systems. In view of the multitude of systems that could be investigated, the initial step of proceeding in this direction should be to establish firm principles for selecting the system. For example, pertinent analyses would be particularly efficient if the tools and methods of this work could be recycled. In this spirit, one could for instance require that RMCT be capable of describing the system or that the matrix feature a percolation transition. The former would narrow the choice to systems with precisely three interaction types—matrix–matrix, matrix–fluid, and fluid–fluid—, while the latter would suggest to focus on systems in which the matrix–fluid interaction includes a hard-core part (cf. Secs. 1.1.3, 2.2, and 2.4).

Allowing for these restrictions, one possible alteration with respect to monodisperse HS-QA systems would be to change the dimensionality (for instance to two or to four dimensions), the prospect being that it renders interesting phenomena more pronounced and/or more accessible. Another conceivable modification would be to consider matrix particles with a diameter differing from that of the fluid particles—a scenario that has already been explored in terms of RMCT by the author and his colleagues in article listed in the Preface. However, possibly the most productive type of change would involve solely the matrix. Specifically, one could suppress the matrix–matrix interaction altogether so as to obtain the matrix structure of the extremely well-investigated Lorentz gas [18, 83, 104, 111–113, 159, 246, 247], which would be interesting not least for the reasons outlined in the context of avenue (i). Alternatively, it would be possible to modify the protocol by which the matrix is obtained (cf. Sec. 2.3)

^{C.2}In the end, two circumstances deterred us from our plan of using HS-QA systems to elucidate the nature of dynamic arrest: firstly, HS-QA systems turned out to be complex and interesting of their own right; secondly, we came to realize that equilibrated-mixture systems as described in the context of avenue (iii) might present a more suitable basis for the envisioned studies.

since, as Krakoviack showed [164], RMCT is in fact capable of describing systems in which the matrix component of a QA-like system is equilibrated in the presence of the fluid particles before being frozen in place. Such “equilibrated mixture” (EM) systems have likewise been studied extensively [131, 148–150, 163] and have been found to exhibit many features similar to those of HS-QA systems. In addition, it has recently been shown by a number of authors [29, 42, 47, 135–137, 253] that EM systems can be interpreted in terms of point-to-set correlation functions (cf. Sec. 1.2.5), meaning that a detailed characterization of these systems may contribute to a fundamental understanding of the glass transition.

Jettisoning the requirements of the theory and of the established tools of analysis, a more drastic path in modifying the investigated systems would be to closer resemble real systems. One example to this end would be to introduce polydispersity, i.e., to consider particles whose sizes obey some distribution. If the polydispersity remains limited to the fluid particles while retaining monodisperse matrix particles, many tools and methods of this work would still be largely applicable—including percolation analyses, which could be conducted separately for each fluid particle (cf. Sec. 2.4 and Appendix A.1). In the more complex case of polydisperse matrix particles, in contrast, many of the tools and methods would require substantial modification [48, 61, 179, 229]. Notwithstanding, a recent article [50] has studied the case of two-dimensional QA systems with polydisperse hard-disk matrix particles and hard-disk tracer fluid particles, the findings including a considerable impact of the polydispersity onto the diffusivity of the tracers. In more general terms, the notion of polydispersity is of significant relevance in the aforementioned context of frustration in glass-forming systems (cf. Sec. 1.3.3.2)—an issue that has been highlighted in a number of recent publications [46, 204, 255, 287].

The list of further conceivable system alterations is of course vast. However, in view of the multitude of possible extension schemes outlined in the above paragraphs, it is not too far-fetched to project that taking on even a fraction of these challenges would easily fill another thesis of this kind. Therefore, we feel that at this point it is due time to conclude this section, and consequently this work, in the hope to have demonstrated that the glass formation of colloids in porous media holds many fascinating phenomena and that the continued pursuit of this research theme would be an occupation well worthwhile.

Appendix

A.1 Network picture of the accessible volume

In this appendix section, we shall be concerned with proving that the Delaunay tessellation is indeed suited to determine the distinct voids present in a hard-sphere quenched-annealed (HS-QA) system. As laid out in Sec. 2.4.1, if we are given some d -dimensional hard-body QA system, then it is in principle relatively simple to identify therein all tuples of d matrix particles that cannot be passed by a fluid particle (cf. Fig. 2.15). A more intricate task is to relate these d -tuples to the volumes—and more specifically to the individual voids—that are present in the system. Such a relation, however, is vital to classify the fluid component of an HS-QA system into free and trapped particles. In Sec. 2.4.3, it is asserted that the Delaunay tessellation constitutes a solution to this problem. To mathematically prove this, we shall first discuss the Voronoi tessellation and its properties the context of accessible volumes, on the basis of which we will then put the tightly-related Delaunay tessellation in the same context. As in Sec. 2.4, in this appendix we will substitute the shortcut “D-” for “Delaunay”, and we will also frequently use the abbreviation “V-” for “Voronoi”.

A.1.1 Voronoi picture

In order to associate the noncrossable matrix-particle tuples with regions of accessible volume, we need to essentially *reverse* the problem at hand, i.e., focus on points located in the “depth” of the voids as opposed to on their boundaries. In the following, we will reproduce, illustrate, and correct a method devised by Kerstein [142] and expanded by Sastry et al. [229] to characterize the connectivity of the accessible volume in terms of a “Voronoi network”. While the tuples may at first seem irrelevant in this method, we will later see that they naturally reappear and assume a vital role.

A.1.1.1 Voronoi tessellation

Since Kerstein’s V-network method is fundamentally based on the concept of the “Voronoi tessellation”, we first need to understand the nature and properties of this very peculiar geometric construction. Like the D-tessellation (cf. Sec. 2.4.2), the V-tessellation is defined on the basis of a set of distinct points that may be arbitrarily

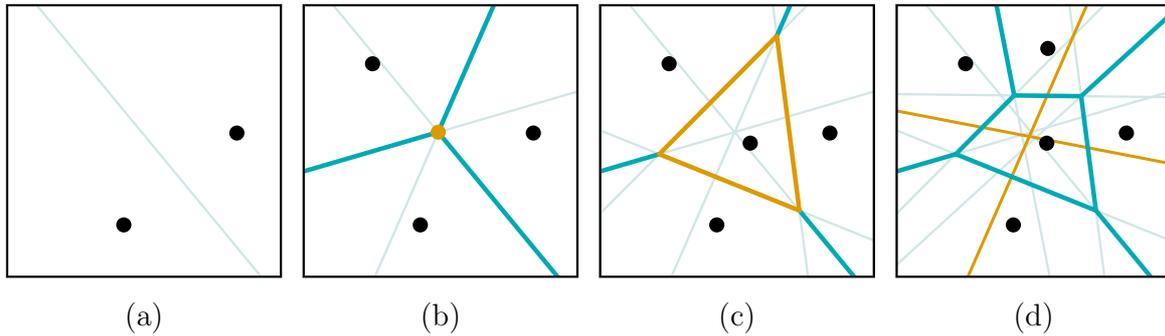


Figure A.1: Voronoi tessellation. Features of a V-tessellation; explanation: see text. Black disks: D-vertices; thick dark-turquoise lines: V-edges; lines of all shades of turquoise: bisectors. In orange color, various features of the V-tessellations are highlighted: a bisector [panel (a)], a V-vertex [panel (b)], a bounded V-cell [panel (c)], and bisectors that are not part of a V-edge [panel (d)].

distributed in space and that (anticipating the relation between the V- and D-tessellations) we will call “Delaunay vertices”. The tessellation—formally introduced by Georgy Voronoi^{A.1} in 1908 [275]—divides the given space in “Voronoi cells” by defining regions in space that are closer to one D-vertex than to all other D-vertices.

Figure A.1 illustrates the nature of V-cells in $d = 2$; generalizations to (and particularities of) higher-dimensional cases will be discussed in the next paragraph. Panel (a) shows a square-shaped area—without periodic boundary conditions (PBCs; see Sec. 2.2.2.1)—that contains two D-vertices (small black disks, symbolizing points). The area is divided into two V-cells, delimited by a “bisector” (highlighted in orange) which contains all points that are equidistant from the two D-vertices. In the following, by the term “Voronoi edge”, we will refer to a line or line segment that is equidistant from two D-vertices and more distant from all other D-vertices. In panel (b), a third D-vertex is introduced, which entails another two bisectors. As one can see, the three lines—now shown in shades of turquoise—meet in a single point (marked by the orange disk). Such points are equidistant from three D-vertices and shall henceforth be referred to as “Voronoi vertices”. Further, panel (b) illustrates that in the general case, some parts of the bisectors *do* contribute to the V-edges (thick dark turquoise lines)

^{A.1}As a matter of fact, the V-tessellation was introduced several times by scientists in various fields [15, 37]. The first scholar to come across this method was the famous mathematician Lejeune Dirichlet, who in the 1850s [65] found similarities of this construction with quadratic forms as discussed by another famous mathematician, Carl Friedrich Gauß [94]. Georgy Voronoi (Ukrainian: Георгій Феодосійович Вороній) is “merely” credited with the original generalization of the tessellation to arbitrary dimension [275]. Nevertheless, in many fields, the name “Voronoi tessellation” has prevailed, equivalently running under the names “Voronoi decomposition” or “Voronoi diagram”. Notably, being a relatively-intuitive method, the construction was also used by a number of early “geologists, foresters, agriculturalists, medical researchers, geographers, crystallographers, and astronomers” [70], many of which, however, were not aware of previous descriptions of this method. Therefore, in some fields, the tessellation is not associated with the name of Voronoi but instead with that of researchers in the field, as is the case for instance in solid-state physics (Wigner and Seitz [283]) and meteorology (Thiessen [257]).

whereas other parts do *not* do so (thin faint turquoise lines). In panel (c), another D-vertex is added, resulting in another three bisectors. Contrary to the V-cells in panels (a) and (b), the V-cell of the newly-added center is entirely bounded by V-edges that coincide with *segments* of bisectors (highlighted in orange). The V-cell therefore assumes the shape of a *polygon* (here a triangle). In panel (d), another D-vertex is added, thereby changing the shape of the central V-cell to a quadrilateral. Most notably, however, among the $\sum_{i=1}^4 i = 10$ bisectors, there are now two (marked in orange) that do not contribute at all to the V-edges. In systems containing much more D-vertices, *most* of the bisectors are of this type, which is the principal challenge to face when actually computing V-tessellations.

In the general, d -dimensional case (which includes the important case $d = 3$), a V-tessellation features much the same properties as in two dimensions. Notable changes include the bisectors, which in d dimensions take the form of $(d-1)$ -dimensional hyperplanes instead of lines in $d = 2$. The V-cells are then bounded by segments of such hyperplanes (“Voronoi faces”) and assume the topology of d -dimensional hyperpolyhedra. While the notion of V-vertices remains intact, in d dimensions they represent points in which $d+1$ hyperplanes intersect. Therefore, a V-vertex in d dimensions is equidistant to $d+1$ D-vertices. From this it follows that there also exist points in which only d hyperplanes intersect (which serves as the general definition of V-edges), or $d-1$ hyperplanes, all the way to three hyperplanes as in two dimensions.

Finally, the V-tessellation—like the D-tessellation—is well defined also in the presence of PBCs (see Sec. 2.4.2); this follows directly from the local nature of the V-cells. The similarity to the D-tessellation also extends to the nearest-image convention, in the presence of which the V-tessellation likewise requires care if the system is small. *Unlike* the D-tessellation—which may in principle involve degeneracies (see Sec. 2.4.2 and Fig. 2.19 therein)—, the V-tessellation is *always unique*.

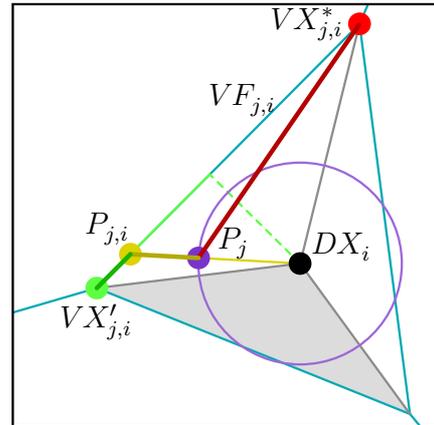
A.1.1.2 Voronoi network

Kerstein [142] uses the peculiar features of V-tessellations to characterize the accessible space in a three-dimensional arrangement of particles. Specifically, he considers the V-tessellation in which the centers of the particles constitute the D-vertices. Since we are concerned only with the space that the matrix particles leave accessible to fluid particles in an HS-QA mixture, for the D-vertices we will consider only *matrix* particle centers.

Kerstein formulates and proves a set of statements (one lemma and three theorems) that, taken together, provide rules for how to interpret the V-tessellation as a network that characterizes the accessible volume. In the following, we will recite these statements (with adapted terminology) and illustrate Kerstein’s proofs.

Lemma 1 (Kerstein): A given point in the interior or on the boundary of a Voronoi cell is connected to a Voronoi vertex by a path which is never closer to the Delaunay vertex than is the given point.

Figure A.2: Proof of Lemma 1. Same two-dimensional V-cell VC_i as in Fig. A.1(c), slightly enlarged. Explanation: see text. Purple disk: given point P_j ; black disk: D-vertex DX_i ; red disk: V-vertex $VX_{j,i}^*$ of VC_i that is most distant from DX_i ; yellow disk: projection $P_{j,i}$ of P_j onto V-face $VF_{j,i}$; green disk: projection of $P_{j,i}$ onto V-vertex $VX'_{j,i}$. Gray lines: delimiters of Kerstein's triangles; gray area: example triangle. Purple circle: points with same distance to DX_i as P_j . Yellow and green lines: projection lines; dark yellow and green lines: desired path connecting P_j and $VX'_{j,i}$; dark red line: path proposed by Kerstein.



Kerstein's proof for Lemma 1 goes as follows. Say that, as illustrated in Fig. A.2, the given point P_j (purple disk) resides in V-cell VC_i (turquoise lines) around D-vertex DX_i (black disk). Then, Kerstein considers triangles (gray lines and gray example area) for which the triangle apex is DX_i and the triangle base is a face of VC_i . If VC_i has n faces, then it can be entirely covered by n such triangles; therefore, P_j must be contained in one such triangle, $Y_{j,i}$. In three dimensions (where the triangles corresponds to pyramids), $Y_{j,i}$ contains at least three V-vertices, one of which (say $VX_{j,i}^*$, red disk) is the point within $Y_{j,i}$ that is farthest from DX_i . Therefore, $VX_{j,i}^*$ is farther from DX_i than is P_j . Without further proof, Kerstein asserts that the path referred to in Lemma 1 is a straight line (dark red line) between P_j and $VX_{j,i}^*$. As is obvious from Fig. A.2, the path suggested by Kerstein may intersect the purple circle—the latter containing the points that have the same distance from DX_i as P_j —, which means that the path is *not* always farther from DX_i than P_j , and consequently that Kerstein's proof is **wrong**.

However, Lemma 1 still holds if we modify the proof as follows^{A.2}. Instead of connecting the given point P_j to $VX_{j,i}^*$, we connect P_j to *some* V-vertex, namely in a way that the path by construction consists of line segments that successively lead farther away from DX_i . First, consider the straight line passing through DX_i and P_j (light-yellow line). Obviously, following this line from DX_i beyond P_j (dark yellow line) we always move farther away from DX_i . The point on this line that is farthest from DX_i but still within VC_i is the intersection point $P_{j,i}$ (yellow disk) of the line with one of VC_i 's faces, $VF_{j,i}$. Next, we find the point on $VF_{j,i}$ that is closest to DX_i by intersecting $VF_{j,i}$ with its normal vector passing through DX_i (dashed light-green line). We then consider the straight line from this point to $P_{j,i}$ (solid light-green line) and follow it beyond $P_{j,i}$ until we exit VC_i (dark solid green line). This line, again, leads only farther away from DX_i . In two dimensions, the exit point is marked by some V-vertex, $VX'_{j,i}$. Since Lemma 1 does not demand that $VX'_{j,i} = VX_{j,i}^*$, we can identify $VX'_{j,i}$ with the required V-vertex and the line $P_j \rightarrow P_{j,i} \rightarrow VX'_{j,i}$ with the

^{A.2}The proof to Lemma 1 provided in this work is *original* to this work, i.e., published nowhere else (yet).

desired path. In d dimensions, the projection procedure described above has to be carried out d times in an analogous way; in $d = 3$, for instance, we first project onto a V-face, then onto a V-edge, and only then onto a V-vertex. Notably, this reasoning is very similar to Kerstein's proof to Theorem 2 (see further below).

Theorem 1 (Kerstein): Every point in the accessible space is connected to some Voronoi vertex by a path contained within the accessible space.

Theorem 1 is a direct consequence of Lemma 1 since by definition every point (whether accessible or not) is contained in some V-cell.

Theorem 2 (Kerstein): If two Voronoi vertices are connected by the accessible space, then they are connected by Voronoi edges.

Theorem 2 is easily proven by a procedure similar to this work's proof to Lemma 1. Suppose that two V-vertices, VX_i and VX_j , are connected by some path in the accessible volume. Then, by $d-1$ successive projections it is possible to map every point of that path onto some V-edge. Since this mapping is affine with fixed points VX_i and VX_j , the projected points (and thus the V-edges) connect the given V-vertices without gaps.

Theorem 3 (Kerstein): A Voronoi edge is contained within the accessible space if and only if its point of closest approach to the plane of the adjacent Delaunay vertices (i.e., the Delaunay vertices of the three Voronoi cells sharing the Voronoi edge) is in the accessible space.

Here, we follow Kerstein's proof: By definition, each point on V-edge VE_i is equidistant from the d D-vertices $\{DX_j\}_i$ and more distant from all other D-vertices. The points equidistant to the $\{DX_j\}_i$ are located on a straight line L_i perpendicular to the plane defined by the $\{DX_j\}_i$. The point $P_{i,L} \in L_i$ that is closest to the $\{DX_j\}_i$ is the intersection of L_i and the plane. If $P_{i,L}$ is a member of VE_i , then $P_{i,L}$ is identical with $P_{i,VE}$, the point of VE_i closest to the $\{DX_j\}_i$. If $P_{i,L}$ is *not* a member of VE_i , then $P_{i,VE}$ is instead constituted by the point on VE_i that is closest to $P_{i,L}$. In either case, if $P_{i,VE}$ is in the accessible space, then all points of VE_i are in the accessible space.

Consider now some arrangement of matrix particles A . For any such arrangement, the stated theorems and lemmas allow to translate the question "What is the connectivity of the accessible space of A " to the question "Which V-edges connect which V-vertices in the V-tessellation of A ". The initial question is thereby recast in terms of a *network percolation* problem as discussed in Sec. 1.3.3.3, namely by treating the V-vertices as "sites" and the V-edges as "bonds" (cf. Ref. [248]). This recasting reduces the investigator's task to identifying V-vertex clusters, i.e., V-vertex groups in which every member is connected to at least one other member by a V-edge. Since

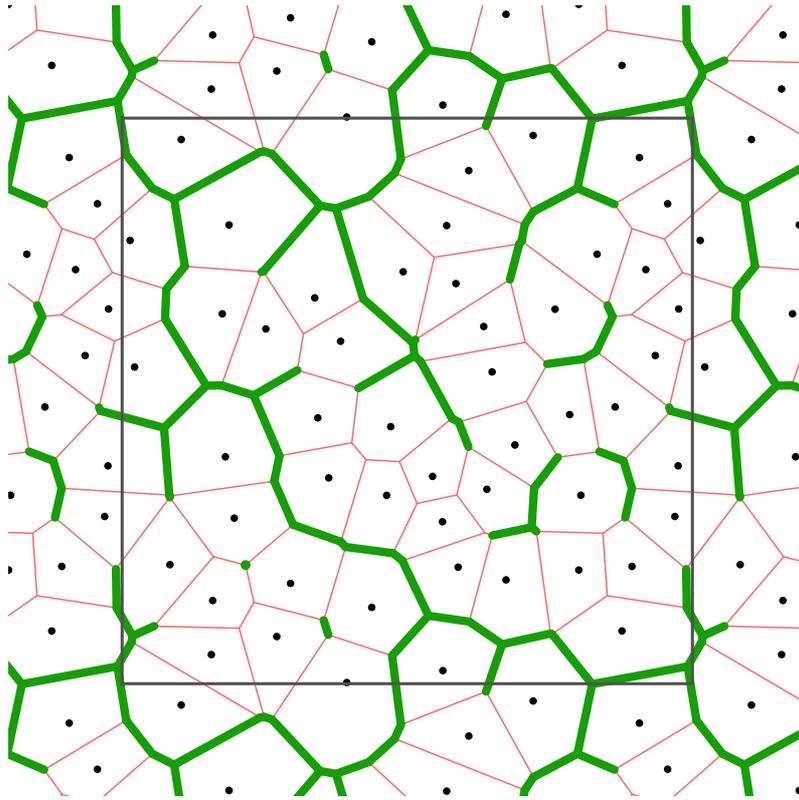


Figure A.3: Voronoi network. Voronoi representation of the same system as in Figs. 2.16, 2.17, and 2.20. Green lines: connecting V-edges; red lines: disconnected V-edges; green dot: isolated accessible V-vertex. Small black disks: reminders of the matrix particle positions. The green elements trace the percolating void and evidence the presence of five traps.

each V-vertex is defined by a local environment of matrix particles, a V-vertex cluster has to count infinitely-many members in order to correspond to a percolating void; correspondingly, clusters with a finite number of members represent traps. Thus, the V-network allows to determine how many traps and percolating voids are present in a given system.

Figure A.3 illustrates the V-network for the same system as shown in Figs. 2.16, 2.17, and 2.20. In the figure, connecting V-edges are marked as thick green lines, whereas disconnected edges are shown as thin red lines. Additionally, shown as a green dot is the lone V-vertex that is accessible but otherwise disconnected from other V-vertices^{A.3}. For reference, the small black disks serve as reminders of the positions of the matrix particles. As expected, some of the green elements (actually the majority in this case) map out the percolating void, while others indicate that there are five traps present in the system.

Albeit certainly valuable, the V-network unfortunately does *not* provide a means

^{A.3}The finder may keep it.

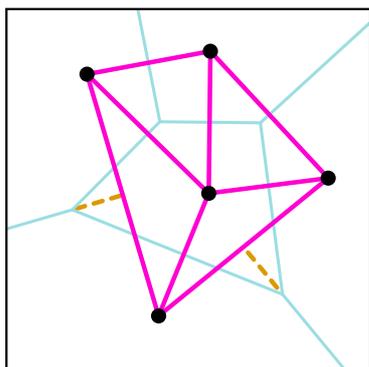


Figure A.4: Delaunay tessellation. Two-dimensional D-tessellation and its Voronoi counterpart; same arrangement of D-vertices as in Figs. 2.18 and A.1(d). Thick pink lines: D-faces; light-turquoise lines: V-edges. Small black disks: D-vertices. Orange dashed lines: extensions of V-edges to their corresponding D-faces.

to determine which void a random accessible point is located in. Therefore, the V-network is not directly suited for identifying whether a given fluid particle in a QA mixture is free or trapped (which is our ultimate aim). Also, the number of V-vertices in a cluster is not an immediate measure of the size of the corresponding void. As shall be detailed in the following, these deficiencies are remedied by the D-tessellation.

A.1.2 Delaunay picture

In this section, we shall give the proof that—as asserted in Sec. 2.4.3—the D-tessellation is suited to determine which void a given accessible point belongs to in a system with particulate obstacles. While a considerable number of authors have applied the D-tessellation in precisely this way [39, 48, 107, 246, 251, 252], it was only Sastry et al. [229] who attempted a systematic validation of this application. Unfortunately, we discovered a major flaw in one of the proofs by the latter authors; this circumstance, however, we shall remedy in this work. In the following, we will first point out the relationship between the D- and V-tessellations, then elaborate on the differences between the associated networks, and finally describe in detail the mechanism by which the D-tessellation associates accessible points with voids.

A.1.2.1 From Voronoi to Delaunay

Our first task is to substantiate the statement that—as mentioned before in Sec. 2.4 and in this appendix—the D- and V-tessellations share a tight relationship. Perhaps the most striking consequence of this relationship is the possibility to construct either one of the tessellations exclusively from information encoded in the respective other tessellation. Using, for instance, as starting point some V-tessellation in d dimensions, the corresponding D-tessellation can be obtained from the following simple rules:

- (i) For each V-edge VE_i , identify the d associated D-vertices $\{DX_j\}_i$. Connect the $\{DX_j\}_i$ such that they form a $(d-1)$ -dimensional hypersurface element DF_i .
- (ii) Interpreting the DF_i 's as D-faces, find all groups of $d+1$ faces that enclose a d -dimensional D-cell.

This prescription is illustrated in Fig. A.4, in which the pink lines indicate the D-tessellation as in Fig. 2.18, and the turquoise lines outline the corresponding V-tessellation as in Fig. A.1(d). It is easily verified that the figure contains precisely one D-face for each V-edge (in $d = 2$, both faces and edges are straight lines or segments thereof), and that each D-face is perpendicular to its corresponding V-edge. Also, it is evident that a D-face does not necessarily intersect its associated V-edge; this is indicated by the orange dashed lines, which extend the respective V-edges to their corresponding D-faces. This particular property of the D-tessellation means that there exist D-cells that do not host a V-vertex, which has significant ramifications for the pertinent network (see Appendix A.1.2.2).

On a more mathematical note, the one-to-one correspondence between V-edges and D-faces as well as between V-vertices and D-cells is tantamount to the D- and V-tessellations being “dual” to each other in the graph-theoretical sense^{A.4}. Since neither of the tessellations allows isomorphisms—relocations of vertices under bond structure preservation—the respective dual construction is *unique* and encodes exactly the same information. The only exception to this rule are degenerate D-tessellations (cf. Sec. 2.4.2), which contain *more* information than their respective Voronoi counterpart. However, the duality still holds in the sense that different realizations of a degenerate D-tessellation (as in Fig. 2.19) correspond to the same V-tessellation.

A.1.2.2 Network pictures: Delaunay \neq Voronoi

As pointed out before, a considerable number of authors have used the D-tessellation to identify voids in condensed-matter systems with obstacle particles [39, 48, 107, 229, 246, 251, 252]. Surprisingly, however, most of these authors provide only little justification for this proceeding, be it in the form of proofs or references. The only exception to this unfortunate finding is the previously-mentioned work by Sastry et al. [229], which forms the basis for many parts of this appendix.

Figure A.5 illustrates the necessity to validate that by the method detailed in Sec. 2.4.3 the D-tessellation indeed identifies voids correctly. The figure shows two different tessellations of the same two-dimensional matrix particle arrangement, namely (a) the D-tessellation, and (b) some arbitrary other triangulation. The colors encode the connectivity picture that emerges from the different tessellations, with noncrossable faces highlighted as thick red lines (similar to Fig. 2.16) and groups of cells flood-filled in yellow if entirely bounded by noncrossable faces. Additionally, inaccessible cells are marked in dark gray, and crossable faces are indicated by thin green lines. Clearly, the two tessellations suggest very different connectivity patterns, which is unfortunate since obviously there exists only a single physically correct pattern for a given arrangement of matrix particles. Therefore, good (geometric) reasons have to be brought forth for favoring one tessellation over the other.

One approach to this puzzle is the following. As laid out in Sec. 2.4.3, the D-tessellation can be interpreted as a network representation of the accessible volume. Since we

^{A.4}http://en.wikipedia.org/wiki/Dual_graph

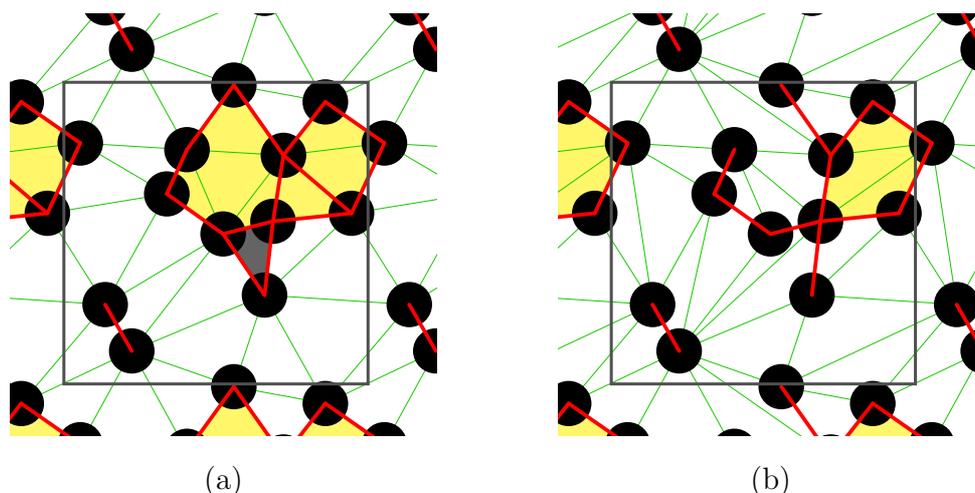


Figure A.5: Connectivity following from different tessellations. Panels (a) and (b) show two different triangular tessellations of the same arrangement of matrix particles. Black disks: matrix particles. Green (red) lines: triangle “faces” that are (non)crossable by a fluid particle when disregarding all but the two matrix particles connected by the face. Yellow areas are entirely bounded by red lines; gray triangles are inaccessible. Like in Figs. 2.16, 2.17, and 2.20, the system is subject to PBCs (cf. Sec. 2.2.2.1), with the dark gray box indicating the “original” volume.

have already validated the network arising from the V-tessellation (Appendix A.1.1.2), a comparison with the D-network may shed light on the validity of the latter. Unfortunately, despite the duality between the tessellations (Appendix A.1.2.1) the two networks happen to *differ* slightly. However, it is premature to discard the D-tessellation solely on the basis of this disparity. Instead, it is prudent to first investigate the nature of the difference in order to assess the repercussions on the D-tessellation’s capability to describe voids. For this investigation consider Fig. A.6, which shows the D- and V-tessellations for an arrangement of five matrix particles, with the matrix particles displayed as in Fig. 2.17, i.e., with possibly-present fluid particles of point size and the accessible volume represented by the white areas.

First, we remember that the sites in the D-network are considered accessible based on point (1) in the beginning of Sec. 2.4.3. This criterion effects a one-to-one correspondence between the sites of the D- and of the V-networks. Since in panel (b) of Fig. A.6, all V-vertices are located in the accessible area (see magnified section at the far right), all D-cells in panel (a) are deemed accessible. However, the same figure evidences that the entire accessible volume is located within merely *one* of the three D-cells (namely D-cell x). From this it follows that according to the above criterion an accessible D-cell can be “empty”—i.e., devoid of accessible volume.

Second, consider the D-faces in Fig. A.6. Here, a disparity between the D- and V-networks is obvious: while the D-face $z-x$ is *noncrossable* [thick red line in panel (a)], the V-edge $z-x$ is *connecting* [thick green lines in panel (b)]. This disparity arises from

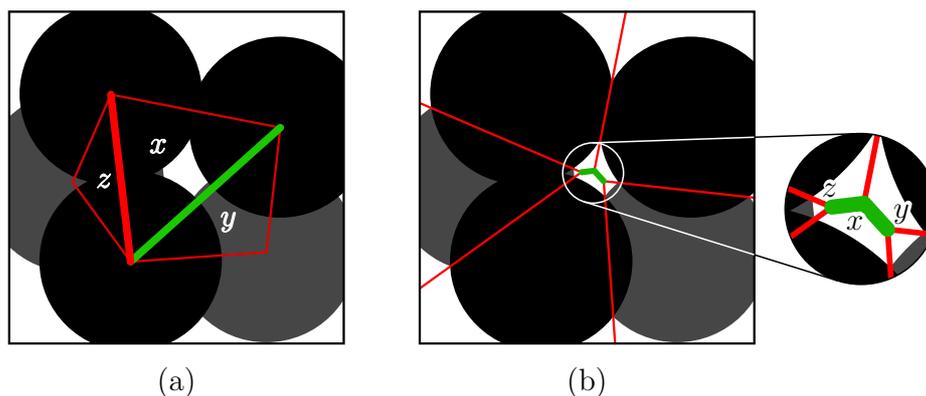


Figure A.6: Differences between the Delaunay and Voronoi networks. Panels (a) and (b) show the same arrangement of five matrix particles. Black and dark gray disks: matrix particles drawn as in Fig. 2.17; white area: accessible space. Panel (a): D-tessellation; red lines: noncrossable D-faces; green line: crossable D-face. Panel (b): V-tessellation; red lines: nonconnecting V-edges; green lines: connecting V-edges. The circle to the very right shows an enlarged view of the central white circle in panel (b). Most noteworthy is the fact that the D-face between D-cells x and z is noncrossable whereas the V-edge between V-vertices x and z is connecting.

the fact that a V-vertex can reside outside of its pertinent D-cell, which is connected to the circumstance that a V-edge does not necessarily intersect its corresponding D-face (cf. Fig. A.4). Therefore, the D- and V-networks generally differ in the “bonds” aspect.

To assess these findings, it is essential to remember that our ultimate aim is to characterize the connectivity of the accessible volume in a way that trapped and free particles can be identified. From this perspective, the “accessible-but-empty” aspect is fairly uncomplicated in that an empty D-cell cannot contribute to any void’s accessible volume—irrespective of the network. The “different bonds” aspect is, unfortunately, not quite as simple when viewed from the same perspective. In fact, it is mainly this issue that the remainder of this appendix shall be devoted to, the ultimate conclusion being that bond differences do not affect the superordinate goal of describing voids. Unfortunately, proving this assertion is a relatively-complicated matter since it requires knowledge about two additional, nontrivial properties of the D-tessellation in the context of accessible volume. However, since for this work it is of paramount importance that the D-tessellation indeed correctly identify voids, we will in the following section (Appendix A.1.2.3) describe and prove these additional properties in detail. Subsequently, in Appendix A.1.2.4, we will assemble the proof for the desired capability of the D-tessellation, which then concludes this appendix.

A.1.2.3 Additional Delaunay tessellation properties

In this section, we will describe and prove two properties exhibited by the D-tessellation in connection with the accessible volume in HS-QA systems. These properties serve to

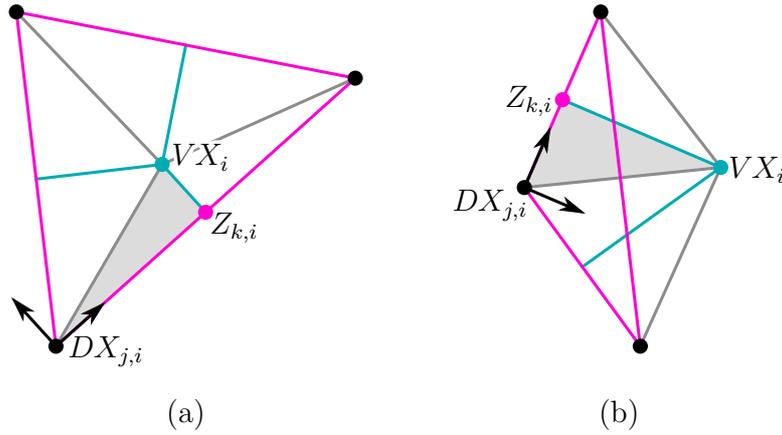


Figure A.7: Subcells for proof of Lemma 2. Two-dimensional scheme for constructing right-angle subcells that entirely cover a D-cell. The vertices of the subcells fulfill the following criteria: one vertex is adjacent to edges that all meet in right angles (pink disks), one vertex is a V-vertex (turquoise disks), and one vertex is a D-vertex (black disks). The pink lines are D-faces, and if only matrix particles at the pictured D-vertices are considered, then the turquoise lines are V-edges. Gray lines complete the boundaries of the subcells; light-gray shades indicate example subcells. For the latter, small arrows indicate the origin and base vectors of the coordinate systems defined by the edges meeting in the right-angle points. Panel (a) shows D-cell x from Fig. A.6, where the corresponding V-vertex resides inside x . Panel (b) shows D-cell z from the same figure, where the corresponding V-vertex is located outside z .

validate that the D-tessellation provides a correct description of the voids in a system. The properties in question have been first described in the elaborate work by Sastry et al. that we quoted before [229]. Like for Kerstein’s lemmas and theorems (see Appendix A.1.1.2), we will recite the properties as described by Sastry and coworkers (with adapted terminology) and provide the pertinent proofs. As in the case of Kerstein, we unfortunately found one of the proofs by Sastry et al. to be flawed; this circumstance we will correct in this section.

Lemma 2 (Sastry): If any point in a Delaunay cell belongs to the accessible space, the corresponding Voronoi vertex belongs to the accessible space.

The proof by Sastry and coworkers starts off by considering a particular D-cell DC_i with D-vertices $\{DX_j\}_i$, D-faces $\{DF_k\}_i$, and corresponding V-vertex VX_i . The central concept of Sastry et al. is to construct several “subcells” from DC_i [229]. Figure A.7 illustrates the two-dimensional analog of their procedure: First, for each $DF_{k,i}$, find the orthogonal projection of VX_i onto that $DF_{k,i}$ and call the projected point $Z_{k,i}$. Then, define nonoverlapping subcells, with VX_i as the first, one of the $\{Z_k\}_i$ as the second, and one of the $\{DX_j\}_i$ as the third vertex. For any subcell defined this way, the respective $Z_{k,i}$ is adjacent to a right angle. As can be seen from Fig. A.7, if VX_i is

located inside DC_i , then all possible subcells combined precisely cover DC_i [panel (a)]; if on the other hand VX_i resides outside DC_i , then all subcells stretch beyond DC_i and merely a subset of the subcells is needed to entirely cover DC_i [panel (b)]. The generalization of this subcell-construction scheme to d dimensions is straightforward: one of the vertices of a subcell is VX_i , one is an orthogonal projection of VX_i onto a $(d-1)$ -dimensional face, one is the projection of such a vertex onto the $(d-2)$ -dimensional boundary of a face, and so on until the d^{th} projection (and vertex) is some $DX_{j,i}$. This way, akin to the $d = 2$ case, each d -subcell features one vertex (the one obtained from the first projection) at which all edges mutually enclose right angles [229].

The right-angle property is central to the proof of Lemma 2 since it allows to define a simple coordinate system in each subcell on the basis of the edges that enclose the right angles [229]. Choosing as the origin the one and only $DX_{j,i}$ that constitutes a vertex of a given subcell, it is possible to scale each of the orthogonal base vectors with $+a$ or $-a$ (with $a \in \mathbb{R}$ being the same for all base vectors) such that VX_i assumes the highest coordinate of all points in that subcell (see Fig. A.7). Thus, for each subcell, VX_i is the point within that subcell that is furthest away from the subcell's $DX_{j,i}$. Now, assume that a given point P_k is located in the accessible space and within D-cell DC_i . Since we can select the subcells such that they wholly cover the D-cell, we can guarantee P_k to be contained in one of the subcells. Say in the following that P_k resides in the subcell associated with D-vertex $DX_{k,i}$. There are then two characteristics of P_k :

- (1) The distance $DX_{k,i} \rightarrow P_k$ is shorter than the distance $DX_{k,i} \rightarrow VX_i$, which follows from the coordinate consideration.
- (2) P_k is not rendered inaccessible by any of the d matrix particles residing at the $\{DX_j\}_i$, which is a simple consequence of P_k being accessible.

Characteristics (1) and (2)^{A.5} directly entail that VX_i is left accessible by the matrix particles at the $\{DX_j\}_i$ if this is the case for P_k . By the very definition of a V-vertex, there then exists no other matrix particle that is closer to VX_i than those residing at the $\{DX_j\}_i$, meaning that if the matrix particles at the $\{DX_j\}_i$ leave VX_i accessible, then VX_i is in the accessible space. Thus Lemma 2 is proven.

Lemma 2 merely makes a statement about the conditional accessibility of two *points*. In the context of the present work, however, it is of paramount importance that there also exist a connecting *path* between these two points. Therefore, we will next consider a theorem by Sastry and coworkers that in its proof implies the existence of such a path.

Theorem 4 (Sastry): Given a set of Voronoi vertices that belong to a void, the union of Delaunay cells corresponding to the Voronoi vertices completely encloses the void.

^{A.5}In the proof to Lemma 2 by Sastry et al., the important logical step in characteristic (2) goes entirely unmentioned.

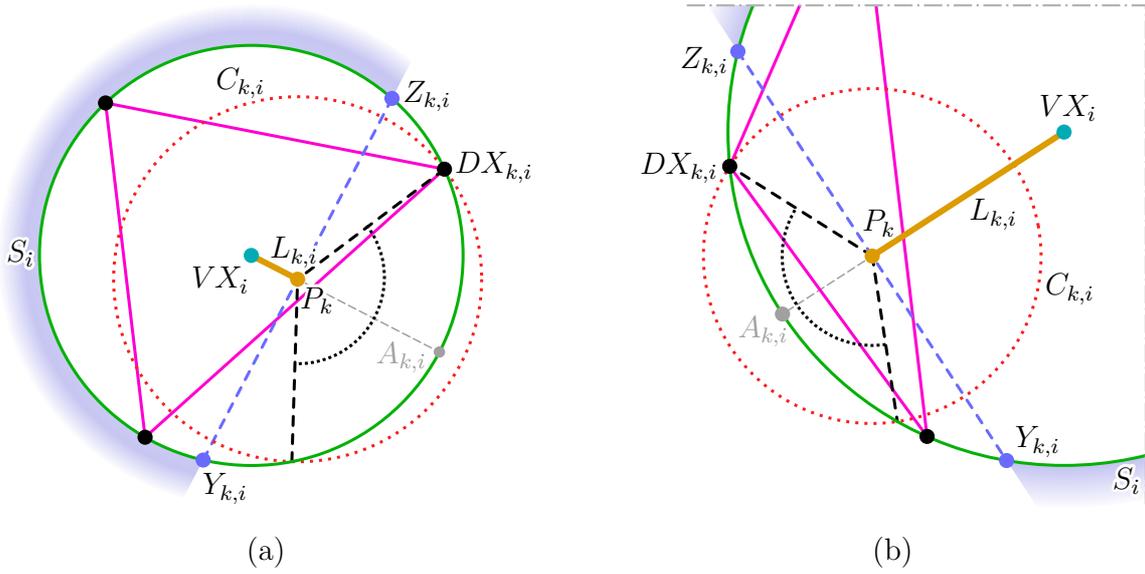


Figure A.8: Proof of Lemma 3 and Theorem 4. Panel (a): same D-cell as in Fig. A.7(a). Panel (b): enlarged section of the D-cell in Fig. A.7(b). Black disks: D-vertices (at each of which a matrix particle resides); pink lines: D-faces of D-cell DC_i ; green circle: surface S_i of DC_i 's sphere vide. Orange disk: given point P_k ; turquoise disk: DC_i 's V-vertex VX_i ; orange line: line $L_{k,i}$ connecting P_k and VX_i . Light-gray disk: point $A_{k,i}$ on S_i that is closest to P_k ; dashed light-gray line: line connecting $A_{k,i}$ and P_k . Light-blue area: points that are outside S_i and are closer to some point on $L_{k,i}$ than to P_k . Blue dashed line: line perpendicular to $L_{k,i}$ and passing through P_k ; blue disks: intersection of the latter line with S_i , marking the points of the light-blue area that are closest to $L_{k,i}$. Red dotted circle: circle around P_k that passes through $DX_{k,i}$, the D-vertex most proximate to P_k . Black dashed lines: connecting lines between P_k and the intersection points of S_i and the red circle; black dotted arc: indicator for the angle between the black dashed lines. In panel (a), the V-vertex resides inside the D-cell, whereas in panel (b), it is located outside of the D-cell.

As Sastry et al. show [229], Theorem 4 is a straightforward consequence of a specific precondition. This precondition, which we here state as a separate lemma, reads as follows:

Lemma 3: An accessible point within a Delaunay cell belongs to the same void as the Voronoi vertex of the Delaunay cell.

From Lemma 2, we already know that the V-vertex VX_i of a D-cell DC_i is accessible if there is any accessible point P_k in DC_i . Therefore, the task in proving Lemma 3 (and thus Theorem 4) is to show that there exists a connecting *path* between VX_i and every accessible P_k within DC_i . Proving Lemma 3 therefore amounts precisely to showing that the requirement formulated further above holds.

For their pertinent proof, Sastry et al. invoke the scheme of *falsification*, which in the present context means to consider the case that no path exists between P_k and VX_i

and to show that this cannot be the case. Unfortunately, while this proof scheme is perfectly valid, Sastry and coworkers execute it in an erroneous way. To understand their logical mistake, we first follow their reasoning. Consider Fig. A.8(a), which shows the same D-cell as Fig. A.7(a). Sastry et al. name the straight line $L_{k,i}$ that connects P_k and VX_i (thick orange line) as their candidate for the sought-after path. Then, they find the spatial point $A_{k,i}$ (light-gray disk) that is closest to any of the points of $L_{k,i}$ and that is—at the same time—an allowed position for another matrix particle. By the definition of the D-tessellation, $A_{k,i}$ has to be located outside (or at most on the surface) of the sphere vide corresponding to DC_i (green circle in Fig. A.8(a), cf. Sec. 2.4.2). As is easily verified, $A_{k,i}$ is constituted by the intersection point between the extension of $L_{k,i}$ beyond P_k (dashed light-gray line) and the surface S_i of the empty sphere. Clearly, $A_{k,i}$ is the point on S_i that is closest not only to P_k but to *any* point of $L_{i,j}$ (except for VX_i , which by definition is equidistant to all points of S_i). Moreover, P_k is the point of $L_{i,j}$ that is closest to $A_{k,i}$. Therefore, if a matrix particle resides at $A_{k,i}$ and P_k is accessible, $L_{k,i}$ is accessible in its entirety. This, Sastry and coworkers **incorrectly** consider as proof of Theorem 4.

The logical mistake lies in the following. Contrary to the assertion by Sastry et al., the falsification scheme actually requires to prove that *no* additional matrix particle placed *anywhere* (subject to the empty-sphere constraint) renders the path $L_{k,i}$ disconnected. However, Sastry and coworkers instead focus on merely *one possible choice* for the location of the additional matrix particle (namely one the closest to P_k) and show that in this case $L_{k,i}$ is connecting. This however, does not suffice to preclude the existence of *other* matrix particle locations for which P_k is accessible but $L_{k,i}$ is disrupted. In the following, however, we will prove that such locations indeed do not exist^{A.6}.

Suppose, as before, that P_k is the given (accessible) point within D-cell DC_i , that $L_{k,i}$ is the straight line connecting P_k and VX_i (the V-vertex associated with DC_i), and that $DX_{k,i}$ is the vertex of DC_i that is closest to P_k . To prove Lemma 3, we need to put constraints on the possible position of an additional matrix particle that would leave P_k accessible but would render $L_{k,i}$ nonconnecting. Specifically, such a position would have to be

- (I) located outside of the empty sphere of DC_i ,
- (II) farther from P_k than from some other point on $L_{k,i}$, and
- (III) far enough from P_k to leave the latter accessible.

Let us consider these conditions for a two-dimensional example. Figure A.8(a) shows the same D-cell as Fig. A.7(a), i.e., a D-cell for which the corresponding V-vertex resides inside the D-cell. As indicated by the light-blue crescent, the points that fulfill both condition (I) and (II) cover a half-plane with parts of a disk removed. The points $Z_{k,i}$ and $Y_{k,i}$ (small blue disks) mark the locations within this area that are closest to $L_{k,i}$. These points are obtained by intersecting the surface S_i of the sphere

^{A.6}The proof to Theorem 4 provided in this work is *original* to this work, i.e., published nowhere else (yet).

vide (green circle) with the line that is perpendicular to $L_{k,i}$ and passes through P_k (dashed blue line). Note that by the latter construction, $Z_{k,i}$ and $Y_{k,i}$ are on the *surface* of the light-blue area, meaning that actually it is P_k that is the point on $L_{k,i}$ closest to these two points. Strictly speaking, this is in violation of condition (II). However, to meet condition (II) it suffices to move only infinitesimally far away from those points “into” the light-blue area. Next, we show that by the above construction, the distance from P_k to $Z_{k,i}$ is always larger than that from P_k to $DX_{k,i}$. Consider a circle $C_{k,i}$ (red dotted circle) that is centered about P_k and runs through $DX_{k,i}$. Clearly, for $Z_{k,i}$ and $Y_{k,i}$ to reside inside $C_{k,i}$, the angle marked by the black dotted arc would have to be larger than 180° . However, the latter is possible only if P_k resides outside the D-cell—which is in clear contradiction to our assumption. From this it follows that if a matrix particle resides at $Z_{k,i}$ and renders any of the points of $L_{k,i}$ inaccessible, then the matrix particle located at $DX_{k,i}$ certainly renders P_k inaccessible—which again contradicts our assumption.

Figure A.8(b)—which shows a section of the D-cell in Fig. A.7(b)—evidences that the same reasoning holds if VX_i is located outside DC_i . However, if at this point we considered Lemma 3 proven, we would commit same mistake as Sastry et al. in that we would have shown merely for a particular case that $L_{k,i}$ is entirely accessible. Therefore, consider now that *none* of the candidate locations in the light-blue area is closer to P_k than $Z_{k,i}$ is (notably $Y_{k,i}$ and $Z_{k,i}$ are equidistant from P_k). Reiterating the reasoning in the previous paragraph, this means that *none* of these candidate locations can render a point on $L_{k,i}$ inaccessible without $DX_{k,i}$ rendering P_k inaccessible. This entails that conditions (I)–(III) cannot be fulfilled simultaneously, which finally proves Lemma 3 (and thus Theorem 4).

The generalization of the above argument to d dimensions is straightforward. The most significant modifications concern the set of the points that adhere to conditions (I) and (II) and are closest to the given point P_k , and the set of the points that mark the intersection of S_i and $C_{k,i}$. The points in both of these sets actually form the surface of a $(d-1)$ -dimensional hypersphere, meaning that whereas in $d = 2$ both sets encompass precisely two points, in three dimensions each set describes a *circle*.

Finally, two simple corollaries follow directly from Lemma 3 and Theorem 4, namely that

- (i) a D-cell overlaps at most one void [229], and that
- (ii) if a V-vertex is inaccessible, then no point within the corresponding D-cell is accessible.

A.1.2.4 Accessible volume in void: Delaunay = Voronoi

We will now, finally, forge the proof that the D-tessellation indeed provides a valid description of the voids in an HS-QA system (cf. Sec. 2.4.3). To achieve this, we will make use of the properties detailed in Appendix A.1.2.3 and of the findings of

Appendix A.1.2.2, the latter having been that the “sites” are identical whereas the “bonds” may differ among the D- and V-networks.

As the last prerequisite of the proof, we need to determine under which circumstances a specific bond can be different in the D- and V-networks. For this, consider a D-face DF_i and a V-edge VE_i that represent the same bond in the two networks. Following from the definition of V-edges, each point on VE_i is equidistant from the D-vertices $\{DV_j\}_i$ that define DF_i . The point M_i that is equidistant from and closest to the $\{DV_j\}_i$ is obtained by intersecting DF_i with the straight line that VE_i is a segment of. According to step (2) in the beginning of Sec. 2.4.3, whether or not DF_i is crossable depends solely upon whether M_i is accessible. Theorem 3, on the other hand, entails that the connectivity of VE_i is determined by M_i only if VE_i intersects DF_i . If such an intersection is *not* the case, then whether or not VE_i is connecting depends upon the point $M'_i \in VE_i$ that is closest to M_i . Since M'_i is always more distant from the $\{DV_j\}_i$ than M_i , it follows that VE_i is *always* connecting if DF_i is crossable. Therefore, the only possible way for the D- and V-networks to differ is for some connecting V-edge to correspond to a noncrossable D-face.

On the basis of this finding, we are finally in the position to validate the void picture resulting from the D-tessellation. The ensuing reasoning is as follows. Suppose that the V-edge VE_{ij} links the V-vertices VX_i and VX_j , and that the corresponding D-cells DC_i and DC_j are joined by D-face DF_{ij} . Suppose further that both D-cells are accessible, and that both DC_i and DC_j do actually contain accessible volume. (We merely need to consider this case here since—as laid out in Appendix A.1.2.2—“empty” D-cells are irrelevant to the void picture.) Suppose, finally, that the V- and D-networks that the above elements are part of are identical except possibly for VE_{ij} and DF_{ij} . There are then precisely two cases for the latter two elements:

- (1) The V-edge and the D-face in question render the networks *entirely identical*, meaning that either the former is connecting and the latter is crossable or VE_{ij} is nonconnecting and DF_{ij} is noncrossable. Since we know the V-network to correctly identify all voids (Appendix A.1.1.2), the same property trivially holds for the D-network in this case.
- (2) The V-edge and the D-face in question cause the networks to be *different*. From the prerequisite established in this section’s second paragraph, we know that the only way for VE_{ij} and DF_{ij} to differ is that the former is connecting whereas the latter is noncrossable. Considering the accessible volume, by Lemma 3, any accessible point inside DC_i has to be in the same void as VX_i ; the same is true for VX_j and accessible points within DC_j . By the assumption that VE_{ij} is connecting, we know that VX_i and VX_j reside in the same void. Combining these statements, under the given conditions the accessible volume inside DC_i belongs to the same void as the accessible volume inside DC_j —irrespective of whether the D-face is crossable or not^{A.7}.

^{A.7}Interestingly, since the crossability of D-faces is physically meaningful, a noncrossable D-face DF_{ij}

In combination with corollary (i) in Appendix A.1.2.3 (“a D-cell overlaps at most one void”), the above two cases mean that the D-tessellation is capable of associating *every accessible point in the system with the correct void*. Therefore, by the methods in Secs. 2.4.3 and 2.4.5 the D-network delivers a correct representation of the void connectivity, which finally allows to reliably identify free and trapped particles.

A.2 Finite-size effects in simulations

Due to the compulsory averaging of observables over multiple matrix realizations (Sec. 2.5.1.5), the large number of investigated state points (Chapter 3), and limited computational resources, most of the simulations in this work had to be conducted on relatively-small hard-sphere quenched-annealed (HS-QA) systems. Unfortunately, it is possible that observables in small systems do not (as usually intended) resemble the same observables in larger systems, the cause being not mere statistical errors but the very size of the system. This unwanted behavior is commonly referred to as “finite-size effects” [9, 85, 218], and the purpose of this appendix is to ensure that for all of the state points considered in this work, the size of the simulated systems was sufficient to avoid the presence of such effects.

The first origin of problems in this context lies in the existence of large length scales in systems at certain state points (combinations of ϕ_m and ϕ_f). Specifically, such length scales occur whenever the packing fractions of the fluid, ϕ_f , and/or of the matrix, ϕ_m , are elevated, with the former causing large dynamic correlations in the process of dynamic arrest (Sec. 1.2.5), and the latter leading to a fractal structure of voids in the process of localization (Sec. 1.3.3.3). Since the reliability of reproducing large length scales in simulations is dependent primarily upon the *overall* size of a system, we decided to perform most simulations with the same *total* number of particles, $N = N_m + N_f$. This decision is reasonable also considering the fact that the computational effort for advancing an HS-QA system by a certain amount of simulated time depends predominantly upon N for a wide range of state points (see Appendix A.3.2). As a trade-off between admissible program run time and desired system size, we opted for the specific value $N = 1000$ (cf. Sec. 3.1), and it is this value of N that we shall put to a test here.

The second issue at call is the large disparity between ϕ_m and ϕ_f for a number of state points considered in Chapter 3. Upon fixing the total number of particles (see above), such a disparity entails a comparatively low value of either N_m or N_f , neither of which is desirable since it will cause an imprecise calculation of the properties pertaining to either the matrix or the fluid. Among the systems considered in this work, the most extreme case for low N_m is the state point $(\phi_m, \phi_f) = (0.050, 0.505)$, in which our choice of a constant $N = 1000$ entailed a mere $N_m = 90$. The corresponding

between two nonempty D-cells requires that the connection between the accessible volume in the D-cells somehow “bypass” DF_{ij} . This means that the path has to leave and enter the two D-cells through faces other than DF_{ij} and thereby involve *at least three* D-cells.

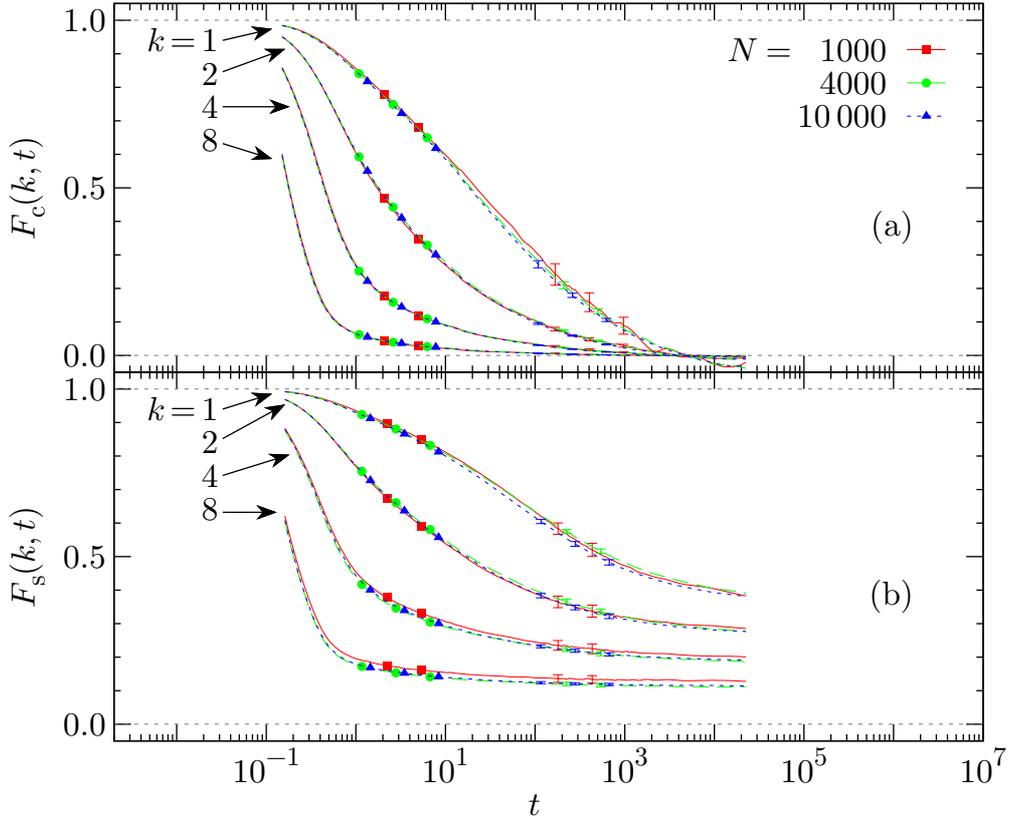


Figure A.9: Finite-size effects in intermediate scattering functions (ISFs). Assessment of the state point $(\phi_m, \phi_f) = (0.2500, 0.0125)$ close to the percolation transition of the accessible volume and to the fluid tracer case. Panel (a): connected ISF, $F_c(k, t)$; panel (b): single-particle ISF, $F_s(k, t)$. Both panels: wave vector $k \in \{1, 2, 4, 8\}$; total particle numbers $N = 1000$ (red solid line, squares), $N = 4000$ (green dashed line, circles), and $N = 10000$ (blue dotted line, triangles).

extremal case for low N_f is the state point $(\phi_m, \phi_f) = (0.3000, 0.0025)$, which for $N = 1000$ left only *eight* fluid particles to move in a matrix of 992 particles. Hence, of the problems caused by disparate packing fractions, low values of N_f is the much more severe one.

In order to curtail the latter issue, we chose to increase N whenever $N = 1000$ would effect one of the particle numbers N_m or N_f to be below the (arbitrary) limit $N^* = 50$. The previous brief analysis shows that in none of our systems, the matrix particle number is below N^* , which limits the discussion to state points for which N_f is low. In the following, we will investigate whether or not the choices of N and N^* are appropriate; this we will do by considering a state point representative of the problem of both low N_f and a large length scale in the void structure. Specifically, we will inspect various observables at $(\phi_m, \phi_f) = (0.2500, 0.0125)$, a state point close to the percolation transition of the voids (cf. Sec. 3.3) and to the tracer case $\phi_f \rightarrow 0$. Using $N = 1000$, this system attains $N_f = 48$, which is reasonably close to N^* .

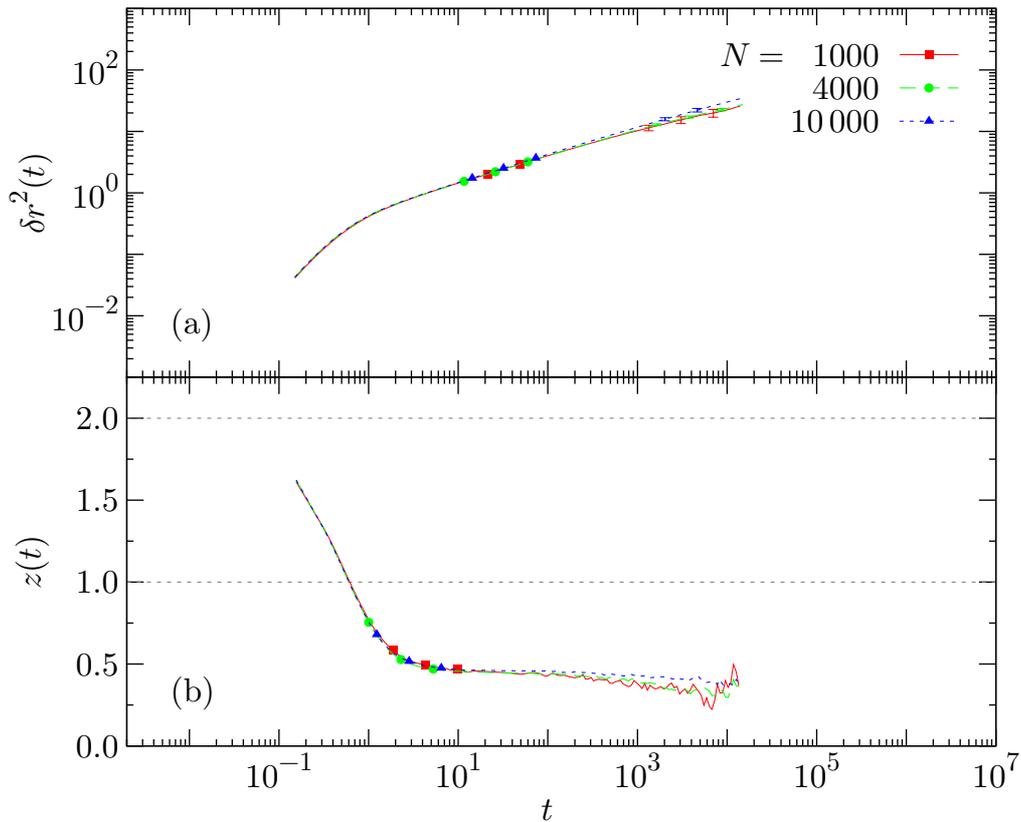


Figure A.10: Finite-size effects in mean squared displacement. Assessment of $(\phi_m, \phi_f) = (0.2500, 0.0125)$, the same state point as in Fig. A.9. Panel (a): mean squared displacement (MSD), $\delta r^2(t)$; panel (b): MSD's logarithmic derivative, $z(t)$. Both panels: $N = 1000$ (red solid line, squares), $N = 4000$ (green dashed line, circles), and $N = 10000$ (blue dotted line, triangles).

To investigate finite-size effects, we computed the quantities $F_c(k, t; N)$, $F_s(k, t; N)$, $\delta r^2(t; N)$, and $z(t; N)$, which are now taken to also depend upon the total particle number in addition to the arguments defined in Sec. 2.5.3. Specifically, we considered the cases $N = 1000$, 4000 , and 10000 , and as usual, we averaged the observables over ten matrix configurations.

Figure A.9 visualizes the dependence of two intermediate scattering functions (ISFs) upon N , namely that of the connected ISF, $F_c(k, t; N)$, and of the single-particle ISF, $F_s(k, t; N)$. The ISFs are shown for selected wave vectors, where any finite-size effects should be most prominent for the smallest k since the latter corresponds to the longest wave length. Indeed, for $k = 1$ and $t > 10^1$, there are noticeable discrepancies between the curves of $F_c(k, t; N)$ for $N = 1000$, 4000 , and 10000 ; however, this is hardly surprising considering that even for $N = 10000$, the corresponding length scale $2\pi/k = 2\pi$ is only slightly smaller than half of the edge length of the simulation box. In fact, already for $k = 2$, there is no distinguishable difference between $F_c(k, t; N=1000)$, $F_c(k, t; N=4000)$, and $F_c(k, t; N=10000)$. On the other hand, $F_s(k, t; N)$ displays small consistent discrepancies between the curves for $N = 1000$,

4000, and 10 000; however, even the worst-case deviation—which occurs at $k = 8$ and $t = t_{\max} = 2.25 \times 10^4$ —is no more than $\sim 12\%$ between $F_s(k, t; N=1000)$ and $F_s(k, t; N=10\,000)$. It is, however, not surprising that the single-particle ISF curves exhibit larger deviations from one another than the curves of the connected ISF, the reason being that $F_c(k, t; N)$ approaches zero relatively quickly whereas $F_s(k, t; N)$ does not completely relax. Also, we point out that all discrepancies in the self ISF for different N are well within the range of statistical uncertainty.

Similarly, Fig. A.10 illustrates the dependence upon N of the mean squared displacement (MSD, cf. Sec. 2.5.3.1), $\delta r^2(t; N)$, and its logarithmic derivative, $z(t; N)$. For the MSD, the differences between $\delta r^2(t; N=1000)$ and $\delta r^2(t; N=10\,000)$ for $t = t_{\max}$ amount to $\sim 24\%$; this discrepancy can be considered minor since throughout this work, we are interested in $\delta r^2(t; N)$ on *logarithmic* scales. The derivative $z(t; N)$, on the other hand, is clearly more susceptible to stochastic errors, which dominate finite-size effects for $t > 3.6 \times 10^3$ and $N = 1000$. For $t = 3.6 \times 10^3$, we find a difference of 17% between $z(t; N=1000)$ and $z(t; N=10\,000)$, which—with some care—permits quantitative interpretation.

In conclusion, our limited analysis suggests that a minimum system size of $N = 1000$ and the statistics of $N_f \geq N^*$ is sufficient to yield reasonable results for all state points and observables considered. None of the investigated quantities exhibited a deviation of more than 20% between $N = 1000$ and 10 000, which lends credibility to the results presented and the conclusions drawn in Chapter 3.

A.3 Validation and performance of numeric algorithms

A.3.1 Validation of molecular-dynamics algorithm

In computer investigations of statistical-mechanical systems, it is imperative that the core simulation algorithm generate correct data, the reason being that the validity of conclusions drawn on these data hinges on their correctness. In order to warrant the proper operation of the core program—which in this work is the molecular-dynamics (MD) algorithm of Sec. 2.2—we employed two independent approaches: monitoring of conserved quantities, and comparison with previous works.

A.3.1.1 Conserved properties

For the validation of this work's MD algorithm, we considered three quantities that are by definition conserved in event-driven MD: energy, linear momentum, and angular momentum (cf. Sec. 2.2.1.1). Formally, these three invariants can be expressed as

$$\begin{aligned} \sum_{i=1}^N (m_i/2) |\mathbf{v}_i|^2 &= \text{const}, \\ \sum_{i=1}^N m_i \mathbf{v}_i &= \text{const}, \quad \text{and} \\ \sum_{i=1}^N m_i \mathbf{r}_i \times \mathbf{v}_i &= \text{const}, \end{aligned} \tag{A.4}$$

where N is the total number of particles in the system, \mathbf{r}_i , \mathbf{v}_i , and m_i denote the location, velocity, and mass of particle i , respectively, and the symbol “ \times ” as usual denotes a cross product. The first two lines in Eq. (A.4) reflect the conditions from which the collision physics in Sec. 2.2.1.4 was derived; the conservation of angular momentum follows from the absence of external torques and provides an additional independent check. Notably, to machine precision, the properties in Eq. (A.4) are conserved *exactly* in event-driven MD.

In the case of systems with a matrix of quenched particles, the equalities in Eq. (A.4) require some modifications in order to qualify as valid checks. The problem lies in the fact that despite their special properties, the matrix particles have to be included in the respective sums. As alluded to in Sec. 2.2.1.4, mathematically it is efficient to describe the collision behavior of the matrix particles by considering the limit of an infinitely-large mass; this way, the product of mass and velocity after a collision remains well defined. In computer simulations, however, the immobility of a particle is most conveniently achieved by simply setting its velocity strictly to zero, which, unfortunately, renders its contribution to the sums in Eq. (A.4) strictly zero and thereby invalidates the conservation laws.

One way to remedy this complication is as follows. If a particle i is artificially and rigorously fixed in space, then this information can be stored as a mere flag without loss of flexibility. Any information about the velocity and mass of i is then irrelevant to the MD algorithm, meaning that the corresponding memory fields can be filled with arbitrary numbers—for instance with values that, if used in the sums in Eq. (A.4), restore the conservation properties. The most convenient way to achieve this is to fix the “mass” field of the matrix particles to some value (unity, for instance), and to then adjust their “velocity” entries such that the artificial momentum accounts for the recoil from collisions between mobile and immobile particles.

In addition to the three dynamic conservation laws in Eq. (A.4), a property “conserved” by definition in event-driven MD is the absence of overlaps between particles. Therefore, the MD code can generally be assumed to be flawed if such overlaps occur^{A.8}. The only exception to this rule is minute overlaps due to numerical round-off errors, which regularly occur on real computers (cf. Footnote 2.17). We implemented an independent overlap-check routine with a simplified version of the underlying distance evaluation—one without bucket lists—in order to increase the robustness of the validation. Unfortunately, this choice comes at the expense of $\mathcal{O}(N^2)$ operations for a single checking of the complete system for the presence of overlaps.

^{A.8}An interesting, straightforward consequence of Eqs. (2.12) and (2.16) is that the check for the next collision of a particle (Sec. 2.2.1.2) always yields a time in the “past” if the checked particle overlaps another particle. Since by construction, such a time is automatically the highest-priority item in the list of future collisions (Sec. 2.2.2.5), processing that event will cause the algorithm to run “backwards” for a short time. Due to the fact that such a reverse operation—even if brief—probably results in additional particle overlaps in the system, it is likely to mark the beginning of a sustained time reversal of the algorithm. Unfortunately, while easy and computationally cheap to identify, the start of such a backward operation is not directly informative since the causative MD algorithm error may have taken place many operations earlier.

In the context of computational expense, it is important to note that the validation criteria discussed in this section have been applied merely to representative extended test runs. In production runs, all such routines were switched off in order to enhance the algorithm's performance.

A.3.1.2 Comparison with literature

As laid out at length in Chapters 1 and 2, the primary objective of this work is to investigate the *dynamic* features (i.e., the physical time evolution) of hard-sphere systems. Since the time-dependent quantifiers of a system are highly sensitive to errors in the MD routine, we complemented the consistency checks in Appendix A.3.1.1 by a comparison of dynamic observables as obtained from sample runs of our MD algorithm and from investigations reported in the literature. We shall here illustrate selected results that are specifically concerned with the velocity auto-correlation function, an observable defined by

$$\psi(t) = \frac{1}{N} \sum_{i=1}^N \frac{\langle \mathbf{v}_i(t) \cdot \mathbf{v}_i(0) \rangle}{\langle \mathbf{v}_i(0) \cdot \mathbf{v}_i(0) \rangle}, \quad (\text{A.5})$$

where N is the number of particles, $\mathbf{v}_i(t)$ is the velocity vector of particle i at time t , and the symbol $\langle \dots \rangle$ denotes a thermodynamic average (cf. Sec. 2.5.1). The velocity auto-correlation function, like any single-particle property, is self-averaging (i.e., it can be averaged over different particles; see Sec. 2.5.3.1), which alleviates unwanted statistical errors; also, $\psi(t)$ is well investigated for hard spheres (see for instance Refs. [4, 7, 8, 285]). Since the mathematical and physical basis of correlation functions is discussed in more detail in Sec. 2.5, we shall here only focus on our effort to validate the MD algorithm.

Figure A.11 displays two distinct sets of data that both represent $\psi(t)$ in one-component hard-sphere systems at various densities. Lines, on the one hand, denote results that were obtained by Williams et al. (these data were digitally extracted from Fig. 1 in Ref. [285]); symbols, on the other hand, indicate data generated by this work's MD algorithm for the same external parameters. In both cases, the system under investigation is a collection of identical hard spheres, the only control parameter being the volume fraction ϕ_f as defined in Sec. 2.5. Notably, for all investigated values of ϕ_f , the system is in its fluid phase, and both sets of data employ the same unit of time τ as defined in Sec. 2.5.1.6, which facilitates the comparison.

As can be seen from Fig. A.11, there is an excellent agreement between the $\psi(t)$ values from Ref. [285] and from this work's MD algorithm; the slight residual differences are easily explained by the imperfect extraction of data from a published plot as well as by the small statistical errors in all data. Provided that the results from Ref. [285] are reliable themselves, the coinciding data suggest that other observables evaluated from similar runs of our MD code are valid as well.

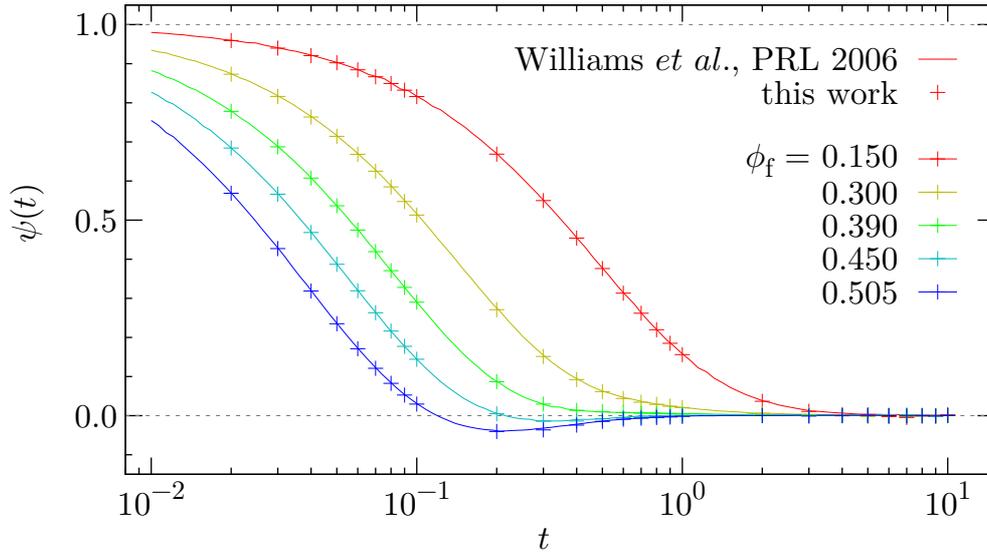


Figure A.11: Validation through comparison with literature data. Velocity auto-correlation function, $\psi(t)$, for a system of identical hard spheres at various volume fractions ϕ_f . Lines: results by Williams et al. [285] ($\sim 11\,000$ particles); symbols: data from this work’s MD algorithm (1000 particles).

A.3.2 Performance of molecular-dynamics algorithm

In Sec. 2.2.2, we asserted that the event-driven MD algorithm detailed in Sec. 2.2 requires $\mathcal{O}(N \log N)$ operations to advance a simulation with a total of $N = N_m + N_f$ particles by a fixed time interval Δt . In this section, we assess this assertion by investigating the collision rate $R_s(N, \phi_m, \phi_f)$, i.e., the number of collisions computed by the MD algorithm in one second of real time in a simulation at the state point (ϕ_m, ϕ_f) .

Figure A.12 shows R_s for systems of varying N at three different combinations of ϕ_m and ϕ_f . The first state point—the red solid line—is tantamount to a moderately dense fluid located in the quasi-bulk regime at $(\phi_m, \phi_f) = (0.050, 0.400)$; the second state point—the green solid line—contains a balanced number of matrix and fluid particles at $(\phi_m, \phi_f) = (0.100, 0.100)$; and the third state point—the blue solid line—is in the regime of the matrix dominating a dilute fluid at $(\phi_m, \phi_f) = (0.200, 0.025)$. The values for each point of the solid curves in Fig. A.12 have been obtained by measuring the time that the MD algorithm needed to complete 10^6 collisions on an Intel Xeon X5650 CPU. As for any observable, the systems have been equilibrated (albeit only up to $\delta r^2(t) = 1$ since equilibration is not as important in this context), and the values have been averaged over multiple—here: ten—matrix realizations. These realizations also form the data basis of the error bars in Fig. A.12, the latter indicating the standard error of the mean (cf. Secs. 2.5.1.5 and 2.5.5).

The first, expected feature of the MD algorithm—which is immediately evident from Fig. A.12—is its decreasing performance as N is increased. Bearing in mind that $\mathcal{O}(1)$ events per particle are required for some simulation time Δt to elapse

(cf. Sec. 2.2.2), this dependence is expected to be $R_s \propto 1/(\log N)$ for fixed (ϕ_m, ϕ_f) . However, this dependence is not quite followed by R_s for all of the depicted state points; this can be seen from a comparison with the dotted lines, each of which indicates an idealized $1/(\log N)$ law that at $N = 10^2$ coincides with the solid line of the same color. Interestingly, relative to those idealized laws the algorithm *exceeds* the expected collision rate for $10^2 < N < 10^4$, whereas for $10^4 < N < 10^6$ its performance is roughly in accord with the prediction. These findings result from two circumstances. Firstly, the MD algorithm—as any simulation program—involves operations that do not depend upon system size. Since the time fraction spent on such operations is larger in smaller systems, the algorithm performs relatively poorly for small systems, which in turn explains the apparent overachievement for moderate system sizes. The second effect is—in short—an interplay of the amount of simulation-relevant data with the sizes of the various data caches of the computing system^{A.9}. This effect entails the pronounced decrease for $N \approx 10^4$ and probably also the drop-off for $N \geq 10^6$. All in all, however, the pattern of the collision rate is consistent with a $1/(\log N)$ dependence—especially taking into account the large range of the investigated values of N .

The second feature visible from Fig. A.12 is the higher performance of the algorithm for systems with a larger proportion of matrix particles. This is hardly surprising in view of the fact that matrix particles can simply be skipped in the particle advancement (Sec. 2.2.1.3) and—partially—in the next event-search (Sec. 2.2.1.2) routines. However, the effect is not as pronounced as one might expect, with R_s increasing only approximately twofold upon decreasing the fraction of fluid particles by a factor of eight (from 89% fluid particles in the system of the red curve to 11% in the system of the blue curve). Therefore, the performance of the MD algorithm in terms of R_s can be considered approximately constant for the state points of interest to this work.

In the context of state-point dependence, however, it is due to note that R_s alone is not quite suited to indicate the “speed” of a simulation. The natural quantification

^{A.9}In order to perform computations, a computer’s CPU has to load data from some memory structure. The memory structures of modern computers are—simply put—arranged in a “cascade” along which data flow “towards” the CPU^{A.10}. Usually, the CPU’s data access is limited to the lowermost cascade level, typically called the “L1 cache”. If there is a “cache miss”—i.e., requested data are not available in that cache—, then data first have to be loaded to the L1 cache from the “L2 cache” (one level up the cascade); if the L2 cache also returns a cache miss, then data first have to be moved to the L2 cache from the “L3 cache”—and so on. This architecture is motivated by the different speeds at which data can be retrieved from the caches, with the L1 cache being fastest, and each level being *orders of magnitude* faster than the respective one above. Unfortunately, for physical and engineering reasons, faster caches are generally smaller, which entails the following effect on the performance of a program. Say a simulation A operates on an amount of data that precisely fits into the L2 cache, meaning that in this simulation, the CPU is supplied with information at the speed of the L2 cache. If a simulation B operates on *more* data than simulation A , then frequent accesses to the L3 cache are necessary. Since the L3 cache is much slower than the L2 cache, the CPU obtains data at a correspondingly slower rate, which effectively renders the computing system slower for simulation B than for simulation A . Notably, the impact of this effect is difficult to estimate a priori since it depends upon a number of factors such as the type of computations and the “intelligence” of the caches.

^{A.10}http://en.wikipedia.org/wiki/CPU_cache

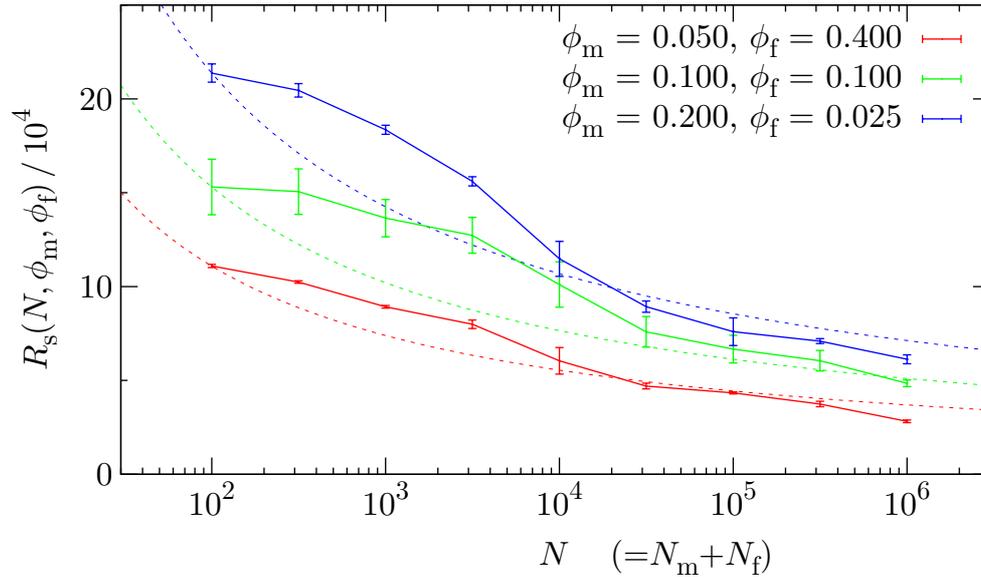


Figure A.12: Performance of the molecular-dynamics algorithm. Solid curves: number of collisions, $R_s(N, \phi_m, \phi_f)$, computed per second on an Intel Xeon X5650 CPU by this work’s event-driven MD algorithm for systems with $N = N_m + N_f$ particles in total. Dotted curves: expected $1/(\log N)$ law, adjusted to fit simulation data at $N = 10^2$. Red: quasi-bulk regime; green: equal amounts of fluid and matrix; blue: fluid dominated by matrix. Error bars: standard error of the mean (ten system realizations).

of this property is instead the ratio $\Theta = \tau/s$, i.e., the number of simulated time units that elapse per real-time second. Yet, if we denote by R_τ the average number of collisions per simulated time unit τ , then we can “recycle” R_s in the expression

$$\Theta(N, \phi_m, \phi_f) = \frac{R_s}{R_\tau}. \quad (\text{A.6})$$

Like R_s , the rate R_τ varies with the state point and with the size of the system; however, since $R_\tau(N, \phi_m, \phi_f)$ is obviously simply proportional to N , the information of interest is encoded in the dependence of R_τ upon ϕ_m and ϕ_f . It is therefore more insightful to consider the quantity $r_\tau(\phi_m, \phi_f) = R_\tau(N, \phi_m, \phi_f)/N$; and in fact, inserting this equality into Eq. (A.6) renders the factor N that we neglected in this section’s third paragraph visible. Notably, r_τ does not vary with temperature since the time unit τ is designed to render all dynamic quantities independent of temperature; this, once again, reflects the athermal nature of hard-sphere systems (cf. Secs. 1.1.3.2 and 2.5.1.6).

Figure A.13 shows the values of $r_\tau(\phi_m, \phi_f)$ for the same state points as in the kinetic diagram for the connected intermediate scattering function (Fig. 3.13). The different symbols encode different value ranges of r_τ , and the thick red solid line interpolates through points for which $r_\tau \simeq 24$. For comparison, the figure also includes lines of dynamic arrest as defined by two different criteria: the mean squared displacement $\delta r^2(t)$, shown as a green dotted line (see Sec. 3.2.1), and the connected intermediate scattering function $F_c(k, t)$, indicated by the blue dotted line (as in Sec. 2.5.3.3,

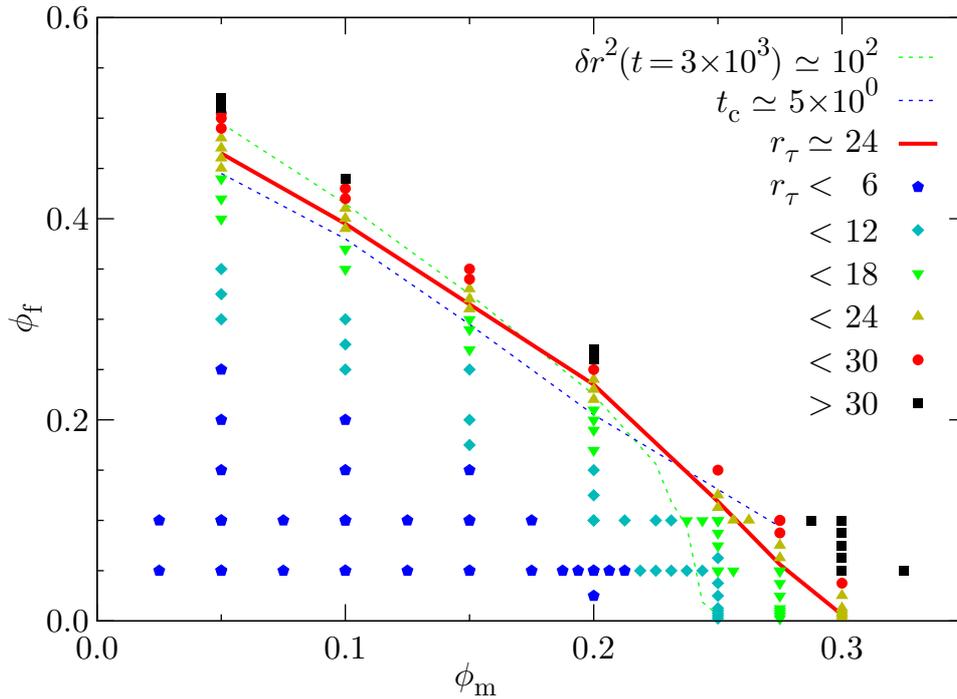


Figure A.13: Collisions per particle and per simulated time unit. Symbols: different ranges of the average number of collisions $r_\tau(\phi_m, \phi_f)$ undergone by each particle per simulation time unit τ . Thick red solid line: interpolation through points for which $r_\tau \simeq 24$. Shown for comparison are the criteria for dynamic arrest based on the mean squared displacement $\delta r^2(t)$ (green dotted line; see Sec. 3.2.1) and on the connected intermediate scattering function $F_c(k, t)$ (blue dotted line; as in Sec. 2.5.3.3, but with $t_c \simeq 5 \times 10^0$).

but with $t_c \simeq 5 \times 10^0$). As can be clearly seen from the figure, r_τ increases with both ϕ_m and ϕ_f ; this is to be expected since with every added particle—mobile or not—the mean free path between successive collisions of a fluid particle decreases. Strikingly, as ϕ_f is increased in the region $\phi_m < 0.2$, the growth of r_τ goes hand-in-hand with the approach of dynamic arrest. However, deviations from this correlation appear for $\phi_m > 0.2$, with r_τ in this regime assuming an intermediary position in that its increase is less pronounced than the slowing of the single-particle properties but more rapid than the collective slow-down. This is interesting in itself concerning the relationship between the collision rate and the pressure of a hard-sphere system^{A.11}. In any case, an unfortunate consequence of r_τ being larger in the vicinity of dynamic arrest is that in this regime not only relaxation times are large, but also relatively-long real times are required to advance a simulation by a fixed simulated time span.

^{A.11}In ordinary hard-sphere (HS) systems, the pressure is directly proportional to the rate of collisions [115]. Unfortunately, pressures in QA systems are a considerably more complicated matter [145, 224]; therefore, we did not attempt to evaluate precise values for the pressure in our HS-QA systems.

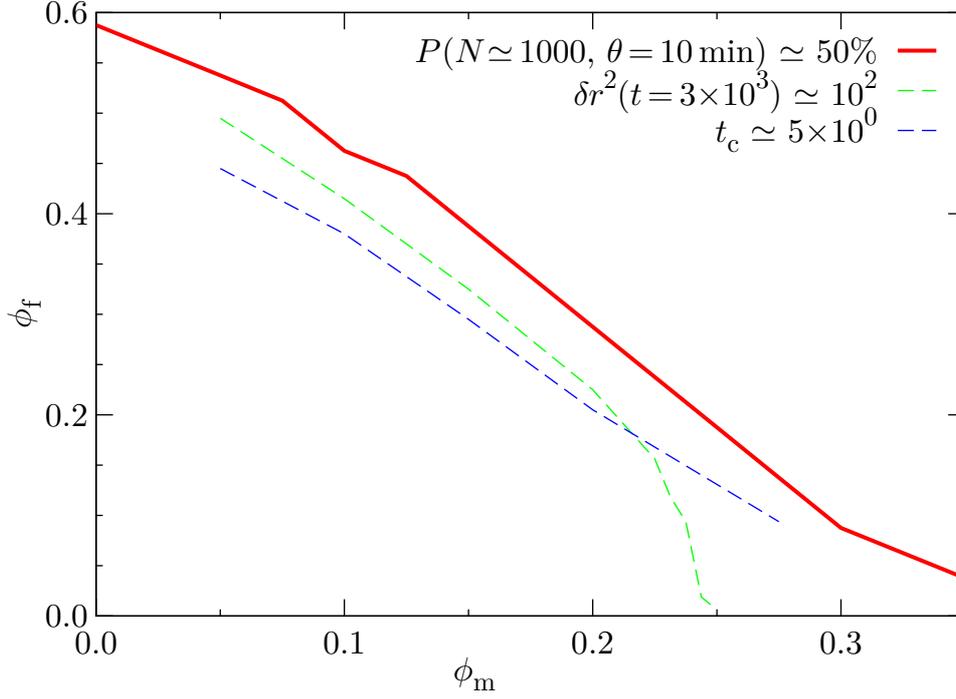


Figure A.14: Performance of the system-setup routine. Comparison of accessible system parameters with selected dynamic features of the system. Thick red solid line: probability $P(N, \theta) \simeq 50\%$ to set up a system instance with $N = N_m + N_f \simeq 1000$ particles within $\theta = 10$ CPU minutes (Intel Xeon X5650) using the custom algorithm described in Sec. 2.3.3. Upper right: $P(N, \theta) < 50\%$, lower left: $P(N, \theta) > 50\%$. Superimposed are the criteria for dynamic arrest based on the mean squared displacement $\delta r^2(t)$ (green dashed line; see Sec. 3.2.1) and on the connected intermediate scattering function $F_c(k, t)$ (blue dashed line; as in Sec. 2.5.3.3, but with $t_c \simeq 5 \times 10^0$).

A.3.3 Performance of initial-states algorithm

In Sec. 2.3.3, we introduced a custom method for establishing initial states of hard-sphere (HS) QA systems. Prior to a large-scale application of that algorithm, it is prudent to systematically investigate at which packing fractions of the matrix (ϕ_m) and of the fluid (ϕ_f) we can expect the algorithm to succeed in setting up systems. In Figure A.14, we consider the case of systems with $N = N_m + N_f \simeq 1000$ particles (the case that dominates throughout our work), and we determine whether the custom algorithm accomplishes the set-up of instances of such systems within the arbitrary time frame of $\theta = 10$ minutes when running on an Intel Xeon X5650 CPU. The thick red solid line in Fig. A.14 marks the state points at which the set-up of systems was achieved with probability $P(N \simeq 1000, \theta = 10 \text{ min}) \simeq 50\%$. To the lower left of this line, the probability to obtain systems is larger than 50%, whereas to the upper right, $P < 50\%$. For comparison, Fig. A.14 also includes the dynamic arrest criteria based on the mean squared displacement $\delta r^2(t)$ (green dashed line; see Sec. 3.2.1) and on the connected intermediate scattering function $F_c(k, t)$ (blue dashed line; as in Sec. 2.5.3.3,

but with $t_c \simeq 5 \times 10^0$).

As is evident from Fig. A.14, our setup routine generates a sufficient number of systems instances for all state points that exhibit interesting features in the *single-particle* dynamic properties. The *collective* dynamics, however, cannot be fully investigated since in the regime of large ϕ_m and moderate ϕ_f , even our highly-optimized algorithm fails to create valid HS-QA systems (see also Sec. 2.5.3.3). While this is unfortunate and unsatisfactory, the inability of our algorithm to set up systems merely reflects a fundamental feature of HS-QA systems, namely that there exist state points for which only few or even no system instances can be realized.

Notably, in practice, a probability of 50% is inadequate. Ideally, none of the setup attempts should be rejected since as little bias as possible should be exerted on the statistics. However, already at relatively-low ϕ_m , it is strictly impossible to achieve a probability $P = 100\%$ in finite systems, the reason being that there exist peculiar matrix configurations that prohibit all fluid particles to be inserted. Nonetheless, by prolonging the CPU time θ , it is possible to increase P since more theoretically possible system configurations can actually be realized. Throughout this work, only state points have been considered for which $P \geq 90\%$; typically, we allowed for $\theta = 1$ hour to achieve this threshold.

A.3.4 Validation of Delaunay-tessellation algorithm

Since in this work, a number of conclusions were drawn on the basis of a separation of QA fluids into free and trapped particles (see Sec. 3.4), it is essential that the algorithm in Sec. 2.4.4 generate flawless Delaunay tessellations. To ascertain this, we pursued three different approaches, which we shall illustrate in the following.

The first of our methods is to reverify that each D-cell in a Delaunay tessellation is indeed associated with a *sphère vide* (cf. Sec. 2.4.2). In order for this verification to be somewhat independent and more robust, we reimplemented the particle distance evaluation that underlies the emptiness check; like in Sec. A.3.1.1, this is accomplished by looping over all particles instead of using bucket lists. Since this procedure requires $\mathcal{O}(N^2)$ operations to validate all D-cells, this check was conducted only for a limited number of representative test runs.

The second method is based on the fact that in a Delaunay tessellation under periodic boundary conditions (PBCs, see Sec. 2.2.2.1), each d -dimensional D-cell is formed by precisely $d+1$ distinct D-faces, and that—regardless of dimensionality—a D-face always borders two distinct D-cells. Checking these two features for all D-faces and D-cells is straightforward if the Delaunay tessellation is available as a whole. A simple, easily-verifiable consequence of these properties is the fact that a Delaunay tessellation under PBCs always contains $(d+1)/2$ times as many D-faces as D-cells.

The third method is based on the so-called “Euler characteristic”^{A.12}, a rule for the number of constituent elements of constructs like hyperpolyhedra, graphs, and

^{A.12}<http://mathworld.wolfram.com/EulerCharacteristic.html>

tessellations. A construct of dimensionality d contains elements for each dimensionality from the range $\{d, d-1, \dots, 0\}$; for example, a two-dimensional Delaunay tessellation consists of D-cells ($d = 2$), D-edges ($d = 1$), and D-vertices ($d = 0$). Denoting the number of elements of dimension d by the symbol E_d , the Euler characteristic states that the elements of a construct obey the relation^{A.12}

$$\Pi_d = \sum_{i=0}^d (-1)^i E_i = 1 - (-1)^{d+1} - 2g \quad , \quad (\text{A.7})$$

which in the specific cases $d = 2$ and $d = 3$ reduces to

$$\Pi_2 = E_0 - E_1 + E_2 = 2 - 2g \quad \text{and} \quad (\text{A.8})$$

$$\Pi_3 = E_0 - E_1 + E_2 - E_3 = -2g \quad . \quad (\text{A.9})$$

The symbol g in the above equations denotes the so-called ‘‘genus’’^{A.13}, a quantity that essentially encodes the number of holes in the construct of interest. As laid out in Sec. 2.2.2.1 and Fig. 2.7 therein, PBCs render the topology of a simulated d -dimensional space to be that of a $(d+1)$ -dimensional torus; since for a three-dimensional torus the genus is $g_{\text{torus}}^{d=3} = 1$, the right-hand side of Eq. (A.8) is zero. The interested reader may verify Eq. (A.8) with $g = g_{\text{torus}}^{d=3}$ for instance for Fig. A.5(a), the latter containing 12 D-vertices, 36 D-edges, and 24 D-cells.

Unfortunately, this work’s author did not succeed in discovering literature that would state the genus of a four-dimensional torus as defined by a cube under PBCs. However, a strong indication for Eq. (A.9) to be fulfilled is if Π_3 takes the same value for each sampled Delaunay tessellation. Empirically, in each investigated case we found that $g_{\text{torus}}^{d=4} = 0$, which unfortunately is at odds with the expectation that a four-dimensional torus has at least one hole, i.e., that $g_{\text{torus}}^{d=4} > 0$.

Verifying the Euler characteristic and the number of D-cells adjacent to D-faces (and vice versa) is computationally cheap. Therefore, these checks have been performed for every Delaunay tessellation constructed by the algorithm in Sec. 2.4.4.4.

A.3.5 Performance of Delaunay-tessellation algorithm

In this section, we briefly investigate the performance of the algorithm outlined in Sec. 2.4.4.4 for constructing Delaunay tessellations. The quantification of the performance is conducted in terms of R_s , the number of Delaunay cells constructed per second of real time. R_s may vary with the size of the system—here measured by the number of matrix particles N_m —and with ϕ_m , the packing fraction of the matrix. The latter dependence is nonnegligible since ϕ_m affects the uniformity of the particle distribution, with nonuniform distributions containing comparatively large Delaunay cells with correspondingly large circumhyperspheres. In order to verify the empty-sphere property for such D-cells, a relatively-large number of buckets need to be considered

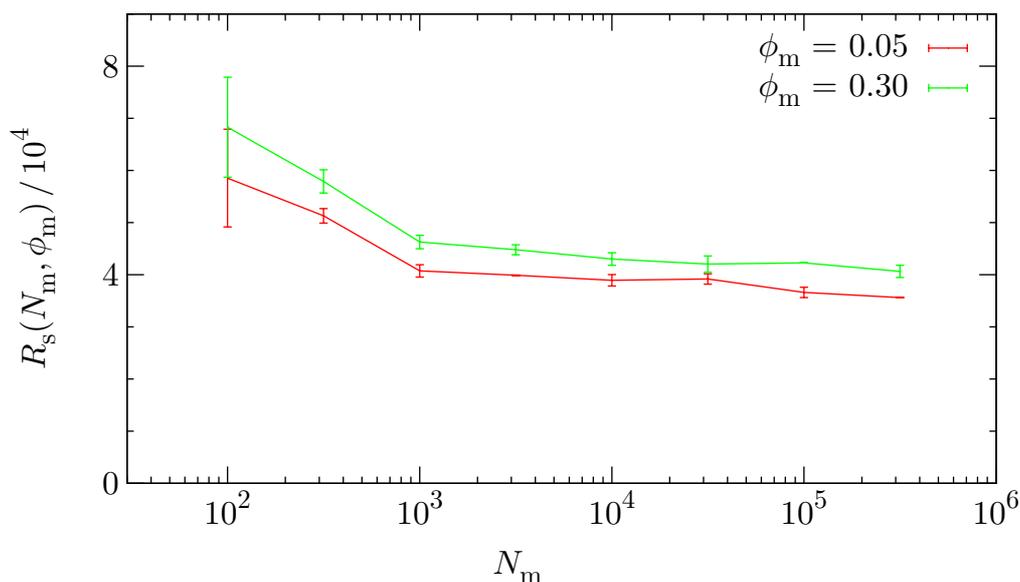


Figure A.15: Performance of the Delaunay-tessellation algorithm. Number of constructed Delaunay cells per second, $R_s(N_m, \phi_m)$, for $\phi_m = 0.05$ (red curve) and $\phi_m = 0.30$ (green curve) at various matrix particle numbers N_m . Error bars indicate the standard error of the mean over ten different system realizations.

in the nearest-neighbor search (cf. Sec. 2.4.4.3), which has repercussions on the algorithm’s performance.

In Fig. A.15, the rate R_s is considered for two different values of the matrix packing fraction—one at the lower and one at the upper end of the ϕ_m values of interest to this work (cf. Chapter 3). Each depicted value represents an average over ten different realizations, and for every realized system, a complete Delaunay tessellation has been constructed. The first observation from the figure is that R_s is essentially independent of ϕ_m for the QA systems of interest; this can be seen from the difference of corresponding values on the red and green curves, which in no case is more than 20%. Concerning the dependence upon system size, the figure shows that for $N_m \geq 10^3$ the performance of the tessellation algorithm is virtually independent of N_m . This is in accordance with the assertion in the end of Sec. 2.4.4.4 that the algorithm requires $\mathcal{O}(1)$ operations to construct one D-cell.

However, deviations from this behavior occur for the (relatively-irrelevant) cases of very small systems; in this N_m range, the algorithm surprisingly yields considerably-*higher* D-cell construction rates than for larger N_m . This is contrary to the expectation that for small systems, the algorithm perform sub-par due to the large time fraction spent on system size-independent operations (cf. Sec. A.3.2). Similar to the event-driven MD algorithm, however, there is also the effect that the amount of data relevant to the tessellation algorithm interplays with the capacities of the computer’s various memory containers (see again Appendix A.3.2 and Footnote A.9 therein). Taking into

A.13 <http://mathworld.wolfram.com/Genus.html>

account that the combined number of D-cells and D-faces exceeds N_m by one and a half orders of magnitude (see Sec. 3.3.1) and that the data structures for Delaunay tessellation elements and for MD-simulation particles require comparable amounts of memory, the sharp performance decrease of the MD algorithm at $3 \times 10^3 < N < 3 \times 10^4$ (cf. Fig. A.12) suggests that such a drop-off occur at $10^2 < N_m < 10^3$ for the tessellation algorithm—which is precisely what is seen in Fig. A.15.

References

- [1] ADAM G and GIBBS JH. *On the Temperature Dependence of Cooperative Relaxation Properties in Glass-Forming Liquids*. J. Chem. Phys. **43**, 139 (1965).
- [2] AKIMOTO T and MIYAGUCHI T. *Role of infinite invariant measure in deterministic subdiffusion*. Phys. Rev. E **82**, 030102 (2010).
- [3] ALCOUHLABI M and MCKENNA GB. *Effects of confinement on material behaviour at the nanometre size scale*. J. Phys.: Condens. Matter **17**, R461 (2005).
- [4] ALDER BJ, GASS DM, and WAINWRIGHT TE. *Studies in Molecular Dynamics. VIII. The Transport Coefficients for a Hard-Sphere Fluid*. J. Chem. Phys. **53**, 3813 (1970).
- [5] ALDER BJ and WAINWRIGHT TE. *Phase Transition for a Hard Sphere System*. J. Chem. Phys. **27**, 1208 (1957).
- [6] ALDER BJ and WAINWRIGHT TE. *Studies in Molecular Dynamics. I. General Method*. J. Chem. Phys. **31**, 459 (1959).
- [7] ALDER BJ and WAINWRIGHT TE. *Velocity Autocorrelations for Hard Spheres*. Phys. Rev. Lett. **18**, 988 (1967).
- [8] ALDER BJ and WAINWRIGHT TE. *Decay of the Velocity Autocorrelation Function*. Phys. Rev. A **1**, 18 (1970).
- [9] ALLEN MP and TILDESLEY DJ. *Computer Simulation of Liquids* (Oxford University Press, Oxford, 1987).
- [10] ÁLVAREZ M, LEVESQUE D, and WEIS JJ. *Monte Carlo approach to the gas-liquid transition in porous materials*. Phys. Rev. E **60**, 5495 (1999).
- [11] ANGELL CA. *Formation of Glasses from Liquids and Biopolymers*. Science **267**, 1924 (1995).
- [12] ANONYMOUS. *What is random packing?* Nature **239**, 488 (1972).
- [13] ANTL L, GOODWIN J, HILL R, OTTEWILL R, OWENS S, PAPWORTH S, and WATERS J. *The preparation of poly(methyl methacrylate) latices in non-aqueous media*. Colloids and Surfaces **17**, 67 (1986).
- [14] ASHKIN A, DZIEDZIC JM, BJORKHOLM JE, and CHU S. *Observation of a single-beam gradient force optical trap for dielectric particles*. Opt. Lett. **11**, 288 (1986).
- [15] AURENHAMMER F. *Voronoi diagrams – a survey of a fundamental geometric data structure*. ACM Comput. Surv. **23**, 345 (1991).
- [16] BABU S, GIMEL JC, and NICOLAI T. *Tracer Diffusion in Colloidal Gels*. J. Phys. Chem. B **112**, 743 (2008).
- [17] BÉNICHOU O, CHEVALIER C, KLAFTER J, MEYER B, and VOITURIEZ R. *Geometry-controlled kinetics*. Nat. Chem. **2**, 472 (2010).

- [18] BAUER T, HÖFLING F, MUNK T, FREY E, and FRANOSCH T. *The localization transition of the two-dimensional Lorentz model*. Eur. Phys. J. Spec. Top. **189**, 103 (2010).
- [19] BAYER M, BRADER JM, EBERT F, FUCHS M, LANGE E, MARET G, SCHILLING R, SPERL M, and WITTMER JP. *Dynamic glass transition in two dimensions*. Phys. Rev. E **76**, 011508 (2007).
- [20] BENGZELIUS U, GOTZE W, and SJOLANDER A. *Dynamics of supercooled liquids and the glass transition*. J. Phys. C: Solid State Phys. **17**, 5915 (1984).
- [21] BENNETT CH. *Serially Deposited Amorphous Aggregates of Hard Spheres*. J. Appl. Phys. **43**, 2727 (1972).
- [22] BERNAL JD. *Geometry of the Structure of Monatomic Liquids*. Nature **185**, 68 (1960).
- [23] BERNE BJ and PECORA R. *Dynamic light scattering: with applications to chemistry, biology, and physics* (Wiley, New York, 1976).
- [24] BERRY H and CHATÉ H. *Anomalous subdiffusion due to obstacles: A critical survey*. arXiv:1103.2206 (2011).
- [25] BERTHIER L. *Dynamic heterogeneity in amorphous materials*. Physics **4**, 42 (2011).
- [26] BERTHIER L and BIROLI G. *Theoretical perspective on the glass transition and amorphous materials*. Rev. Mod. Phys. **83**, 587 (2011).
- [27] BERTHIER L, BIROLI G, BOUCHAUD JP, CIPELLETTI L, EL MASRI D, L'HÔTE D, LADIEU F, and PIERNO M. *Direct Experimental Evidence of a Growing Length Scale Accompanying the Glass Transition*. Science **310**, 1797 (2005).
- [28] BERTHIER L and KOB W. *The Monte Carlo dynamics of a binary Lennard-Jones glass-forming mixture*. J. Phys.: Condens. Matter **19**, 205130 (2007).
- [29] BERTHIER L and KOB W. *Static point-to-set correlations in glass-forming liquids*. Phys. Rev. E **85**, 011102 (2012).
- [30] BIANCHI E, BLAAK R, and LIKOS CN. *Patchy colloids: state of the art and perspectives*. Phys. Chem. Chem. Phys. **13**, 6397 (2011).
- [31] BINDER K and KOB W. *Glassy materials and disordered solids: an introduction to their statistical mechanics* (World Scientific, New Jersey, 2005).
- [32] BINDER K and STAUFFER D. *Statistical theory of nucleation, condensation and coagulation*. Adv. Phys. **25**, 343 (1976).
- [33] BINDER K and YOUNG AP. *Spin glasses: Experimental facts, theoretical concepts, and open questions*. Rev. Mod. Phys. **58**, 801 (1986).
- [34] BIROLI G, BOUCHAUD JP, CAVAGNA A, GRIGERA TS, and VERROCCHIO P. *Thermodynamic signature of growing amorphous order in glass-forming liquids*. Nat. Phys. **4**, 771 (2008).
- [35] BORN M and VON KARMAN T. *Über Schwingungen in Raumgittern*. Phys. Z. **13**, 297 (1912).
- [36] BRAMBILLA G, EL MASRI D, PIERNO M, BERTHIER L, CIPELLETTI L, PETEKIDIS G, and SCHOFIELD AB. *Probing the Equilibrium Dynamics of Colloidal Hard Spheres above the Mode-Coupling Glass Transition*. Phys. Rev. Lett. **102**, 085703 (2009).
- [37] BROSTOW W, DUSSAULT JP, and FOX BL. *Construction of Voronoi polyhedra*. J. Comput. Phys. **29**, 81 (1978).
- [38] BROWN R. *A brief account of microscopical observations made in the months of June, July and August 1827, on the particles contained in the pollen of plants; and on the general existence of active molecules in organic and inorganic bodies*. Phil. Mag. **4**, 161 (1828).

- [39] BRYANT S and BLUNT M. *Prediction of relative permeability in simple porous media*. Phys. Rev. A **46**, 2004 (1992).
- [40] CAIRNS RJR, OTTEWILL RH, OSMOND DWJ, and WAGSTRAFF I. *Studies on the preparation and properties of lattices in nonpolar media*. J. Colloid Interface Sci. **54**, 45 (1976).
- [41] CAMMAROTA C and BIROLI G. *Aging and relaxation near random pinning glass transitions*. EPL **98**, 16011 (2012).
- [42] CAMMAROTA C and BIROLI G. *Ideal glass transitions by random pinning*. Proc. Natl. Acad. Sci. USA **109**, 8850 (2012).
- [43] CANDELIER R, WIDMER-COOPER A, KUMMERFELD JK, DAUCHOT O, BIROLI G, HARROWELL P, and REICHMAN DR. *Spatiotemporal Hierarchy of Relaxation Events, Dynamical Heterogeneities, and Structural Reorganization in a Supercooled Liquid*. Phys. Rev. Lett. **105**, 135702 (2010).
- [44] CARNAHAN NF and STARLING KE. *Equation of State for Nonattracting Rigid Spheres*. J. Chem. Phys. **51**, 635 (1969).
- [45] CHANG R, JAGANNATHAN K, and YETHIRAJ A. *Diffusion of hard sphere fluids in disordered media: A molecular dynamics simulation study*. Phys. Rev. E **69**, 051101 (2004).
- [46] CHARBONNEAU B, CHARBONNEAU P, and TARJUS G. *Geometrical Frustration and Static Correlations in a Simple Glass Former*. Phys. Rev. Lett. **108**, 035701 (2012).
- [47] CHARBONNEAU P and TARJUS G. *Decorrelation of the static and dynamic length scales in hard-sphere glass formers*. Phys. Rev. E **87**, 042305 (2013).
- [48] CHAREYRE B, CORTIS A, CATALANO E, and BARTHÉLEMY E. *Pore-Scale Modeling of Viscous Flow and Induced Forces in Dense Sphere Packings*. Transport Porous Med. **92**, 473 (2011).
- [49] CHÁVEZ-ROJO MA, JUÁREZ-MALDONADO R, and MEDINA-NOYOLA M. *Diffusion of colloidal fluids in random porous media*. Phys. Rev. E **77**, 040401 (2008).
- [50] CHO HW, KWON G, SUNG BJ, and YETHIRAJ A. *Effect of Polydispersity on Diffusion in Random Obstacle Matrices*. Phys. Rev. Lett. **109**, 155901 (2012).
- [51] CLARKE AS and JÓNSSON H. *Structural changes accompanying densification of random hard-sphere packings*. Phys. Rev. E **47**, 3975 (1993).
- [52] COHEN MH and TURNBULL D. *Molecular Transport in Liquids and Glasses*. J. Chem. Phys. **31**, 1164 (1959).
- [53] COSLOVICH D. *Connections between structure, dynamics, and energy landscape in simple models of glass-forming liquids*. Ph.D. thesis, Università di Trieste, Trieste (2008).
- [54] COSLOVICH D. *Locally preferred structures and many-body static correlations in viscous liquids*. Phys. Rev. E **83**, 051505 (2011).
- [55] COSLOVICH D, STRAUSS L, and KAHL G. *Hopping and microscopic dynamics of ultrasoft particles in cluster crystals*. Soft Matter **7**, 2127 (2011).
- [56] CROCKER JC and GRIER DG. *Methods of Digital Video Microscopy for Colloidal Studies*. J. Colloid Interface Sci. **179**, 298 (1996).
- [57] CRUZ DE LEÓN G and ARAUZ-LARA JL. *Static structure and colloidal interactions in partially quenched quasibidimensional colloidal mixtures*. Phys. Rev. E **59**, 4203 (1999).
- [58] CRUZ DE LEÓN G, SAUCEDO-SOLORIO JM, and ARAUZ-LARA JL. *Colloidal Interactions in Partially Quenched Suspensions of Charged Particles*. Phys. Rev. Lett. **81**, 1122 (1998).

- [59] DE GENNES PG. *Soft matter*. Rev. Mod. Phys. **64**, 645 (1992).
- [60] DE GENNES PG and PROST J. *The Physics of Liquid Crystals* (Oxford University Press, Oxford, 1993).
- [61] DE MICHELE C. *Optimizing event-driven simulations*. Comput. Phys. Commun. **182**, 1846 (2011).
- [62] DEBENEDETTI PG and STILLINGER FH. *Supercooled liquids and the glass transition*. Nature **410**, 259 (2001).
- [63] DEL GADO E, ILG P, KRÖGER M, and ÖTTINGER HC. *Nonaffine Deformation of Inherent Structure as a Static Signature of Cooperativity in Supercooled Liquids*. Phys. Rev. Lett. **101**, 095501 (2008).
- [64] DELAUNAY B. *Sur la sphère vide. A la mémoire de Georges Voronoï*. Bull. Acad. Sci. USSR VII: Class Math. Nat. Sci. **6**, 793 (1934).
- [65] DIRICHLET GL. *Über die Reduction der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen*. J. Reine Angew. Math. **1850**, 209 (1850).
- [66] DONATI C, DOUGLAS JF, KOB W, PLIMPTON SJ, POOLE PH, and GLOTZER SC. *Stringlike Cooperative Motion in a Supercooled Liquid*. Phys. Rev. Lett. **80**, 2338 (1998).
- [67] DONG W, KRAKOVIACK V, and ZHAO SL. *Fluids Confined in Porous Media: A Soft-Sponge Model*. J. Phys. Chem. C **111**, 15910 (2007).
- [68] DULLIEN FAL. *Porous media: fluid transport and pore structure* (Academic Press, New York, 1992), second edition.
- [69] DWYER RA. *A Faster Divide-and-Conquer Algorithm for Constructing Delaunay Triangulations*. Algorithmica **2**, 137 (1987).
- [70] DWYER RA. *Higher-Dimensional Voronoi Diagrams in Linear Expected Time*. Discrete Comput. Geom. **6**, 343 (1991).
- [71] EDIGER MD, ANGELL CA, and NAGEL SR. *Supercooled Liquids and Glasses*. J. Phys. Chem. **100**, 13200 (1996).
- [72] EDIGER MD and HARROWELL P. *Perspective: Supercooled liquids and glasses*. J. Chem. Phys. **137**, 080901 (2012).
- [73] EDWARDS SF and ANDERSON PW. *Theory of spin glasses*. J. Phys. F: Metal Phys. **5**, 965 (1975).
- [74] EINSTEIN A. *Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen*. Ann. Phys. **322**, 549 (1905).
- [75] FENZ W, MRYGLOD IM, PRYTULA O, and FOLK R. *Concentration and mass dependence of transport coefficients and correlation functions in binary mixtures with high mass asymmetry*. Phys. Rev. E **80**, 021202 (2009).
- [76] FIEGE A, ASPELMEIER T, and ZIPPELIUS A. *Long-Time Tails and Cage Effect in Driven Granular Fluids*. Phys. Rev. Lett. **102**, 098001 (2009).
- [77] FINNEY JL. *Random Packings and the Structure of Simple Liquids. I. The Geometry of Random Close Packing*. Proc. R. Soc. Lond. A **319**, 479 (1970).
- [78] FLENNER E and SZAMEL G. *Dynamic Heterogeneity in a Glass Forming Fluid: Susceptibility, Structure Factor, and Correlation Length*. Phys. Rev. Lett. **105**, 217801 (2010).

- [79] FLENNER E, ZHANG M, and SZAMEL G. *Analysis of a growing dynamic length scale in a glass-forming binary hard-sphere mixture*. Phys. Rev. E **83**, 051501 (2011).
- [80] FOFFI G, DE MICHELE C, SCIORTINO F, and TARTAGLIA P. *Scaling of Dynamics with the Range of Interaction in Short-Range Attractive Colloids*. Phys. Rev. Lett. **94**, 078301 (2005).
- [81] FORREST JA and DALNOKI-VERESS K. *The glass transition in thin polymer films*. Adv. Colloid Interface Sci. **94**, 167 (2001).
- [82] FORTUNE S. *A sweepline algorithm for Voronoi diagrams*. Algorithmica **2**, 153 (1987).
- [83] FRANOSCH T, SPANNER M, BAUER T, SCHRÖDER-TURK GE, and HÖFLING F. *Space-resolved dynamics of a tracer in a disordered solid*. J. Non-Cryst. Solids **357**, 472 (2011).
- [84] FRENKEL D. *Playing Tricks with Designer “Atoms”*. Science **296**, 65 (2002).
- [85] FRENKEL D and SMIT B. *Understanding Molecular Simulation: From Algorithms to Applications* (Academic Press, San Diego, 2002).
- [86] FULCHER GS. *Analysis of recent measurements of the viscosity of glasses*. J. Amer. Ceram. Soc. **8**, 339 (1925).
- [87] GALLO P, ATILI A, and ROVERE M. *Mode-coupling behavior of a Lennard-Jones binary mixture upon increasing confinement*. Phys. Rev. E **80**, 061502 (2009).
- [88] GALLO P, PELLARIN R, and ROVERE M. *Mode Coupling relaxation scenario in a confined glass former*. EPL **57**, 212 (2002).
- [89] GALLO P, PELLARIN R, and ROVERE M. *Slow dynamics of a confined supercooled binary mixture: Direct space analysis*. Phys. Rev. E **67**, 041202 (2003).
- [90] GALLO P, PELLARIN R, and ROVERE M. *Slow dynamics of a confined supercooled binary mixture. II. Q space analysis*. Phys. Rev. E **68**, 061209 (2003).
- [91] GALLO P and ROVERE M. *Lennard-Jones binary mixture in disordered matrices: exploring the mode coupling scenario at increasing confinement*. J. Phys.: Condens. Matter **23**, 234118 (2011).
- [92] GASSER U, WEEKS ER, SCHOFIELD A, PUSEY PN, and WEITZ DA. *Real-Space Imaging of Nucleation and Growth in Colloidal Crystallization*. Science **292**, 258 (2001).
- [93] GÖTZE W. *Recent tests of the mode-coupling theory for glassy dynamics*. J. Phys.: Condens. Matter **11**, A1 (1999).
- [94] GAUSS CF. *Recension der „Untersuchungen über die Eigenschaften der positiven ternären quadratischen Formen von Ludwig August Seeber, Dr. der Philosophie, ordentl. Professor an der Universität in Freiburg. 1831. 248 S. in 4“*. J. Reine Angew. Math. **1840**, 312 (1840).
- [95] GELB LD, GUBBINS KE, RADHAKRISHNAN R, and SLIWINSKA-BARTKOWIAK M. *Phase separation in confined systems*. Rep. Prog. Phys. **62**, 1573 (1999).
- [96] GIMEL JC and NICOLAI T. *Self-diffusion of non-interacting hard spheres in particle gels*. J. Phys.: Condens. Matter **23**, 234115 (2011).
- [97] GIVEN JA. *Liquid-state methods for random media: Random sequential adsorption*. Phys. Rev. A **45**, 816 (1992).
- [98] GIVEN JA and STELL G. *Ornstein–Zernike formalism for reaction rates in random media*. J. Chem. Phys. **94**, 3060 (1991).
- [99] GIVEN JA and STELL G. *Comment on: Fluid distributions in two-phase random media: Arbitrary matrices*. J. Chem. Phys. **97**, 4573 (1992).

- [100] GIVEN JA and STELL GR. *The replica Ornstein–Zernike equations and the structure of partly quenched media*. Physica A **209**, 495 (1994).
- [101] GLEITER H. *Nanostructured materials: basic concepts and microstructure*. Acta Mater. **48**, 1 (2000).
- [102] GLOTZER SC and SOLOMON MJ. *Anisotropy of building blocks and their assembly into complex structures*. Nat. Mater. **6**, 557 (2007).
- [103] GOLDSTEIN M. *Viscous Liquids and the Glass Transition: A Potential Energy Barrier Picture*. J. Chem. Phys. **51**, 3728 (1969).
- [104] GOTZE W and SJOGREN L. *Relaxation processes in supercooled liquids*. Rep. Prog. Phys. **55**, 241 (1992).
- [105] GRAHAM T. *Liquid Diffusion Applied to Analysis*. Philos. T. R. Soc. **151**, 183 (1861).
- [106] GREEN MS. *Markoff Random Processes and the Statistical Mechanics of Time-Dependent Phenomena. II. Irreversible Processes in Fluids*. J. Chem. Phys. **22**, 398 (1954).
- [107] GREENFIELD ML and THEODOROU DN. *Geometric analysis of diffusion pathways in glassy and melt atactic polypropylene*. Macromolecules **26**, 5461 (1993).
- [108] GREINER W, NEISE L, and STÖCKER H. *Thermodynamik und Statistische Mechanik* (Deutsch, Frankfurt am Main, 1993), second editon.
- [109] GRIER DG. *A revolution in optical manipulation*. Nature **424**, 810 (2003).
- [110] HÖFLING F, BAMBERG KU, and FRANOSCH T. *Anomalous transport resolved in space and time by fluorescence correlation spectroscopy*. Soft Matter **7**, 1358 (2011).
- [111] HÖFLING F and FRANOSCH T. *Crossover in the Slow Decay of Dynamic Correlations in the Lorentz Model*. Phys. Rev. Lett. **98**, 140601 (2007).
- [112] HÖFLING F and FRANOSCH T. *Anomalous transport in the crowded world of biological cells*. Rep. Prog. Phys. **76**, 046602 (2013).
- [113] HÖFLING F, FRANOSCH T, and FREY E. *Localization Transition of the Three-Dimensional Lorentz Model and Continuum Percolation*. Phys. Rev. Lett. **96**, 165901 (2006).
- [114] HALES TC, HARRISON J, MCCLAUGHLIN S, NIPKOW T, OBUA S, and ZUMKELLER R. *A revision of the proof of the Kepler conjecture*. arXiv:0902.0350 (2009).
- [115] HANSEN JP and McDONALD IR. *Theory of Simple Liquids* (Academic Press, London, 1986), second editon.
- [116] HANSEN JP and VERLET L. *Phase Transitions of the Lennard-Jones System*. Phys. Rev. **184**, 151 (1969).
- [117] HEDGES LO, JACK RL, GARRAHAN JP, and CHANDLER D. *Dynamic Order–Disorder in Atomistic Models of Structural Glass Formers*. Science **323**, 1309 (2009).
- [118] HELLMANN M, HEERMANN DW, and WEISS M. *Anomalous reaction kinetics and domain formation on crowded membranes*. EPL **94**, 5 (2011).
- [119] HELLMANN M, KLAFTER J, HEERMANN DW, and WEISS M. *Challenges in determining anomalous diffusion in crowded fluids*. J. Phys.: Condens. Matter **23**, 234113 (2011).
- [120] HENCH LL and WEST JK. *The Sol–Gel Process*. Chem. Rev. **90**, 33 (1990).
- [121] HORBACH J, VOIGTMANN T, HÖFLING F, and FRANOSCH T. *Localization phenomena in models of ion-conducting glass formers*. Eur. Phys. J. Spec. Top. **189**, 141 (2010).

- [122] HOSHEN J and KOPELMAN R. *Percolation and cluster distribution. I. Cluster multiple labeling technique and critical concentration algorithm.* Phys. Rev. B **14**, 3438 (1976).
- [123] HUNTER GL and WEEKS ER. *The physics of the colloidal glass transition.* Rep. Prog. Phys. **75**, 066501 (2012).
- [124] HUNTER RJ. *Foundations of Colloid Science* (Oxford University Press, Oxford, 2001).
- [125] HURLEY MM and HARROWELL P. *Kinetic structure of a two-dimensional liquid.* Phys. Rev. E **52**, 1694 (1995).
- [126] IKEDA A and MIYAZAKI K. *Mode-Coupling Theory as a Mean-Field Description of the Glass Transition.* Phys. Rev. Lett. **104**, 255704 (2010).
- [127] IN 'T VELD PJ, STONE MT, TRUSKETT TM, and SANCHEZ IC. *Liquid Structure via Cavity Size Distributions.* J. Phys. Chem. C **104**, 12028 (2000).
- [128] ISING E. *Beitrag zur Theorie des Ferromagnetismus.* Zeitschr. f. Physik A **31**, 253 (1925).
- [129] JARDAT M, HRIBAR-LEE B, DAHIREL V, and VLACHY V. *Self-diffusion and activity coefficients of ions in charged disordered media.* J. Chem. Phys. **137**, 114507 (2012).
- [130] JODREY WS and TORY EM. *Computer simulation of close random packing of equal spheres.* Phys. Rev. A **32**, 2347 (1985).
- [131] JUÁREZ-MALDONADO R and CHÁVEZ-ROJO MA. *Theoretical unification between Quenched-Annealed and Equilibrated-Mixture Systems.* arXiv:1108.6291 (2011).
- [132] KAMIEN RD and LIU AJ. *Why is Random Close Packing Reproducible?* Phys. Rev. Lett. **99**, 155501 (2007).
- [133] KAPFER SC, MICKEL W, MECKE K, and SCHRÖDER-TURK GE. *Jammed spheres: Minkowski tensors reveal onset of local crystallinity.* Phys. Rev. E **85**, 030301 (2012).
- [134] KÄRGER J. *Straightforward derivation of the long-time limit of the mean-square displacement in one-dimensional diffusion.* Phys. Rev. A **45**, 4173 (1992).
- [135] KARMAKAR S, LERNER E, and PROCACCIA I. *Direct estimate of the static length-scale accompanying the glass transition.* Physica A **391**, 1001 (2012).
- [136] KARMAKAR S and PARISI G. *Random pinning glass model.* Proc. Natl. Acad. Sci. USA **110**, 2752 (2013).
- [137] KARMAKAR S and PROCACCIA I. *Exposing the static scale of the glass transition by random pinning.* arXiv:1105.4053 (2011).
- [138] KAUZMANN W. *The Nature of the Glassy State and the Behavior of Liquids at Low Temperatures.* Chem. Rev. **43**, 219 (1948).
- [139] KAWASAKI T and TANAKA H. *Structural origin of dynamic heterogeneity in three-dimensional colloidal glass formers and its link to crystal nucleation.* J. Phys.: Condens. Matter **22**, 232102 (2010).
- [140] KEGEL WK and VAN BLAADEREN A. *Direct Observation of Dynamical Heterogeneities in Colloidal Hard-Sphere Suspensions.* Science **287**, 290 (2000).
- [141] KELTON K, EHRENREICH H, and TURNBULL D. *Crystal Nucleation in Liquids and Glasses.* In: *Solid State Physics*, vol. 45, p. 75 (Academic Press, 1991).
- [142] KERSTEIN AR. *Equivalence of the void percolation problem for overlapping spheres and a network problem.* J. Phys. A: Math. Gen. **16**, 3071 (1983).

- [143] KIERLIK E, MONSON PA, ROSINBERG ML, SARKISOV L, and TARJUS G. *Capillary Condensation in Disordered Porous Materials: Hysteresis versus Equilibrium Behavior*. Phys. Rev. Lett. **87**, 055701 (2001).
- [144] KIERLIK E, ROSINBERG ML, TARJUS G, and MONSON P. *The pressure of a fluid confined in a disordered porous material*. J. Chem. Phys. **103**, 4256 (1995).
- [145] KIERLIK E, ROSINBERG ML, TARJUS G, and MONSON PA. *Phase diagrams of single-component fluids in disordered porous materials: Predictions from integral-equation theory*. J. Chem. Phys. **106**, 264 (1997).
- [146] KIM IC and TORQUATO S. *Diffusion of finite-sized Brownian particles in porous media*. J. Chem. Phys. **96**, 1498 (1992).
- [147] KIM K. *Effects of pinned particles on the structural relaxation of supercooled liquids*. EPL **61**, 790 (2003).
- [148] KIM K, MIYAZAKI K, and SAITO S. *Slow dynamics in random media: Crossover from glass to localization transition*. EPL **88**, 36002 (2009).
- [149] KIM K, MIYAZAKI K, and SAITO S. *Molecular dynamics studies of slow dynamics in random media: Type A–B and reentrant transitions*. Eur. Phys. J. Spec. Top. **189**, 135 (2010).
- [150] KIM K, MIYAZAKI K, and SAITO S. *Slow dynamics, dynamic heterogeneities, and fragility of supercooled liquids confined in random media*. J. Phys.: Condens. Matter **23**, 234123 (2011).
- [151] KIM K and MUNAKATA T. *Glass transition of hard sphere systems: Molecular dynamics and density functional theory*. Phys. Rev. E **68**, 021502 (2003).
- [152] KIRKPATRICK S. *Percolation and Conduction*. Rev. Mod. Phys. **45**, 574 (1973).
- [153] KNUTH DE. *The Art of Computer Programming: Searching and Sorting*, vol. 3 (Addison-Wesley, Reading, 1973).
- [154] KOB W. *Supercooled liquids, the glass transition, and computer simulations*. Lecture notes for the Les Houches Summer School (2002).
- [155] KOB W and ANDERSEN HC. *Testing mode-coupling theory for a supercooled binary Lennard-Jones mixture. II. Intermediate scattering function and dynamic susceptibility*. Phys. Rev. E **52**, 4134 (1995).
- [156] KOB W and ANDERSEN HC. *Testing mode-coupling theory for a supercooled binary Lennard-Jones mixture: The van Hove correlation function*. Phys. Rev. E **51**, 4626 (1995).
- [157] KOB W, ROLDÁN-VARGAS S, and BERTHIER L. *Non-monotonic temperature evolution of dynamic correlations in glass-forming liquids*. Nat. Phys. **8**, 164 (2012).
- [158] KOHLRAUSCH R. *Theorie des elektrischen Rückstandes in der Leidener Flasche*. Ann. Phys. **167**, 179 (1854).
- [159] KRAKOVIACK V. *Liquid–Glass Transition of a Fluid Confined in a Disordered Porous Matrix: A Mode-Coupling Theory*. Phys. Rev. Lett. **94**, 065703 (2005).
- [160] KRAKOVIACK V. *Liquid–glass transition of confined fluids: insights from a mode-coupling theory*. J. Phys.: Condens. Matter **17**, S3565 (2005).
- [161] KRAKOVIACK V. *Mode-coupling theory for the slow collective dynamics of fluids adsorbed in disordered porous media*. Phys. Rev. E **75**, 031503 (2007).
- [162] KRAKOVIACK V. *Tagged-particle dynamics in a fluid adsorbed in a disordered porous solid: Interplay between the diffusion–localization and liquid–glass transitions*. Phys. Rev. E **79**, 061501 (2009).

- [163] KRAKOVIACK V. *Statistical mechanics of homogeneous partly pinned fluid systems*. Phys. Rev. E **82**, 061501 (2010).
- [164] KRAKOVIACK V. *Mode-coupling theory predictions for the dynamical transitions of partly pinned fluid systems*. Phys. Rev. E **84**, 050501 (2011).
- [165] KRIVOV SV. *Optimal dimensionality reduction of complex dynamics: The chess game as diffusion on a free-energy landscape*. Phys. Rev. E **84**, 011135 (2011).
- [166] KUBO R. *Statistical-Mechanical Theory of Irreversible Processes. I. General Theory and Simple Applications to Magnetic and Conduction Problems*. J. Phys. Soc. Jpn. **12**, 570 (1957).
- [167] KURITA R and WEEKS ER. *Glass transition of two-dimensional binary soft-disk mixtures with large size ratios*. Phys. Rev. E **82**, 041402 (2010).
- [168] LECHNER W and DELLAGO C. *Accurate determination of crystal structures based on averaged local bond order parameters*. J. Chem. Phys. **129**, 114707 (2008).
- [169] LEE DT and SCHACHTER BJ. *Two algorithms for constructing a Delaunay triangulation*. Int. J. Comp. Inf. Sci. **9**, 219 (1980).
- [170] LEE LL. *An accurate integral equation theory for hard spheres: Role of the zero-separation theorems in the closure relation*. J. Chem. Phys. **103**, 9388 (1995).
- [171] LEOCMACH M and TANAKA H. *Roles of icosahedral and crystal-like order in the hard spheres glass transition*. Nat. Comm. **3**, 974 (2012).
- [172] LIKOS CN. *Effective interactions in soft condensed matter physics*. Phys. Rep. **348**, 267 (2001).
- [173] LIU AJ and NAGEL SR. *Nonlinear dynamics: Jamming is not just cool any more*. Nature **396**, 21 (1998).
- [174] LOMBA E, GIVEN JA, STELL G, WEIS JJ, and LEVESQUE D. *Ornstein–Zernike equations and simulation results for hard-sphere fluids adsorbed in porous media*. Phys. Rev. E **48**, 233 (1993).
- [175] LORENZ CD and ZIFF RM. *Precise determination of the critical percolation threshold for the three-dimensional “Swiss cheese” model using a growth algorithm*. J. Chem. Phys. **114**, 3659 (2001).
- [176] LUBACHEVSKY BD. *How to simulate billiards and similar systems*. J. Comput. Phys. **94**, 255 (1991).
- [177] LUBACHEVSKY BD and STILLINGER FH. *Geometric properties of random disk packings*. J. Stat. Phys. **60**, 561 (1990).
- [178] LUBCHENKO V and WOLYNES PG. *Theory of Structural Glasses and Supercooled Liquids*. Annu. Rev. Phys. Chem. **58**, 235 (2007).
- [179] LUCHNIKOV VA, MEDVEDEV NN, OGER L, and TROADEC JP. *Voronoi–Delaunay analysis of voids in systems of nonspherical particles*. Phys. Rev. E **59**, 7205 (1999).
- [180] MADDEN WG. *Fluid distributions in random media: Arbitrary matrices*. J. Chem. Phys. **96**, 5422 (1992).
- [181] MADDEN WG and GLANDT ED. *Distribution functions for fluids in random media*. J. Stat. Phys. **51**, 537 (1988).
- [182] MATTSSON J, WYSS HM, FERNANDEZ-NIEVES A, MIYAZAKI K, HU Z, REICHMAN DR, and WEITZ DA. *Soft colloids make strong glasses*. Nature **462**, 83 (2009).

- [183] MAUS A. *Delaunay triangulation and the convex hull of n points in expected linear time*. BIT **24**, 151 (1984).
- [184] MCGUFFEE SR and ELCOCK AH. *Diffusion, Crowding & Protein Stability in a Dynamic Molecular Model of the Bacterial Cytoplasm*. PLoS Comput. Biol. **6**, e1000694 (2010).
- [185] MCKENNA GB. *International workshop on dynamics in confinement: A personal summary*. J. Phys. IV France **10**, 343 (2000).
- [186] MCKENNA GB. *Status of our understanding of dynamics in confinement: Perspectives from Confit 2003*. Eur. Phys. J. E **12**, 191 (2003).
- [187] MCKENNA GB. *Confit III. Summary and perspectives on dynamics in confinement*. Eur. Phys. J. Spec. Top. **141**, 291 (2007).
- [188] MEDALIA O, WEBER I, FRANGAKIS AS, NICASTRO D, GERISCH G, and BAUMEISTER W. *Macromolecular Architecture in Eukaryotic Cells Visualized by Cryoelectron Tomography*. Science **298**, 1209 (2002).
- [189] MERONI A, LEVESQUE D, and WEIS JJ. *Correlation functions of hard sphere fluids adsorbed in porous media*. J. Chem. Phys. **105**, 1101 (1996).
- [190] METROPOLIS N, ROSENBLUTH AW, ROSENBLUTH MN, TELLER AH, and TELLER E. *Equation of State Calculations by Fast Computing Machines*. J. Chem. Phys. **21**, 1087 (1953).
- [191] METZLER R and KLAFTER J. *The random walk's guide to anomalous diffusion: a fractional dynamics approach*. Phys. Rep. **339**, 1 (2000).
- [192] METZLER R and KLAFTER J. *The restaurant at the end of the random walk: recent developments in the description of anomalous transport by fractional dynamics*. J. Phys. A: Math. Gen. **37**, R161 (2004).
- [193] MEYER A, HORBACH J, KOB W, KARGL F, and SCHOBER H. *Channel Formation and Intermediate Range Order in Sodium Silicate Melts and Glasses*. Phys. Rev. Lett. **93**, 027801 (2004).
- [194] MEYERS M, MISHRA A, and BENSON D. *Mechanical properties of nanocrystalline materials*. Prog. Mater. Sci. **51**, 427 (2006).
- [195] MILLER S and LUDING S. *Event-driven molecular dynamics in parallel*. J. Comput. Phys. **193**, 306 (2003).
- [196] MITTAL J, ERRINGTON JR, and TRUSKETT TM. *Using available volume to predict fluid diffusivity in random media*. Phys. Rev. E **74**, 040102 (2006).
- [197] MORENO AJ and COLMENERO J. *Anomalous dynamic arrest in a mixture of large and small particles*. Phys. Rev. E **74**, 021409 (2006).
- [198] MORENO AJ and COLMENERO J. *Is there a higher-order mode coupling transition in polymer blends?* J. Chem. Phys. **124**, 184906 (2006).
- [199] MOSAYEBI M, DEL GADO E, ILG P, and ÖTTINGER HC. *Probing a Critical Length Scale at the Glass Transition*. Phys. Rev. Lett. **104**, 205704 (2010).
- [200] NIJBOER BRA and RAHMAN A. *Time expansion of correlation functions and the theory of slow neutron scattering*. Physica **32**, 415 (1966).
- [201] ODRIOZOLA G. *Replica exchange Monte Carlo applied to hard spheres*. J. Chem. Phys. **131**, 144107 (2009).
- [202] ORNSTEIN LS and ZERNIKE F. *Accidental deviations of density in mixtures*. Proc. K. Akad. Wet. (Amsterdam) **17**, 793 (1914).

- [203] PANYUKOV S and RABIN Y. *Statistical physics of polymer gels*. Phys. Rep. **269**, 1 (1996).
- [204] PÉREZ-ÁNGEL G, SÁNCHEZ-DÍAZ LE, RAMÍREZ-GONZÁLEZ PE, JUÁREZ-MALDONADO R, VIZCARRA-RENDÓN A, and MEDINA-NOYOLA M. *Equilibration of concentrated hard-sphere fluids*. Phys. Rev. E **83**, 060501 (2011).
- [205] PARISI G and ZAMPONI F. *Mean-field theory of hard sphere glasses and jamming*. Rev. Mod. Phys. **82**, 789 (2010).
- [206] PASCHINGER E and KAHL G. *Structure and thermodynamic properties of a binary liquid in a porous matrix: The formalism*. Phys. Rev. E **61**, 5330 (2000).
- [207] PASCHINGER E, LEVESQUE D, KAHL G, and WEIS JJ. *On the phase separation of a binary fluid in a porous medium*. EPL **55**, 178 (2001).
- [208] PAUSCHENWEIN GJ. *Phase behavior of colloidal systems*. Ph.D. thesis, Technische Universität Wien, Wien (2008).
- [209] PEDOE D. *Circles: a mathematical view* (Cambridge University Press, Cambridge, 1995).
- [210] PERCUS JK and YEVICK GJ. *Analysis of Classical Statistical Mechanics by Means of Collective Coordinates*. Phys. Rev. **110**, 1 (1958).
- [211] PHAM KN, PUERTAS AM, BERGENHOLTZ J, EGELHAAF SU, MOUSSAÏD A, PUSEY PN, SCHOFIELD AB, CATES ME, FUCHS M, and POON WCK. *Multiple Glassy States in a Simple Model System*. Science **296**, 104 (2002).
- [212] PREPARATA FP and SHAMOS MI. *Computational geometry: an introduction* (Springer, New York, 1985).
- [213] PRIOUR JR DJ. *Percolation through Voids around Overlapping Spheres, a Dynamically based Finite Size Scaling Analysis*. arXiv:1208.0328 (2012).
- [214] PUSEY PN and VAN MEGEN W. *Phase behaviour of concentrated suspensions of nearly hard colloidal spheres*. Nature **320**, 340 (1986).
- [215] PUSEY PN and VAN MEGEN W. *Observation of a glass transition in suspensions of spherical colloidal particles*. Phys. Rev. Lett. **59**, 2083 (1987).
- [216] RAHMAN A. *Correlations in the Motion of Atoms in Liquid Argon*. Phys. Rev. **136**, A405 (1964).
- [217] RAMÍREZ-GONZÁLEZ PE, LÓPEZ-FLORES L, ACUÑA-CAMPA H, and MEDINA-NOYOLA M. *Density-Temperature-Softness Scaling of the Dynamics of Glass-Forming Soft-Sphere Liquids*. Phys. Rev. Lett. **107**, 155701 (2011).
- [218] RAPAPORT DC. *The event scheduling problem in molecular dynamic simulation*. J. Comput. Phys. **34**, 184 (1980).
- [219] RAPAPORT DC. *The Art of Molecular Dynamics Simulation* (Cambridge University Press, Cambridge, 2004), second editon.
- [220] REICHMAN DR and CHARBONNEAU P. *Mode-coupling theory*. J. Stat. Mech. **2005**, P05013 (2005).
- [221] REISS H, FRISCH HL, and LEBOWITZ JL. *Statistical Mechanics of Rigid Spheres*. J. Chem. Phys. **31**, 369 (1959).
- [222] REYNOLDS PJ, STANLEY HE, and KLEIN W. *Large-cell Monte Carlo renormalization group for percolation*. Phys. Rev. B **21**, 1223 (1980).

- [223] RITORT F and SOLLICH P. *Glassy dynamics of kinetically constrained models*. Adv. Phys. **52**, 219 (2003).
- [224] ROSINBERG ML, TARJUS G, and STELL G. *Thermodynamics of fluids in quenched disordered matrices*. J. Chem. Phys. **100**, 5172 (1994).
- [225] ROYALL CP, WILLIAMS SR, OHTSUKA T, and TANAKA H. *Direct observation of a local structural mechanism for dynamic arrest*. Nat. Mater. **7**, 556 (2008).
- [226] RUSSEL WB, SAVILLE DA, and SCHOWALTER WR. *Colloidal Dispersions* (Cambridge University Press, Cambridge, 1989).
- [227] SANTEN L and KRAUTH W. *Absence of thermodynamic phase transition in a model glass former*. Nature **405**, 550 (2000).
- [228] SANZ E, VALERIANI C, ZACCARELLI E, POON WCK, PUSEY PN, and CATES ME. *Crystallization Mechanism of Hard Sphere Glasses*. Phys. Rev. Lett. **106**, 215701 (2011).
- [229] SASTRY S, CORTI DS, DEBENEDETTI PG, and STILLINGER FH. *Statistical geometry of particle packings. I. Algorithm for exact determination of connectivity, volume, and surface areas of void space in monodisperse and polydisperse sphere packings*. Phys. Rev. E **56**, 5524 (1997).
- [230] SASTRY S, DEBENEDETTI PG, and STILLINGER FH. *Statistical geometry of particle packings. II. "Weak spots" in liquids*. Phys. Rev. E **56**, 5533 (1997).
- [231] SAUSSET F and LEVINE D. *Characterizing Order in Amorphous Systems*. Phys. Rev. Lett. **107**, 045501 (2011).
- [232] SCHÖLL-PASCHINGER E, LEVESQUE D, WEIS JJ, and KAHL G. *Phase diagram of a symmetric binary fluid in a porous matrix*. Phys. Rev. E **64**, 011502 (2001).
- [233] SCHMELZER J. *Nucleation theory and applications* (Wiley-VCH, Weinheim, 2005).
- [234] SCHMIDT M. *Test particle limit for the pair structure of quenched-annealed fluid mixtures*. Phys. Rev. E **79**, 031405 (2009).
- [235] SCHNYDER SK, HÖFLING F, FRANOSCH T, and VOIGTMANN T. *Long-wavelength anomalies in the asymptotic behavior of mode-coupling theory*. J. Phys.: Condens. Matter **23**, 234121 (2011).
- [236] SCHOLZ C, WIRNER F, GÖTZ J, RÜDE U, SCHRÖDER-TURK GE, MECKE K, and BECHINGER C. *Permeability of Porous Materials Determined from the Euler Characteristic*. Phys. Rev. Lett. **109**, 264504 (2012).
- [237] SCHRÖDER-TURK GE, MICKEL W, SCHRÖTER M, DELANEY GW, SAADATFAR M, SENDEN TJ, MECKE K, and ASTE T. *Disordered spherical bead packs are anisotropic*. EPL **90**, 34001 (2010).
- [238] SCHWABL F. *Statistische Mechanik* (Springer, Berlin, 2000).
- [239] SCHWANZER DF, COSLOVICH D, KURZIDIM J, and KAHL G. *Effects of porous confinement on the structural properties of the Gaussian core model*. Mol. Phys. **107**, 433 (2009).
- [240] SCIORTINO F and KOB W. *Debye-Waller Factor of Liquid Silica: Theory and Simulation*. Phys. Rev. Lett. **86**, 648 (2001).
- [241] SCIORTINO F and TARTAGLIA P. *Glassy colloidal systems*. Adv. Phys. **54**, 471 (2005).
- [242] SCIORTINO F, TARTAGLIA P, and ZACCARELLI E. *Evidence of a Higher-Order Singularity in Dense Short-Ranged Attractive Colloids*. Phys. Rev. Lett. **91**, 268301 (2003).

- [243] SCOTT GD and KILGOUR DM. *The density of random close packing of spheres*. J. Phys. D: Appl. Phys. **2**, 863 (1969).
- [244] SENGUPTA S and KARMAKAR S. *Distribution of Diffusion Constants and Stokes–Einstein Violation in supercooled liquids*. arXiv:1301.1181 (2013).
- [245] SHIBA H and KAWASAKI T. *Spatiotemporal heterogeneity of local free volumes in highly supercooled liquid*. arXiv:1202.3509 (2012).
- [246] SPANNER M, HÖFLING F, SCHRÖDER-TURK GE, MECKE K, and FRANOSCH T. *Anomalous transport of a tracer on percolating clusters*. J. Phys.: Condens. Matter **23**, 234120 (2011).
- [247] SPANNER M, SCHNYDER SK, HÖFLING F, VOIGTMANN T, and FRANOSCH T. *Dynamic arrest in model porous media: intermediate scattering functions*. Soft Matter **9**, 1604 (2013).
- [248] STAUFFER D and AHARONY A. *Perkolationstheorie: eine Einführung* (VCH, Weinheim, 1995).
- [249] STREKALOVA EG, LUO J, STANLEY HE, FRANZESE G, and BULDYREV SV. *Confinement of Anomalous Liquids in Nanoporous Matrices*. Phys. Rev. Lett. **109**, 105701 (2012).
- [250] SU P and DRYSDALE RLS. *A Comparison of Sequential Delaunay Triangulation Algorithms*. In: *Proceedings of the eleventh annual symposium on Computational geometry*, SCG '95, p. 61 (ACM, 1995).
- [251] SUNG BJ and YETHIRAJ A. *Lateral Diffusion and Percolation in Membranes*. Phys. Rev. Lett. **96**, 228103 (2006).
- [252] SUNG BJ and YETHIRAJ A. *The effect of matrix structure on the diffusion of fluids in porous media*. J. Chem. Phys. **128**, 054702 (2008).
- [253] SZAMEL G and FLENNER E. *Glassy dynamics of partially pinned fluids: An alternative mode-coupling approach*. EPL **101**, 66005 (2013).
- [254] TAMMANN G and HESSE W. *Die Abhängigkeit der Viskosität von der Temperatur bei unterkühlten Flüssigkeiten*. Anorg. Allg. Chem. **156**, 245 (1926).
- [255] TANAKA H, KAWASAKI T, SHINTANI H, and WATANABE K. *Critical-like behaviour of glass-forming liquids*. Nat. Mater. **9**, 324 (2010).
- [256] TANEMURA M, OGAWA T, and OGITA N. *A New Algorithm for Three-Dimensional Voronoi Tessellation*. J. Comput. Phys. **51**, 191 (1983).
- [257] THIESSEN AH and ALTER JC. *District No. 10, Great Basin*. Mon. Wea. Rev. **39**, 1082 (1911).
- [258] TODA M, KUBO R, and SAITÔ N. *Statistical Physics I: Equilibrium Statistical Mechanics* (Springer, Berlin, 1983).
- [259] TORQUATO S and JIAO Y. *Robust algorithm to generate a diverse class of dense disordered and ordered sphere packings via linear programming*. Phys. Rev. E **82**, 061302 (2010).
- [260] TORQUATO S and STILLINGER FH. *Jammed hard-particle packings: From Kepler to Bernal and beyond*. Rev. Mod. Phys. **82**, 2633 (2010).
- [261] TORQUATO S, TRUSKETT TM, and DEBENEDETTI PG. *Is Random Close Packing of Spheres Well Defined?* Phys. Rev. Lett. **84**, 2064 (2000).
- [262] TRAPPE V, PRASAD V, CIPELLETTI L, SEGRE PN, and WEITZ DA. *Jamming phase diagram for attractive particles*. Nature **411**, 772 (2001).
- [263] TRUSKETT TM, TORQUATO S, SASTRY S, DEBENEDETTI PG, and STILLINGER FH. *Structural precursor to freezing in the hard-disk and hard-sphere systems*. Phys. Rev. E **58**, 3083 (1998).

- [264] VAN HOVE L. *Correlations in Space and Time and Born Approximation Scattering in Systems of Interacting Particles*. Phys. Rev. **95**, 249 (1954).
- [265] VAN MEGEN W and UNDERWOOD SM. *Glass transition in colloidal hard spheres: Mode-coupling theory analysis*. Phys. Rev. Lett. **70**, 2766 (1993).
- [266] VEGA C, KAMINSKY RD, and MONSON PA. *Adsorption of fluids in disordered porous media from integral equation theory*. J. Chem. Phys. **99**, 3003 (1993).
- [267] VERLET L. *Computer “Experiments” on Classical Fluids. II. Equilibrium Correlation Functions*. Phys. Rev. **165**, 201 (1968).
- [268] VISSCHER WM and BOLSTERLI M. *Random Packing of Equal and Unequal Spheres in Two and Three Dimensions*. Nature **239**, 504 (1972).
- [269] VOGEL H. *Temperaturabhängigkeitsgesetz der Viskosität von Flüssigkeiten*. Phys. Z. **22**, 645 (1921).
- [270] VOIGTMANN T. *Mode Coupling Theory of the Glass Transition in Binary Mixtures*. Ph.D. thesis, Technische Universität München, München (2002).
- [271] VOIGTMANN T. *Mode-Coupling Theory of the Colloidal Glass Transition*. In: *Soft Matter: From Synthetic to Biological Materials*, no. 39 in IFF Spring School, p. D3.1 (Forschungszentrum Jülich GmbH, Jülich, 2008).
- [272] VOIGTMANN T and HORBACH J. *Slow dynamics in ion-conducting sodium silicate melts: Simulation and mode-coupling theory*. EPL **74**, 459 (2006).
- [273] VOIGTMANN T and HORBACH J. *Double Transition Scenario for Anomalous Diffusion in Glass-Forming Mixtures*. Phys. Rev. Lett. **103**, 205901 (2009).
- [274] VON SMOLUCHOWSKI M. *Zur kinetischen Theorie der Brownschen Molekularbewegung und der Suspensionen*. Ann. Phys. **326**, 756 (1906).
- [275] VORONOI G. *Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Deuxième mémoire. Recherches sur les paralléloèdres primitifs*. J. Reine Angew. Math. **1908**, 198 (1908).
- [276] WATANABE MS. *Percolation with a periodic boundary condition: The effect of system size for crystallization in molecular dynamics*. Phys. Rev. E **51**, 3945 (1995).
- [277] WEEKS ER, CROCKER JC, LEVITT AC, SCHOFIELD A, and WEITZ DA. *Three-Dimensional Direct Imaging of Structural Relaxation Near the Colloidal Glass Transition*. Science **287**, 627 (2000).
- [278] WEEKS JD, CHANDLER D, and ANDERSEN HC. *Role of Repulsive Forces in Determining the Equilibrium Structure of Simple Liquids*. J. Chem. Phys. **54**, 5237 (1971).
- [279] WEI QH, BECHINGER C, and LEIDERER P. *Single-File Diffusion of Colloids in One-Dimensional Channels*. Science **287**, 625 (2000).
- [280] WERTHEIM MS. *Exact Solution of the Percus–Yevick Integral Equation for Hard Spheres*. Phys. Rev. Lett. **10**, 321 (1963).
- [281] WEYSER F, PUERTAS AM, FUCHS M, and VOIGTMANN T. *Structural relaxation of polydisperse hard spheres: Comparison of the mode-coupling theory to a Langevin dynamics simulation*. Phys. Rev. E **82**, 011504 (2010).
- [282] WIDMER-COOPER A, HARROWELL P, and FYNEWEVER H. *How Reproducible Are Dynamic Heterogeneities in a Supercooled Liquid?* Phys. Rev. Lett. **93**, 135701 (2004).
- [283] WIGNER E and SEITZ F. *On the Constitution of Metallic Sodium*. Phys. Rev. **43**, 804 (1933).

-
- [284] WILLIAMS G and WATTS DC. *Non-symmetrical dielectric relaxation behaviour arising from a simple empirical decay function*. Trans. Faraday Soc. **66**, 80 (1970).
- [285] WILLIAMS SR, BRYANT G, SNOOK IK, and VAN MEGEN W. *Velocity Autocorrelation Functions of Hard-Sphere Fluids: Long-Time Tails upon Undercooling*. Phys. Rev. Lett. **96**, 087801 (2006).
- [286] WOOD WW and JACOBSON JD. *Preliminary Results from a Recalculation of the Monte Carlo Equation of State of Hard Spheres*. J. Chem. Phys. **27**, 1207 (1957).
- [287] XU WS, SUN ZY, and AN LJ. *Structure, compressibility factor, and dynamics of highly size-asymmetric binary hard-disk liquids*. J. Chem. Phys. **137**, 104509 (2012).
- [288] ZACHARIASEN WH. *The Atomic Arrangement In Glass*. J. Am. Chem. Soc. **54**, 3841 (1932).
- [289] ZHANG L and VAN TASSEL PR. *Theory and simulation of adsorption in a templated porous material: Hard sphere systems*. J. Chem. Phys. **112**, 3006 (2000).
- [290] ZHAO SL, DONG W, and LIU QH. *Fluids in porous media. I. A hard sponge model*. J. Chem. Phys. **125**, 244703 (2006).
- [291] ZIFF RM, CUMMINGS PT, and STELL G. *Generation of percolation cluster perimeters by a random walk*. J. Phys. A: Math. Gen. **17**, 3009 (1984).
- [292] ZINCHENKO AZ. *Algorithm for Random Close Packing of Spheres with Periodic Boundary Conditions*. J. Comput. Phys. **114**, 298 (1994).

Curriculum Vitae

Jan Kurzidim, born in Sindelfingen, Germany

Education

	Ph.D., Physics (<i>anticipated</i>)
12/2007 – present	Vienna University of Technology (Vienna, Austria)
10/2006 – 11/2007	University of Vienna (Vienna, Austria)
	M.Sc., Physics (<i>“with Honors”; GPA: 3.92/4</i>)
08/2003 – 05/2005	University of Kansas (Lawrence, Kansas, United States)
10/2000 – 07/2003	University of Stuttgart (Stuttgart, Germany)
	Secondary school degree (<i>“Abitur”; grade average: 1.0</i>)
09/1990 – 06/1999	Goldberg-Gymnasium Sindelfingen (Sindelfingen, Germany)

Professional experience

11/2011 – 07/2012	Book author & designer “Разом: Ukrainisch für Anfängerinnen und Anfänger”
12/2007 – 04/2011	Ph.D. fellow Austrian Science Fund (FWF)
01/2007 – 11/2007	IT assistant National Fund of the Republic of Austria for Victims of National Socialism (Vienna, Austria)
11/2005 – 09/2006	Software engineer IBM Research and Development (Böblingen, Germany)
06/2004 – 05/2005	Research assistant University of Kansas (Lawrence, Kansas, United States)
08/2003 – 05/2004	Teaching assistant University of Kansas (Lawrence, Kansas, United States)

Other occupations

07/1999 – 05/2000	Military service Staff of the II. GE/US Corps (Ulm, Germany)
-------------------	--

Mehr, wenn ihr mich wiederseht.
Ihr müsst unbedingt gucken, wie's weitergeht.

