ABSTRACT

Title of dissertation: SPATIALLY HETEROGENEOUS DYNAMICS AND STRING-LIKE CORRELATED MOTION IN SUPERCOOLED LIQUIDS AND POLYMERS

Yeshitila Gebremichael, Doctor of Philosophy, 2004

Dissertation directed by: Professor Sharon C. Glotzer
Chemical Engineering Department
University of Michigan, Ann Arbor, MI

Professor John D. Weeks
Chemical Physics Program
University of Maryland, College Park, MD

Dense liquids above their glass transition exhibit spatially heterogeneous dynamics (SHD) in which regions within the liquid exhibit enhanced or diminished mobility relative to the average on some time scale. The spatially heterogeneous nature of local dynamics in supercooled liquids is fairly well established both experimentally and computationally. However, many questions remain concerning why and how this complex dynamics arises. Here we address these questions and present results of a detailed investigation of SHD in models of a one-component supercooled liquid and a low-molecular-weight polymer melt, via molecular dynamics simulation.

We find that particles or chain segments (monomers) with high mobility exhibit a correlated motion in which they move in a quasi-one dimensional "string-like" paths that aggregate into larger, ramified clusters. These dynamical clusters grow in size with decreasing temperature. The mean string and cluster sizes show a transient nature, with peaks at the late-β/early-α relaxation time of the mode-coupling theory (MCT). The size distribution of the strings shows an exponential behavior, while that of the clusters approaches a power law near T_{MCT}. We further investigate the microscopic details of local particle dynamics in order to understand the mechanisms by which particles move along string-like correlated paths. We find that the degree of coherence, i.e., the simulataneous motion by consecutive particles along a string, depends on the length of the string.

We also explore the thermodynamic behavior of the one-component liquid via the inherent structure formalism to study the connection between the dynamical strings and clusters we have investigated and the "cooperatively rearranging region (CRR)" of the Adam-Gibbs (AG) theory. We find that the average cluster size is linearly related to the inverse of the configurational entropy S_{conf}, as observed in simulated water. However, we also find a similar linear relationship between the average string size and configurational entropy.

SPATIALLY HETEROGENEOUS DYNAMICS AND STRING-LIKE CORRELATED MOTION IN SUPERCOOLED LIQUIDS AND POLYMERS

by

Yeshitila Gebremichael

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements of the degree of
Doctor of Philiosophy
2004

Advisory Committee:

Professor John Weeks, Chair/Advisor
Professor Sharon Glotzer, Advisor
Professor Sandra Greer
Professor Victor Munoz
Professor Sergei Sukharev

ACKNOWLEDGEMENTS

I extend my sincere gratitude and appreciation to many people who made this journey an experience that can be remembered fondly. Special thanks are due to my supervisor Professor Sharon Glotzer who inspired me with her knowledge, ambition, and commitment to science. In particular, I am deeply indepted for her guidance, advice, and encouragement during my studies. Her expertise has been extremely invaluable for the success of my research.

I am also grateful to Professor John Weeks for taking on the responsibility of being my internal advisor at the Chemical Physics Program. I wish to thank him and the other members of my dissertation committee, Professors Sandra Greer, Victor Munoz, and Sergei Sukharev for comments and suggestions.

I wish to express my gratitude to all people with whom I had scientific collaborations and discussions. In particular, I am highly indepted to Dr. Thomas Schrøder, who introduced me to computer programming, Dr. Francis Starr, who has been a great asset for my scientific growth, and Dr. Michael Vogel, who assisted me in tackling several complex tasks. I am also thankful to Professor Joerg Baschnagel and Dr. Martin Aichele for their collaborative work that lead to a part in my thesis, Dr. Naida Lacevic for helpful discussions, and Mr. Magnus Bergroth

for providing me with a minimization code.

I had the opportunity to meet many people both at the university of Maryland and at the university of Michigan who made my graduate school experience memorable. I am especially thankful to Mrs. Diane Mancuso whose unfailing support and kindness made administrative processes easier. I am privileged to have known her. I am also thankful to Professor Michael Coplan for his support and encouragement, and Mr. Tong Zhao for his friendship. Here at Michigan, I am also grateful to all my officemates for their support and friendship. I am especially thankful to Mr. Mark Horsch and Dr. Zhenli Zhang for their friendship as well as the wonderful scientific discussions that made me appreciate researches outside my specific area.

Most of all, I owe special thanks to my families and friends for their love and support. I would like to extend my love and appreciation to my daughter Hanna, my wife Elizabeth, and my mother Tsige for their unreserved love and patience. This whole journey would have been different without their love and support. Their love has been the precious gift of life to me, and experiencing it will always remain to be my greates acheivement in life than anything else.

To Hanna, Elizabeth, and Tsigiye

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

When a liquid is cooled, it can solidify in two distinct ways. The familiar route produces an ordered crystal. An equally important but subtle transformation occurs when crystallization is avoided during cooling. This can be achieved, for instance, by a rapid quenching of the liquid below its freezing point. Other ways include reactive precipitation, electrolyte deposition (starting from a liquid), ion implantation, and chemical vapor deposition, to name a few [1, 2]. As a result, a wide variety of materials are capable of glass formation (vitrification) under certain conditions. For example, in addition to the commonly available inorganic oxides such as SiO_2, GeO_2, B_2O_3, etc, that commonly form glasses, other materials such as organic polymers (e.g. poly(ethylene oxide), polystyrene, poly(vinyl chloride)), low molecular weight organic compounds (e.g., glycerol, glucose, o-terphenyl), molten salts, metallic liq-

uids, and water are known to form glasses [1]. Even biomacromolecules like proteins are known to exhibit a dynamic transition known as a protein glass transition, or sometimes referred to as the "slaved" glass transition to emphasize the influence of the solvent (e.g. water) in which the proteins are usually embedded [2, 3] .

In general, a glass can be formed by any substance, provided nucleation is suppressed. The question is not whether a given substance vitrifies or not, but under what condition it can do so. For example, metallic liquids are known to vitrify, but bulk metallic glasses (BMG) were not found until recently due to the large quenching rates ($\approx 10^6 - 10^8$ K s^{-1} as compared with $0.1 - 1$ K s^{-1} for 'easy glass formers' such as SiO_2 and B_2O_3 [4]) required to produce them .

A liquid that succeeds in avoiding crystallization below its melting temperature T_m, but that is not yet a glass, is referred to as being in the supercooled liquid state, which is metastable with respect to the crystalline state. As the supercooled liquid is cooled to lower temperatures, its density and viscosity increase, and the molecules in the liquid move more and more slowly. At sufficiently low temperatures, the molecules in the liquid will rearrange so slowly that they can not adequately sample configurations in the available time allowed by the cooling rate. The liquid's structure therefore becomes frozen on the laboratory time scale. In other words, the characteristic time for the structural relaxation becomes comparable to the duration of a macroscopic experiment (e.g., of the order of 100 seconds) [5]. On this and shorter time scales the supercooled liquid is structurally arrested, and is called a glass.

The temperature at which a liquid falls out of equilibrium and transforms

into a glass is known as the glass transition temperature, T_g. This occurs across a narrow transformation range where the rate of change of volume or enthalpy with respect to temperature decreases abruptly to a value comparable to that of the crystalline solid [6]. Fig. 1 illustrates the temperature dependence of a liquid's volume (or enthalpy) as it is rapidly cooled under isobaric conditions. Unlike T_m, T_g is not unique for a given substance. It depends on the cooling rate; the slower a liquid is cooled, the longer the time available for configurational sampling at each temperature, and hence the colder it can become before falling out of liquid-state equilibrium [7]. Consequently, T_g decreases with the cooling rate. Fig. 1 shows different values of T_g resulting from different cooling rates. Typically, the dependence of T_g upon cooling rate is weak; an order of magnitude change in cooling rate may change T_g by only $3-5$ K [8]. Despite its (weak) dependence on the cooling rate, when defined consistently, T_g is an important material property that can be used, e.g., in estimating the mechanical properties of materials.

Because glass formation can be achieved by a variety of materials, understanding the nature of glasses and supercooled liquids has impact in areas as diverse as environmental, biological, pharmaceutical, technological and many other fields. Environmentally, for example, glasses are important geological materials on earth, where rapid cooling of magma produces about a billion cubic meters of glass each year [9]. Further, it has been argued that most of the universe's water exists in the glassy state [1, 2]. In the area of life sciences, the dynamics of proteins is intimately related to that of glass-forming liquids, one example being the protein glass transition mentioned above [2, 3]. Additionally, the concept of the energy landscape,

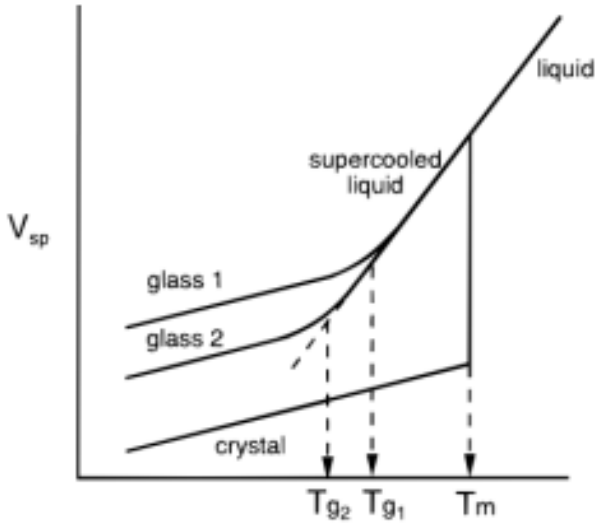

Figure 1.1: Schematic representation of the specific volume as a function of T for a liquid that can both crystalize and form a glass. T_m is the melting temperature, and T_{g1} and T_{g2} represent the glass transition temperatures of the liquid resulting from relatively faster (glass 1) and slower (glass 2) cooling rates, respectively. Adapted from Ref. [8].

originally developed for glass-forming liquids, is extensively used in the study of the protein folding problem [10].

In the pharmaceutical area, the use of supercooled water as a storage medium is considered to be a way of preserving the biological activity of isolated proteins in solution during the storage process for the purpose of therapeutic or biochemical applications. The effectiveness of supercooling as a means of prolonging the shelf life of proteins hinges on the availability of reliable techniques for preventing freezing [5], which in turn relies on understanding the dynamic and thermodynamic behavior of supercooled liquids, in this case water. Along this line, saccharide glasses are used

to preserve biological structures (tissues, cells, enzymes) for storage and transportation [11]. Additionally, glassy pharmaceuticals are known to be more rapidly taken up by the body than pharmaceuticals in crystalline form [11].

In the technological arena, on the other hand, polymers can be cited as one simple example where the knowledge of the glass transition temperature T_g has a significant role in harnessing them for technological uses. All synthetic polymers are solids that are at least partially amorphous. Depending on their technological applications, polymers can be used either above or below their glass transition. For example, hard plastics like polystyrene and poly(methyl methacrylate) are used below their glass transitions, while rubber elastomers like polyisoprene and polyisobutylene are used above their glass transitions. Therefore, T_g is an important material property that is useful in estimating the mechanical properties of a polymeric material.

The above few examples demonstrate the wide range of applicability of glasses that spans from the archetypal inorganic window glasses to a biologically active material. Despite their wide applicability, the formation of glasses is not well understood. In particular, several aspects of their relaxation properties are still a mystery. Part of the mystery lies in the dramatic slowing down of dynamics upon glass formation. When the liquid is cooled towards its glass transition temperature T_g, several dynamical quantities such as relaxation times, diffusivities or viscosities change by up to 14 orders of magnitude. Yet structurally, glasses look almost indistinguishable from the liquid from which they are formed, when subjected to typical structural measures such as the radial distribution function. The primary issue of the glass

transition phenomena is thus *understanding the cause of the dramatic change in the dynamics without any significant change in the structure.* Of course, a dramatic change in the dynamics can also be observed when a liquid crystallizes. But in this case the change in the dynamics can be attributed to the change in the structure, i.e., a change from a disordered liquid to an ordered solid.

In the past, several theories have been proposed to explain the origin of slowing dynamics and the cause of the glass transition under deep supercooling. Among these are the entropy theory of Adams, Gibbs, and DiMarzio [12, 13], the free volume theory of Turnbull, Cohen and Grest [14, 15, 16], the mode-coupling theory (MCT) of Götze and co-workers [17], and the frustration-limited domain theory of Kivelson, Tarjus and co-workers [18], to name a few. These theories have explained several aspects of the relaxation of liquids upon cooling towards T_g. However, thus far, a comprehensve theory that captures all the salient features of this phenomenon is still missing. In fact, nearly a decade ago, Anderson [19] described this problem as being "*the deepest and most interesting unsolved problem in solid state physics*". Still presently, this phenomenon remains a challenge to the scientific community, although much progress has been made over the past several decades in the understanding of many aspects of the glass transition.

Part of the reason for the difficulty in resolving the mystery of glass formation, and the lack of a comprehensive theory that has universal acceptance, lies in the absence of a detailed microscopic description of this complex phenomenon. Such a picture has begun to emerge only recently with the advent of novel experimental techniques (e.g., confocal microscopy [20, 21], 4D-NMR experiments [22], and oth-

ers), and the advancement of computational capabilities. Our research, being part of the general effort in understanding the dynamic and thermodynamic properties of glass-forming liquids, is dedicated to exploring the supercooled state by making use of computer simulation of model systems. In particular, we strive to shed light on the microscopic behavior of glass-forming liquids that may contribute to our understanding of the origin of slow dynamics upon cooling towards T_g.

The thesis is organized as follows: In Chapter 2 we provide a brief background on the properties of glass-forming liquids, where the theories describing their relaxation behavior is discussed. In this chapter, we also present in more detail the context for our studies with respect to the general effort on the study of glass formation. In Chapter 3 we describe the two models used in our simulations: a model of a polymer melt and a liquid metal. In this chapter we describe the structural and dynamic properties of these systems. In Chapter 4 and 5 we discuss the cooperative dynamics of chain segments in polymer melts, and their tendency to form clusters to escape from the cages formed by neighboring chains. In Chapter 6, we explore a similar behavior for a model of liquid metals known as the Dzugutov liquid, with an emphasis on a detailed description of string-like particle motion, where we investigate the local rearrangements of particles that lead to the formation of the strings. In Chapter 7, we explore the thermodynamic properties of the Dzugutov liquid using the inherent structure formalism. This final chapter describes part of ongoing research effort to make a connection between a microscopic description of dynamics in terms of strings and clusters, and the macroscopic quantity, configurational entropy, thereby bridging the spatially heterogeneous dynamics observed

in glass-forming liquids and the well known Adam-Gibbs theory. Conclusions are given in chapter 8. Relevant analysis code is included in the Appendix.

Chapter 2

Background

In this chapter we discuss the relaxation properties of glass-forming liquids and the different viewpoints describing the glass transition. We begin our presentation by a brief discussion of the salient features of supercooled liquids and glasses. We then present different theories of glass formation, followed by experimental and computational observations that describe the most prominent features of glass-forming liquids. Our description is focused on those properties and theories that are most relevant to our discussion in the following chapters. Moreover, here and in the remainder of the thesis, our description is restricted to the dynamics and thermodynamics of glass-forming liquids approaching the glass transition from high T, i.e., glass transition viewed 'from the liquid', which is the scope of our research.

2.1 Canonical features of glass-forming liquids

The temperature dependence of the transport properties of liquids such as viscosity η, diffusivity D, and conductivity σ, as well as relaxation times like shear mechanical relaxations τ_s, dielectric relaxations τ_D, and other characterisitic times have been extensively studied for all classes of glass-forming liquids (See Ref. [23] for review). With few exceptions, the temperature dependence of these quantities is found to deviate from an Arrhenius behavior that is familiar in most physical processes. Such deviation is known to be one of the most important canonical features of glass-forming liquids. To characterize this non-Arrhenius behavior, a wide variety of mathematical equations have been proposed that describe the temperature dependence of relaxation times or viscosity in glass-forming liquids. Among these are $\tau \propto \exp(B/(T-T_0)^{3/2})$ of Bendler and Shlesinger [24], $\tau \propto \exp(T_0^2/T^2)$ of Bässler and Richert [25, 26], $\tau \propto (T-T_c)^{-\gamma}$ with $T_c < T_g$ of Colby [27] and with $T_c > T_g$ of the idealized mode-coupling theory, to mention a few equations that require only three fitting parameters. Here T_c is a critical temperature. There are also other mathematical equations that provide an excellent fit to the temperature dependence of transport properties, but they contain four or more fitting parameters, e.g. the Cohen-Grest equation [28].

The most frequently used equation to characterize the temperature dependence of the transport coefficients of supercooled liquids is the equation proposed by Vogel, Tammann, and Fulcher (VTF) [29],

$$\tau = A \exp(\frac{B}{T - T_0}), \tag{2.1}$$

where B is the activation energy of the system, and T_0 is the so-called Vogel temperature. When $T_0 = 0$, the familiar Arrhenius equation results. As will be discussed shortly, T_0 is related to the ideal glass transition temperature. An important aspect of this equation is its implication that the relaxation time τ diverges at a nonzero temperature, $T_0 > 0$, which in turn provokes the existence of an underlying phase transition. In practice, however, this has never been observed because of the finite cooling rate that causes the system to fall out of equilibrium and enter the glassy state before T_0 is reached. The VTF equation is mathematically equivalent to the Williams-Landel-Ferry (WLF) relation that is used to describe the temperature dependence of viscosity or relaxation times in polymers [30].

Another salient feature of glass-forming liquids is the decoupling between translational diffusion and viscosity, and between rotational and translational diffusion [31, 32, 33]. Translational and rotational diffusion coefficients are related to the viscosity through the familiar Stokes-Einstein-Debye equations as [34]

$$D_T = \frac{k_B T}{6\pi\eta R} \tag{2.2}$$

$$D_R = \frac{k_B T}{8\pi\eta R^3} \tag{2.3}$$

where D_T and D_R are the translational and rotational diffusion coefficients, re-

spectively, and k_B is the Boltzmann's constant, T is temperature, R is radius of sphere, and η is viscocity. At higher temperatures, both D_T and D_R are proportional to T/η in glass-forming liquids, in agreement with the Stokes-Einstein (SE) and Stokes-Einstein-Debye (SED) equations [35]. However, upon deep supercooling (approximately below $1.2\,T_g$), several experiments [33, 36, 37] have revealed that the inverse relation between D_T and η breaks down. Molecules are observed to translate faster than what is expected based on their viscosity, resulting in translational diffusion that is two orders of magnitude faster than predicted from the measured viscosity [35]. On the other hand, the inverse relation between the rotational motion and viscosity continues to conform reasonably well near T_g [35]. This means that, as the temperature is lowered, molecules on average translate progressively more for every rotation they execute [38], the consequence of which is the decoupling of translational and rotational motion.

Another characteristic feature of glass-forming liquids is the non-exponential relaxation behavior of macroscopic responses to perturbations. Experimentally, there are numerous ways in which a system of interest can be perturbed from its equilibrium state for the investigation of structural relaxations [39]. These include mechanical stress where, e.g., the stress in response to an imposed deformation is measured, electrical stress where, e.g., the dielectric relaxation or polarization is measured in response to an applied elctric field or voltage jump, and a thermal stress where, e.g., the fluctuation in enthalpy, volume or entropy is measured after a temperature jump. All these responses involve some sort of molecular rearrangements, and are used to study the relaxation behavior.

Almost universally, all measured response or correlation functions of viscous liquids close to T_g exhibit a non-exponential relaxation [23]. The temporal behavior of these non-exponential response functions are often described by the streched exponential, or Kohlraush-Williams-Watts (KWW) [40] function

$$\Phi(t) = \exp[-(t/\tau)^{\beta}], \tag{2.4}$$

where $\Phi(t)$ represents the correlation function that measures the fluctuation of a given physical quantity, τ is the characteristic relaxation time, and β is the stretching exponent, which measures the extent of exponentiality. For example, if $\beta = 1$ then an exponential behavior is recovered. For many glass formers, the KWW equation describes rather well a major portion of the primary relaxation process that is responsible for the glass transition [23]. Deviations from the KWW form are often found in the short time ($t << \tau$) and in the long time ($t >> \tau$) regimes [8]. Apart from these deviations, KWW type decays appear to be a universal behavior of relaxation in disordered matter.

In general, the above discussed canonical features of glass-forming liquids may be interconnected with one another, and, of course, with the dramatic slowing down of dynamics upon deep supercooling. Next, we discuss some of the theories that have been somewhat successful, and widely applied, in explaining different aspects of the above relaxation behaviors.

2.2 Theories of glass formation

Theories of glass formation can be catagorized under thermodynamic or dynamic view points. In the thermodynamic view point, the experimentally observable glass transition is viewed as a kinetically controlled manifestation of an underlying singularity [5], which is explained in purely thermodynamic terms. This view point underlies both the entropy and the free volume theories, where the entropy theories view the experimentally observable glass transition as a manifestation of an underlying entropy crisis at a temperature known as the Kauzmann temperature T_K [5]. This is a temperature at which the entropy difference between the liquid and the corresponding crystal vanishes. When a liquid is cooled below its melting temperature, the entropy of the liquid S_{liquid} decreases at a much faster rate than that of the crystal S_{crystal}. One expects that, upon extrapolation of the liquid's entropy to much lower T, the difference between S_{liquid} and S_{crystal} vanishes at a nonzero temperature T_K [41]. Upon further extrapolation for $T < T_K$, S_{liquid} becomes lower than S_{crystal} for the same T.

Although it is strange to imagine a liquid with smaller entropy than crystal, thermodynamics places no restriction on the sign of the entropy difference between the liquid and the crystal. In fact, a system of hard spheres, e.g. colloids, freezes to a solid that has higher entropy than the liquid [5]. The entropy crisis rather arises from further extrapolation of the crystalline and the amorphous entropies much below T_K, assuming such an extrapolation is possible. In this case, $S_{\text{crystal}} \to 0$ as $T \to 0$. The entropy of the disordered phase, on the other hand, would approach

negative values. Negative entropies are inconsistent with the classical expression $S = k_B ln\Omega$, since the number of accessible states Ω can not be less than unity. This impossible scenario, first pointed out by Kauzmann in 1948 [42], constitutes the entropy crisis and has come to be known as the Kauzmann paradox since then.

As a resolution to this paradox, T_K is viewed as an absolute limit, or underlying singularity, below which the liquid can not exist. Succeeding to avoid crystallization thus far, the supercooled liquid can only escape the apparent entropy crisis by undergoing a sharp glass transition at T_K. For this reason T_K is called the ideal glass transition temperature [43]. In practice however, T_K is not attainable experimentally, because vitrification intervenes at some higher temperature $T_g > T_K$.

An important critique of the validity of the extrapolation of the liquid entropy below T_g was put forward by Stillinger [44] based on the energy landscape paradigm. This paradigm is based on the idea, first proposed by Goldstein [45], that molecular motions in a deeply supercooled liquid consists of thermally driven anharmonic vibrations about deep potential energy minima, and infrequent transitions between different minima. Stillinger and co-workers [46, 47, 48] extended this idea by developing the concept of inherent structures, which are the local potential energy minima in a multidimensional potential energy hypersurface (or 'energy landscape') about which the system vibrates.

For the simple case of N particles with no internal degrees of freedom, the multidimensional hypersurface is a $3N + 1$-dimensional object in which each point represents the coordinates of the N particles and the corresponding value of potential energy U, i.e., $(U, \mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_N)$ [45]. The implication of this paradigm is that

vibrational and configurational contributions to the supercooled liquid's entropies (or other properties) are separable, where the configurational entropy is associated with the number of accessible basins in the potential energy hyperspace [44]. A basin is defined as a collection of points in the multidimensional hypersurface that lead to the same inherent structure upon steepest descent quenching of configurations [44]. Based on the examination of the basin enumeration function $\gamma(u)$ that is defined as the number of potential energy basins whose depth lies in the range between $u \pm du/2$, where $u \equiv U/N$, i.e., total potential energy per particle, Stillinger concluded that in nonpolymeric supercooled liquids the rate of entropy loss predicted by extrapolating liquid properties from above T_g cannot persist indefinitely. Instead, as $T \to 0$, the entropy of the supercooled liquid approaches zero smoothly, resulting in a sharp change in the entropy versus temperature curve.

The arguments for the existence or lack of the ideal glass transition are not yet definitive [5]. Nevertheless, in many current studies, the ideal glass transition temperature is estimated from extrapolation or fitting of measurements of transport coefficients such as η or D as a function of T, and is used as a reference temperature for the lower limit of transformation from a liquid to a disordered solid state. In general, the entropy-based view point of vitrification aims at quantifying the above pictures. An important theory in this category that has proven to be useful in the interpretation of the transport and relaxation properties of supercooled liquids is the Adam-Gibbs theory [12]. This theory provides a relation between the relaxtion time τ (equivalently, the viscosity η or the self diffusion coefficient D) and the configurational entropy $S_{\rm conf}$ of the system as

$$\tau \propto \exp\left(\frac{C}{TS_{conf}}\right) \tag{2.5}$$

where C is a constant. In the derivation of the above equation, Adam and Gibbs invoked the concept of a cooperatively rearranging region (CRR), which is defined as a group of molecules that, as a result of energy fluctuations, rearranges itself into different configurations independently of its environment [5]. The minimum size of the cooperatively rearranging region $z*$ is related to the system's configurational entropy through the relation

$$z* = \frac{s^*_{conf} N_A}{S_{conf}}. \tag{2.6}$$

In this equation, $z*$ is the minimum size of a CRR in one mole of molecules that consists of $n = N_A/z$ cooperatively rearranging regions, which are asssumed to have equal sizes. Each of these regions contributes s_{conf} to the configurational entropy S_{conf} of the whole system so that $S_{\text{conf}} \approx n s_{conf}$. This additivity follows from the assumption that a cooperatively rearranging region is independent of its surroundings. Based on this concept Adam and Gibbs derived an expression for the relaxation time τ as

$$\tau \propto \exp(\frac{A}{T - T_K}). \tag{2.7}$$

This equation is identical to the VTF equation (Eq. 2.1) discussed in the last section, with the identification of the Vogel temperature T_0 to be the same as T_K. In general, the Adam-Gibbs theory interprets the slowing down of dynamics in terms of the decrease in the number of configurations that the system is able to sample upon deep supercooling. That is, as the liquid is supercooled, the cooperatively rearranging regions grow, and relaxation requires the concerted participation of a larger and larger number of particles that progressively increases with decreasing temperature. This increased cooperativity upon cooling is reflected in the loss of configurational entropy, which in turn is manifested by the increase in the relaxation time or viscosity. Since the entropy difference between the supercooled liquid and the crystal, or equivalently the configuratioanl entropy $S_{\rm conf}$, vanishes at the Kauzmann temperature, the theory thus predicts structural arrest to occur at T_K.

Another theory in the category of the thermodynamic view point is the free volume theory proposed by Turnbull and Cohen [14], and later revisited by Grest and Cohen [15, 16]. According to the theory, vitrification occurs when the volume available for transitional molecular motion falls below a critical value. If v is the volume per molecule, and v_o is the volume per molecule excluded from all other molecules, the excess volume is $v - v_o$. A part of the excess volume that can be randomly distributed without any energy penality is referred to as the free volume v_f [49]. Accordingly, the ideal glass transition is viewed as a phenomenon that occurs when the free volume v_f vanishes [14], and thus the excess volume falls below the critical volume δv_c, which is part of the excess volume that can not be randomly distributed. The main result of the Cohen-Turnbull theory is the relation between

the translational diffusion coefficient and free volume, i.e., $D \propto \exp(-\frac{C}{v_f})$, where C is the product of a geometric factor that corrects for the overlap of the free volume and the minimum free volume capable of accomodating another molecule after the original displacement in the cage. The theory predicts that D, and hence the possiblity for translational motion, will vanish when $v_f = 0$.

The above models of glass formation predict thermodynamic transitions at temperatures below the laboratory glass transition temperature T_g. There are also other models that similarly predict a low-temperature phase transition. Some of these models are based on the study of spin glasses that are known to have much similarity with that of structural glasses. By making an analogy with the spin-glass problem, Binder and Young [50] argued for a thermodynamic transition at T_K. Kirkpatrick, Thirumalai and Wolynes concluded that the universal behavior of supercooled liquids arises from proximity to the underlying random first-order transition which is found in the mean-field theories of spin glasses [51]. Another model that predicts a low-temperature phase transition for glass formation and that is not based on the spin glass models is the recent theory by Colby [27], where the author utilized a critical phenomena description of phase transitions for understanding glass formation. Using the idea that free volumes diffuse randomly, the author constructed a scaling description of glass formation, in which he predicted a phase transition at a temperature T_c roughly 10 K below T_g.

A nonthermodynamic view of glass formation that can be categorized under the dynamic viewpoint describes the glass transition as a purely dynamic singularity that occurs upon deep supercooling. Mode-coupling theory belongs to this

class. In simple terms, this theory views vitrification as a transition from ergodic to nonergodic behavior in the relaxation dynamics of density fluctuations [17]. The theory starts from well known microscopic dynamics [52], and derives a set of dynamic equations for the density correlation function $F(q,t)$, more precisely for $\Phi_q(t) = F(q,t)/S(q)$. The equation gives a self-consistent description of the dynamics of particles in the system. The solution of this equation is completely determined by the knowledge of the static structure factor $S(q)$ which, in principle, can be calculated from the microscopic interactions by means of statistical mechanics.

The main predictions of MCT is that for certain values of density and temperature, Φ decays to zero, while for other values it decays to a finite, nonzero number. The former condition is identified with the liquid or ergodic behavior, while the latter is identified with glass, i.e., non-ergodic behavior. MCT thus predicts a critical temperature $T_{\rm MCT}$ (or critical density) where a dynamic transition from an ergodic to nonergodic phase takes place. Since this transition is accompanied by a diverging relaxation time of the time correlation functions, it was identified with the glass transition temperature. However, such a sharp transition, with the exception of colloidal systems [53], has never been observed either in simulations or experiments [54]. In fact, in most real systems $T_{\rm MCT} \approx 1.2\, T_g$.

The lack of a singularity at $T_{\rm MCT}$ and the breakdown of the ideal MCT near $T_{\rm MCT}$ is attributed to the onset of hopping processes that restore ergodicity below $T_{\rm MCT}$, but that are neglected in the idealized version of the theory. The extended version of the theory [55] incorporates this effect, but, lacks a detailed microscopic description. This model is an ad hoc addition to the ideal MCT, and the relevant

coupling parameters have to be determined by fitting to experimental data [8]. Despite its failure to predict the glass transition temperature T_g, MCT succeeds in describing many aspects of the relaxation behavior of moderately supercooled liquids. In fact it is regarded as the only theory so far that, starting from first principles, succeeds in describing the transport properties of supercooled liquids above T_{MCT}.

Another alternative view point of glass formation that has also contributed to the current understanding of some aspects of supercooling and the glass transition is the frustration limited domain theory of Kivelson and co-workers [18]. Similar to MCT, the interpretations of this model invoke a narrowly avoided singularity above T_g. The basic physical ingredient of the model is the concept of structural frustration. According to the theory, molecules in a liquid tend to arrange themselves into a locally preferred structure that minimizes the local free energy. This preferred local structure is different from the structure in the actual crystalline phase, and the spatial extension of it is prevented because of geometrical frustration owing to the fact that the local structure does not tile space. An example of such energetically favored but non-space-tiling local structure is the icosahedral packing. This geometric frustration causes the system to build up strain. Below some temperature T^*, because of the competition between the short-range tendency to order and the strain generated by the frustration, the liquid breaks up into frustration-limited domains, thereby avoiding a phase transition (singularity) at T^*. The avoided transition temperature T^* acts as a critical point, below which two length scales emerge. One is the locally preferred, but strained, frustation-limited domain, and the other is a

domain that governs density fluctuations in the absence of frustration. With this model, reasonable agreement between the prediction of the theory and experimental data has been demonstrated on the temperature dependence of viscosity [18], but at the expense of introducing a number of fitting parameters [56].

The theories of glass formation are by no means restricted to the above models. There are still several other models that are relevant in improving our understanding of the relaxation properties of glass-forming liquids. Some of these models (e.g., the coupling model of Ngai and coworkers [57], the replica model of Parisi and co-workers are long-standing, while others (e.g., the trap model of Bouchaud and co-workers [58], the cluster model of Fan and Fecht [59], the non-topographic view of Garrahan and Chandler [60, 61], etc.) are catagorically new emerging theories. The non-topographic model of Garrahan and Chandler appears to be relevant in describing some aspects of our computational results. We will, therefore, discuss this model later in the context of our results.

2.3 Spatially heterogeneous dynamics

From the wide range of models and theories that exist, it may be reasonable to expect that our search for understanding the mystery of glass formation is still far from being over. However, recently some important concepts are gaining nearly universal acceptance in almost all new emerging theories, as well as in explaining several experimental results related to the study of glass formation. The most prominent ones are the concepts of spatially heterogeneous dynamics (or dynamical heterogeneity)

and cooperativity. In fact, these two concepts are essentially inseparable in almost all instances. In some cases, however, the concept of cooperativity is discussed with out invoking heterogeneity, e.g., the Adam-Gibbs theory.

Spatially heterogeneous dynamics (SHD) refers to the fact that upon deep supercooling different regions within a liquid consists of subensembles of particles that exhibit temporarily enhanced or diminished mobility relative to the average. These regions can be only a few nanometers away from each other [32]. Those particles with higher mobility than the average are commonly referred to as "mobile" or "fast" particles, while those with low mobility are referred to as "immobile" or "slow" particles. Although not stated explicitly, the concept of dynamical heterogeneity is not an entirely new idea. For example, the "two-fluid" model of Cohen and Grest [15, 16] in their free volume theory involves such an idea, where the existence of "solidlike" and "liquidlike" regions have been hypothesized. Stillinger et al., on the other hand, proposed a "fluidized domain model", where they presented a description of diffusive motion in strongly supercooled liquids as spatially localized thermal excitations of particles in an otherwise solid-like matrix [35].

The concepts of dynamical heterogeneity and/or cooperativity are invoked for providing explanations for an increasing number of experimental observations on the macroscopic properties of supercooled liquids and glasses [32, 62]. In particular, the notion of heterogeneity primarily arose from an effort to rationalize the non-exponential relaxation responses discussed above. At the molecular level, the non-exponential relaxation can be explained by two fundamentally different scenarios. One can imagine that the deviation from an exponential pattern is due

to the presence of a heterogeneous set of environments that relax exponentially, with the relaxation times varying significantly among the different sets. This is the heterogeneous scenario. Alternatively, one can imagine that each molecule in the supercooled liquid relaxes nearly identically in an intrinsically non-exponential manner, such that the local and the ensemble-averaged dynamics will be the same. This is the homogeneous scenario.

The issue of homogeneous versus heterogeneous description of non-exponentiality has provoked quite a large number of experimental techniques that are targeted in understanding the macroscopic responses, and hence the nature of slowing dynamics, at the molecular level. The most important techniques [62] include the multi-dimensional nuclear magnetic resonance (NMR) [22], deep photobleaching, and dielectric and magnetic hole burning, which are collectively referred to as 'dynamic hole burning' experiments [32, 62]. These experimental techniques allow the dynamics of subensembles to be selectively observed, where the subensembles are selected based on the distribution of relaxation times.

The observation of a subensemble that has a relaxation time different from the ensemble average is already a proof of the presence of dynamical heterogeneity, albeit without spatial information [32]. The above experiments also characterize the persistence time, often referred to as the rate memory or rate exchange time, of a particular relaxation. This quantity measures for how long a particular subset survives without evolving into the average set, providing the lifetime of heterogeneities. Other experimental techniques performed to detect dynamical heterogeneity include methods such as solvation dynamics [63], atomic force microscopy [64], single-molecule

spectroscopy [65] and confocal microscopy [20, 21].

In all these experiments the dynamics of supercooled liquids approaching T_g is found to be dynamically heterogeneous. The emergence of dynamical heterogeneity not only explains the deviation from the exponential behavior of macroscopic responses, but also other canonical features. For example, the decoupling of translational and rotational motion upon supercooling has been related to the emergence of dynamical heterogeneity [32, 62, 66, 67]. Therefore, understanding the nature of dynamical heterogeneity is believed to be an important step towards understanding the molecular mechanisms leading to the formation of glasses [62]. Consequently, dynamical heterogeneity gained a great deal of attention in the search for the origin of slowing dynamics, which in turn became the source for several new questions [62]: How do we quantify dynamical heterogeneity? What is the best measure that reveals this behavior? How does this property change with T? Is there any transient nature in this dynamical anomaly? How big are the heterogeneities? What is the persistence time associated with it? How does the dynamics vary between the fastest and the slowest regions? And why are dynamics spatially heterogeneous?

Despite the overwhelmingly large experimental evidence for the existence of dynamical heterogeneity, the spatial character of heterogeneity has been inferred unambiguously by only a few experiments. The first experiment to quantify a length scale of heterogeneity is the experiment of Tracht et al. that combined multidimensional NMR with the effect of spin diffusion [68]. This experiment makes use of the transfer of magnetization between ^{13}C and ^{1}H nuclei via cross polarization that is only efficient for nuclei that are sufficiently close to each other [32]. In this work,

the characteristic size ξ_H of the slow region of the heterogeneous dynamics is found to be in the range of $2-4$ nm for polyvinylacetate at $T_g + 10$ K [68]. Subsequently, similar measurements were conducted by Reinsberg et al. [69] on glycerol. They found that $\xi_H \approx 1$ nm. However, these experiments did not provide information on the temperature dependence of this length scale, which, in fact, is an important piece of information needed for understanding the fundamental origin of slow dynamics near T_g. One would like to know if there is a growing length scale associated with these dynamical regions upon cooling towards T_g.

The existence of a growing length scale had been an issue for some time [70]. Despite theoretical predictions for a growing range of correlation length, e.g., the CRR model of the Adam-Gibbs theory, there had been a number of unsuccessful attempts to unambiguously identify such regions. For example, Ernst et al. [71] examined the relaxation of near neighbor pairs and bond orientational correlations using computer simulation and found no indication of a growing length upon cooling. A similar result was reported by Dasgupta et al. [72] using a four-point spatial correlation function. The main challenges in identifying such length scales is in finding a suitable correlation function that clearly signifies any increase in length scale upon supercooling. Recently, a number of correlation functions and their corresponding generalized susceptibilities have been devised to search for growing length scales. One of the pioneering work in the search for such length scales is that of Mountain [70], where the author found a hydrodynamic length L that is rapidly increasing with decreasing T [73].

A more direct approach in which a specific domain is selectively studied was

used by Glotzer and co-workers [74, 75, 76, 77, 78, 79, 80, 81, 82]. For example, by studying the slow particles in the simulation of binary LJ mixtures [83], Lacevic et al. analyzed the properties of the generalized susceptibility $\chi_4(t)$ that is related to the time-dependent, four-point density correlation function $G_4(\mathbf{r}_1, \mathbf{r}_2, t)$. The main result of their study revealed a growing spatial correlation between localized particles as T approaches $T_{\rm MCT}$. Additional evidence for an increasing length scale associated with solid-like properties in supercooled liquids has also been reported in Refs. [84, 85].

To study the regions consisting of mobile particles, a generalized susceptibility $\chi_U(t)$ that corresponds to the displacement-displacement correlation function $G_u(\mathbf{r}, t)$ has been proposed [78, 80]. This quantity reflects the correlation of highly mobile particles since, by construction, particles with large displacements contribute most to the function. The analysis of $\chi_U(t)$ indicates that dynamics in the mobile domains is also highly correlated over a distance that grows upon cooling towards $T_{\rm MCT}$. $\chi_U(t)$ is found to have a maximum at shorter time scales than $\chi_4(t)$.

The growing dynamical correlation length quantified by the generalized susceptibility $\chi_U(t)$ is found to be in agreement with the simulation results obtained by directly analyzing the size of clusters formed by the mobile particles [76]. It has been found that highly mobile particles move cooperatively forming dynamical clusters whose length scale increase with decreasing T towards the glass transition temperature T_g. This is actually one of the central themes of our research where we further investigate the transient nature and temperature dependence of these clusters in an effort to understand the spatial and temporal aspects of dynamical heterogeneity.

Such a direct observation of correlated motion that is growing upon cooling is an interesting discovery since, at least conceptually, it confirms the notion of cooperativity as a cause for the slowing down of dynamics upon cooling, as proposed by the phenomenological CRR theory of Adam, Gibbs, and DiMarzio [12, 13].

The above finding regarding the clustering of mobile particles has also been confirmed experimentally by Weeks et al. [21] using confocal microscopy on dense colloidal particles. Analogous to molecular liquids, colloids exhibit a glass transition. In these materials, the parameter controlling the glass transition is density instead of temperature. As the density increases beyond some critical value, particles in the dense liquid become trapped within the "cage" formed by their neighbors leading to the colloidal glass transition. Hard sphere colloids are known to obey several predictions of MCT [53], and are good candidates for studying glass transition phenomena using confocal microscopy. In particular, if dyed with appropriate labels, the individual particle motion can be traced in real space as a function of time by rapid imaging through confocal laser scanning microscopy, and thus provide microscopic information that can be utilized to investigate the cooperative nature of dynamics in these systems. Accordingly, Weeks et al. [21] demonstrated that highly mobile particles move in cooperatively rearranging dynamical clusters whose length scales increase with increasing density towards the corresponding glass transition density.

As will be shown in later chapters, a closer inspection of these clusters reveals that within any cluster of mobile particles, smaller subsets move together in a correlated way whereby several particles replace each other along quasi-one dimensional paths forming elementary units referred to as "strings". These strings appear to

be the rudimentary element of cooperative motion, and are found to play an important role in different aspects of dynamical heterogeneity (see chapter 6 for more detail). It is therefore essential to conduct a detailed investigation on the nature and formation of these elementary units. It is our belief that understanding these dynamical units will have contribution to our understanding of the nature of cage rearrangement and cooperative motion, which in turn may shed light on the origin of dynamical heterogeneity.

As mentioned earlier, despite a large number of theoretical, computational and experimental advances, there are many issues of dynamical heterogeneity that remain to be understood, the main challenge being understanding the origin of dynamical heterogeneity itself. To unveil this challenging phenomenon, it is necessary to trace particle motions at the microscopic level. Because strings organize to form larger, ramified clusters that represent one of the domains of a dynamically heterogeneous system, a microscopic understanding of the local rearrangement of particles that form strings will be a prerequisite to rationalizing the macroscopic properties observed in glass-forming liquids. In fact, as will be described in later chapters, these dynamical objects are relevant in the development of new emerging theories as well as in understanding some aspects of the well-established theories of glass formation. Therefore, part of our research is dedicated to tracking particle motion at the microscopic level, and understanding *how* particles are moving upon supercooling. This is, in fact, a ground work for understanding *why* particles are moving in a certain way. Is there any driving force, local structure, or other factor that makes the particles reorganize in a certain way? In order to answer these and similar other

questions, one needs to know in the first place *how* the particles are rearranging. A combination of these two may provide, hopefully, a rigorous account for the origin of dynamical heterogeneity, and hence the slowing down of dynamics.

Chapter 3

Models of glass-forming liquids

In this thesis, we study in detail two different models of glass-forming liquids - a bead-spring polymer melt and a Dzugutov liquid. This chapter is devoted to the description of these systems. First, the models are described, and then we discuss their static properties, measured in terms of the pair correlation function $g(r)$ or the static structure factor $S(q)$, and their dynamic properties, measured in terms of the mean square displacement (MSD), the structure factor $F(\mathbf{q}, t)$ and the van Hove correlation function $G(\mathbf{r}, t)$. Prior to this we briefly describe the basics of the molecular dynamics simulation method that we used to generate our data.

3.1 Simple models for glassy dynamics

The relaxation behavior of glass-forming liquids has been studied computationally using a wide range of model systems that are conceived to imitate a particular class of material, or a specific substance. In addition to the systems of polymer melt and the Dzugutov liquid we studied, some other examples that model a class of material include the binary Lennard-Jones (LJ) mixtures [75, 86], and the 2D and 3D binary-mixtures of purely repulsive soft-core potentials [87, 88, 89]. One example of the latter model is the Weeks-Chandler-Andersen (WCA) [90] model in which a repulsive potential is obtained from the application of purturbation theory to split the LJ potential into repulsive and attractive components. Other simple models that have proved to be useful for studying glass-forming liquids are those resulting from the hard-sphere potentials [91]. These models are often considered as representatives for colloidal systems [91]. It is also interesting to note that simple models like spin glasses [50, 51, 92] and lattice gas models [93] have been utilized in studying glass forming behaviors. Spin glasses are considered to reproduce phenomenologically several features of structural glasses, and have a considerable impact in understanding glassy phenomena [51, 92, 94].

Specific materials like water [95, 96], silica [97, 98], lithium metasilicate (Li_2SiO_3) [99] and others have also been modeled for studying glassy dynamics. In all these models one first selects a generic potential that describes a given class of material. The parameters in the potential are then tuned to reproduce the macroscopic property, e.g., the structure factor, of the real material to be studied. These parameters

may be obtained either from experimental data, or from *ab-initio* calculations, and their validity is checked at different state points within a particular ensemble (NPT, NVT, etc). In this way, different classes of materials as diverse as molecular liquids, colloids, polymers, ionic glasses, oxides and others have been explored in an effort to understand the relaxation behavior of supercooled liquids and glasses. Once the models are set, a simulation method is selected among the different techniques available that are applicable to the problem. Traditionally, this field is investigated using molecular dynamics (MD) and Monte Carlo (MC) simulation methods, with few exceptions of Brownian dynamics for colloids (BD).

Our analysis of the dynamic and thermodynamic properties of supercooled liquids is based on the data obtained from MD simulations of a polymer melt, and a one-component liquid obtained from the Dzugutov potential (to be described shortly). The latter was conceived as a structural model for metallic liquids. Our choice for these systems is motivated by our desire to investigate the clustering behavior and string-like motion of a homogeneous system that does not involve different compositions as in, e.g., the binary LJ mixture, although the primary investigations of these dynamical behaviors were done using the binary LJ system. For a detailed study of these phenomena, which we intend to accomplish, one would like a system in which composition is not an issue. Additionally, the observations made in the LJ system should be tested in different systems to establish the universality of the phenomena across a variety of systems.

Historically, we began exploring these dynamical anomalies on a system of polymer melts using data generated by Bennemann et al. [100]. Due to their struc-

tural complexity, polymers constitute a class of material that have reduced crystallization tendency. Upon cooling, several polymeric materials undergo a glass transition, and hence they are good candidates for studying glass forming behavior. However, the issue of connectivity present in the polymer melts motivated us further to study a clean one-component glass-forming system. The Dzugutov liquid is one good choice in that respect. Following the work of Dzugutov et al. [101], we generated data for this system. This system has been studied by Dzugutov and co-workers for its glass-forming behavior, and it has been found that it behaves as a typical glass former [101, 102, 103].

3.2 Molecular dynamics simulation

Molecular dynamics simulation is a method by which the classical equation of motion of a system of interacting particles are solved. Given the initial coordinates and momenta of a set of particles, the position and momenta of the particles at any later time are obtained by solving Newton's equations of motion, i.e., for a system of N particles one solves a set of $3N$ coupled, second order, differential equations

$$m_i \ddot{\mathbf{r}}_i = -\nabla V_N(\mathbf{r}^N) \tag{3.1}$$

where m_i and $\ddot{\mathbf{r}}_i$ are the mass and the acceleration of particle i. When the interaction potentials $V_N(\mathbf{r}^N)$ are known, the positions and velocities of all particles are calculated by numerically integrating the above equations of motion using a

finite difference method. There are several algorithms for integrating the equations of motion using finite difference methods [104, 105]. These include the Verlet algorithm [106], the leap-frog algorithm [107], the Beeman algorithm [108], etc. Among these, the Verlet algorithm is probably the most widely used integration algorithm [104]. The integration algorithms use an appropriate integration time step δt that is large enough to cover a given amount of phase space, but small enough to guarantee stability. The maximum time step chosen in the MD simulations is limited by the time scale of the shortest motion in the system, e.g, the mean time between collisions in an atomic fluid, or the shortest period of vibrations in molecular or polymeric liquid.

MD simulations are carried out in a particular ensemble, such as the micro-canonical (NVE), canonical (NVT) or isothermal-isobaric (NPT) ensemble. The choice of a specific ensemble is dictated by the nature of the problem under study. For example, a constant temperature simulation is required if we wish to understand how the behavior of a system changes with temperature while its density remains fixed. Examples of such problems include the unfolding of a protein. To simulate a system under constant T or constant P conditions (or both), the system will be coupled to an "external reservoir", e.g., heat bath or piston, that is designed to regulate these quantities as in real experiments. Such methods are commonly referred to as thermostat or barostat, where different methods like the extended system method (e.g. Nosé-Hoover thermostat [109], Anderson barostat [110], Parrinello-Rahman barostat [111]), or scaling method (e.g., Berendsen thermostat/barostat [112]) are utilized to maintain T or P (see Ref [104, 105] for details).

For a given ensemble, the positions and velocities of all particles are calculated at every time step, from which the trajectories (and sometimes velocities) of the particles are saved at a predetermined time interval for further analysis. The static and dynamic properties of the system are finally determined using the generated data. On the other hand, since many thermodynamic quantities are expressible as statistical averages of certain functions of the coordinates and momenta, the thermodynamic properties of the system can also be obtained from the generated data. That is, if $\mathscr{F}(\mathbf{r}^N, \mathbf{p}^N)$ is a function of the $6N$ dimensional phase space $(\mathbf{r}^N, \mathbf{p}^N)$, and if F is the associated thermodynamic quantity, then

$$F = \left\langle \mathscr{F}(\mathbf{r}^N, \mathbf{p}^N) \right\rangle, \tag{3.2}$$

where the angular bracket denotes the statistical ensemble average. For an ergodic system this is equivalent to a time average

$$F = \langle \mathscr{F} \rangle_t = \lim_{\tau \to \infty} \frac{1}{\tau} \int_0^\tau \mathscr{F}[\mathbf{r}^N(t), \mathbf{p}^N(t)] dt. \tag{3.3}$$

In computer simulations, this equivalence is exploited to calculate several statistical averages, where the integral is replaced by discrete time steps. Therefore, information on thermodynamic quantities such as temperature, pressure, potential energy, and many more can be easily extracted from the time average of the corresponding instantaneous functions. For example, temperature and pressure are calculated from

the instantaneous temperatures $\mathscr{T}(t)$ and pressures $\mathscr{P}(t)$ that are derived using the $3N$ coordinates and momenta generated by the simulation as follows:

$$T = \langle \mathscr{T}(t) \rangle_t = \frac{1}{\mathrm{N_{MD}}} \sum_{t=1}^{\mathrm{N_{MD}}} \mathscr{T}(t) \tag{3.4}$$

where,

$$\mathscr{T}(t) = \frac{1}{3Nk_B} \sum_{i}^{N} \frac{1}{m_i} |\mathbf{p}_i(t)|^2, \tag{3.5}$$

and N_{MD} is the total number of time steps. Note that Eq. 3.5 is a result of the virial theorem. In a similar way, the pressure P is obtained from the virial equation expressed as

$$\frac{\beta P}{\rho} = 1 - \frac{\beta}{3N} \left\langle \sum_{i=1}^{N} \mathbf{r}_i(t) \cdot \nabla V_N[\mathbf{r}^N(t)] \right\rangle \tag{3.6}$$

where the averaging is done as in Eq. 3.4.

3.3 Simulation models

In this section we describe the two models we investigated. The first model is polymer melt, and the second model is a one-component Dzugutov liquid.

3.3.1 Polymer melt

We studied a coarse-grained, bead spring model of a polymer melt to investigate the spatially heterogeneous motion of monomers. This is a simplified model in which chains with chemical monomers and realistic potentials are replaced by chains of beads (LJ particles) connected by nonlinear springs. The system consists of 120 polymer chains, each of which is composed of 10 monomers (beads) with mass m set to unity. All monomers interact by means of a truncated LJ potential [105],

$$V_{LJ}(r) = \begin{cases} 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] + C, & \text{if } r \leq 2r_{\min} \\ 0, & \text{if } r > 2r_{\min} \end{cases} \tag{3.7}$$

where C is a constant that guarantees the potential vanishes continuously at $r = 2r_{\min}$, $r_{\min} = 2^{1/6}\sigma$ is the position where the LJ potential is minimum. The parameters ϵ and σ are set to unity. These parameters define the well depth of the potential, and the hard-core diameter of the LJ particles, respectively. In addition to the LJ potential, the nearest-neighbor monomers along the backbone of a chain are bonded to each other by a finitely extensible, nonlinear elastic (FENE) potential given by [113]

$$V_{FENE}(r) = -(k/2)R_0^2 \ln\left[1 - (r/R_0)^2\right]. \tag{3.8}$$

The parameters of the FENE potential are chosen as $k = 30$ and $R_0 = 1.5$ [113]. These values guarantee a certain stiffness of bonds while avoiding high-frequency

modes and crossings. For these values, the superposition of the two potentials (LJ and FENE) leads to a steep effective bond potential with a minimum at about $0.96\,\sigma$ [114]. This minimum is slightly smaller than the length favored by the pure LJ potential. The presence of these two incompatible preferred length scales prevents long-range ordering (i.e., crystallization) at low temperatures by causing packing frustration.

For this system, the radius of gyration R_g varies very little over the whole temperature range, i.e., R_g ranges between $2.09 \leq R_g \leq 2.23$ for a temperature range as large as $0.48 \leq T \leq 2.0$ [114]. This indicates that the chains do not show any tendency of becoming stiffer as the temperature is lowered. Note that, because the simulation was not intended for a specific polymer, all the quantities mentioned above are quoted in reduced units. It is a common practice in computer simulations to express quantities in dimensionless reduced units unless one is interested in simulating a particular system. In that case, the reduced units may be converted to their corresponding actual dimensions. In general, in the reduced units, length is expressed in units of σ, temperature in units of ϵ/k_B, and time in units of $\sigma\sqrt{m/\epsilon}$. For Argon atom these units correspond to $\sigma = 0.34$ nm, $\epsilon/k_B = 119.8\ K$, $m = 39.95$ g mol^{-1} and hence time $\tau = 2.149$ ps.

The simulation data we analyzed consisted of eight state points with an average pressure $p = 1$, and temperature T ranging between $0.46 \leq T \leq 0.7$ [115]. The corresponding density ρ is adjusted between $1.04 \geq \rho \geq 0.98$ to follow the isobaric path. This simulation is done in two steps (see Benneman et al. [100] for details). For any temperature T, the simulation is first run under isobaric condition (NPT

ensemble) at $p = 1$ to obtain an equilibrium density corresponding to that state point. Then, the simulation is continued in the canonical (NVT) ensemble using the Nosé-Hoover thermostat at the fixed density obtained from the NPT ensemble. As a result, ρ is different for the different temperatures. Strictly speaking, this simulation method may be regarded as an NVT ensemble simulation with an isobaric cooling. But, because of the fact that the average pressure is monitored to be constant within 5% at all T, it may still be regarded as an NPT ensemble simulation as stated by the authors. In any event, Bennemann et al. [100] performed a comparative study of constant volume and constant pressure cooling methods, and the two methods show no qualitative difference in the measured observables. The integration of the equation of motion was performed using the Heun algorithm, with a time step of $\delta t = 0.002$ [100]. For reference, the ideal glass transition temperature is $T_0 = 0.34 \pm 0.02$, and the critical temperature T_{MCT} of the MCT is $T_{\mathrm{MCT}} = 0.45 \pm 0.01$ [100, 114].

3.3.2 Monoatomic Dzugutov liquid

The second model we investigated is a monoatomic system described by the Dzugutov potential [101], which is designed to prevent the nucleation of the ground state crystal structures. This potential evolved from a pair potential that was originally developed for metallic liquids [116]. In the original potential, the parameters were optimized to reproduce the static structure factor $S(q)$ of liquid lead measured in a neutron scattering experiment close to the melting point $T_{\mathrm{m}} = 623K$. This pair potential includes, in addition to terms describing the strong short-range interac-

tions and the usually predicted Friedel oscillation [117], a soft repulsive component representing the screened Coulomb repulsion between the ions.

In its present form, the Dzugutov potential is characterized by the presence of two repulsive regimes and one attractive region. Its main repulsive part is identical to that of the LJ potential, but the Dzugutov potential features an additional maximum at a range typical of next-nearest-neighbor coordination distances in closed-packed crystals [118]. This maximum suppresses crystallization by disfavoring closed-packed ordering. On the other hand, the maximum is located in a region between the distances bounded by the first and the second neighbor shells in the icosahedral polytope [101, 119], causing the preferred local order in the system to be icosahedral. This type of local order is known to play an essential role in the glass formation of some simple systems [119, 120, 121], since the 'frustration' inherent in packing icosahedra in Euclidean three-dimensional space makes it impossible to form a long-range crystalline structure in which each atom has such an environment. Hence, a system described by the Dzugutov potential is a good glass-former, and can be studied in the supercooled regime prior to nucleation of the crystal. Near and below T_g this system is known to exhibit a first sharp diffraction peak and a split second peak in the structure factor [102]. These are common features of metallic glasses, which have an inherent structure that can be accounted for by icosahedral coordination of the first neighbor shell. These systems are, however, multicomponent systems whose structure is mostly dictated by the presence of short-range chemical ordering. The Dzugutov liquid can thus be perceived as a one-component reference system for multicomponent metallic glass-formers, whose relaxation on supercooling

involves both topological and chemical ordering. Hence, the model provides a unique opportunity to separate the contribution of these processes to the formation of glasses.

The explicit form of the Dzugutov pair potential is expressed as [101]

$$\begin{aligned} V &= V_1 + V_2 \\ V_1 &= A(r^{-m} - B)\exp\left(\frac{c}{r-a}\right), r < a \\ V_1 &= 0, r \geq a, \\ V_2 &= B\,\exp\left(\frac{d}{r-b}\right), r < b, \\ V_2 &= 0, r \geq b, \end{aligned} \tag{3.9}$$

m	A	c	a	B	d	b
16	5.82	1.1	1.87	1.28	0.27	1.94

Table 3.1: Parameters of the Dzugutov pair potential.

where the parameters are compiled in Table 3.1. In Fig. 3.1, we plot the Dzugutov potential together with the LJ potential, where the latter is shifted up by an amount $0.419\,\epsilon$ to align the minima for the sake of comparison. Both potentials have minima at the same position, but unlike the LJ potential, the Dzugutov potential has an additional repulsive piece.

The Dzugutov model has been used in studies of supercooled liquids [101, 102,

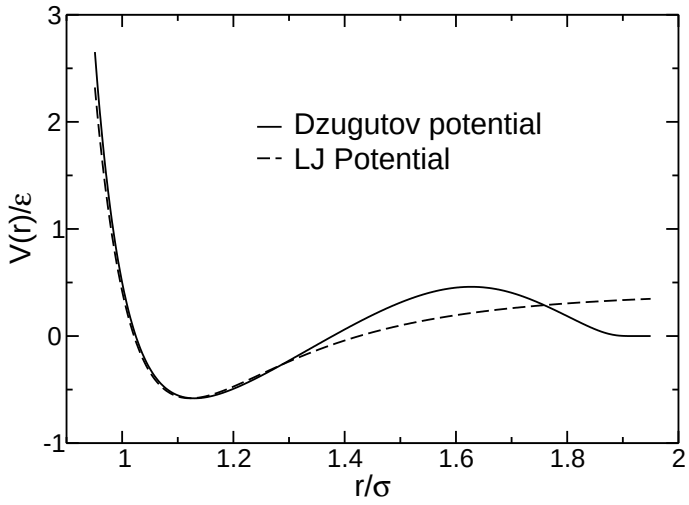

Figure 3.1: Dzugutov potential plotted together with LJ potential, where the LJ potential has been shifted up by $0.419\,\epsilon$ to emphasize that the two potentials have minima at the same position. The Dzugutov potential has a maximum at a distance $r \approx 1.6\,\sigma$.

103] as well as in simulations of freezing [122, 123], where the observed solid structure for sufficiently long relaxation upon supercooling is found to be a monoatomic dodecagonal quasicrystal. By construction, however, such transformation can be delayed, and the potential stabilizes the one-component liquid in a metastable supercooled state, allowing a time window long enough for the observation of the essential dynamical properties [102]. In terms of its glass transition behavior, the model is known to be a fragile liquid [102]. The supercooled regime, characterized by the super-Arrhenius slowing down of the diffusion coefficient D, is found to set in at around $T = 0.8$ [102]. The critical temperature of MCT, estimated from a power law fitting of D, is $T_{\mathrm{MCT}} = 0.4$ [102].

Our MD simulations are performed for a system of 17576 particles in a tem-

perature range 0.42 - 1.6. For all state points studied, the simulations are done under isothermal conditions using a Berendsen thermostat, and at a constant density $\rho = 0.85$ (NVT ensemble). Periodic boundary conditions are used in all three spatial directions. To prepare the system, the liquid is cooled and equilibriated in a stepwise manner starting from $T = 1.6$. At each T studied, several independent samples are prepared to improve statistics. All analysis of bulk dynamic properties is conducted over the entire range of T, however, our detailed study of string-like motion is restricted to the lowest temperature simulated, $T = 0.42$. The integration was done using the velocity Verlet algorithm, and the integration time step used in the simulation is 0.01. All units are quoted in LJ reduced units: length in units of σ, temperature T in units of ϵ/k_B and time in units of $\sigma\sqrt{m/\epsilon}$. The mass m and the distance σ are set to unity. The simulations carried out for the present study, prior to post analysis, required roughly 1300 cpu hours on a AMD Athlon 2000+ MP Myrinet cluster.

3.4 Static properties

3.4.1 Pair correlation function

A fundamental static property that can probe the structure of a fluid in real space is the pair distribution function $g_N^{(2)}(\mathbf{r}_1, \mathbf{r}_2)$. It measures the extent to which the structure of a fluid deviates from an ideal gas. For an isotropic and homogeneous system $g_N^{(2)}(\mathbf{r}_1, \mathbf{r}_2)$ is a function of only the separation $r_{12} = |\mathbf{r}_1 - \mathbf{r}_2|$, and $g_N^{(2)}(\mathbf{r}_1, \mathbf{r}_2)$

is simply written as $g(r)$. The pair correlation function $g(r)$ can be expressed in terms of the local particle density

$$\rho(\mathbf{r}) = \sum_{\mathbf{i=1}}^{\mathbf{N}} \delta(\mathbf{r} - \mathbf{r_i}) \tag{3.10}$$

as follows,

$$g(r) = \frac{V}{N^2} \left\langle \sum_{i=1}^{N} \sum_{j \neq i}^{N} \delta(\mathbf{r} + \mathbf{r}_j - \mathbf{r}_i) \right\rangle. \tag{3.11}$$

in which case $g(r)$ is related to the static two point density-density autocorrelation function $G(\mathbf{r})$ that is defined as

$$G(\mathbf{r}) = \frac{1}{N} \int \langle \rho(\mathbf{r}' + \mathbf{r}) \rho(\mathbf{r}) \rangle \, d\mathbf{r}' \tag{3.12}$$

$$= \rho g(\mathbf{r}) + \delta(\mathbf{r}). \tag{3.13}$$

The time dependent generalization of $G(\mathbf{r})$ is the van Hove correlation function that plays an important role in the description of the dynamic properties, which will be discussed in the next section.

In Fig. 3.2(a) and 3.2(b) we show $g(r)$ for the two models we studied as calculated using Eq. 3.11. The overall shape of these correlation functions resembles that of typical liquids [105], where one observes a sharp first nearest-neighbor peak followed by smaller, oscillating peaks that decrease and asymptotically approach a value of one at large r. However, each of the pair correlation functions shows a distinct signature that is peculiar to the system under investigation. For example,

$g(r)$ of the polymer melt (Fig. 3.2(a)) exhibits a split in the first nearest neighbor peak reflecting the two competing preferred length scales of the model. The first of the two peaks is due to the length scale corresponding to the bond length $r_{\mathrm{bond}} = 0.96$ [114], while the second is due to the minimum of the LJ potential at $r_{\mathrm{min}} = 2^{1/6}$. The temperature dependence of $g(r)$ is also shown in the figure. It is clear from the figures that, upon cooling towards T_g, the structural change depicted by $g(r)$ is not so dramatic. As will be shown in the following sections, this is in contrast to the dynamic changes observed on cooling, presenting an example for the main source of the challenge in glass formation discussed in the introduction.

The radial distribution function $g(r)$ of the Dzugutov liquid is shown in Fig. 3.2(b), where a first sharp peak and a splitting of the second peak of the pair correlation function is manifested characterizing typical metallic liquids. The splitting of the second peak becomes more pronounced near T_{MCT}, reflecting the peculiar feature of these systems. In general, $g(r)$ provides the information, in real space, that the structure of the systems under study are representatives for models of a polymer melt (Fig. 3.2(a)) and a structural metallic liquid (Fig. 3.2(b)).

3.4.2 Static structure factor

Another static quantity that provides information on the structure of a fluid, and that can be measured by experiments like X-ray or neutron scattering is the static structure factor, $S(\mathbf{q})$. It is defined as

$$S(\mathbf{q}) = \frac{1}{N} \langle \rho(\mathbf{q}) \rho(-\mathbf{q}) \rangle \tag{3.14}$$

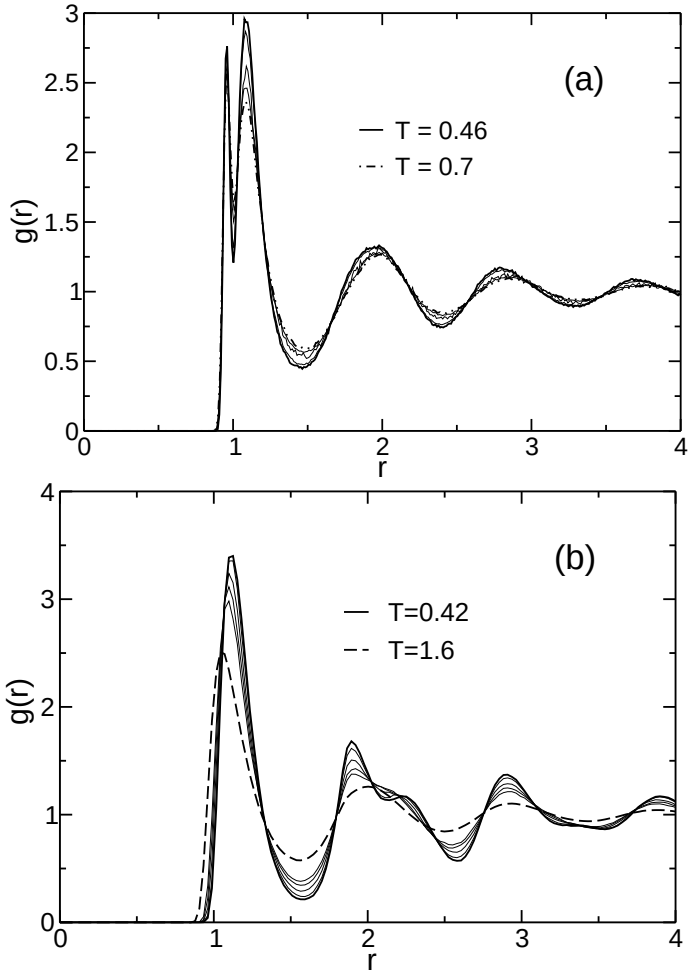

Figure 3.2: The pair correlation function $g(r)$ of (a) the polymer melt and (b) the Dzugutov liquid for different temperatures. The temperatures are, when viewed from top to bottom at the main peak, $T = 0.46$, $T = 0.5$, $T = 0.6$, $T = 0.65$, and $T = 0.7$ for the polymer melt, and $T = 0.42$, $T = 0.46$, $T = 0.55$, $T = 0.65$, $T = 0.75$ and $T = 1.6$ for the Dzugutov liquid.

where $\rho(\mathbf{q})$ is the Fourier transform of $\rho(\mathbf{r})$

$$\begin{aligned}\rho(\mathbf{q}) &= \int \exp(-i\mathbf{q}\cdot\mathbf{r})\rho(\mathbf{r})d\mathbf{r} \\ &= \sum_{i=1}^{N} \exp(-i\mathbf{q}\cdot\mathbf{r}_i). \end{aligned} \tag{3.15}$$

Then, $S(\mathbf{q})$ is expressed as

$$S(\mathbf{q}) = \frac{\mathbf{1}}{\mathbf{N}} \left\langle \sum_{\mathbf{i=1}}^{\mathbf{N}} \sum_{\mathbf{j=1}}^{\mathbf{N}} \exp[-\mathbf{iq} \cdot (\mathbf{r_i} - \mathbf{r_j})] \right\rangle \tag{3.16}$$

From the above equation, it can be shown that $S(\mathbf{q})$ can be expressed as the Fourier transform of $g(\mathbf{r})$ as

$$S(\mathbf{q}) = \mathbf{1} + \rho \int \exp(-\mathbf{iq} \cdot \mathbf{r}) \mathbf{g}(\mathbf{r}) \mathbf{dr}. \tag{3.17}$$

For an isotropic system the above equation reduces to

$$S(q) = 1 + 4\pi\rho \int r^2 g(r) \frac{\sin qr}{qr} dr. \tag{3.18}$$

This last equation provides a second method for calculating $S(q)$. For most practical purposes Eq. 3.18 is more efficient than the direct method Eq. 3.16. The latter is computationally less intensive than the former. However, when one is interested in accurately determining the value of $S(q)$ in the small q regime, the latter method is considered a method of choice, since in the former the lowest q value accessible by the Fourier transformation is limited by the size of the simulation box. As a result, it is impossible to reach substantially low q values for a small system.

The structure factor for the polymer melt is computed in Ref. [100]. In Fig. 3.3(a) we show the plot of a similar calculation. As shown in the figure, similar to the pair correlation function, $S(q)$ exhibits a behavior typical of a liquid. The plot

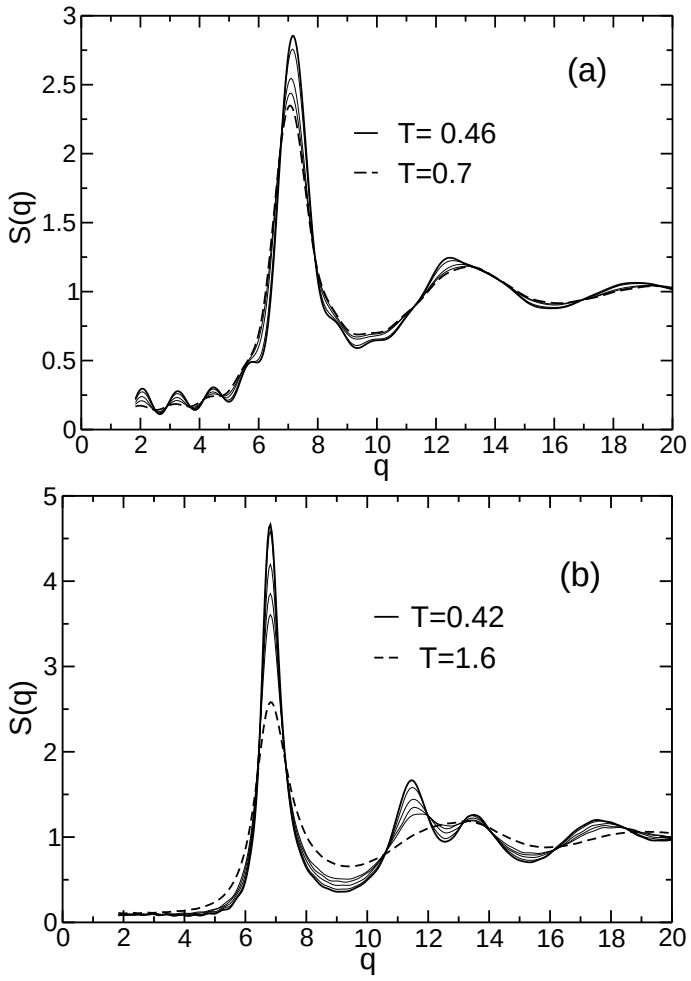

Figure 3.3: The static structure factor $S(q)$ of (a) the polymer melt and (b) the Dzugutov liquid for different temperatures. The temperatures are, when viewed from top to bottom at the main peak, $T = 0.46$, $T = 0.5$, $T = 0.6$, $T = 0.65$, and $T = 0.7$ for the polymer melt, and $T = 0.42$, $T = 0.46$, $T = 0.55$, $T = 0.65$, $T = 0.75$ and $T = 1.6$ for the Dzugutov liquid. Note that S(q) is calculated using Eq. 3.18. The oscillation observed at the low q value of $S(q)$ in the polymer melt is an artifact of the small system size, while this is not a problem for the Dzugutov liquid, which is approximately 17 times larger.

of $S(q)$ for the Dzugutov liquid is shown in Fig. 3.3(b). As expected, $S(q)$ exhibits a first sharp diffraction peak and a split second peak, which are typical features of metallic glasses [119, 120]. In addition to structure, $S(q)$ also provides information on the isothermal compressibility κ_T from the relation $S(0) = \rho \mathrm{k_B T} \kappa_\mathrm{T}$. The very low value of $S(q \to 0)$ apparent from Fig. 3.3 are typical for all liquids far from any critical point [124] reflecting the very low compressibility of the simulated system, as expected.

3.5 Dynamic properties

3.5.1 Mean square displacement

The simplest parameter that can be easily calculated from simulation data, but that has a wealth of information on the dynamic properties of a system, is the mean square displacement, $\langle r^2(t) \rangle$. It is defined as

$$\left\langle r^2(t) \right\rangle \equiv \left\langle |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \right\rangle \tag{3.19}$$

where the angular brackets represent an ensemble average. At short times, the mean square displacement increases quadratically with time, i.e., $\langle r^2(t) \rangle \propto t^2$. This can be understood by noting that

$$\mathbf{r}_i(t) = \mathbf{r}_i(0) + \int_0^t \mathbf{v}_i(t')dt'. \tag{3.20}$$

For short times, $\mathbf{v}_i(t)$ can be approximated to be constant, say, $\mathbf{v}_i$, and $|\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 =$

$\mathbf{v}_i^2 t^2$. Hence,

$$\left\langle r^2(t) \right\rangle = \left\langle \mathbf{v}_i^2 \right\rangle t^2 = \left(\frac{3k_B T}{m} \right) t^2, \tag{3.21}$$

where we used the equipartition theorem in the last part of the above equation. A more rigorous approach for arriving at this conclusion may be found in Ref. [4, 125].

In the long time limit, i.e., $t \to \infty$, the mean square displacement scales linearly with time, $\langle r^2(t) \rangle \propto t$. This can be seen if one writes the mean square displacement in terms of the velocity autocorrelation function. From Eq. 3.20 it is clear that

$$\left| \mathbf{r}_i(t) - \mathbf{r}_i(0) \right|^2 = \int_0^t dt' \int_0^t dt'' \mathbf{v}_i(t') \cdot \mathbf{v}_i(t''). \tag{3.22}$$

For long times, the mean square displacement can be shown to reduce to [125]

$$\left\langle r^2(t) \right\rangle = 2t \int_0^\infty d\tau \left\langle \mathbf{v}(0) \cdot \mathbf{v}(\tau) \right\rangle. \tag{3.23}$$

On the other hand, the self-diffusion coefficient D can be expressed in terms of the Green-Kubo relation as the time integral of the velocity autocorrelation function [125],

$$D = \frac{1}{3} \int_0^\infty d\tau \left\langle \mathbf{v}(0) \cdot \mathbf{v}(\tau) \right\rangle, \tag{3.24}$$

Then, comparing Eq. 3.23 and 3.24 we find that $\langle r^2(t) \rangle$ is related to the self-diffusion coefficient D by

$$\left\langle r^2(t) \right\rangle = 6Dt, \tag{3.25}$$

which is the well-known Einstein relation. In general, the motion of particles is ballistic at short times, i.e., the particles move as if they were free, and is diffusive at long times.

In Fig. 3.4(a) and 3.4(b), we show the plots of the mean square displacement for the two systems at different T. The plot for the Dzugutov liquid depicts that, at high T, a ballastic regime is followed by a diffusive regime as expected. But as the temperature is lowered towards $T_{\rm MCT}$, the ballistic and the diffusive regimes are separated by a plateau at intermediate times. This is the typical behavior of glass-forming liquids, where the plateau observed at low temperatures is a result of temporary "caging" of each particle by its neighbors. With decreasing temperature, the plateau regime extends to longer times indicating that, upon deep supercooling, the particles are caged for a longer period of time. In this regime the motion of particles is restricted within a cage. This regime corresponds to the β-relaxation regime of the MCT. The α-relaxation regime, on the other hand, corresponds to the time when the particles break out of their cage and become diffusive, or sub-diffusive in the case of polymer melts (described below).

The behavior of $\langle r^2(t)\rangle$ is slightly different in the polymer melts. To describe this dynamics we first denote the mean square displacement of the monomers by $\langle r^2_{\rm m}(t)\rangle$ and that of the center of mass of the polymer chains by $\langle r^2_{\rm c}(t)\rangle$ as

$$\left\langle r^2_{\rm m}(t)\right\rangle = \langle|\mathbf{r}_i(t) - \mathbf{r}_i(0)|\rangle\,; \qquad \left\langle r^2_{\rm c}(t)\right\rangle = \langle|\mathbf{R}_{cm}(t) - \mathbf{R}_{cm}(0)|\rangle\,, \tag{3.26}$$

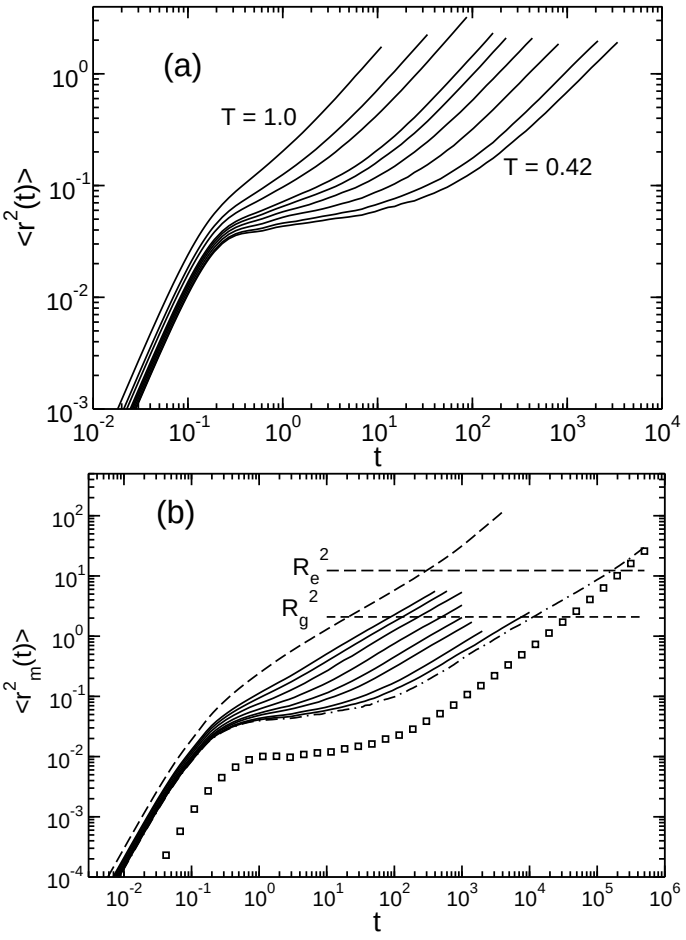

Figure 3.4: (a) Time dependence of the mean square displacement (MSD) $\langle r^2(t)\rangle$ of particles in the Dzugutov liquid for different temperatures. The temperatures are, from left to right, $T =$ 1.0, 0.75, 0.65, 0.55, 0.52, 0.49, 0.46, 0.43, 0.42 (b) Time dependence of the MSD $\langle r_{\rm m}^2(t)\rangle$ of monomers (lines) at all T studied, and the MSD $\langle r_{\rm c}^2(t)\rangle$ of the center of mass (shown by $\Box$) at $T = 0.46$. The temperatures are, from left to right, $T =$ 1.0, 0.7, 0.65, 0.6 0.55, 0.52, 0.50 0.48, 0.47, 0.46. The dashed horizontal lines indicate the values of the radius of gyration $R_{\rm g}^2$ (= 2.09) and of the end-to-end distance $R_{\rm e}^2$ (= 12.3).

where $\mathbf{r}_i$ represents the position of a monomer, and $\mathbf{R}_{cm}$ represents the position of the center of mass of a chain. For our system in which m is set to unity, $\mathbf{R}_{cm}(t)$ can be defined as $\mathbf{R}_{cm}(t) = (1/M)\sum_{i=1}^{M} \mathbf{r}_i(t)$, where $M(= 10)$ is the total number of monomers in a chain. The plot of these quantities is shown in Fig. 3.4(b), where $\langle r_{\rm m}^2(t)\rangle$ is plotted for all T studied while $\langle r_{\rm c}^2(t)\rangle$ is plotted for the lowest T simulated.

As can be seen from the figure, at short times both $\langle r_{\rm m}^2(t)\rangle$ and $\langle r_{\rm c}^2(t)\rangle$ increase quadratically with t, showing a ballistic regime as seen in other simple liquids. With decreasing T towards $T_{\rm MCT}$, the ballistic regime is followed by a plateau regime. Upon further progress in time (or, immediately after the ballistic regime for $T = 1$), a distinct regime emerges that is specific to the polymer melt. In this regime the mean square displacement of the monomers $\langle r_{\rm m}^2(t)\rangle \sim t^x$, where $x = 0.63 \pm 0.02$. This regime is referred to as the sub-diffusive regime. The diffusive regime in this system is reached only after long simulation times. Specially, for low T, since the plateau regime itself extends over a long period of time upon cooling, the diffusive regime is reached at a time much longer than that at high T.

The sub-diffusive nature of the long time dynamics of the polymer melt is a consequence of chain connectivity. This is in contrast to the short and intermediate time scales, where $\langle r_{\rm m}^2(t)\rangle$ is so small that the monomers feel no or little effect of the chain connectivity. As a result, $\langle r_{\rm m}^2(t)\rangle$ is essentially identical to those observed in simple supercooled liquids, and the plateau is simply due to the transient caging of each monomer by neighboring monomers as in simple liquids. The sub-diffusive relaxation behavior is also predicted by the Rouse model. In the Rouse model, however, $x = 0.5$. This deviation shows that the model behaves slightly differently

than what is expected from the ideal Rouse model. Unlike $\langle r_{\mathrm{m}}^2(t)\rangle$, the mean square displacement of the center of mass $\langle r_{\mathrm{c}}^2(t)\rangle$ directly crosses over to the diffusive regime for long times. In this case the sub-diffusive regime is absent because the center of mass is not subject to chain connectivity.

3.5.2 van Hove correlation function

Another quantity of interest that describes the dynamics of a system of particles by measuring the time dependent spatial correlations of particles position is the van Hove correlation function $G(\mathbf{r}, t)$, which is defined in terms of the density-density time correlation function as

$$G(\mathbf{r}, t) = \frac{1}{\rho} \langle \rho(\mathbf{r}, t)\rho(\mathbf{0}, 0)\rangle \,. \tag{3.27}$$

That is,

$$G(\mathbf{r}, t) = \frac{1}{N} \left\langle \sum_{i=1}^{N} \sum_{j=1}^{N} \delta[\mathbf{r} + \mathbf{r}_j(0) - \mathbf{r}_i(t)] \right\rangle . \tag{3.28}$$

The physical meaning of this function is that $G(\mathbf{r}, t)d\mathbf{r}$ is proportional to the probability of finding a particle i in a region $d\mathbf{r}$ around a point $\mathbf{r}$ at time t given that there was a particle j at the origin at time $t = 0$ [125]. The above summation can be decomposed into summations over single terms ($j = i$) and cross terms ($j \neq i$), which leads to the separation of $G(\mathbf{r}, t)$ into the "self" (s) and "distinct" (d) parts as,

$$G(\mathbf{r},t) = G_s(\mathbf{r},t) + G_d(\mathbf{r},t) \tag{3.29}$$

where

$$G_s(\mathbf{r},t) = \frac{1}{N}\left\langle \sum_{i=1}^{N} \delta[\mathbf{r} + \mathbf{r}_i(0) - \mathbf{r}_i(t)] \right\rangle \tag{3.30}$$

$$G_d(\mathbf{r},t) = \frac{1}{N}\left\langle \sum_{i\neq j}^{N} \delta[\mathbf{r} + \mathbf{r}_j(0) - \mathbf{r}_i(t)] \right\rangle \tag{3.31}$$

At $t = 0$ we find that $G(r,0) = \rho g(\mathbf{r}) + \delta(\mathbf{r})$, which is simply the static two-point density-density autocorrelation discussed in the last section. Thus, $G_s(\mathbf{r},0) = \delta(\mathbf{r})$ and $G_d(r,0) = \rho g(\mathbf{r})$. For isotropic fluids both G_s and G_d will be a function of the scalar quantity r. In what follows we discuss $G_s(r,t)$ for our systems.

$G_s(r,t)$ measures the probability distribution of the particle displacements in a time interval t. To investigate the nature of particles dynamics in a system, it is customary to compare $G_s(r,t)$ with the Gaussian distribution function [125] $G_0(r,t)$, defined as

$$G_0(r,t) = \left(\frac{3}{2\pi\langle r^2(t)\rangle}\right)^{3/2} \exp\left(-\frac{3r^2}{2\langle r^2(t)\rangle}\right). \tag{3.32}$$

At sufficiently high temperatures, since the motion of particles is sampled from the Maxwell-Boltzmann distribution, $G_s(r,t)$ is well approximated by the Gaussian

distribution. But at low T, the Gaussian approximation holds only at short ($t \to 0$) or long ($t \to \infty$) time limits. At intermediate times, $G_s(r,t)$ is significantly different from $G_0(r,t)$ [74, 86]. The deviation from Gaussian behavior can be quantified by the non-Gaussian parameter α_2, which is known to show a maximum at a time $t^*_{\alpha_2}$ in the late-β/early-α relaxation regime [74, 76, 86, 126]. We postpone the discussion of α_2 to the following chapters where we describe the spatially heterogeneous dynamics of each of the systems. Here, we show the self part of the van Hove correlation function for the two liquids.

Fig. 3.5(a) shows a plot of $4\pi r^2 G_s(r,t)$ for the polymer melt at $T = 0.46$. As shown in the figure, $G_s(r,t)$ shows Gaussian behavior at early ($t = 2.24$) and late ($t = 14000$) times. These times correspond to the time scales in the ballistic and sub-diffusive regimes, where the time $t = 14000$ belongs to the α-relaxation regime of the polymer melt and corresponds to the time when $\langle r^2_{\rm m}(t)\rangle \approx 2.38$. At an intermediate time $t^*_{\alpha_2}$, a significant deviation from Gaussian behavior is observed. The long tail of $4\pi r^2 G_s(r,t)$ seen at this time indicates the presence of particles that have moved much larger distances than one would expect from a Gaussian distribution [74], cf. Fig. 3.5(a). These highly mobile particles will be the focus of our later analysis.

A similar behavior is observed for the Dzugutov liquid. Fig. 3.5(b) shows the self part of the van Hove correlation function for the Dzugutov liquid at $T = 0.42$. In the insets, $4\pi r^2 G_s(r,t)$ and $4\pi r^2 G_0(r,t)$ are compared for two dynamically significant time scales. For $t = t^*_{\alpha_2}$ a long tail is observed as in the polymer case. However, for $t = 3393.31$, in the α-relaxation regime, the development of a significant

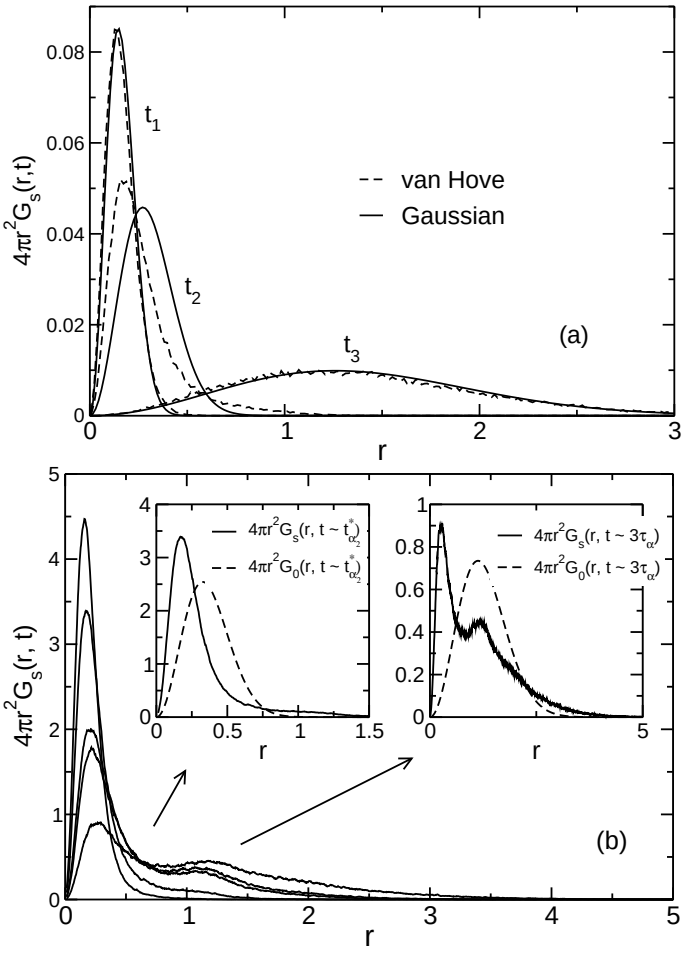

Figure 3.5: (a) The self part of the van Hove correlation function $G_s(r,t)$ of the polymer melt at $T = 0.46$. $G_s(r,t)$ is plotted together with the Gaussian distribution function, $G_o(r,t)$ for the times $t_1 = 2.24$, $t_2 = t^*_{\alpha_2}$, and $t_3 = 14000$. (b) $G_s(r,t)$ of the Dzugutov liquid at $T = 0.42$. $G_s(r,t)$ is plotted for the times $t = 11.01, 140.77(\sim t^*_{\alpha_2}), 811.15, 1115.3(\sim \tau_\alpha), 3399.31$. In the inset we plot $G_s(r,t)$ together with $G_o(r,t)$ for two selected times, $t = 140.77$ and $t = 3399.31$. These times correspond to the late-β/early-α relaxation regime, and the α-relaxation regime, respectively (see Fig. 3.4).

secondary peak in $4\pi r^2 G_s(r,t)$ is striking, cf. Fig. 3.5(b). At this time $\langle r^2(t)\rangle \approx 2.0$. Such a secondary peak was also observed for a binary LJ mixture at $T \approx T_{MCT}$ [128, 129, 130], and it is commonly interpreted as indicating single-particle "hopping" [131, 132], although it has been demonstrated that the hopping indicated by the secondary peak is not due to transitions over single energy barriers [130]. The lack of a secondary peak in the polymer melt may be attributed to the presence of connectivity in this system.

3.5.3 Intermediate scattering function

Another important quantity for describing the dynamic behavior of a system of particles is the intermediate scattering function $F(q,t)$. It measures the density-density correlation in Fourier space and is defined as

$$F(\mathbf{q},t) = \frac{1}{N}\left\langle \rho(\mathbf{q},t)\rho(-\mathbf{q},0)\right\rangle , \tag{3.33}$$

where $\rho(\mathbf{q},t)$ is the Fourier transform of $\rho(\mathbf{r},t) = \sum_{i=1}^{N}\delta(\mathbf{r},t)$. Using the expression for $\rho(\mathbf{r},t)$, $F(\mathbf{q},t)$ can be expressed as,

$$F(\mathbf{q},t) = \frac{1}{N}\left\langle \sum_{i=1}^{N}\sum_{j=1}^{N}\exp[-i\mathbf{q}\cdot\mathbf{r}_i(0)]\exp[i\mathbf{q}\cdot\mathbf{r}_j(t)]\right\rangle , \tag{3.34}$$

The above equation can be easily found by taking the Fourier transform of the van Hove correlation function $G(\mathbf{r},t)$,

$$F(\mathbf{q},t) = \int G(\mathbf{r},t)\exp(-i\mathbf{q}\cdot\mathbf{r})d\mathbf{r}. \tag{3.35}$$

Because $G(\mathbf{r},t)$ can be decomposed into the self and distinct parts, the intermediate scattering function can also be decomposed into these parts. However, due to their connections with scattering experiments, only the self $F_s(\mathbf{q},t)$ and the full $F(\mathbf{q},t)$ intermediate scattering functions are considered to be useful in the study of dynamic properties. Additionally, these quantities serve as important tools for investigating the relaxation behavior of supercooled liquids in terms of MCT, where they are borne in the development of the theory. The calculation of these dynamical variables using computer simulations has been a means for testing several aspects of MCT. Here we use $F_s(\mathbf{q},t)$ for investigating the relaxation behavior of our systems, and for testing the main predictions of MCT, especially the nature of the power law divergence of relaxation times upon cooling [17].

$F_s(\mathbf{q},t)$ is obtained from the Fourier transformation of $G_s(\mathbf{r},t)$ that yields an expression given by

$$F_s(\mathbf{q},t) = \frac{1}{N}\left\langle \sum_{i=1}^{N} \exp[-i\mathbf{q}\cdot(\mathbf{r}_i(t) - \mathbf{r}_i(0))] \right\rangle \tag{3.36}$$

$F(\mathbf{q},t)$ and $F_s(\mathbf{q},t)$ are usually referred to as the coherent and incoherent correlation functions, respectively, because of their correspondence, through the dynamic structure factor $S(\mathbf{q},\omega)$, with the coherent and incoherent parts of the neutron scattering

cross-section. $S(\mathbf{q},\omega)$ is related to $F(\mathbf{q},t)$ through a Fourier transform as

$$S(\mathbf{q},\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\mathbf{q},t)\exp(i\omega t)dt. \tag{3.37}$$

A similar transformation of $F_s(\mathbf{q},t)$ yields the self dynamic structure factor $S_s(q,w)$. The intermediate scattering function thus provides insight into the dynamic properties of a given system upon supercooling that can be related to experiments. Here, we demonstrate the properties of $F_s(\mathbf{q},t)$ for the Dzugutov liquid. A similar behavior is observed for the polymer melt. (see Ref. [100] for the details)

In Fig. 3.6(a) we show $F_s(\mathbf{q},t)$ of the Dzugutov liquid for different T evaluated at the wave vector $q_m = 6.82$, which corresponds to the value of q at which $S(q)$ is maximum, cf. Fig. 3.3. At high T, i.e., $T > 0.8$, the relaxation of $F_s(\mathbf{q},t)$ is exponential as found in "normal" liquids that can be described in terms of Brownian dynamics. As the temperature is lowered, $F_s(\mathbf{q},t)$ exhibits a two step relaxation, which is a typical behavior of glass forming liquids that is found in several experiments [133] and simulations [86], including the polymer model we studied. The temperature at which the two-step relaxation is first observed is considered as an indication for the onset of caging, and hence for the Dzugutov liquid this temperature corresponds to $T \approx 0.8$ as seen from the figure.

It is now well established for supercooled liquids [86] that the short-time decay, the plateau and the long-time decay of $F_s(\mathbf{q},t)$ corresponds to vibrations, the β-regime and α-regime, respectively. The average relaxation time of the α-

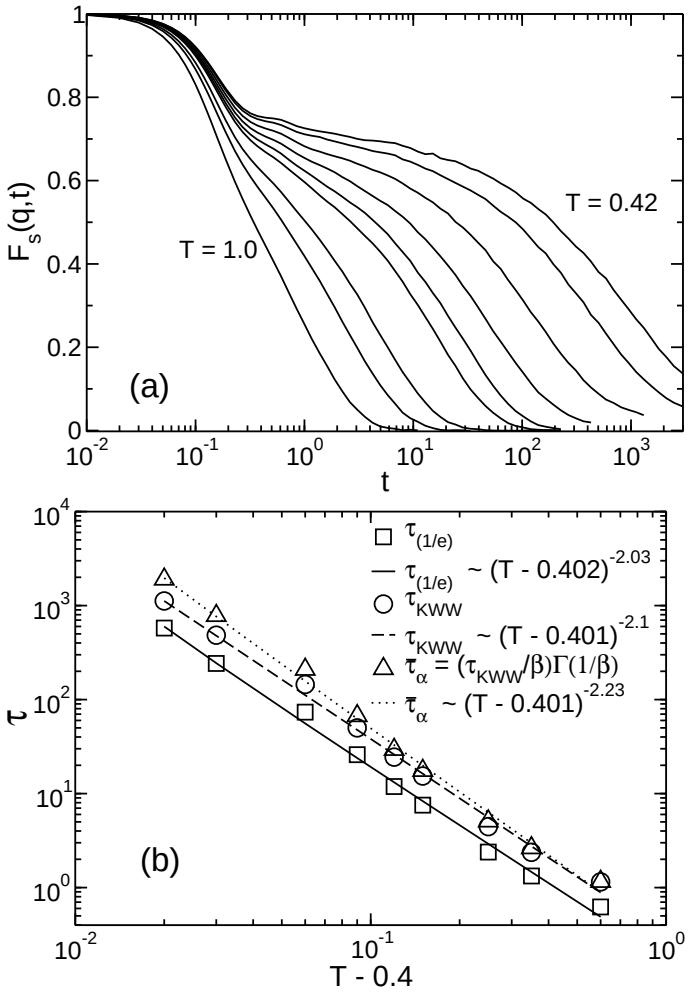

Figure 3.6: (a) The self part of intermediate scattering function $F_s(q,t)$ of the Dzugutov liquid plotted as a function of time for different temperatures. The temperatures from left to right are$T =$ 1.0, 0.75, 0.65, 0.55, 0.52, 0.49, 0.46, 0.43, 0.42 $F_s(q,t)$ is evaluated at the wave vector $q_m = 6.82$ that corresponds to the first peak position of the corresponding static structure factor. (b) Temperature dependence of the α-relaxation time, where the relaxation time is characterized by extracting $\tau_{1/e}$ or $\overline{\tau}_\alpha$, cf. text. A fit for $\tau_{\rm KWW}$ is also shown for comparison.

process, τ_α, can be characterized either by extracting $\tau_{1/e}$, i.e., the time where $F_s(q,t)$ has decayed to $1/e$ of its initial value, or by fitting the long-time decay of $F_s(q,t)$ to a Kohlrausch-Williams-Watts (KWW) function, $A\exp[-(t/\tau_{\rm KWW})^\beta]$, and calculating the mean time constant $\overline{\tau}_\alpha$ from the fit parameters according to $\overline{\tau}_\alpha = (\tau_{\rm KWW}/\beta)\,\Gamma(1/\beta)$, where $\Gamma(x)$ is the Γ-function [127]. MCT predicts that any definition of the relaxation time that measures the time scale of the α-relaxation process exhibits the same T dependence. Fig. 3.6(b) shows that the temperature dependence of $\tau_{1/e}$ and $\overline{\tau}_\alpha$ are well described by a power-law $(T-T_{\rm MCT})^{-\gamma}$. While $T_{\rm MCT} \approx 0.4$ results consistently from both approaches (also, from a power-law fit of the diffusion coefficient [102]), somewhat different exponents $\gamma_{(1/e)} = 2.03$ and $\gamma_{ave} = 2.33$ are found, where $\gamma_{(1/e)}$ and γ_{ave} are the exponents corresponding to $\tau_{1/e}$ and $\overline{\tau}_\alpha$, respectively. The stronger T-dependence of $\overline{\tau}_\alpha$ can be traced back to the variation of the streching parameter β with change in T, i.e., the time-temperature superposition principle may not hold. For our system, as observed in other systems [130], we find a value of $\beta \approx 0.5$ for the lowest T, which increases with temperature.

Chapter 4

Spatially heterogeneous dynamics in a polymer melt: Dynamic clusters

This chapter is the first part of the study of dynamical heterogeneity in the model of a polymer melt, where we quantify the spatial extent of cooperative motion in highly mobile dynamical subensembles. We measure the clusters formed by the mobile particles, and investigate the transient nature and temperature dependence of the mean cluster size formed by these dynamical units. We first present the calculation of the non-Gaussian parameter that measures the extent of deviation from Gaussian behavior. The deviation from Gaussian behavior, and the increase in the size of the non-Gaussian parameter, is considered to be a signature of dynamical heterogeneity. The work presented in this chapter is published in Ref. [134].

4.1 Cooperativity and spatial correlation

Stretched exponential relaxation as well as the decoupling of translational and rotational motion in glass-forming liquids have been attributed to the emergence of dynamical heterogeneity [32, 62, 66, 67]. Several experiments have shown that it is possible to select subensembles of slow or fast particles close to T_g, and this has been a simple proof for the existence of dynamical heterogeneity, but only few experiments have suceeded in measuring the spatial extent of dynamical heterogeneity directly. Quantifying the correlation length of these dynamical subensembles is an important aspect of glass transition phenomena, since several theories postulate a divergence of a characteristic length scale at the glass transition. For example, the length scale of dynamical heterogeneity can be related to the size of a cooperatively rearranging region of the Adam-Gibbs theory, which is postulated to diverge at the Kauzmann temperature T_K where the configurational entropy vanishes.

Generally, it has been a central goal of theories of the glass transition to account for the bulk phenomena in terms of the microscopic dynamical motion of the molecules of the liquids [62]. In this respect, computer simulations have played a vital role for achieving this goal. The only experimental technique that provides the same level of detail as in computer simulations is confocal laser scanning microscopy, where detailed information on the trajectories of individual colloidal particles is directly accessible. Thus, insight into the detailed nature of the cooperative motion of particles in supercooled liquids have been especially provided by computational studies.

In several computer simulations [135] and also the microscopy studies of colloidal suspensions [20, 21], the existence of dynamically heterogeneous regions has been directly observed and quantified. In these studies, the connection between dynamical heterogeneity and cooperativity, in which particles move together along the same path in a correlated fashion, has been further elucidated [75, 21, 136, 137]. For example, in Ref. [74, 75, 76, 77], it was shown in a binary LJ mixture that at any given moment, most particles can be found trapped in "cages" formed by their neighbors, while roughly 5-6% constitute a highly mobile subset that is breaking out of these cages. They showed that these mobile, "escaping" particles move cooperatively in string-like paths, forming clusters of strings which grow in size with decreasing temperature on approaching the mode-coupling crossover temperature T_{MCT}. The experiments on suspensions of hard sphere colloids confirmed the clustering of highly mobile subset of particles at densities below the colloidal glass transition density, where the tendency for the increase in the cluster size is also confirmed on approaching the glass transition.

In this chapter we quantify the characteristic length of the spatially heterogeneous dynamics of monomers in a model polymer melt by examining the dynamical clusters formed by the mobile monomers. In this way we investigate the cooperative nature of molecular motion in a polymeric system upon approaching T_{MCT}. The primary purpose of this study is to explore the nature of dynamical clustering, observed in the simulation of a binary LJ mixture, in a different, structurally homogeneous system. In doing so, we especially investigate the transient nature of this clustering that was not shown in earlier works. In other words, we study

how the dynamic length scale associated with mobile domains changes during the relaxation of the system. This is vital information for understanding the nature of relaxation in glass-forming supercooled liquids. For example, some insight into the two-step, non-exponential relaxation of the density-density correlation function may be obtained by examining the transient nature and temperature dependence of these clusters. Therefore, the present chapter is dedicated to examining this property.

4.2 Non-Gaussian parameter

As mentioned in Chapter 3, the dynamic properties of a system of particles can be understood by studying the self part of the van Hove correlation function $G_s(r,t)$, which is the probability for finding a particle at a distance r at time t. In the short and long time limits this quantity can be described by the Gaussian distribution function $G_0(r,t)$. For the polymer melt we realize that in the limit $t \to 0$, the monomers move ballistically, and hence $G_{\rm s}(\mathbf{r},t)$ is proportional to the Maxwell-Boltzmann distribution [125], which has a Gaussian form. In the opposite limit $t \to \infty$, the polymers behave as if they were isolated Brownian particles subjected to a heat bath, and hence diffuse freely. Due to chain connectivity the monomers must follow the diffusive motion of the center of mass, and $G_{\rm s}(\mathbf{r},t)$ is again Gaussian.

At intermediate times, however, there may be deviations from Gaussian behavior. A possible means to measure these deviations is the non-Gaussian parameter $\alpha_2(t)$ [125]

$$\alpha_2(t) = \frac{3\left\langle [\mathbf{r}_i(t) - \mathbf{r}_i(0)]^4 \right\rangle}{5\left\langle [\mathbf{r}_i(t) - \mathbf{r}_i(0)]^2 \right\rangle^2} - 1 \,, \tag{4.1}$$

where $\mathbf{r}_i(t)$ is the position of monomer i at time t. Similar to $\alpha_2(t)$, which quantifies deviations of the monomer dynamics from Gaussian behavior, one can also measure these deviations for the chain motion by calculating a non-Gaussian parameter $\alpha_2^{\mathrm{p}}(t)$ for the polymers as

$$\alpha_2^{\mathrm{p}}(t) = \frac{3\left\langle [\mathbf{R}_c(t) - \mathbf{R}_c(0)]^4 \right\rangle}{5\left\langle [\mathbf{R}_c(t) - \mathbf{R}_c(0)]^2 \right\rangle^2} - 1 \ , \tag{4.2}$$

where $\mathbf{R}_c(t)$ is the position of the center of mass of chain c at time t.

Figure 4.1 shows $\alpha_2(t)$ and $\alpha_2^{\mathrm{p}}(t)$ for various T. As expected, these quantities are zero at short times ($t \to 0$), then become positive, exhibit a maximum, and finally go to zero at long times ($t \to \infty$). As T decreases, the positions of the maxima, $t^*_{\alpha_2}$ for $\alpha_2(t)$ and $t^*_{\alpha_2^{\mathrm{p}}}$ for $\alpha_2^{\mathrm{p}}(t)$, shifts towards longer times, and the height of the maxima increase. Furthermore, for all T, we find that $t^*_{\alpha_2}$ corresponds to times in the late-β/early-α relaxation regime. Such behavior is often observed in computer simulations [86, 138, 139, 140, 141, 142] and experiments [20, 21, 143] on glass-forming liquids. Notice that the peak height is larger for $\alpha_2(t)$ than for $\alpha_2^{\mathrm{p}}(t)$ and that $t^*_{\alpha_2}$ and $t^*_{\alpha_2^{\mathrm{p}}}$ are different. $t^*_{\alpha_2^{\mathrm{p}}}$ is shifted by about half a decade to longer times for $T \leq 0.52$.

The small amplitude of $\alpha_2^{\mathrm{p}}(t)$ may be attributed to the difference in packing of the monomers and of the chains. The monomers of our model exhibit an oscillatory pair-distribution function $g(r)$ [144] whose shape and range are very similar to those found in simple dense liquids (see Fig. 3.2(a)). In contrast to that, the pair-distribution function $g_{\mathrm{cm}}(r)$ for the centers of mass is fairly structureless, and resembles the $g(r)$ of a gas (see Ref. [145, 146]). This reflects the fact that polymers

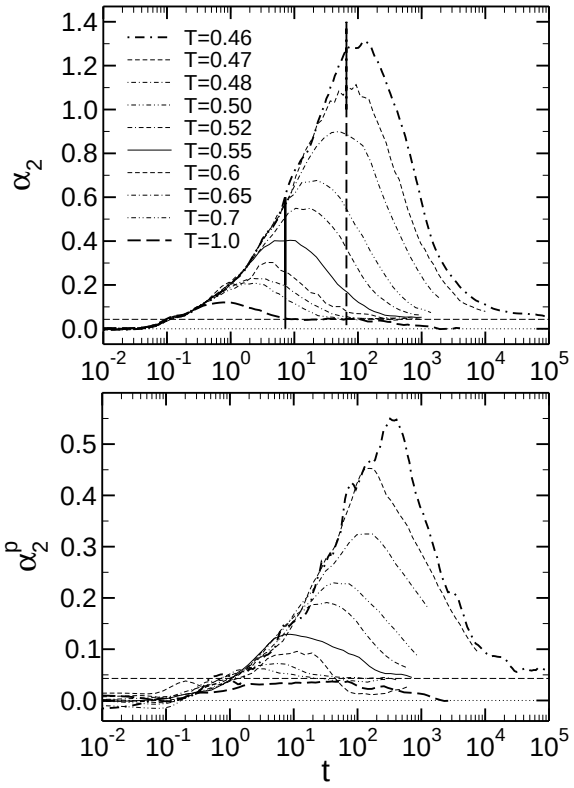

Figure 4.1: Non-Gaussian parameter of the monomers $\alpha_2(t)$ (upper panel) and of the center of mass of the polymers $\alpha_2^{\rm p}(t)$ (lower panel) versus time for different temperatures. $\alpha_2(t)$ and $\alpha_2^{\rm p}(t)$ are defined in Eqs. (4.1) and (4.2). The temperature ranges from the high-T, normal liquid state above onset of caging ($T = 1$; thick dashed lines in both panels), to the supercooled state of the melt slightly above $T_{\rm MCT} \simeq 0.45$ ($T = 0.46$; thick dash-dotted lines). Temperature decreases from the bottom curve to the top curve in both panels. The dashed horizontal line ($= 0.043$) in the upper panel indicates a possible intermediate plateau toward which all $\alpha_2(t)$-curves could converge. This line is also included in the lower panel. The two vertical lines in the upper panel indicate the times $t_{\rm clu}^{\rm max}$ (to be discusses) where clusters are maximum for $T = 0.46$ at $t_{\rm clu}^{\rm max} = 65.85$ (dashed line) and for $T = 0.55$ at $t_{\rm clu}^{\rm max} = 7.22$ (straight line).

are soft, strongly interpenetrating objects and that the effective interaction between the centers of mass is weak. If this interaction were zero, there would be no resulting force on the center of mass, and the chain would diffuse freely (outside the ballistic regime). The small, non-zero value of $\alpha_2^{\mathrm{p}}(t)$ at $T = 1$ may thus be related to a weak force arising from the presence of other chains in the volume occupied by a polymer [147].

On cooling the melt toward T_{MCT} a pronounced maximum in time occurs for both non-Gaussian parameters. Since $g_{\mathrm{cm}}(r)$ is (nearly) temperature independent [145], the maximum of α_2^{p} cannot be attributed to enhanced inter-chain interactions at low T. The similarity between α_2 and α_2^{p} rather suggests that the coupling between monomer and chain dynamics [148] drives the behavior of $\alpha_2^{\mathrm{p}}(t)$. If the monomers of a chain are trapped in their cages and prevented from moving, the center of mass cannot move either. On the other hand, if a sufficient number of monomers move far during the time t, a large displacement of the center of mass results. As many monomers of the same chain are involved in this motion, a large displacement of the center of mass should take a longer time than for a single monomer. This explains why $t^*_{\alpha_2^{\mathrm{p}}}$ is larger than $t^*_{\alpha_2}$.

In addition to the maximum, α_2 shows two conspicuous features. First, there is a small, temperature independent step (bump) at $t \approx 0.1$. This time corresponds to the crossover of the monomer mean square displacement $\langle r_{\mathrm{m}}^2(t)\rangle$ from the ballistic to the plateau regime. The step can be more or less pronounced, depending on the microscopic properties studied [86, 138, 139, 149], and it may be thought of as a signature of a small correlation present as the dynamics changes from the

ballistic region to the plateau where the free motion of the particles is hindered as a result of caging. Second, $\alpha_2(t)$ relaxes toward a plateau at long times before decaying to zero. This behavior is clearly visible for $T = 1.0$, while lower T are only indicative of a similar trend. Figure 4.1 suggests that the plateau value is the same for all T, but that the time when it is reached increases on cooling. For $T = 1$ the plateau is attained when the MSD of all monomers is about 1, and this occurs at $t \approx 10$. For $T = 0.46$, the plateau is only reached if $t \gtrsim 10^5$. This time corresponds to displacements of the order of the chain size for $T = 0.46$, i.e., $R_{\rm g}^2 < \langle r_{\rm m}^2(t)\rangle < R_{\rm e}^2$ ($R_{\rm g}$ and $R_{\rm e}$ refer to the radius of gyration and the end-to-end distance of a polymer chain, respectively). Because the motion of the monomers becomes diffusive for $\langle r_{\rm m}^2(t)\rangle > R_{\rm e}^2$, one can speculate that the length of the plateau decreases with increasing T.

For all T the plateau occurs if $\langle r_{\rm m}^2(t)\rangle \gtrsim 1$. This corresponds to times where the Rouse model [150] is believed to describe the dynamics of non-entangled chains in the melt [151]. In this model the displacements of the monomers and of the chains follow a Gaussian distribution, in which case, $\alpha_2(t)$ and $\alpha_2^{\rm p}(t)$ should vanish at all times. The finite value of the plateau points to small but systematic deviations from Rouse behavior. This value is approximately the same for both α_2 and $\alpha_2^{\rm p}$, and roughly agrees with the maximum of $\alpha_2^{\rm p}$ found at $T = 1$. The latter observation could imply that the occurrence of the plateau is related to the weak interactions between the centers of mass alluded to above.

In summary, the interpretation of Fig. 4.1 suggests that deviations from Gaussian behavior in our model might have two origins: The weak, temperature indepen-

dent interaction between the centers of mass leads to small deviations in the long-time sub-diffusive regime. Preceding the sub-diffusive regime strong, T-dependent deviations occur due to the caging and subsequent correlated motion of monomers as observed in dense simple atomic liquids and colloids. This drives the sluggish glass-like relaxation of the monomer and, as a consequence, also that of the center of mass. Our subsequent analysis will focus on this correlated nature of the monomer dynamics.

4.3 Clusters of mobile monomers

4.3.1 Definition of mobility

In order to study the clusters formed by mobile monomers, a criterion must be set to define "mobility". One common practice is calculating the scalar displacement $r_i(t)$ of individual particles in the system, and then ranking the particles according to their displacement, where a certain fraction ϕ of them with the largest scalar displacements are finally selected to be mobile. The challenge is, however, in choosing ϕ that can best signify the nature of cooperativity, and ensure reproducibility.

In the system we studied, it is known from, e.g., calculations of the mean square displacement $\langle r_{\mathrm{m}}^2(t)\rangle$ of the monomers, that on intermediate times, monomers on average are trapped or localized in cages formed by their neighbors (Fig. 3.4, Chapter 3). In a glass-forming binary LJ mixture, it was demonstrated that, on these time scales, most of the particles can be found oscillating inside their cages, with only

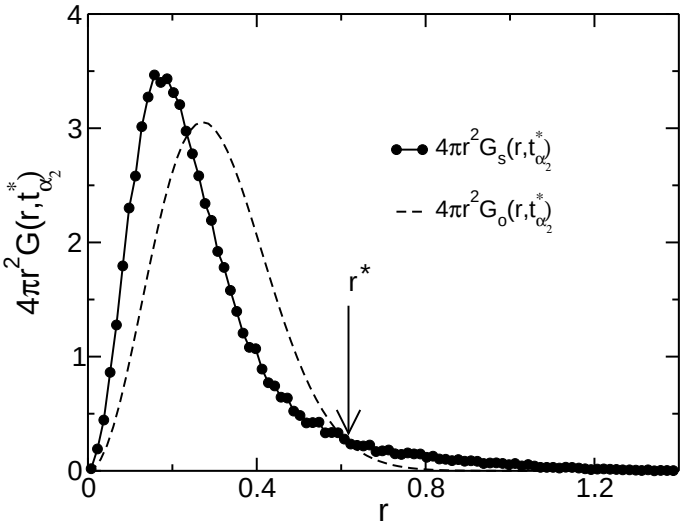

Figure 4.2: The radially averaged van Hove correlation function $G_s(r,t)$, at $t = t^*_{\alpha_2}$ for $T = 0.46$, plotted with the Gaussian distribution $G_0(r, t^*_{\alpha_2})$. Mobile monomers are defined as those monomers that moved a distance greater than r^*.

approximately 5-6% of the particles undergoing significant displacement. At a later time, of course, a different subset of particles can be found moving beyond their cage. In the range of T studied in that work, however, as in the present study, the distribution of particle (or monomer) displacements as measured by the self van Hove distribution function is continuous and unimodal, exhibiting at most a long tail to large displacements (in some liquids, this tail becomes a secondary peak at sufficiently low T [129]). This makes the identification of mobile particles substantially more difficult than if the distribution of particle displacements were, say, bimodal, in which case there would be an obvious criterion for identifying mobile particles.

Here we describe the method we used in identifying mobile monomers following

the work of Ref. [74, 76, 77] (Later we check if this method captures the dynamic signature.) These studies identify the mobile subset as those particles which, in a time interval $t^*_{\alpha_2}$, move further than some distance r^*, where $t^*_{\alpha_2}$ is the time interval when the non-Gaussian parameter $\alpha_2(t)$ is maximum [152], as discussed in the last section. They define r^* as the distance where the self part of the van Hove correlation function $G_s(r, t^*_{\alpha_2})$ and the Gaussian distribution $G_0(r, t^*_{\alpha_2})$, calculated with the measured value of $\left\langle r^2(t^*_{\alpha_2})\right\rangle$, cross each other as depicted in Fig. 4.2. The plot shows $G_s(r, t^*_{\alpha_2})$ together with $G_0(r, t^*_{\alpha_2})$ to identify r^*. The fraction ϕ of the monomers that are mobile is defined by integrating $G_s(r, t^*_{\alpha_2})$ for $r \geq r^*$, *i.e.*

$$\phi \equiv \int_{r^*}^{\infty} 4\pi r^2 G_s(r, t^*_{\alpha_2}) dr. \tag{4.3}$$

By using this method on a binary LJ mixture, ϕ was found to constitute approximately 5.5% of the total number of particles, independent of T and ρ [74, 75, 76, 77]. Following the same procedure, we find $6.2\% \leq \phi \leq 6.8\%$ (Table 4.1). Thus for convenience we will fix the fraction at an intermediate value of $\phi = 6.5\%$ for all analysis.

Once ϕ is selected, the subset of monomers that is considered mobile in each time interval t is identified by ranking the scalar displacement of all monomers in t, and choosing the 6.5% with the largest value. In any given t, the number of mobile monomers defined in this way is necessarily the same, but the membership will generally be different since a monomer that is mobile in one time interval may be caged in the next, and vice versa.

T	0.46	0.47	0.48	0.5
ϕ^I	$6.4 \pm 0.5\%$	$6.2 \pm 0.4\%$	$6.8 \pm 0.5\%$	$6.5 \pm 0.3\%$
ϕ^{II}	$5 \pm 1\%$	$5.5 \pm 0.5\%$	$5.5 \pm 0.5\%$	$6 \pm 1\%$
S_0^I	2.66 ± 0.05	2.58 ± 0.02	2.7 ± 0.03	2.55 ± 0.02
S_0^{II}	2.06 ± 0.06	2.22 ± 0.16	2.26 ± 0.2	2.36 ± 0.04

T	0.52	0.55	0.6	0.7
ϕ^I	$6.7 \pm 0.6\%$	$6.7 \pm 0.6\%$	$6.5 \pm 0.7\%$	$6.3 \pm 0.7\%$
ϕ^{II}	$6.5 \pm 0.5\%$	$7 \pm 1\%$	$7.5 \pm 0.5\%$	$8 \pm 1\%$
S_0^I	2.64 ± 0.01	2.61 ± 0.01	2.76 ± 0.03	2.62 ± 0.02
S_0^{II}	2.64 ± 0.24	2.81 ± 0.53	3.27 ± 0.26	3.31 ± 0.03

Table 4.1: ϕ^I refers to the fraction ϕ of highly mobile monomers at a time $t^*_{\alpha_2}$ when α_2 is maximum (Sec. 4.3.1), and ϕ^{II} refers to the fraction that maximizes the normalized weight-averaged cluster size for each T (Sec. 4.3.3). The normalization factor S_0 is the initial (or correspondingly, the random) value of the weight-averaged cluster size of the fraction considered. S_0^I refers to the value of S_0 at each T for $\phi = 6.5\%$, used to evaluate $S(t)$ using the procedure outlined in Sec. 4.3.1. S_0^{II} refers to the value of S_0 at each T for a fraction corresponding to ϕ^{II}. Note that S_0^I is nearly constant: $S_0^I = 2.65 \pm 0.1$ for all T, while S_0^{II} varies considerably because ϕ is different for each state point. The error bars in calculating ϕ^I reflect the uncertainty in estimating r^* for the evaluation of ϕ from Eq. 4.3. The error bars in estimating ϕ^{II} reflect the range of ϕ over which the fraction that maximizes S could be identified with confidence.

4.3.2 Mean cluster size

To estimate the typical distance over which mobile monomers are correlated, we define clusters [76, 77] as groups of highly mobile monomers that are within the first neighbor shell of each other [153], where the first neighbor shell is defined by the distance of the first minimum ($r = 1.5$) of the pair correlation function $g(r)$. An example of the clusters formed by the 6.5% most mobile monomers is shown in Fig. 4.3 for $T = 0.46$ at early, and at intermediate time intervals. We see that the typical cluster size depends upon the time window of observation t; smaller clusters appear at early t, and larger clusters appear at intermediate t, when $\langle r^2_{\rm m}(t)\rangle$ of the monomers crosses over from the plateau regime to the sub-diffusive regime. Such transient clustering of mobile particles has also been observed experimentally in dense collodial suspensions using a confocal microscope [21], by looking at the 5% most mobile particles as in Refs. [76, 77].

To quantify the clustering of mobile monomers, we calculate the weight-averaged mean cluster size S_w defined as [156],

$$S_w(t) = \frac{\langle n^2(t)\rangle}{\langle n(t)\rangle} = \frac{\sum n^2(t)P(n(t))}{\sum n(t)P(n(t))}. \tag{4.4}$$

Here $P(n)$ is the probability of finding a cluster of size n, and $nP(n)$ is the probability that a randomly chosen mobile monomer belongs to a cluster of size n. $S_w(t)$ defined in this way is the average size of a cluster to which a randomly chosen mobile monomer belongs.

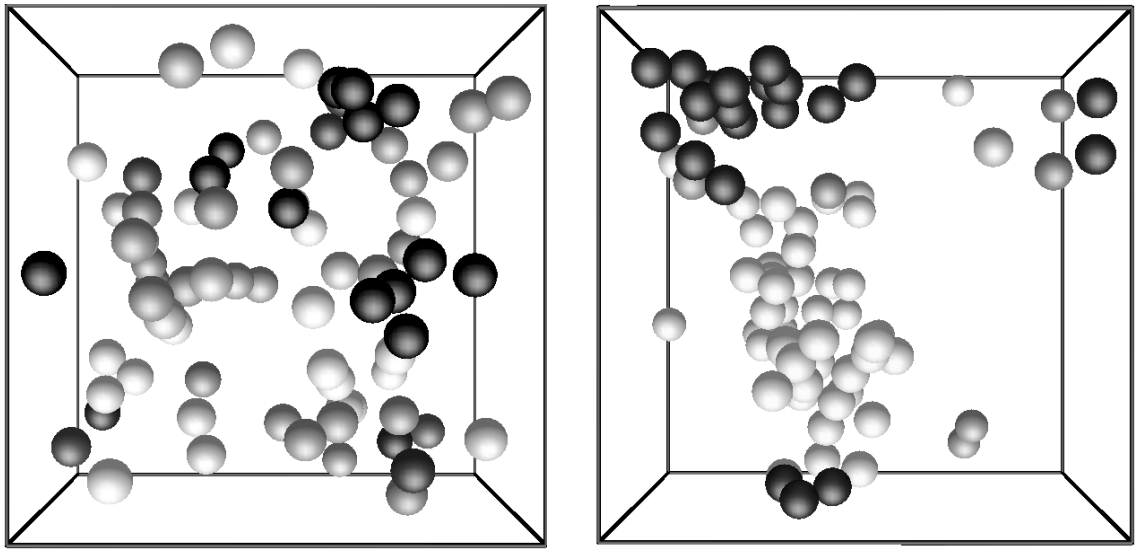

Figure 4.3: Clusters formed by the 6.5% most mobile monomers at early time $t = 0.002$ (left panel), and at intermediate time $t = 45.4$ (right panel), for $T = 0.46$. Each monomer is represented as a sphere, and connectivity information has been suppressed. Monomers belonging to the same cluster are colored the same shade of gray. Note that only the most mobile monomers out of the 1200 total monomers are shown in the figure.

We normalize S_w by the average size S_0 of clusters formed by mobile particles at the initial t (i.e., one MD time step), and find that, at each T, S_0 (reported in Table 4.1) coincides with the average cluster size found by selecting monomers randomly [155]. This demonstrates that short-time monomer motion is uncorrelated, as found previously for this system using an alternative (non cluster-based) analysis approach [79], and as found in both the LJ liquid referred to previously [78] and a colloidal suspension [20, 21]. Following convention [76, 156], any spanning clusters present in a given snapshot are omitted from the calculation to minimize finite

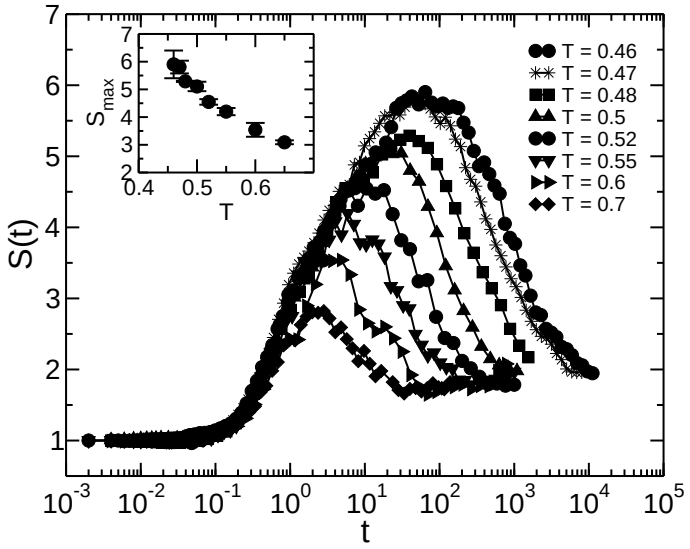

Figure 4.4: Normalized weight-averaged mean cluster size $S(t)$ for $\phi = 6.5\%$ for all T. The inset shows $S_{\max} \equiv S(t_{\rm clu}^{\max})$ versus T, where $t_{\rm clu}^{\max}$ is the time at which S is maximal.

size effects. In Fig. 4.4, we show the normalized mean cluster size $S \equiv S_w/S_0$ for several T. We find that the clusters formed by the most mobile monomers "grow" and "shrink" as the window of observation increases. Furthermore, the maximum amplitude of $S(t)$ shown in the inset of Fig. 4.4 increases with decreasing T, indicating that the monomer motion becomes increasingly spatially correlated as the melt becomes colder and more dense.

The behavior of $S(t)$ can be interpreted by comparing it with $\langle r_{\rm m}^2(t)\rangle$ (Fig. 3.4(b)). Consider, for example, the behavior of S at $T = 0.46$. For small t ($t \lesssim 2 \times 10^{-1}$), the monomers' motion is ballistic, and so the probability of finding large clusters is negligibly small, since we are simply choosing the most highly mobile monomers from the tail of the Maxwell-Boltzmann velocity distribution, and these monomers

are randomly distributed in space; at these short times $G_s(r,t)$ is well-approximated by a Gaussian. At slightly longer t ($2 \times 10^{-1} \lesssim t \lesssim 10$, the plateau regime) the motion of monomers is restricted to "rattling" within the cage formed by neighboring monomers. Thus, big clusters are less likely to be formed since the particles do not move large distances, and are consequently less likely to affect the motion of others. Between the plateau and sub-diffusive regimes ($10 \lesssim t \lesssim 10^3$), when the monomers begin to escape from their cages [82], the motion of one monomer becomes highly influenced by the motion of others around it; a monomer cannot move unless its neighbors also move, causing large clusters to be formed.

At the longest time scale we probe ($t \gtrsim 10^3$), the monomers' displacement is sub-diffusive (characterized by $\langle r_{\mathrm{m}}^2(t) \rangle \sim t^{0.62 \pm 0.03}$)[114, 157]. At this time scale, the probability of finding large clusters again decreases, indicating an increased tendency towards uncorrelated motion. Nevertheless, $S_w(t)$ is still greater than the random value S_0. This may be due to the presence of some correlation since the monomers are not yet completely diffusive, or possibly to polymer specific effects, or both.

The behavior of $S(t)$ is qualitatively similar to that of a generalized susceptibility $\kappa_U(t)$ calculated for this same system in Ref. [79]. κ_U is related to the volume integral of the displacement-displacement correlation function (essentially a density-density correlation function, but with every particle's contribution weighted by its scalar displacement in t) in the same way as the isothermal compressibility κ_T in a fluid is related to the volume integral of the density-density correlation function. Accordingly, $\kappa_U(t)$ is proportional to the fluctuations in the total system displacement at time t, in the same way as κ_T is proportional to the fluctuations in the

number of particles in the fluid.

The similarity between S and κ_U is not surprising, since the scalar displacements of the most highly mobile monomers in t are included in κ_U. However, the peak time of $S(t)$, which coincides with the crossover between the plateau and subdiffusive regimes, precedes that of $\kappa_U(t)$ (by less than a factor of 10 at $T = 0.46$). This suggests that the cooperative motion of monomers, which requires clustering and allows the monomers to escape from their cages, is a precursor to the more global dynamical heterogeneity measured by κ_U.

4.3.3 Variable fraction

The method outlined in the last section for selecting a fraction of mobile monomers ensures a clearly defined and reproducible subset of the most mobile monomers in a given time window, and can be easily applied to any system. However, there is no *a priori* reason why this should be the definition of choice, and in particular whether this fraction is more spatially correlated than some other fraction. The "ideal" fraction is the one that most clearly and naturally captures dynamical correlation. To search for this "natural" fraction, and to check if it is substantially different from the fraction used in the previous section, we select a subset of highly mobile monomers by varying ϕ, and then choosing that fraction which maximizes $S(t)$ for all t.

We find that $S(t)$ is maximum for ϕ in the range $5\% - 8\%$ for all T considered, e.g., S is maximum at $\phi = 5\%$ for $T = 0.46$ and at 8% for $T = 0.7$. Table 4.1 shows

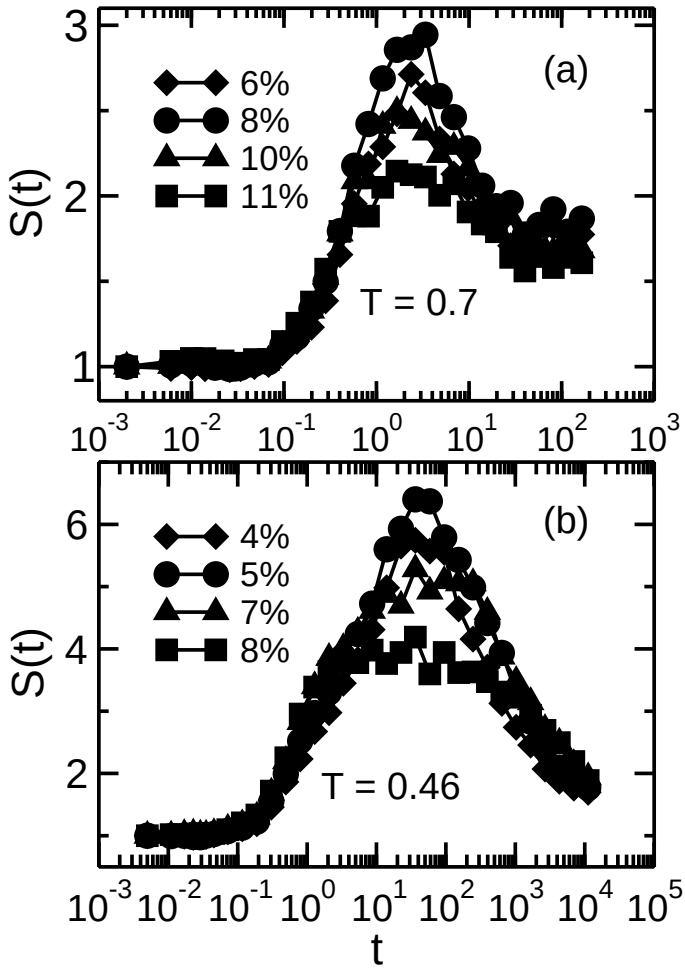

Figure 4.5: Normalized weight-averaged cluster size $S(t)$ as a function of time window for selected ϕ for (a) $T = 0.7$, and (b) $T = 0.46$. For these two state points, $S(t)$ is maximized by $\phi = 8\%$ and $\phi = 5\%$, respectively. Only a few selected fractions are shown for the clarity of the graph.

a complete list of fractions that maximize S at each T. In Fig. 4.5 we show $S(t)$ for $T = 0.46$ and $T = 0.7$ for four values of ϕ. We have checked S at 1% intervals of ϕ to determine the fraction ϕ that maximizes $S(t)$. However, for the sake of clarity, we show only a few representative ϕ, including the ϕ that yields a maximal value of

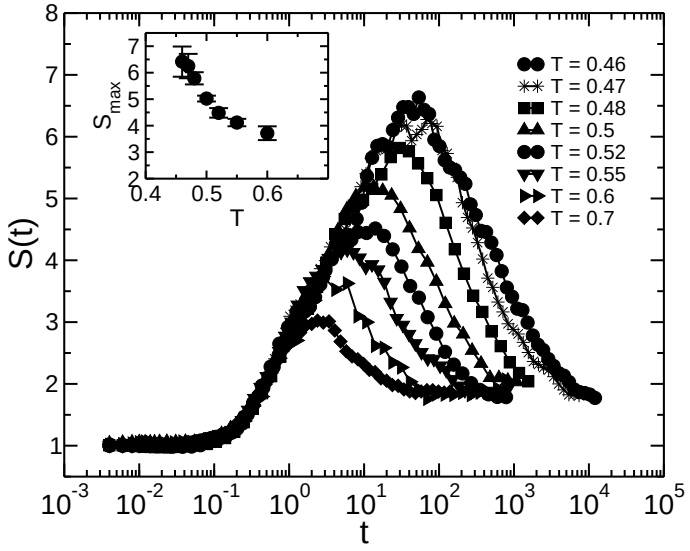

Figure 4.6: Normalized weight-averaged cluster size as a function of time for different temperatures using the fraction ϕ of mobile monomers that provides the largest average cluster for the given temperature. The inset shows $S_{\max} \equiv S(t_{\rm clu}^{\max})$ versus T, where $t_{\rm clu}^{\max}$ is the time at which S is maximal.

S.

Using those fractions ϕ^{II} reported in Table 4.1 that maximize the cluster size for each T, we calculate $S(t)$ for each T, as shown in Fig. 4.6. We find that this second method does not alter the qualitative features of the time and temperature dependence of S found by the first method. However, there are slight quantitative differences in the values of S. The peak values of S obtained from the variable fraction method are slightly larger (at most by $\approx 13\%$) than the method that at each T uses fixed $\phi = 6.5\%$. Since this difference in S from the two methods is not dramatic, we perform all subsequent analysis using a fixed fraction $\phi = 6.5\%$ for all T.

4.3.4 Temperature dependence of peak average cluster size

We next focus on the temperature dependence of the maximum value $S_{\max}$ of $S(t)$, and the time $t_{\rm clu}^{\max}$ at which $S(t)$ is maximum. Examination of Fig. 3.4(b) shows that $t_{\rm clu}^{\max}$ is in the time window when monomers escape from their cages, as characterized by the increase of $\langle r_{\rm m}^2(t)\rangle$ from the plateau towards the sub-diffusive regime. This time also corresponds to the late-β/early-α relaxation regime [17]. The shift in $t_{\rm clu}^{\max}$ to longer t as T decreases reflects the increase in the time scale necessary for a monomer to break free from its cage, which requires the participation of larger groups of monomers on increased cooling towards T_g. At each T, this peak time is close to, but slightly earlier than, the time scale where $\alpha_2(t)$ is maximum.

The T dependence of $t_{\rm clu}^{\max}$ can be studied by fitting the data by various functional forms. One choice is to fit the data by a power law. In the temperature regime we study, other characteristic times, such as τ_α, follow power law behavior, as predicted by the mode-coupling theory [17]. Additionally, the time $t_{\alpha_2}^*$ when the non-Gaussian parameter is maximum has been found to follow a power law in $(T - T_{\rm MCT})$ [141]. Also, Ref. [79] found that the time when the correlations as measured by a displacement-displacement correlation function are maximum, can be fit by a power law in $(T - T_{\rm MCT})$ [79]. Motivated by these findings, we fit the data by

$$t_{\rm clu}^{\max} \sim (T - T_{\rm MCT})^{-x}. \tag{4.5}$$

Fig. 4.7(a) shows the best fit obtained by fixing $T_{\rm MCT} = 0.45$. $t_{\rm clu}^{\max}$ shows a reasonable power-law behavior with $x = 1.47 \pm 0.16$ in the temperature regime where MCT holds. The deviation from power law at the lowest T, which is commonly

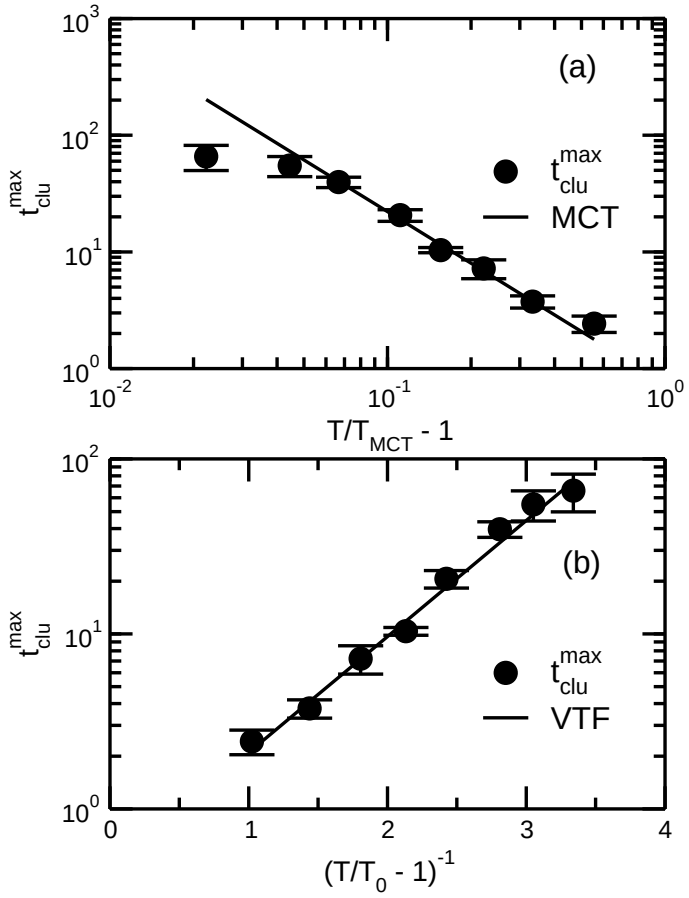

Figure 4.7: Temperature dependence of $t_{\rm clu}^{\rm max}$, using the top 6.5% mobile monomers, fitted to (a) a power law (Eq. 4.5), with $T_{\rm MCT} = 0.45$, yielding $\gamma = 1.47 \pm 0.16$, and (b) a VTF expression (Eq. 4.6), yielding $E = 0.54 \pm 0.07$, and $T_0 = 0.35 \pm 0.02$ plotted on linear-log axis.

observed for dynamical quantities, is expected due to the breakdown of MCT near $T_{\rm MCT}$ [158, 159].

Another functional form for the T-dependence of dynamical quantities that often holds in supercooled liquids is given by the well known Vogel-Tammann-Fulcher (VTF) equation [29],

$$t_{\rm clu}^{\rm max} \sim \exp\left(\frac{E}{T-T_0}\right), \tag{4.6}$$

The fit of this expression to our data is shown in Fig. 4.7(b). We find a reasonable agreement with the VTF form with $T_0 = 0.35 \pm 0.02$; this value agrees with the value $T_0 = 0.34 \pm 0.02$ found by fitting τ_α, defined as the time at which the incoherent (self) part of the intermediate scattering function $F_q^{inc}(\tau_\alpha) = 0.3$ [114]. Note that τ_α occurs at a later time than $t_{\rm clu}^{\rm max}$.

MCT predicts the β-time scale t_ϵ satisfies the relation $t_\epsilon \sim (T - T_{\rm MCT})^{-1/2a}$ where a is uniquely determined by fixing any other exponent used by MCT. However, unambigious identification of t_ϵ with the appropriate scaling has been notoriously difficult in simulations of supercooled liquids. Within an MCT analysis of the simulation data of the same system, it was found that $\gamma = 1.95$, and hence MCT predicts $a = 0.352$ [100, 114], and thus $1/2a = 1.42$, which is within numerical uncertainity of the exponent x determined from $t_{\rm clu}^{\rm max}$. Thus, for the polymer melt studied, $t_{\rm clu}^{\rm max}$ follows the predicted scaling law for the β time scale, and signifies a connection between dynamic heterogeneity and MCT. The correspondence of $t_{\rm clu}^{\rm max}$ with t_ϵ, and the fact that $t_{\rm clu}^{\rm max}$ and τ_α would appear to diverge at the same temperature if the functional forms continued to hold to lower T, suggests that the two time scales represent a hierarchy of events in the relaxation process, consistent with the MCT prediction of two time scale relaxation process.

The observation of a growing cluster size shown in the inset of Fig. 4.4 and

4.6 is consistent with the results of Ref. [76], and with earlier hypotheses that dynamics in supercooled fluids involves the motion of molecules within "cooperatively rearranging regions" [12, 13, 160, 161], whose size grows as the glass transition is approached on cooling. In the Adam-Gibbs theory [12], the smallest possible size z^* that can give rise to a cooperative rearrangement is inversely proportional to the configurational entropy of the system [5], which is a measure of the number of mechanically stable states sampled by the system. The direct connection between z^* and the mean cluster size of mobile monomers in our analysis is not trivially obvious for the polymer melt studied, although there are some indications for other systems, as will be described in Chapter 7. However, we can discuss the implications of growing cluster sizes in the spirit of the Adam-Gibbs theory, which has proved to be useful for the interpretation of transport and relaxation in supercooled liquids [96, 162, 163, 164, 165].

The Adam-Gibbs theory predicts that a thermodynamic glass transition occurs at a finite T as the configurational entropy vanishes. As a consequence, the theory also predicts z^* diverges at non-zero temperature. However, our result for $S_{\max} \equiv S(t_{\rm clu}^{\max})$ showing Arrhenius T (i.e. VTF with $T_0 = 0$, Fig. 4.8) dependence over the (admittedly narrow) range of temperatures we have simulated implies that the mean cluster size does not diverge at non-zero temperature [166]. This may indicate that the T range studied is too far from T_g to reliably estimate the T at which S might diverge. Alternatively, $S_{\max}$ may not be an appropriate measure of z^* for the polymer melt, which needs to be proven.

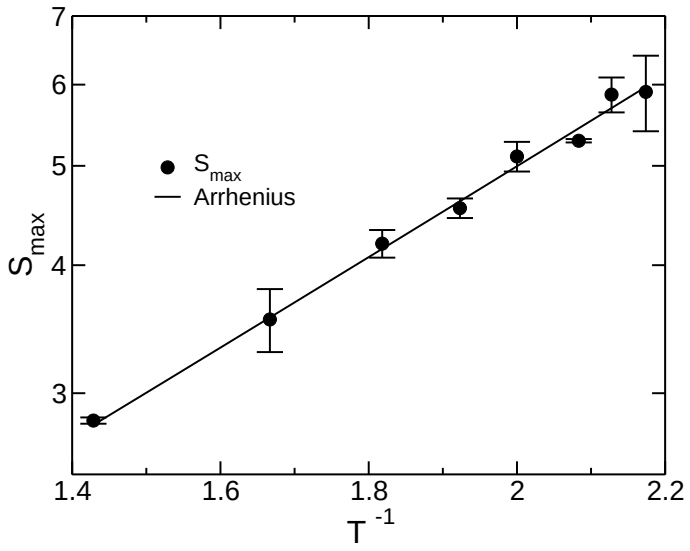

Figure 4.8: Temperature dependence of $S_{\max}$ fitted to the Arrhenius form $S_{\max} \sim \exp\left(\frac{\mathrm{E}}{\mathrm{T}}\right)$, with $E = 0.98 \pm 0.02$, with $S_{\max}$ plotted logarithmically

4.3.5 Cluster size distribution

In the previous section, we examined the average cluster size S. Here we investigate the cluster size distribution $P(n)$, and study both the time and temperature dependence of this quantity. We first consider $P(n)$ at the lowest temperature ($T = 0.46$) for several t (Figs. 4.9 (a) and (b)). At early times, we find that $P(n)$ is dominated by smaller clusters, as expected from the fact that $S(t)$ is small at early t. As t increases through the plateau regime of $\langle r_{\mathrm{m}}^2(t)\rangle$ (Fig. 3.4), larger clusters contribute significantly to $P(n)$. As t continues to increase into the sub-diffusive regime, $P(n)$ again becomes dominated by small clusters.

We now compare $P(n)$ for different T at the time $t_{\mathrm{clu}}^{\max}$. Fig. 4.10 shows that as T decreases, $P(n, t_{\mathrm{clu}}^{\max})$ becomes dominated by larger clusters. This is a consequence of correlated motion of monomers as T approaches T_{MCT}, which is expected from

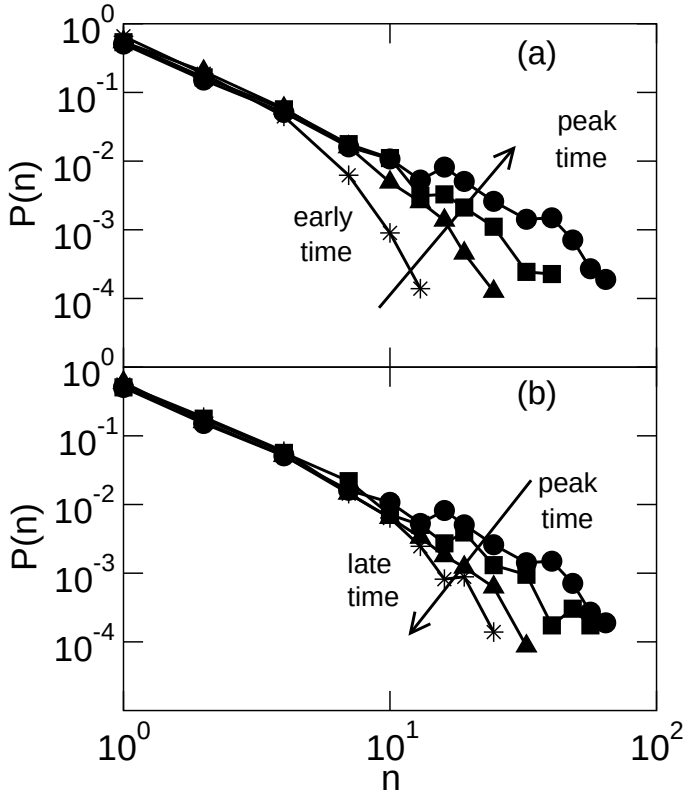

Figure 4.9: Probability distribution $P(n)$ of cluster sizes at $T = 0.46$ for different times as time progresses (a) from early time ($t = 0.02$) to the peak time ($t = 65.9$), and (b) from the peak time ($t = 65.9$) to the sub-diffusive regime ($t = 11939.5$).

the behavior of $S(t)$ presented earlier. We find that $P(n, t_{\rm clu}^{\rm max})$ can be fit by a power law with exponential cutoff [156],

$$P(n) \sim n^{-\tau} \exp\left(-n/n_0(T)\right) \tag{4.7}$$

where $n_0(T)$ is a characteristic cluster size for the given T. The corresponding data collapse is shown in the inset of Fig. 4.10. The collapse is not nearly as good as for the larger system studied in Ref. [76], which may be due to finite size effects. We find $\tau = 1.62 \pm 0.12$ for all T. This value is smaller than the value obtained

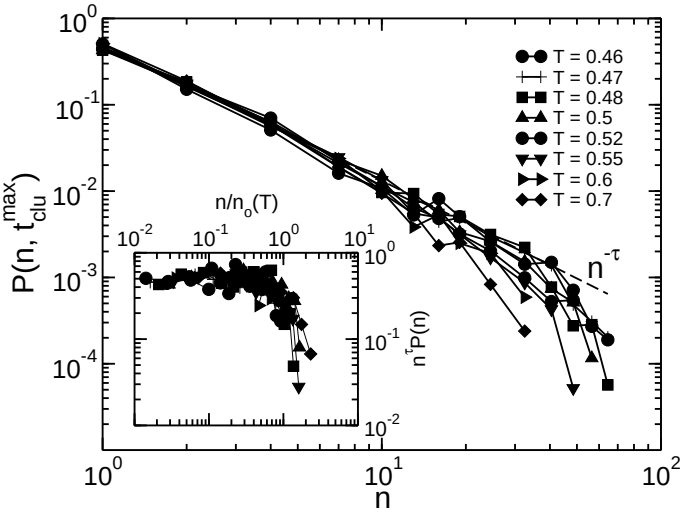

Figure 4.10: $P(n, t_{\rm clu}^{\rm max})$ for the 6.5% most mobile monomers as a function of cluster size n for different T. The dashed line is a simple power law fit $P(n) \sim n^{\tau}$ with $\tau = 1.62$ for $T = 0.46$. The inset shows the same data scaled as indicated to show data collapse.

for a binary mixture of LJ particles ($\tau \approx 1.9$) [76], and for colloids ($\tau = 2.2 \pm 0.2$) [21], suggesting that the exponent value may be non-universal. Fig. 4.11 shows that $n_0(T)$ increases as T decreases, causing the probability distribution to approach a simple power law with decreasing T.

4.3.6 Dynamic correlation length

It is straightforward to calculate the correlation (or connectivity) length ξ of the clusters analyzed in the previous section. In lattice percolation theory, the correlation or connectivity length ξ, given by

$$\xi^2 = \frac{\sum r^2 g(r)}{\sum g(r)}, \tag{4.8}$$

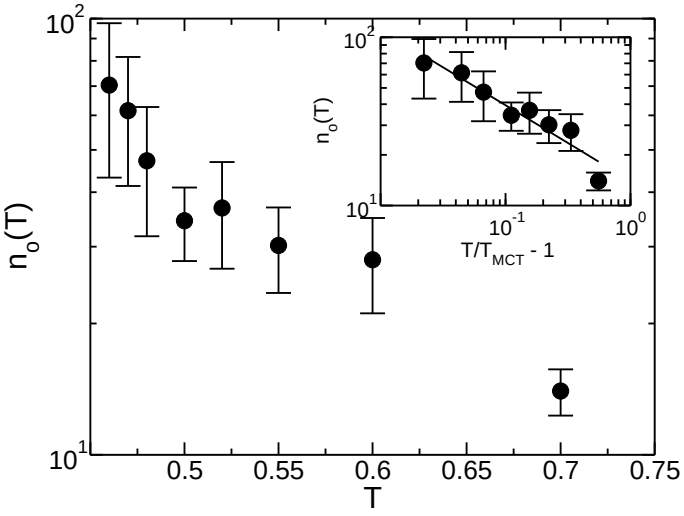

Figure 4.11: A cutoff cluster size $n_0(T)$ obtained from fitting Eq. 4.7 to the data, plotted as a function of T. The error bar is estimated by fixing $\tau = 1.62$, and then determining the range of n_0 values in Eq. 4.7 which reasonably fit the $P(n)$ data. Inset: a power law fit to the $n_0(T)$ data, i.e., $n_0(T) \sim (T - T_{\rm MCT})^{-\gamma}$, using $T_{\rm MCT} = 0.45$. The value of γ obtained from the fit is $\gamma = 0.45 \pm 0.08$.

is defined as the root-mean-square distance between two sites belonging to the same cluster [156], where r is the distance between two sites and $g(r)$ is the pair correlation or pair connectivity function, defined as the probability that a site a distance r from an occupied site belongs to the same cluster. To map this definition onto the off-lattice system we consider, we define ξ as the root-mean-square distance between two monomers in a cluster, where $g(r)$ is the probability that a monomer a distance r from another monomer belongs to the same cluster. The sum in Eq. 4.8 runs over all monomers in the cluster.

Eq. 4.8 may be rewritten in terms of the cluster size n and the radius of

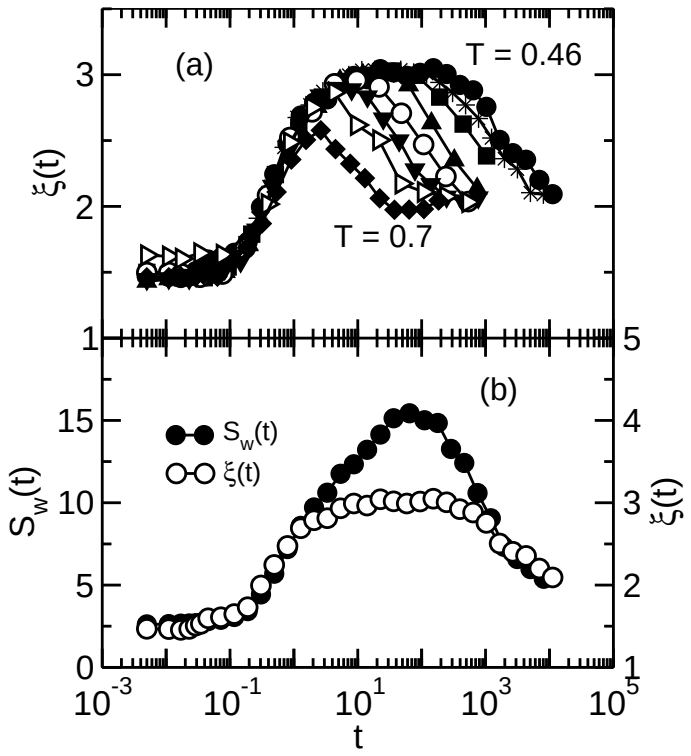

Figure 4.12: (a) Dynamic correlation length $\xi(t)$ for different T. (b) $S_w(t)$ plotted together with $\xi(t)$ for $T = 0.46$ to emphasize that both quantities increase and decrease on the same timescale. Note that here the mean cluster size is not normalized by S_0.

gyration R_n as [156]

$$\xi^2 = \frac{2\sum R_n^2 n^2 P(n)}{\sum n^2 P(n)}. \tag{4.9}$$

R_n is defined by

$$R_n = \frac{\sum_i \sum_j |\mathbf{r}_i - \mathbf{r}_j|^2}{2n^2}, \tag{4.10}$$

where $\mathbf{r}_i$ and $\mathbf{r}_j$ refer to the position of monomers i and j, where i and j are within the same cluster.

Fig. 4.12(a) shows the dynamic correlation length $\xi(t)$ for several T. We

find that $\xi(t)$ exhibits a time and temperature dependence similar to that of $S(t)$ (Fig. 4.12(b)), *i.e.* it grows and shrinks with t, and indicates a dynamic correlation length that increases on cooling. This similarity is not surprising since ξ is related to the average radius of the clusters that contribute significantly to S [156]. At the largest t accessible to our simulations, $\xi(t)$ does not decay to the initial value, also observed for S. The maximum value of $\xi(t)$ appears to saturate to the same value of approximately $\xi_{\max} \approx 3.1$ for $T \lesssim 0.5$. This saturation is likely due to the small system size of the simulation, since the maximum value of ξ for these temperatures approaches half the system length ($\frac{L}{2} \approx 5.25$) in our simulation. Indeed, finite size effects have been reported for simulation studies of dynamical heterogeneity, and in many of the configurations examined in the present work, clusters were found that spanned the entirety of the simulation box. In Fig. 4.13 we plot $\xi(t_{\text{clu}}^{\max})$ as a function of T. It is clear from Fig. 4.13 that we do not detect a tendency toward divergence of ξ in the temperature range studied. In Chapter 6 we investigate spatially heterogeneous dynamics in a substantially larger system, where finite size effects are minimized in the range of temperatures studied.

It is interesting to note that a number of other studies [81, 88, 91, 83] have also calculated a dynamic correlation length. It is thus instructive to compare the behavior of the correlation length found in our system of polymer melt with those. Ref. [88] calculates the static structure factor for particle pairs whose "bond" has been broken, with a dynamic bond criterion based on particle separation. Using the Ornstein-Zernike formalism [124], they extract a correlation length for particle pairs with broken bonds. This correlation length is not unlike ours, since for particles

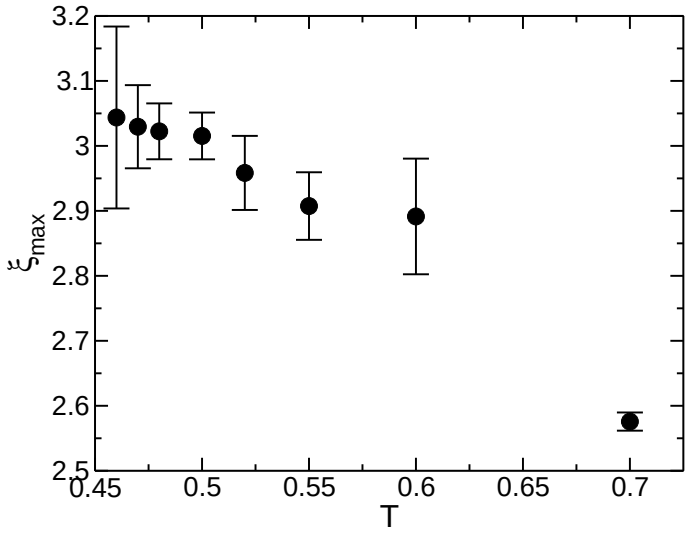

Figure 4.13: Dynamic correlation length $\xi(t_{\rm clu}^{\rm max})$ as a function of T.

to have broken a "bond", they must have moved apart from each other. The main difference is the absence of a well-defined time scale on which this motion occurs. Ref. [88] finds that the correlation length defined in that way grows with decreasing T, but saturates at low T because the correlation length approaches the system size, very similar to what we observe for the polymer melts.

Refs. [78, 81] and [91] calculate pair correlation functions based on the deviation of the displacement of each particle from the average value as a function of t. Ref. [91] finds that the tail of the spatial correlation function may be fit by an exponential, and from this extract a correlation length. Unlike the length calculated in our system of polymer melt, and that calculated in Ref. [81], this correlation length was found to saturate at a (roughly) constant value at times much longer than the α relaxation time. The behavior observed in Ref. [91] is surprising, since it implies that there exist *persistent* spatial correlations in the particle motion on time

scales that exceed all other relaxation times, unexpected for an ergodic liquid. We note that although we do find some "saturation" of the correlation length around the peak time at low T (possibly due to finite-size effects as discussed above), the length decreases at long times (and must decrease to its "random" value at sufficiently long times when the liquid is diffusive).

Chapter 5

Spatially heterogeneous dynamics in a polymer melt: String-like motion

The second part of the study of dynamical heterogeneity in the polymer melt is presented in this chapter where we focus on the transient nature and temperature dependence of the string-like motion in this system. In the LJ system, the nature of cooperative motion was further investigated, and it was discovered that mobile particles follow each other in quasi one-dimensional paths, forming strings. Prior to our study, this analysis has not been performed for a polymer system. In particular, it is interesting to ask whether, if strings exist, chain connectivity plays any role. Therefore the goals of this work were two-fold: (i) to ascertain the tendency (or lack thereof) for monomers in the supercooled melt to follow each other in string-like

paths, and (ii) to study the influence of chain connectivity on dynamical heterogeneity and strings. A study of the transient nature of the strings, which was also lacking in earlier studies, was also performed. The work presented in this chapter, and published in Ref. [126] was carried out in collaboration with M. Aichele, a PhD student at Johannes Gutenberg Universität, Mainz, Germany under the guidance of our advisors S. C. Glotzer and J. Baschnagel.

5.1 Mobile monomers and the role of chain connectivity

Thus far, we have discussed the motion of mobile monomers without distinguishing whether or not they are connected to each other. The interplay of connectivity and mobility is one of the main issues we wish to address. Intuitively, one may expect that the bonds in a chain provide a preferred direction along which mobility can be "transmitted". To investigate this polymer-specific effect we calculate the mean contiguous segment length $N_{\mathrm{c,m}}(t)$, which is defined as the average number of mobile monomers that are consecutively bonded to each other on a given polymer chain, averaged over all chains that contain at least one mobile monomer. Figure 5.1 illustrates this definition and shows that there can be multiple contiguous segments on a single chain. One can simply imagine this quantity as a cluster of bonded mobile particles, since by definition if any two monomers are mobile and bonded then they will definitely be within the first neighbor distance of each other, and

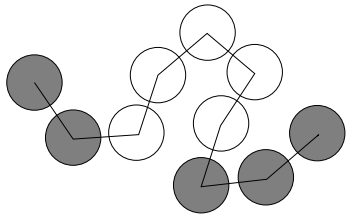

Figure 5.1: A schematic diagram showing how the average length of contiguous segments of mobile monomers, $N_{\mathrm{c,m}}(t)$, is defined. Assume that, at time t, only the shaded monomers are considered to be mobile. The first two monomers constitute a contiguous segment of length 2, followed by five non-mobile monomers, followed by another contiguous segment of length 3. So, the average length of contiguous mobile segments is $N_{\mathrm{c,m}}(t_\mu) = 2.5$.

hence belong to the same cluster. This definition is thus a special subset of the clusters defined above, where here all mobile monomers in a cluster that are not bonded to each other are excluded. In this way we can gain some insight into the contribution of connectivity on cooperativity.

5.1.1 Correlations of mobile monomers in a chain

Figure 5.2 shows $N_{\mathrm{c,m}}(t)$ for all temperatures studied. In the ballistic regime we expect no correlations [79, 134], and find $N_{\mathrm{c,m}} \approx 1.06$. The value $N_{\mathrm{c,m}} \approx 1.06$ could also be obtained by calculating $N_{\mathrm{c,m}}$ after selecting 6.5% of monomers at random and labeling them as "mobile". Therefore, no significant dynamic correlations between bonded nearest neighbors exist in the ballistic regime.

Beyond the ballistic regime $N_{\mathrm{c,m}}(t)$ increases, but never exceeds ~ 1.5 as long

as t is not significantly larger than $t^*_{\alpha_2}$. Thus, in the studied temperature interval, the relaxation mechanism does not correspond to the sliding motion of many consecutive monomers along the backbone of the chain, since that would require $N_{\mathrm{c,m}}$ to be of order N. The small value of $N_{\mathrm{c,m}}$ rather suggests that the relaxation in the β-regime is predominantly determined by the dense local packing of the melt and not by chain connectivity. This is consistent with the degree to which the ideal MCT for simple liquids is successful at describing the dynamics of polymer melts. However, this does not imply that chain connectivity is completely irrelevant. For $T \leq 0.7$ (below the onset of caging), $N_{\mathrm{c,m}}(t)$ exhibits a maximum at $t^{\mathrm{max}}_{\mathrm{seg}}$ (slightly larger than $t^*_{\alpha_2}$) in the time window of the late-β/early-α process, before it first decreases and eventually increases as specifically shown for $T = 0.46$ and $T = 1.0$. This maximum increases, and also shifts to longer t, upon cooling toward T_{MCT}. Thus, in a relative sense, the colder the melt, the larger the tendency for finding bonded mobile monomers along the chain.

For times significantly larger than $t^{\mathrm{max}}_{\mathrm{seg}}$ the spatial correlations between mobile monomers first diminish, where the length of the contiguous segments relaxes back to a minimum. The minimum occurs at $t^{\mathrm{min}}_{\mathrm{seg}}$ which roughly corresponds to the time where $\langle r^2_{\mathrm{m}}(t)\rangle = 1$ (subdiffusive regime). For $t > t^{\mathrm{min}}_{\mathrm{seg}}$, the crossover to free diffusion takes place and $N_{\mathrm{c,m}}$ continuously increases, possibly converging to its upper limit N, because the displacement of the center of mass is predicated upon a concomitant motion of many monomers in the chain.

The occurrence of the maximum and the minimum suggests that there are two relaxation mechanisms at low temperatures: One, occurring when $N_{\mathrm{c,m}}(t)$ is

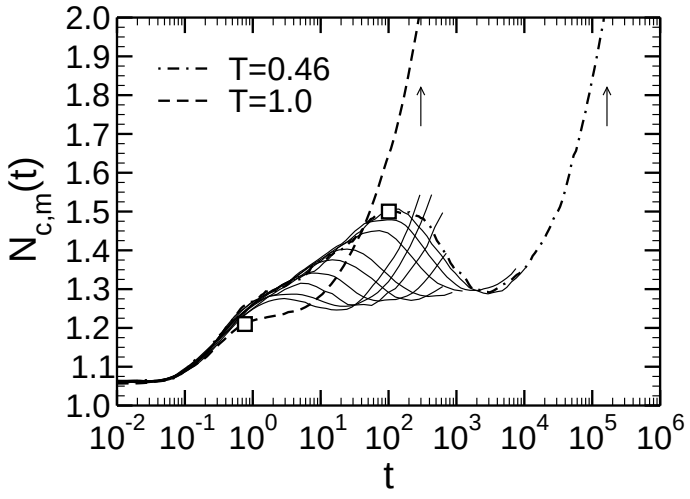

Figure 5.2: Mean contiguous segment length $N_{\rm c,m}(t)$ versus time for all temperatures. Besides $T = 1$ (dashed curve) and $T = 0.46$ (dash-dotted curve) the following temperatures are shown (solid curves from left to right): $T = 0.7$, 0.65, 0.6, 0.55, 0.52, 0.5, 0.48, 0.47. For $T = 1$ and $T = 0.46$, the open squares indicate $t^*_{\alpha_2}$ and the arrows indicate the time when $\langle r^2_{\rm m}(t)\rangle = R^2_{\rm e}$.

a maximum, corresponds to the cage-breaking process. Here, clustering of highly mobile monomers is most pronounced, irrespective of whether they are bonded to each other or not. Cluster formation related to cage breaking is also observed in other non-polymeric systems, e.g., in the simulations of binary LJ-mixture [76] and water [167] close to $T_{\rm MCT}$, as well as in experiments on colloidal suspensions close to the glass transition [21]. Thus, chain connectivity is not necessary for clustering. As in the non-polymeric liquids, the clustering is rather a consequence of the self-generated cooperativity between the local motion of the caged monomers in the cold melt. To a large extent, this cooperativity is lost as $N_{\rm c,m}$ crosses over to the minimum. The minimum and the subsequent steep rise, which correspond to

the second relaxation mechanism, are a signature of Rouse-like, polymer-specific dynamics because they are, at least as precursors, already present at $T = 1$ where no caging occurs.

5.1.2 Mobile end monomers

In addition to $N_{\mathrm{c,m}}$ we also analyzed the mobility of end monomers as compared to central monomers in polymer chains by calculating the fraction $f_{\mathrm{e,m}}$ of mobile monomers that are end monomers. Figure 5.3(a) shows the time evolution of $Nf_{\mathrm{e,m}}/2$ for all temperatures. The factor $N/2$ takes into account that the *a priori* probability of finding an end monomer among the N monomers of a chain is $2/N$. If the mobility of the ends cannot be distinguished from the average, $Nf_{\mathrm{e,m}}/2$ should be 1. This is the case in the ballistic regime, where the monomers are independent of each other, and in the diffusive regime, where they follow the motion of the center of mass. At intermediate times, however, we find $Nf_{\mathrm{e,m}}/2 > 1$, and hence chain ends are more mobile than inner monomers.

We compare $Nf_{\mathrm{e,m}}/2$ with the ratio $\left\langle r^2_{\mathrm{e,\,m}}(t)\right\rangle / \langle r^2_{\mathrm{m}}(t)\rangle$, where $\left\langle r^2_{\mathrm{e,\,m}}(t)\right\rangle$ is the MSD of the end monomers [Fig. 5.3(b)]. The time dependence of $Nf_{\mathrm{e,m}}/2$ closely resembles that of $\left\langle r^2_{\mathrm{e,\,m}}(t)\right\rangle / \langle r^2_{\mathrm{m}}(t)\rangle$. This similarity is not unexpected, since both quantities measure the mobility of the end mobiles with respect to the average. Therefore, to understand the behavior of $Nf_{\mathrm{e,m}}/2$ better, we turn our attention to $\left\langle r^2_{\mathrm{e,\,m}}(t)\right\rangle / \langle r^2_{\mathrm{m}}(t)\rangle$.

When leaving the ballistic regime $\left\langle r^2_{\mathrm{e,\,m}}(t)\right\rangle / \langle r^2_{\mathrm{m}}(t)\rangle$ first increases. This in-

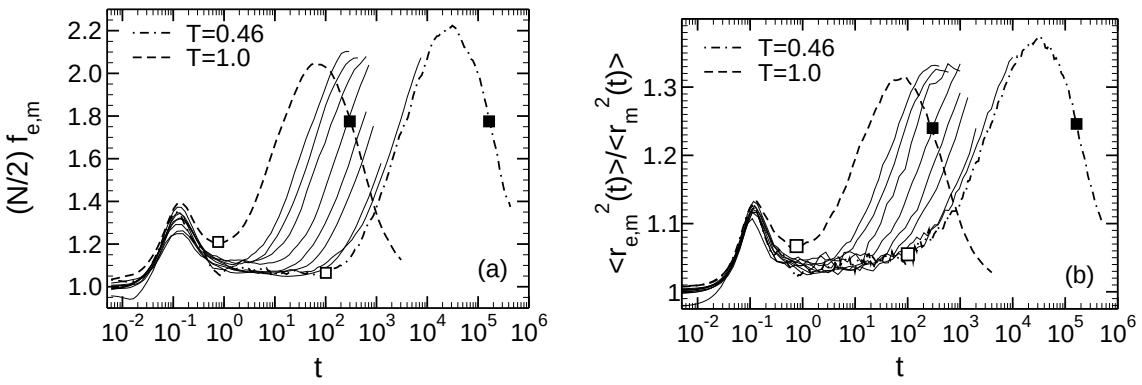

Figure 5.3: Panel (a): Fraction of mobile end monomers $f_{\mathrm{e,m}}$ versus t. $f_{\mathrm{e,m}}$ is multiplied by $N/2$ $(= 5)$ to account for the fact that there are only two ends per chain. Besides $T = 1$ (dashed curve) and $T = 0.46$ (dash-dotted curve) the following temperatures are shown (solid curves from left to right): $T = 0.7$, 0.65, 0.6, 0.55, 0.52, 0.5, 0.48, 0.47. For $T = 1$ and $T = 0.46$, the open squares indicate $t^*_{\alpha_2}$, the filled squares the time when $\langle r^2_{\mathrm{m}}(t)\rangle = R^2_{\mathrm{e}}$. Panel (b): Same as in panel (a), but for $\left\langle r^2_{\mathrm{e,m}}(t)\right\rangle / \langle r^2_{\mathrm{m}}(t)\rangle$. $\left\langle r^2_{\mathrm{e,m}}(t)\right\rangle$ is the MSD of the end monomers.

crease can be understood by the short-time expansion of the MSD of monomer i [125]

$$\left\langle [\mathbf{r}_i(t) - \mathbf{r}_i(0)]^2 \right\rangle = 2t \int_0^t \mathrm{d}t' \left(1 - \frac{t'}{t}\right) \left\langle \mathbf{v}_i(t) \cdot \mathbf{v}_i(0) \right\rangle \approx 3Tt^2 \left[1 - \frac{\langle |\mathbf{F}_i|^2 \rangle}{36T} t^2\right] \quad (t \text{ small}) \,,$$

where $\langle \mathbf{v}_i(t) \cdot \mathbf{v}_i(0)\rangle$ is the velocity auto-correlation function and $\mathbf{F}_i$ the total force on monomer i. Since an end is only bonded to one monomer, $\mathbf{F}_i$ is smaller than for inner monomers. Thus, one expects $\left\langle r^2_{\mathrm{e,m}}(t)\right\rangle / \langle r^2_{\mathrm{m}}(t)\rangle > 1$ (and hence $N f_{\mathrm{e,m}}/2 > 1$) for times just outside the ballistic regime. In our model the ratio continues to increase up to a maximum that occurs around $t \approx 0.13$ for all T. This is close to the time where the velocity auto-correlation function becomes negative [100].

Notice that if the microscopic "collision" time τ_{collison} is defined as the time when the velocity autocorrelation function first changes sign, as was done in Ref [76], this time corresponds to τ_{collison}. The inversion of the initial direction of the velocity is caused by rebounding collisions between a monomer and its neighbors. It is typical of dense liquids and must occur in the same way for end and inner monomers. Therefore, the difference in mobility should diminish and $\left\langle r^2_{\text{e, m}}(t)\right\rangle / \left\langle r^2_{\text{m}}(t)\right\rangle$ should decrease. In fact, the simulation shows that, for longer times, $\left\langle r^2_{\text{e, m}}(t)\right\rangle / \left\langle r^2_{\text{m}}(t)\right\rangle$ first decreases toward a plateau and then, at about $t \approx t^*_{\alpha_2}$, crosses over to a steep rise. The rise reaches a maximum close to the time where the MSD of the center of mass equals R^2_{g}. This roughly corresponds to the Rouse time τ_{R} [150] of our model. Thereafter, the transition to free diffusion takes place.

The enhanced mobility of the end monomers for $t > t^*_{\alpha_2}$ is not unexpected. The Rouse theory predicts $\left\langle r^2_{\text{e, m}}(t)\right\rangle / \left\langle r^2_{\text{m}}(t)\right\rangle = 2$ in the time regime where the monomer displacement follows a $t^{1/2}$ behavior (i.e., for $t \leq \tau_{\text{R}}$) [150]. In the present simulation, the maximum of $\left\langle r^2_{\text{e, m}}(t)\right\rangle / \left\langle r^2_{\text{m}}(t)\right\rangle$ is smaller than 2, partially due to short chain effects. Longer chains may attain the Rouse prediction more closely if entanglements can be neglected [148].

5.2 String-like motion

In this section we present the transient nature and temperature dependence of the strings formed by mobile monomers. To identify these strings we compare snapshots of the monomers configurations at two different times, say at some reference time

$t = t_0$ and at a later time $t = t_0 + t$, and then search for mobile monomers that have replaced neighboring mobile monomers within a radius δ [75]. More precisely, we construct strings by connecting any two mobile monomers i and j if

$$\min\big[|\mathbf{r}_i(t_0 + t) - \mathbf{r}_j(t_0)|, |\mathbf{r}_i(t_0) - \mathbf{r}_j(t_0 + t)|\big] < \delta \; . \tag{5.1}$$

This equation means that monomer i moved from $\mathbf{r}_i(t_0)$ to $\mathbf{r}_i(t_0 + t)$ in time t, while the other monomer j simultaneously approached the initial position of i within a sphere of radius δ. δ must be sufficiently smaller than the Lennard-Jones diameter σ $(= 1)$ to guarantee that j unambiguously replaces i. For the binary LJ-mixture, a good choice was $\delta = 0.6$ [75]. For the polymer model studied $\delta = 0.55$ was selected based on the consideration that the replacement can be unambiguosly identified in that only 0.2% of the replacements result in branching strings. This type of strings arise when more than one monomer j "replace" i simultaneously.

Once strings are identified at any time t, their transient nature and T-dependence may be studied by calculating the mean string length. In principle, there are two different approaches for defining the mean string length that may have relevance here. One definition is provided by the weight averaged string length $L_{\mathrm{w}}(t)$, given by [168]

$$L_{\mathrm{w}}(t) = \frac{\langle l^2 \rangle}{\langle l \rangle} = \frac{\sum_{l=1}^{\infty} l^2 P(l)}{\sum_{l=1}^{\infty} l P(l)} \; , \tag{5.2}$$

where $l \equiv l(t)$ and $P(l)$ are the string length and the probability of finding a string of length l, respectively. This definition is relevant in the context of percolation theory and, as shown in the last section, has been used to analyze the mean size of clusters formed by mobile monomers. A second definition is the number averaged string length $L(t)$, defined by [168]

$$L(t) = \langle l \rangle = \frac{\sum_{l=1}^{\infty} lP(l)}{\sum_{l=1}^{\infty} P(l)} \,, \tag{5.3}$$

where $\sum_{l=1}^{\infty} P(l) = 1$. This definition arises in the context of equilibrium polymers [169], which we discuss later in this chapter as it appears to be relevant to the present study. Thus it is this definition on which we focus in this chapter. For completeness, however, we present the results for $L_{\mathrm{w}}(t)$ at the end of this chapter. In order to investigate whether or not chain connectivity favors the formation of strings, and to understand the interplay of connectivity and mobility, we additionally calculate the number averaged string length $L_{\mathrm{seg}}(t)$ of contiguous segments of mobile monomers in a chain. The comparison of the two quantities should reveal the contribution of chain connectivity to the formation of strings.

Figure 5.4 shows the time evolution of $L(t)$ and $L_{\mathrm{seg}}(t)$. Qualitatively, both quantities behave in the same way. At short times, $L(t)$ and $L_{\mathrm{seg}}(t)$ are equal to one. A string length of one means that the mobile monomers are separated from each other, and do not replace other mobile monomers when they move [170]. For a replacement to occur the shortest distance a monomer must travel is the nearest-

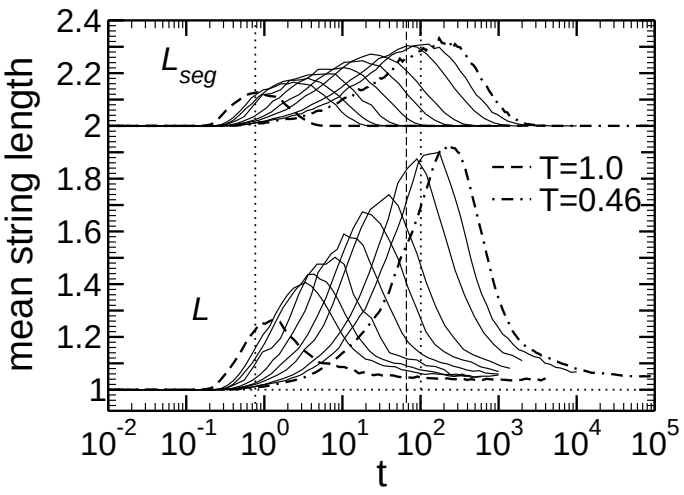

Figure 5.4: Average string length $L(t)$ of all mobile monomers and average string length in contiguous segments of mobile monomers $L_{\rm seg}(t)$ versus t with replacement parameter $\delta = 0.55$. The temperatures shown are (from left to right): $T = 1$, 0.7, 0.65, 0.6, 0.55, 0.52, 0.5, 0.48, 0.47, 0.46. The dotted vertical lines in the lower panel indicate the time when α_2 is maximum for $T = 1$ ($t^*_{\alpha_2} = 0.766$) and $T = 0.46$ ($t^*_{\alpha_2} = 100.894$), whereas the dashed vertical line indicates the time $t^{\rm max}_{\rm clu}$ when the cluster size is maximum for $T = 0.46$ ($t^{\rm max}_{\rm clu} = 65.85$). A string length of one corresponds to an isolated mobile monomer, i.e., no "bond" could be formed between two mobile monomers via the replacement criterion of Eqs. (5.1). Fig. 5.7 shows that despite the small average string length, large strings containing up to 12 monomers occur with significant probability.

neighbor distance (≈ 1) minus δ, which is roughly 0.45.

A string length larger than one implies that mobile monomers tend to replace each other. This trend is present at all temperatures, but becomes more pronounced on cooling. With respect to the MSD of the bulk, the time $t^{\rm max}_{\rm str}$ when $L(t)$ is

maximum corresponds to the time when there is a crossover from a caging regime to a subdiffusive regime, similar to what is observed for other dynamical quantities as discussed in earlier sections. However, the actual times are slightly different. For example, $t_{\rm str}^{\rm max}$ occurs at a slightly later time than $t_{\rm clu}^{\rm max}$.

In principle, one would expect the clusters of mobile monomers and the strings they are comprised of to be maximum at roughly the same time. In the present system, finite size effects may occur at low T, where the clusters become larger than the simulation cell. This could give an estimate of $t_{\rm clu}^{\rm max}$ different from the asymptotic value. It is also possible that in the present systems, $t_{\rm str}^{\rm max}$ is larger than $t_{\rm clu}^{\rm max}$ because there may be a time delay for mobile particles to rearrange themselves in a special one-dimensional path. In any case, despite this small difference in the peak times, both $t_{\rm str}^{\rm max}$ and $t_{\rm clu}^{\rm max}$ occur in the late-β/early-α relaxation regime when particles begin to break out of their cage.

To understand the role of chain connectivity on strings, we compare $L_{\rm seg}(t_{\rm str}^{\rm max})$ and $L(t_{\rm str}^{\rm max})$ in Fig. 5.5 by taking the ratio of the two quantities. A ratio near one (unless both the numerator and denominator are equal to one) implies that the strings are the result of consecutively bonded pairs, i.e., that monomers moving in strings actually move along the backbone of the chain to which they belong, indicating an important contribution from connectivity to the formation of strings. On the other hand, a ratio close to zero implies that string-like motion occurs among non-bonded monomers and that chain connectivity is insignificant. As indicated in the figure, $L_{\rm seg}/L$ decreases as T approaches $T_{\rm MCT}$, suggesting that chain connectivity becomes less important for string-like motion at low T.

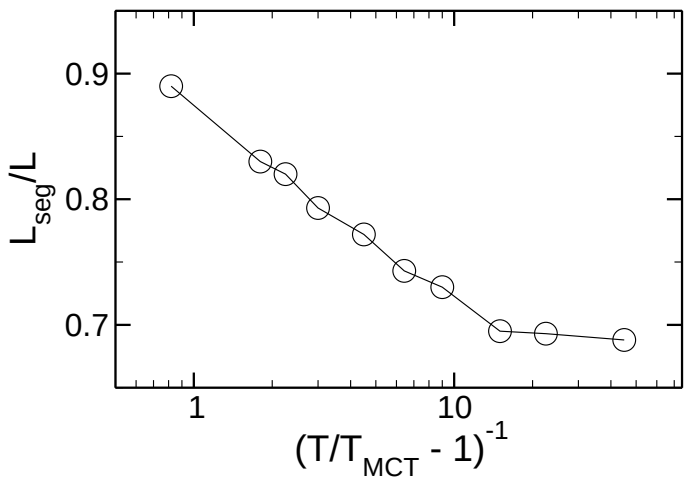

Figure 5.5: Temperature dependence of the ratio of $L_{\rm seg}(t_{\rm str}^{\rm max})$ and $L(t_{\rm str}^{\rm max})$. $t_{\rm str}^{\rm max}$ is the peak time of $L_{\rm seg}$ and L at different temperatures. $T_{\rm MCT} = 0.45$.

The previous analysis was performed with $\delta = 0.55$. When introducing the criterion for defining strings we argued that the precise choice of δ is not crucial, as long as its value is sufficiently small. To illustrate this point, Fig. 5.6 shows the temperature dependence of the maximum average string length, $L(t_{\rm str}^{\rm max})$, for various δ. We find that the strings become longer if δ increases. This is expected, since more particles satisfy the condition given in Eq. 5.1. However, the qualitative features are independent of δ. To support this point further we invoke an analogy, first proposed in Ref. [75], between the strings and equilibrium polymers [169, 171, 172] (see also [173] and references therein).

Equilibrium polymers are systems in which the bonds between monomers are not permanent. They can constantly break and recombine at various points along the backbone or ends of a chain. In chemical equilibrium, a melt of these self-assembling polymers is characterized by an exponential distribution of chain lengths,

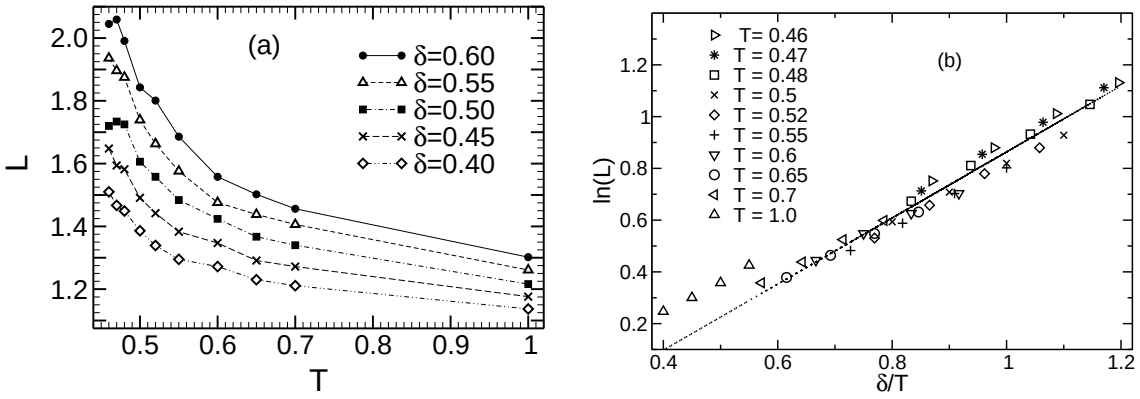

Figure 5.6: Left panel: Average string length L versus T for various δ [see Eqs. (5.1)]. The string length is calculated at $t_{\rm str}^{\rm max}$, where it is maximum. Right panel: Rescaling of L as suggested by the analogy with equilibrium polymers (see text for details). A satisfactory collapse of the data for all T and δ is obtained except for $T = 1$, the temperature below which supercooled liquid dynamics occurs in this model. At this temperature, strings larger than 1 occur very seldomly (see Fig. 5.7). The dashed straight line is a fit through the data for $T \leq 0.7$, yielding $\ln L = -0.23 + 0.73\ \delta/T$.

$P(l) \sim \exp(-l/L)$ (if l is large), and by a mean chain length that increases exponentially with the energy E gained by bond formation, $L \propto \exp(E/T)$.

In our context, the mobile monomers also self-assemble into chains, driven by the sluggish dynamics of the cold melt. The dynamically created bonds can break and recombine at any instant. They are more likely to form, and thus "stronger", the larger the choice of δ. This suggests a correspondence between δ and E, the simplest assumption being $E \propto \delta$. Figure 5.6 shows that this assumption is not unrealistic. Despite the disparity between the theoretical premise of long chains

and the relative shortness of our strings, a reasonable superposition of string lengths, found for various δ and T, is obtained. This implies that any of the values for δ presented could have been chosen for the present analysis.

From the analogy with equilibrium polymers one expects that the strings have an exponential distribution. Figure 5.7 shows the distribution of the string length l found at $t_{\rm str}^{\rm max}$. At the highest temperature, $T = 1$, $P(l)$ is an exponential and decreases rapidly with increasing l. The most frequent string lengths are $l = 1, 2$. Their probability remains essentially unchanged on cooling, whereas longer strings occur much more frequently for $T < 1$. The tail of the distribution appears to remain exponential, further supporting the possible interpretation of strings in the same context as equilibrium polymers. Similar observations of exponential distributions were also made in the simulations of the binary LJ mixture [75], and also the Dzugutov liquid as will be shown in the next chapter.

The weight averaged string length $L_{\rm w}(t)$ is presented in Sec. 5.3. We find that $L_{\rm w}(t)$ and the number average $L(t)$ behave qualitatively in the same way, but are quantitatively different. This difference characterizes the variance of the string length distribution because

$$\frac{\langle (l - \langle l \rangle)^2 \rangle}{\langle l \rangle^2} = \frac{L_{\rm w}}{L} - 1 \; , \tag{5.4}$$

where $\langle \cdot \rangle$ denotes the number average [see Eq. (5.3)]. The ratio $L_{\rm w}/L$, shown in Fig. 5.8, is referred to as the "polydispersity index" in the context of polymerization. We find that the strings are most polydisperse for times of the late-β/early-α process and that this maximum of polydispersity increases with decreasing T.

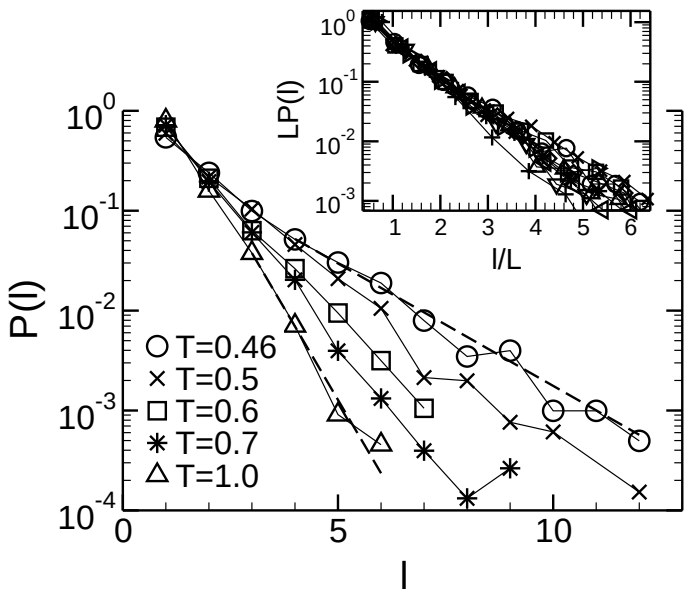

Figure 5.7: Semi-log plot of the probability distribution $P(l)$ of the string length l for various T. $P(l)$ is calculated at $t = t_{\rm str}^{\rm max}$ where L is maximum. All data sets exhibit (roughly) exponential behavior. Inset: $P(l)$ rescaled by the mean value L versus l/L. In addition to the temperatures $T = 0.46$, 0.47, 0.48, 0.50, 0.52, 0.55 for $\delta = 0.55$ the graph also includes $T = 0.46$ and 0.55 for both $\delta = 0.4$ and $\delta = 0.6$. The scaling deteriorates if data at higher T is included.

In carrying out this analysis, the question arises as to whether a string of length one should be included or not in the calculation of mean string lengths. One may argue that $l = 1$ should be excluded from the calculation since by definition string-like motion requires one mobile monomer to replace another, and hence is not really defined for $l < 2$. In other words, a "string" of length one indicates a mobile monomer *not* moving in a string based on the criterion used. To address this issue, we calculate the time evolution of $L(t)$ by excluding strings of size one (see

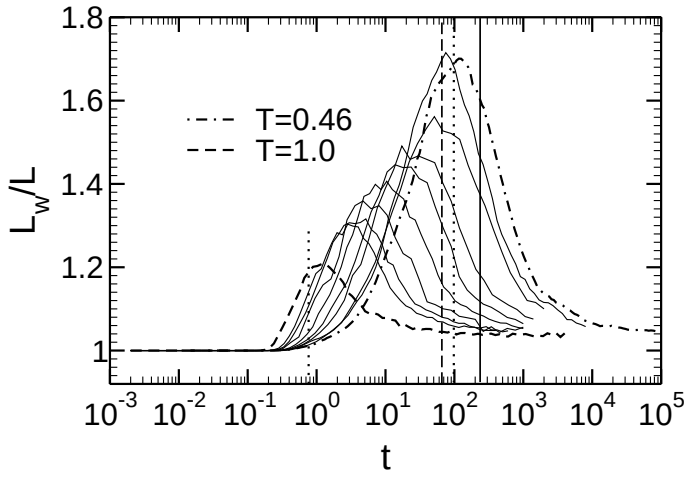

Figure 5.8: The "polydispersity index" given by the ratio of the weight averaged string length L_w and the number averaged string length L plotted as a function of time for all temperatures. The temperatures shown are (from left to right): $T = 1$, 0.7, 0.65, 0.6, 0.55, 0.52, 0.5, 0.48, 0.47, 0.46. The vertical dotted lines indicate $t^*_{\alpha_2} = 0.766$ and $t^*_{\alpha_2} = 100.894$ for $T = 1$ and $T = 0.46$, respectively. For $T = 0.46$ the vertical dashed line shows $t^{\max}_{\text{clu}}$ ($= 65.85$), the vertical solid $t^{\max}_{\text{str}}$ ($= 236.26$).

section 5.3), for comparison. We find no qualitative difference between the two cases, but in the absence of $l = 1$ the mean string length is accordingly larger. Therefore, to maintain the analogy of strings to equilibrium polymerization, in which polymers of length one are included, we restricted our discussion to the mean values that include $l = 1$ and are calculated using the number average.

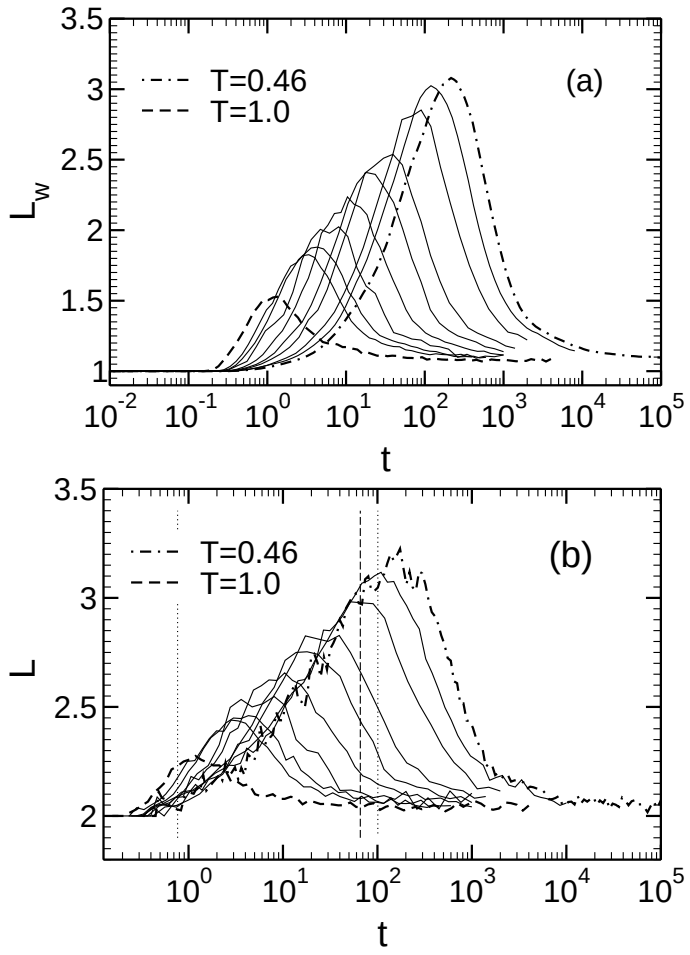

Figure 5.9: (a) Weight averaged string length L_w of all mobile monomers versus t. (b) Number averaged string length L of all mobile monomers versus t calculated by excluding strings of size 1. In both figures, the temperatures shown are (from left to right): $T = 1$, 0.7, 0.65, 0.6, 0.55, 0.52, 0.5, 0.48, 0.47, 0.46. The dotted vertical lines in both panels indicate the time when α_2 is maximum for $T = 1$ ($t^*_{\alpha_2} = 0.766$) and $T = 0.46$ ($t^*_{\alpha_2} = 100.894$). The dashed vertical line indicates the time $t^{\max}_{\text{clu}}$ when the weight averaged cluster size, calculated in Ref. [134], is maximum for $T = 0.46$ ($t^{\max}_{\text{clu}} = 65.85$).

5.3 Weight and number averaged string lengths

To compare the weight averaged string length $L_w(t)$ with the number averaged string length $L(t)$, discussed in Fig. 5.4, we show here the time evolution of L_w, which is calculated using Eq. (5.2), for different T. It is apparent from the figure [Fig. 5.9(a)] that L_w is qualitatively the same as L.

To address the issue of including or excluding strings of size $l = 1$, in Fig. 5.9(b) we show the number averaged string length $L(t)$ for several T calculated by excluding $l = 1$. This calculation yields a similar result to that obtained by including strings with size $l = 1$, but the number average and the weight average (not shown) take larger values when $l = 1$ is excluded [compare Figs. 5.4 and 5.9(b)].

Chapter 6

The formation of clusters and the development of string-like motion in the Dzugutov liquid

The increasing complexity in the dynamics of liquids cooled towards their glass transition has been demonstrated in the last chapters through the study of the average properties of clusters and strings of mobile particles. The purpose of this chapter is to investigate the behavior of these dynamically correlated motions beyond what is known in terms of the average properties. For this reason we investigate the mechanisms involved in the formation of clusters and strings at a finer level. The results presented in this chapter have been published in Refs. [174, 175].

6.1 Relevance of string-like motion

It has been shown in the last chapters that mobile particles aggregate together into clusters that grow in size with decreasing temperature. This observation has been supported through a number of computational [76, 167], and also experimental [21] studies. Within any cluster of mobile particles, smaller subsets move together in a correlated fashion where several particles replace each other along one-dimensional, "string-like" paths [75]. These dynamically correlated structures have been observed in the simulations of a LJ binary mixture [75] and, in the last chapter, we have shown that they are also present in the polymer melt. Similar dynamical objects are also observed in the simulation of 2D binary mixture of soft discs [89] and in the simulation of a nonrandomly frustrated model of spin glasses that are perceived as a model for glass formers [92].

Experimentally, a number of studies find direct and indirect evidence for dynamically correlated groups of particles. Using an approach that corrects multiple scattering noise in inelastic coherent neutron scattering experiments, Russina, Mezei and collaborators [176] explored the microscopic dynamics of a supercooled liquid at small wave numbers and found evidence for collective fast atomic motion on the scale of the intermediate range order found in the static structure. In view of the spatially extended character of the collective excitations, they argued that they may be evidence for strings. A direct experimental observation of string-like motion was made by Marcus et al. [137] and Cui et al. [177] in concentrated quasi-two dimensional colloidal liquids. Using 3D confocal microscopy, Weeks et al. [178] showed

uni-directional motion of neighboring particles in colloidal systems and attributed these motions in part to string-like particle rearrangements.

The concept of string-like motion plays an important role in the development of new emerging theories, as well as in more traditional theories of the glass transition. For example, Garrahan, Chandler and co-workers [60, 61] have recently provided a "non-topographic" description of dynamics in supercooled liquids. Central to the theory is the notion of dynamic facilitation, originally introduced by Fredrickson and Andersen [179]. Dynamic facilitation corresponds to the idea that when particles in a microscopic region of space are mobile, they influence the dynamics of particles in neighboring regions, enabling them to become mobile, thereby allowing mobility to propagate through the system [61]. It has been argued that the observation that highly mobile particles in a supercooled liquid move along correlated strings is a confirmation of this central idea [61]. Our own analysis [175] that will be described below indeed suggests that clusters of highly mobile particles are formed as a result of mobility propagation initiated within a nanoscopic local structure and facilitated through quasi-one dimensional string-like rearrangements [175].

String-like rearrangement of particles has also been recognized in the potential energy landscape or "topographic" view point of dynamics in supercooled liquids [2, 38, 45, 48]. In this picture, the structural relaxation of particles at sufficiently low temperatures is attributed to transitions between local energy minima, or inherent structures (IS), of the multidimensional potential energy hypersurface. In fragile liquids, similar to the scenario described in terms of multifunnel structures [180], basins in configuration space are organized into "metabasins" [38, 48].

Heuer and co-workers [181] recently showed that, upon deep supercooling, a liquid becomes trapped in a single metabasin for an extended period of time, making frequent hops within the metabasin, and infrequent excursions from one metabasin to another. Schrøder et al. [130] showed that transitions between inherent structures involve string-like motion. Further, Denny et al. [58] observed that transitions between metabasins involve cooperative rearrangement of particles. The relevance of string-like motion during inherent structure and metabasin transitions was recently investigated in detail by Vogel et al. [182]. It was demonstrated that although string-like motion facilitates both types of transitions, it is of particular importance for metabasin transitions. All these studies suggest that the concept of string-like motion is essential for understanding how particle rearrangements lead to exploration of configuration space.

Perhaps the most well-known theory that connects dynamical properties of supercooled liquids to their thermodynamic properties is the Adam-Gibbs theory [12]. The main prediction of the theory (see Eq. 2.5) has been tested in simulations and appears to be valid across a wide spectrum of liquids [96, 164, 183, 184]. Despite the validity of the theory, the cooperatively rearranging regions (CRR) have not been definitively identified. It is reasonable to propose [67] that the CRR are associated with the mobile regions of the supercooled liquid. Indeed, a recent study [167] demonstrated a connection between the Adam-Gibbs theory and spatially heterogeneous dynamics in simulations of water. In particular, they showed that the average size of clusters of mobile particles, defined as in Ref. [76], is related to the size of the CRR [167]. Since we know that clusters are also comprised of strings [75], a funda-

mental connection between strings and the CRR of Adam and Gibbs is expected. The validity of this hypothesis will be examined in the following chapter.

The above examples clearly demonstrate the relevance of string-like rearrangements both in new and well-established theories of the glass transition. Nevertheless, little is known about the microscopic details of this dynamical pattern. For example, it is still elusive as to how the short time rattling motion of particles within their temporary cages evolves into structured correlated motion that is manifested as string-like motion along a one-dimensional path. In particular, it is important for the further development of these theories to understand the mechanisms involved in the formation of these local excitations, e.g., how these motions occur, how large strings develop, and to what extent the motion is coherent and cooperative. To investigate these questions, we performed a detailed microscopic analysis of string-like motion in the Dzugutov liquid at temperatures above T_{MCT} [174]. Our analysis answers several of the above questions and provides insight into the most probable mechanism for the formation of strings.

6.2 Average properties of cooperativity

Next we briefly examine the average properties of the quantities that characterize cooperativity in order to test their universality. We first examine the non-Gaussian parameter α_2 calculated for the Dzugutov liquid. As described in the last chapter, α_2 measures the degree to which a system deviates from a Gaussian or random behavior, and hence it quantifies the degree of cooperativity in a system. In Fig. 6.1 we show

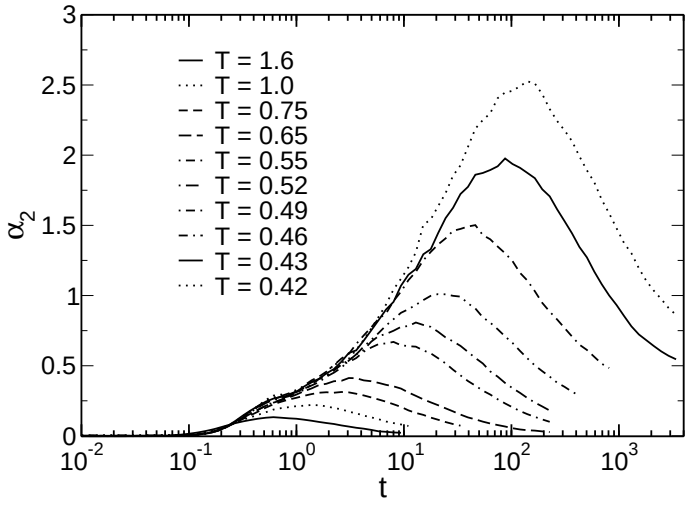

Figure 6.1: Non-Gaussian parameter α_2 as a function of time for different T.

the plot of $\alpha_2(t)$ for various T. As expected, $\alpha_2(t)$ is zero at short times. It then becomes positive with progress in time, and shows a maximum at intermediate time before decreasing towards zero at the late time. The time at which α_2 is maximum corresponds to the late-β/early-α relaxation regime of the MCT. In terms of the MSD of the Dzugutov liquid, this corresponds to the time when the particles break out of their cages and become diffusive. This is similar to the observation in the polymer melt, except that in the polymer melt the crossover regime is between the plateau and sub-diffusive regime. The temperature dependence of $\alpha_2(t)$ for the Dzugutov liquid is also similar to what is observed in other glass-forming liquids [20, 76, 126], where α_2 is found to increase on approaching to $T_{\rm MCT}$.

Note that, at any time interval t, the fact that we are able to identify particles that have moved a much farther distance than expected from a Gaussian distribution, signifies that the system is dynamically heterogeneous. This is measured in terms of the non-Gaussian parameter, and the plot for α_2 shows that dynamical het-

erogeneity becomes more pronounced at the late-β/early-α relaxation regime, and increases with a decrease in T. To study the spatial extent of dynamic heterogeneity in the physical regions of high mobility, we measure the size of clusters formed by the mobile particles. As described in Chapter 4, a fraction ϕ for defining mobility can be selected by integrating the van Hove correlation function $G_s(r,t)$ at the time $t^*_{\alpha_2}$ when α_2 is maximum. For the Dzugutov liquid, this calculation yields $\phi = 6.4\%$ for $T = 0.42$. Nevertheless, since the choice for ϕ does not signify any qualitative difference in the behavior of clusters or strings, we use a fraction $\phi = 5\%$ for the present study, as is traditionally done for several other systems [21, 76, 167].

Next we show the mean cluster size formed by mobile particles for the Dzugutov liquid. At any time interval t, we identify highly mobile particles by monitoring the displacements of all particles within t, and then selecting 5% of the particles with the largest displacements, as in Ref. [21, 74, 75, 76, 167]. Following previous works [76, 134], we define a cluster as a group of highly mobile particles that are within the first neighbor shell of each other, where the first neighbor shell is defined by the distance of the first minimum of $g(r)$, cf. Fig. 3.2(b). In Fig. 6.2 we show examples of typical clusters found in the Dzugutov liquid at $T = 0.42$, where particles in distinct clusters are colored differently. Similar to other systems [21, 76, 134, 167], clusters of different sizes and shapes are observed for the Dzugutov liquid. Thus, a statistical analysis is necessary to determine the transient nature and T dependence of the clusters. We calculate $P(n(t))$, the probability of finding a cluster of size n at a time interval t, and from this, we compute the weight-averaged mean cluster size $S(t)$ using the relation defined in Eq. 4.4 [156].

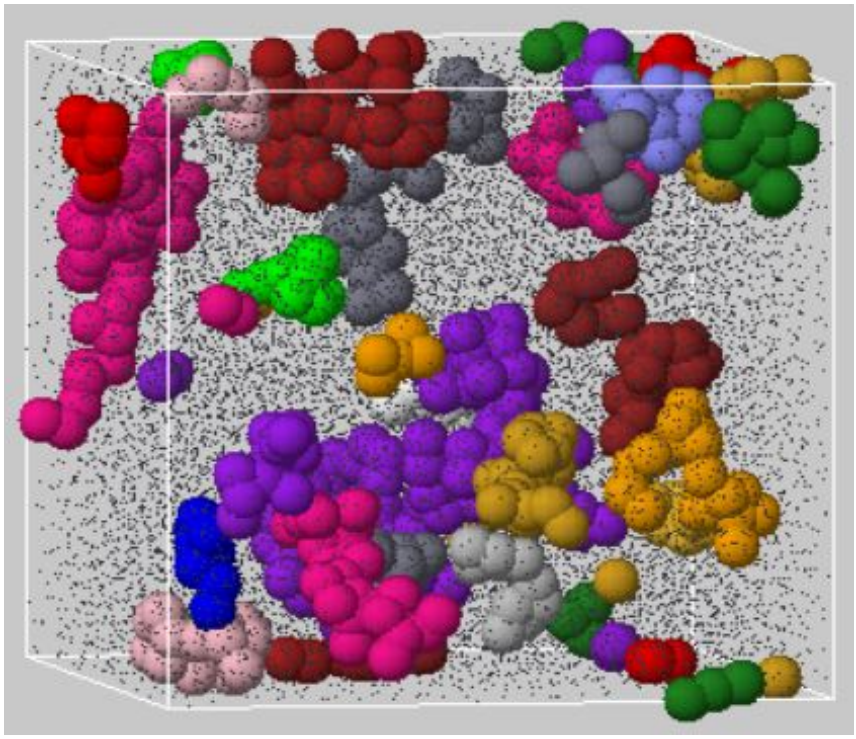

Figure 6.2: Example of typical clusters formed by the 5% most mobile particles that are found at $T = 0.42$ at the time $t = 102.4$. Particles belonging to the same cluster are colored the same. Note that in the simulation all particles have the same size, but for the purpose of visualization all particles not in the subset studied are shown as dots.

In Fig. 6.4, we show $S(t)$ for various T. It can be seen clearly that the mean cluster size increases rapidly upon cooling towards T_{MCT}. Moreover, $S(t)$ shows a peak at an intermediate time $t_{\rm clu}^{\rm max}$ that coincides with the time of the MSD crossover from the caging regime to the diffusive regime, cf. Fig. 3.4(b). We extract $t_{\rm clu}^{\rm max}$ for each T, and fit the data by a power law $t_{\rm clu}^{\rm max} \propto (T - T_{MCT})^{-x}$ as was done for the polymer melt. It is evident from Fig. 6.3 that a power-law, with an exponent $x = 1.39 \pm 0.17$, yields a reasonable fit for the Dzugutov liquid. Within the estimated

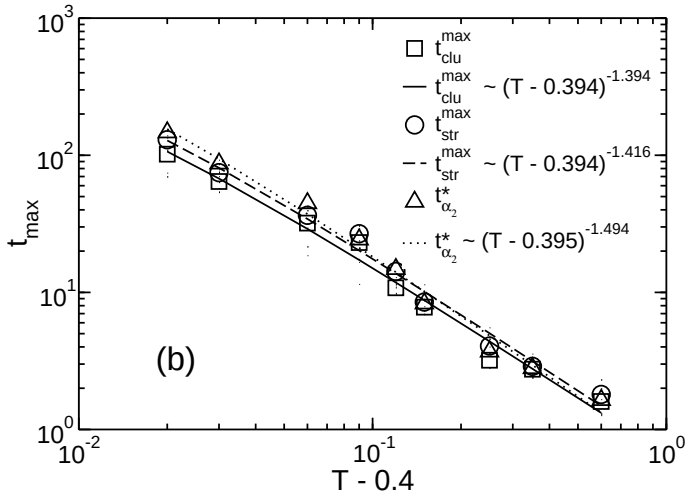

Figure 6.3: Temperature dependence of the peak times $t_{\rm clu}^{\rm max}$ and $t_{\rm str}^{\rm max}$ characterizing the times when the mean cluster size and the mean string size, respectively, are maximum. For completeness, the time $t^*_{\alpha_2}$ when the non-Gaussian parameter (not shown) of this system is maximum is also plotted. In the figure, the lines show results from non-linear curve fitting of each data to a power law $\sim (T - T_{\rm MCT})^{-\gamma}$, where $T_{\rm MCT}$ and γ are used as free fit parameters. In all cases $T_{\rm MCT}$ is close to $T = 0.4$.

numerical error, this is consistent with $x = 1.47 \pm 0.16$ observed in simulations of a polymer melt [134]. Thus, the transient nature of $S(t)$ is similar to that found in previous studies including experimental work on colloids [21]. In all studied cases, the peak of $S(t)$ lies in the late-β/early-α relaxation regime of the MCT and, hence, the dynamical process which manifests itself in the formation of the clusters precedes the long-time structural relaxation.

Another interesting similarity between different systems becomes obvious when inspecting the probability distribution of the cluster size. In accordance with pre-

vious works [21, 76, 134], Fig. 6.4 shows that $P(n)$ at the characteristic time $t_{\rm clu}^{\rm max}$ is well described by a power-law $P(n) \propto n^{-\tau}$ (multiplied by an exponential cutoff for $T > T_{\rm MCT}$ [156]). The exponent $\tau = 1.69 \pm 0.16$ for the Dzugutov liquid at $T = 0.42$ is similar to that found for a polymer melt ($\tau = 1.62 \pm 0.12$) close to T_{MCT} [134], but it is different from the values obtained for a binary LJ mixture ($\tau = 1.9$) [76] and a colloidal system ($\tau = 2.2 \pm 0.2$) [21]. Though the value of the exponent τ may be non-universal, it is noteworthy that, for all systems studied so far, the distribution $P(n)$ at the characteristic time $t_{\rm clu}^{\rm max}$ exhibits a power-law behavior when $T \to T_{MCT}$.

As will be shown in more detail in the following section, the clusters, especially the larger ones, contain several strings. An example for the decomposition of a cluster into several strings is shown in Fig. 6.5 where the different colors indicate distinct strings. The strings are found by comparing snapshots of the particle configurations at two different times, and then identifying those mobile particles that have replaced each other within a radius δ. Again, since the choice of δ does not affect the qualitative features of the strings, provided δ is chosen smaller than the hard-core radius σ, we simply used $\delta = 0.6$ following Donati et al. [75]. In our analysis, however, we took the issue discussed for the polymer model on whether strings of size one should be considered as strings or not a step further, and decided to focus our attention on the non-trivial strings, i.e., strings with size $l \geq 3$. This will simply give us a more stringent criteria on which to base our prediction on the nature of string-like motion, but it does not alter the average property of the strings.

The average property of the strings is studied for the Dzugutov liquid. In analogy to the mean cluster size, the weight-averaged mean string length, L_w, can

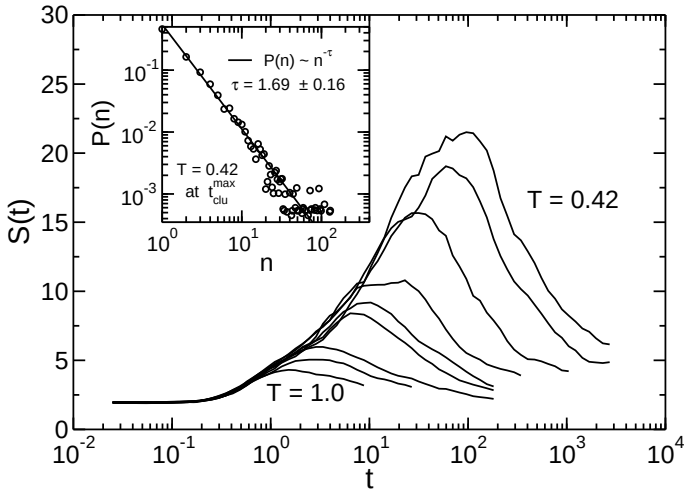

Figure 6.4: The mean cluster size S as a function of time t for temperatures, from left to right, $T = 1.0, 0.75, 0.65, 0.55, 0.52, 0.49, 0.46, 0.43, 0.42$. Inset: probability distribution, $P(n)$, at the time $t_{\rm clu}^{\rm max}$ for $T = 0.42$. The solid line is a power law fit $P(n) \sim n^{-\tau}$. τ is found to be $\tau = 1.69 \pm 0.16$. Note that here the mean cluster sizes are plotted without normalizing them with the random values. The estimates for the random values of the mean cluster sizes are in the range $1.91 - 1.97$ for all T. Therefore, for each T, the normalized mean cluster size will be nearly half the size shown here.

be calculated using Eq. 5.2, described in the last chapter. In Fig. 6.6, $L_w(t)$ is displayed for different T. The time and the temperature dependence of the mean string length are essentially similar to that of the mean cluster size. In particular, $L_w(t)$ peaks at a time $t_{\rm str}^{\rm max}$ which is within the numerical error of $t_{\rm clu}^{\rm max}$, and the maximum value of $L_w(t)$ increases with decreasing T. Thus, long strings are mainly formed in the late-β/early-α relaxation regime where the mean length increases upon cooling. This time regime also corresponds to the time $t_{\alpha_2}^*$ when the non-Gaussian

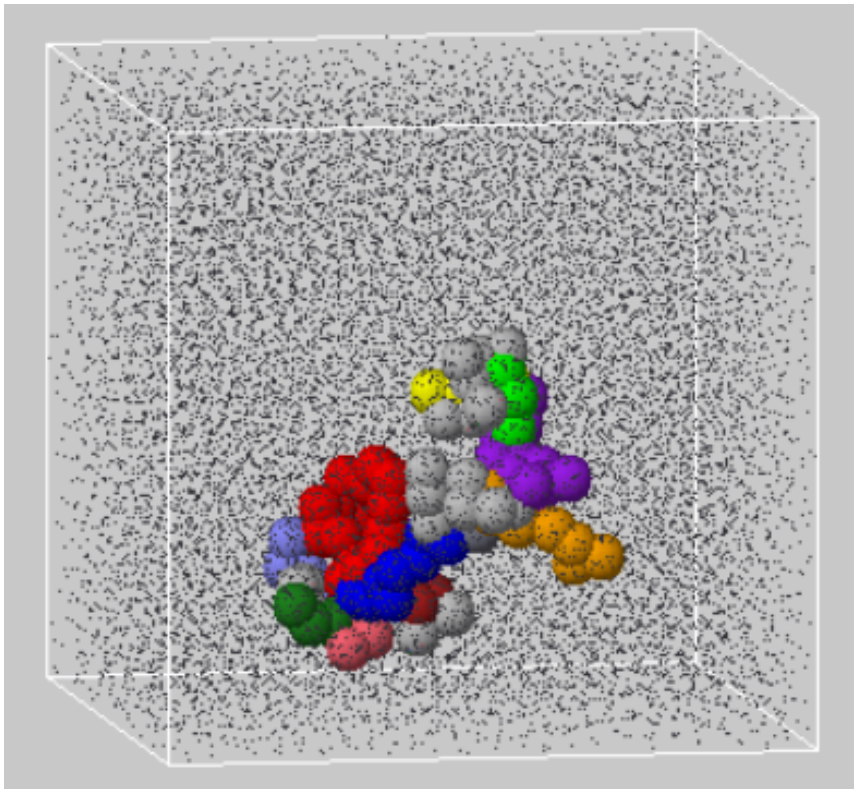

Figure 6.5: Large cluster identified at $T = 0.42$ at a time $t = 102.4$. For the purpose of visualization all particles in the cluster are represented by a sphere of radius 1.0, while all other particles in the system are represented by a sphere of radius 0.1. Particles moving in the same string are given the same color. Those particles in the cluster that are not involved in string-like motion are colored gray.

parameter $\alpha_2(t)$ is maximum. An important difference between strings and clusters exists for the respective size distributions. At the respective peak times for $T = 0.42$, $P(n)$ is well described by a power-law (cf. Fig. 6.4) while $P(l)$ shows an exponential decay. Such behavior was also found in simulations for a binary LJ mixture [75] and the polymer melt we studied [126].

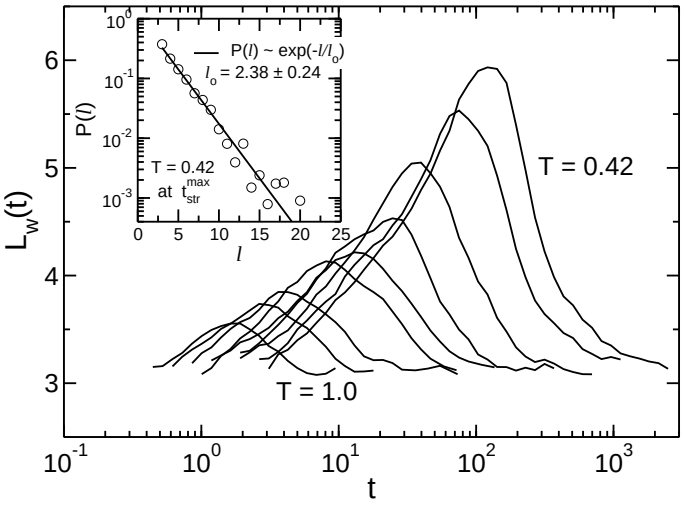

Figure 6.6: The weight-averaged mean string size L_w as a function of time t for temperatures, from left to right, $T = 1.0, 0.75, 0.65, 0.55, 0.52, 0.49, 0.46, 0.43, 0.42$. Inset: probability distribution, $P(l)$, at the time $t_{\rm str}^{\rm max}$ for $T = 0.42$. The solid line represents an exponential fit, $P(l) \sim \exp(-l/l_o)$, where l_o is found to be $l_o = 2.38 \pm 0.24$

6.3 Formation of clusters and mobility propagation

In this section we examine the development of clusters. In Fig. 6.2, we showed examples of several clusters found at $t_{\rm clu}^{\rm max}$. But, it is not immidiately apparent how each of these clusters are formed. Here we seek to understand the formation of these clusters by monitoring individual particle trajectories for the clusters identified at $t_{\rm clu}^{\rm max}$. As an example, we single out one of the clusters and monitor the trajectories of the constituting particles. First, we demonstrate that the enhanced mobility of the particles in this cluster is transient in nature. To quantify the time scale on

which the information about the initial mobility is lost, we monitor the MSD of the selected subset of particles during successive time intervals of $t_{\rm clu}^{\rm max}$, i.e., we compute the quantities

$$\left\langle r_n^2(t) \right\rangle \;=\; \left\langle \left[\mathbf{r}_j\left(nt_{\rm clu}^{\rm max}+t\right) - \mathbf{r}_j\left(nt_{\rm clu}^{\rm max}\right)\right]^2 \right\rangle , \tag{6.1}$$

where $\mathbf{r}_j(t)$ denotes the position of the j-th particle of the cluster and the brackets represent the average over the selected subensemble. The results depicted in Fig. 6.7 indicate that a significantly higher mobility than the average exists during the time interval when the particles are identified as mobile ($n = 0$). On the other hand, the curves for $1 \leq n < 5$ lie only slightly above the one representing the ensemble average, and for even larger n, there is no evidence for systematic deviation from the average. Comparing the time scale of this re-equilibration with the α-relaxation time $\tau_\alpha \approx 30 t_{\rm clu}^{\rm max}$, it becomes clear that the clusters of highly mobile particles lose their mobilities on time scales much less than τ_α, and therefore do not "dissolve" due to the structural relaxation of the bulk. Instead, the dynamical process which leads to the formation and decay of the clusters takes place on a time scale much shorter than the structural relaxation. This conclusion is consistent with the finding that the peak time of the mean cluster size $t_{\rm clu}^{\rm max}$ lies at intermediate times less than τ_α, in the late-β/early-α relaxation regime of the MCT.

To gain further insights into the formation of the clusters, we show snapshots of the selected cluster for different times $t < t_{\rm clu}^{\rm max}$ in Fig. 6.8. In this figure, the mobile

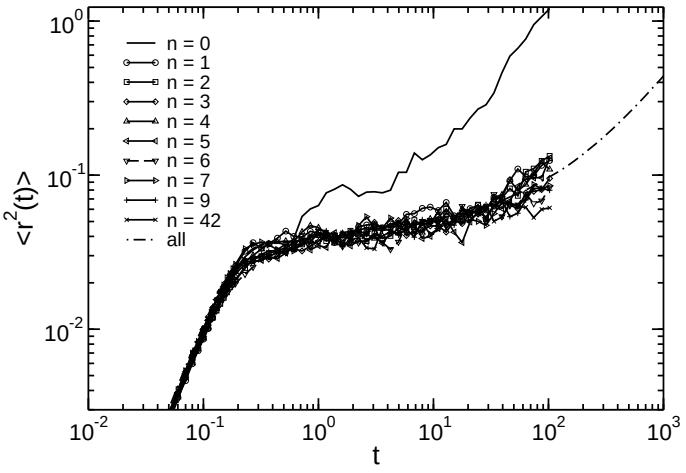

Figure 6.7: Mean square displacement $\langle r_n^2(t) \rangle$ for different time windows between $nt_{\max}$ and $(n+1)t_{\rm clu}^{\max}$, where $(n = 0, 1, 2, 3, 4, 5, 6, 7, 9, 42)$, averaged over all particles in one of the largest clusters (containing 94 particles) identified at $t_{\rm clu}^{\max}$. The dashed-dotted line shows the mean square displacement $\langle r^2(t) \rangle$ averaged over all particles in the system.

particles at each time are colored red. Inspection of the snapshots for early times indicates a few mobile particles that are randomly distributed within the cluster. As time progresses, the mobile particles become organized in small groups which grow with time. Thus, one may speculate that the particles in these groups assist each other to become mobile, e.g., by moving cooperatively, and in this way mobility propagates through out the cluster. This will be investigated further below.

We now focus on the mobile particles identified during the formation of the cluster. In particular, we demonstrate that many of these particles are indeed involved in string-like motion, thereby establishing a relation between strings and clusters. In Fig. 6.9, we display the configurations of the selected cluster at two

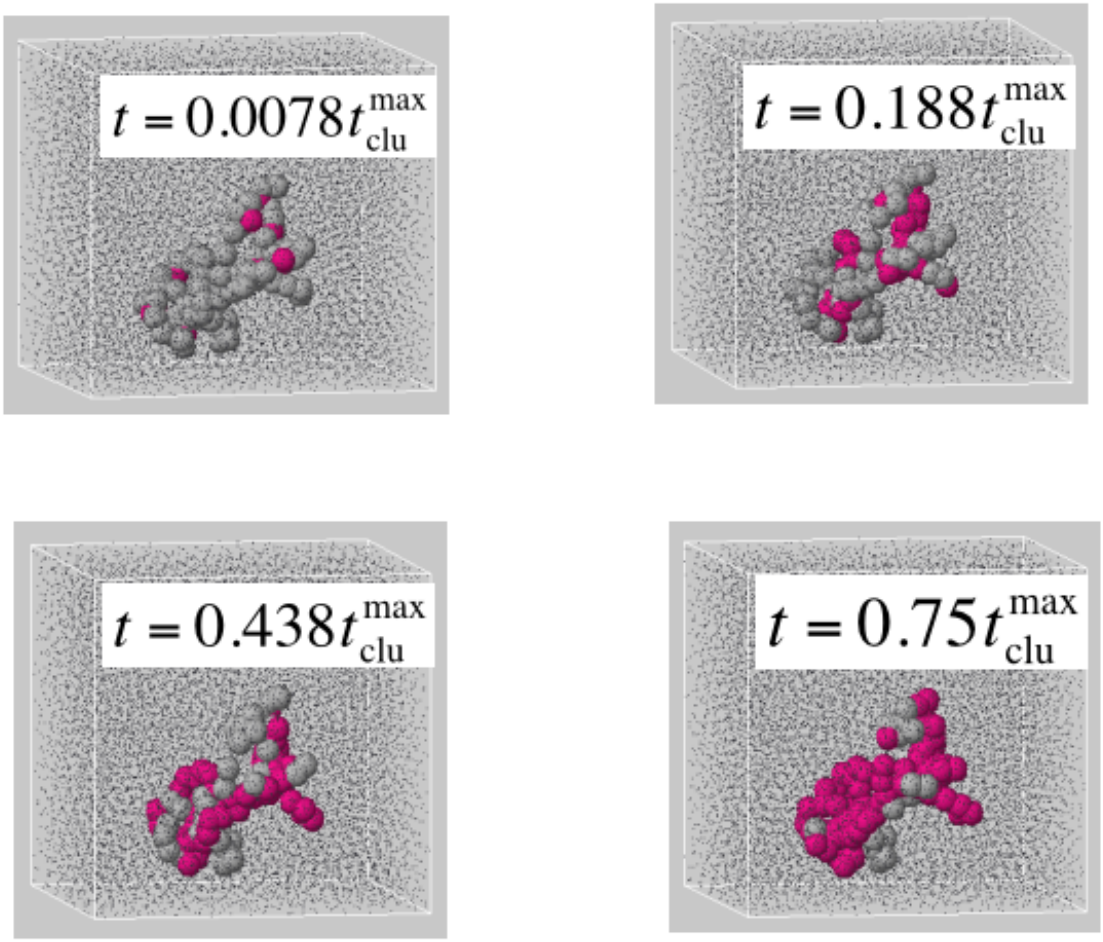

Figure 6.8: Snapshots of one of the largest clusters identified for times $t = 0.0078 t_{\mathrm{clu}}^{\mathrm{max}}$, $0.188 t_{\mathrm{clu}}^{\mathrm{max}}$, $0.438 t_{\mathrm{clu}}^{\mathrm{max}}$, and $0.75 t_{\mathrm{clu}}^{\mathrm{max}}$, as indicated on the figures. Particles belonging to the studied cluster are shown as spheres of radius 1.0. The mobility of all particles in this cluster is monitored for times $t < t_{\mathrm{clu}}^{\mathrm{max}}$, and at any t the most mobile particles are colored red. Note that all 17576 particles in the simulation box are identical. Here only particles within the cluster are magnified for visualization, and all other particles are shown at reduced size.

different times $t < t_{\mathrm{clu}}^{\mathrm{max}}$, marking for each time two different subsets of particles. On the left hand side, all mobile particles at the respective times are colored, while, on the right hand side, only the mobile particles involved in string-like motion are marked where different colors indicate distinct strings. Note that the panels on the left hand side have already been shown in Fig. 6.8. Comparing the corresponding snapshots it can be immediately realized that many of the mobile particles are actually replacing each other and moving in strings. Moreover, it is evident that the lengths of the strings increase with time. What we learn from this analysis is that the selected cluster is formed as a result of mobility propagation starting from distributed points within the cluster. The propagation of this mobility is facilitated through the development of quasi-one dimensional string-like dynamical processes where groups of particles within the cluster move along a single path. The number of particles involved in this string-like rearrangement increases with time.

These findings suggest that string-like motion is an important channel for relaxation within the highly mobile domains of a dynamically heterogeneous system. To corroborate this fact, we compute the fraction $f(t)$ of mobile particles that are involved in non-trivial string-like motion, which means strings consisting of three or more particles. Fig. 6.10 shows that the fraction grows significantly when T is decreased. For example, at $t_{\mathrm{clu}}^{\mathrm{max}}$, roughly 70% of the mobile particles are involved in string-like motion for $T = 0.42$. Further, $f(t)$ has a maximum at a temperature-dependent time $t_f \approx t_{\mathrm{clu}}^{\mathrm{max}}$, cf. Fig. 6.4. These results indicate that string-like motion is an important channel for the relaxation of mobile particles and becomes increasingly significant with decreasing T.

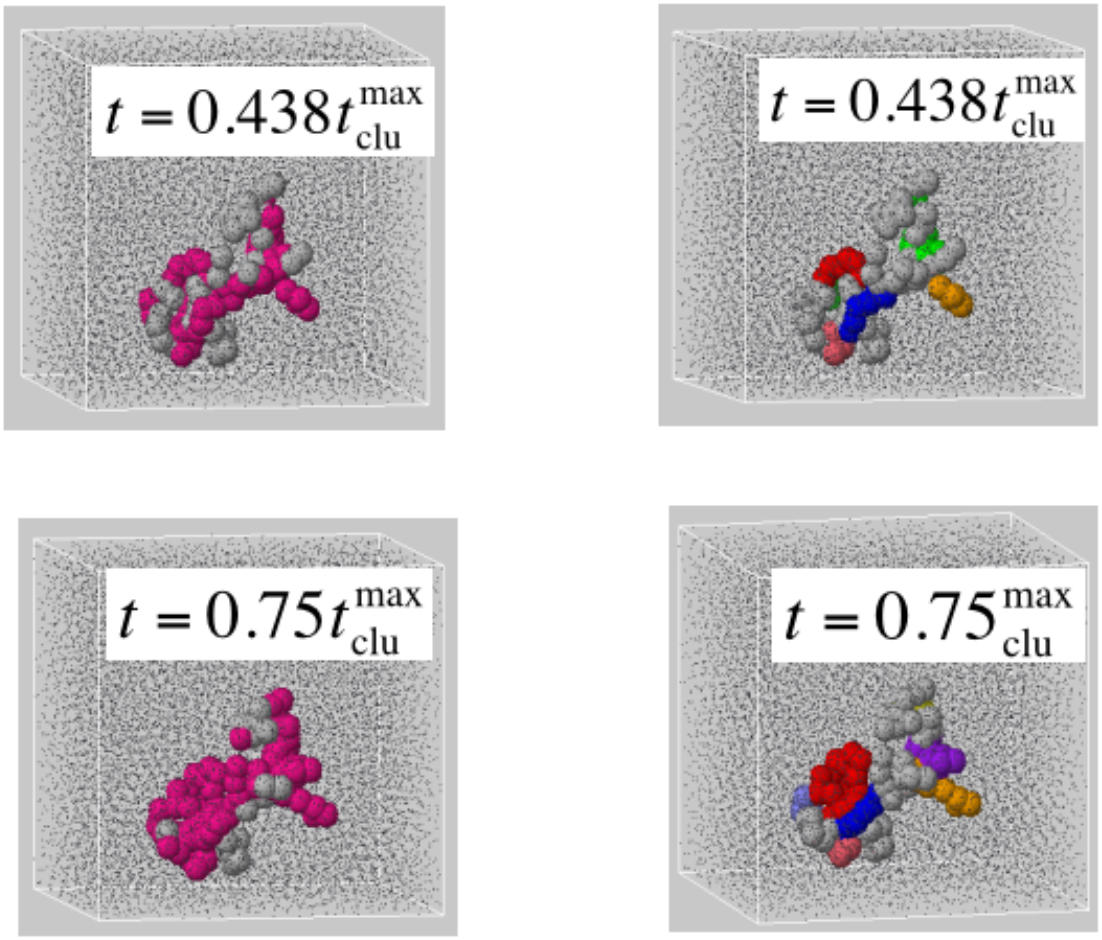

Figure 6.9: Snapshots of configurations at $t = 0.438 t_{\mathrm{clu}}^{\mathrm{max}}$ and $0.75 t_{\mathrm{clu}}^{\mathrm{max}}$. Particles belonging to the studied cluster are shown as spheres. On the left hand side all mobile particles at the respective time are colored. On the right side, only mobile particles involved in string-like motion are marked, where distinct strings are colored differently.

6.4 Mechanism of string-like motion

Thus far, we have explored the average properties of strings, and have shown that string-like motion is an important channel for relaxation in the domains of mobile

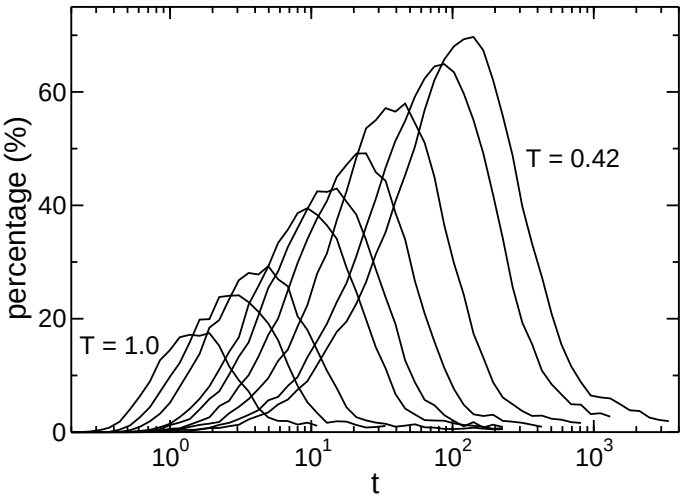

Figure 6.10: The fraction $f(t)$, expressed in percentage, of mobile particles that move in non-trivial strings for temperatures $T = 1.0, 0.75, 0.65, 0.55, 0.52, 0.49, 0.46, 0.43, 0.42$.

particles in this model liquid. In this section we investigate further the details of string-like motion beyond what the length distribution and mean length reveal. What we intend to accomplish is to trace the string-like motion with an increasing amount of microscopic detail, in order to understand precisely how particles move in strings. In doing so, we study a number of issues relevant for cage rearrangement, cooperative motion and dynamical heterogeneity.

As we have seen, strings are largest at times in the late-β/early-α relaxation regime of the MCT, indicating that the motion of mobile particles is highly cooperative on this time scale. However, the mechanism by which this cooperativity is realized is not apparent, i.e., the strings found at $t_{\rm str}^{\rm max}$ may result from a series of local rearrangements at shorter times. For example, one can imagine that the strings are formed as a result of: (i) a "coherent" type of motion where all particles in a

string move simultaneously in a single event, or in a time interval shorter than the dephasing time of the rattling motions within the local cages. (ii) a sequential type of motion where the particles in a string follow each other in a strictly ordered manner along the "backbone" of the string, i.e., the head of the string moves first and the tail last, but at relatively widespread time intervals. Or, (iii) a non-sequential, temporally random type of motion where single events in which the individual particles move into available empty space dominate on short time scales before some structured path emerges. Of course, the actual mechanism may also be a combination of all these processes where the prevailing mechanism depends on both the temperature and the length of the string. This is what we aim to determine in this section.

To investigate these processes, we first examine the individual motion of particles in strings, where we assess the squared displacement of each particle in the strings. Then, we investigate the relative motion of pairs of particles in strings that replace each other, where the relative motion is either with respect to their current positions or the original position of the replaced particle in the pair. In all cases we begin our analysis by inspecting several representative strings. Then, we calculate various ensemble averaged quantities to obtain information about the typical behavior. To do this, we first identify strings found in the lowest T studied, i.e., $T = 0.42$, in the time interval $t_{\mathrm{str}}^{\mathrm{max}}$ from some reference time t_0. Then, starting from the origin of this time interval we monitor the trajectories of the particles during the formation of the strings.

6.4.1 Analysis of typical examples

Single particle motion

First, we assess the individual motion of particles in strings during the formation of the strings by showing the square displacements of the constituting particles, $r_i^2(t)$, for two typical examples, see Fig. 6.11. For the string considered in Fig. 6.11(a), all particles move together within a short period of time by about one inter-particle distance along a single path to replace the neighboring particle, suggesting a nearly coherent type of motion. In contrast, for the string studied in Fig. 6.11(b), two particles move forward as a pair, whereas the jumps of the other particles occur individually at later times. In particular, the delay between the individual jumps is much larger than the dephasing time due to the cage-rattling motions for which an estimate, based on the onset of the plateau regime of the MSD, yields $t_{ph} \approx 5$, cf. Fig. 3.4(a). Thus, the motion of particles in this string is not coherent.

As can be seen in Fig. 6.12, the situation becomes even more complicated for large strings. In this figure, the square displacements $r_i^2(t)$ of the individual particles constituting one large string are organized in three panels to emphasize that sub-units of the string, which we call "microstrings", can be identified within which the particles move nearly simultaneously. For example, the particles labeled 6, 7, 8, 9, and 10 jump together as a unit nearly at the same time, while the motions of the other sub-units occur at different times. All these examples show that the string-like motion realized at some time $t_{\rm str}^{\rm max}$ is a consequence of diverse processes at shorter times, some of which are coherent and some of which are not. To unveil

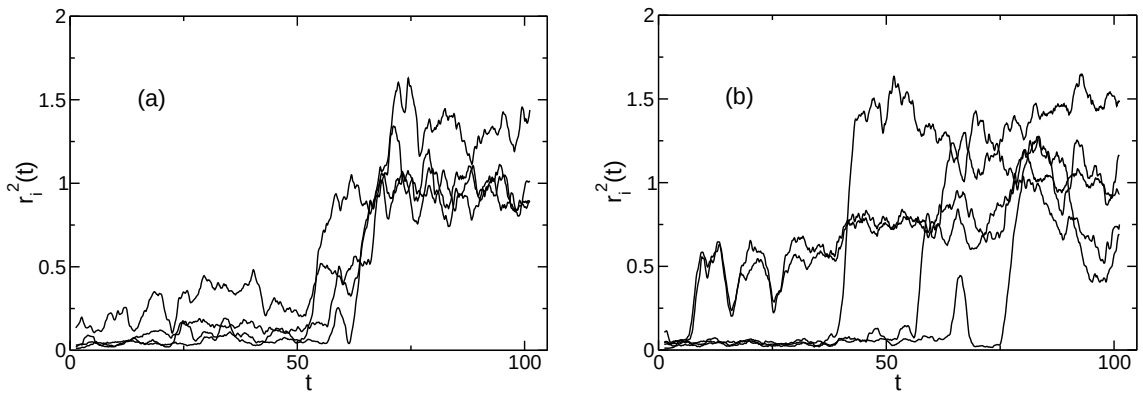

Figure 6.11: Square displacement $r_i^2(t)$ of particles in strings that represent typical examples of (a) coherent motion, and (b) non-coherent motion. Note that here, and in all other figures in this section depicting displacements of particles, the data has been smoothed using running averaging scheme in which several successive data points are replaced by their average to remove vibrational motion. Each data point is an average of 40 successive data points (equivalent to 200 MD steps) or a time range of $t = 2$.

these complicated processes we next inspect the motion of pairs of particles within strings in which one is replaced by the other ("replacing pairs").

Replacing pairs

According to their definition, strings consist of pairs of mobile particles in which one particle replaces the other in a time interval t. Valuable insights into the mechanism of string-like motion can be derived by inspecting the relative motions of these pairs. Suppose particles i and j constitute such a pair in a string k that has been identified in a time interval $t_{\rm str}^{\rm max}$, and let us further assume that j replaces i, i.e., the condition

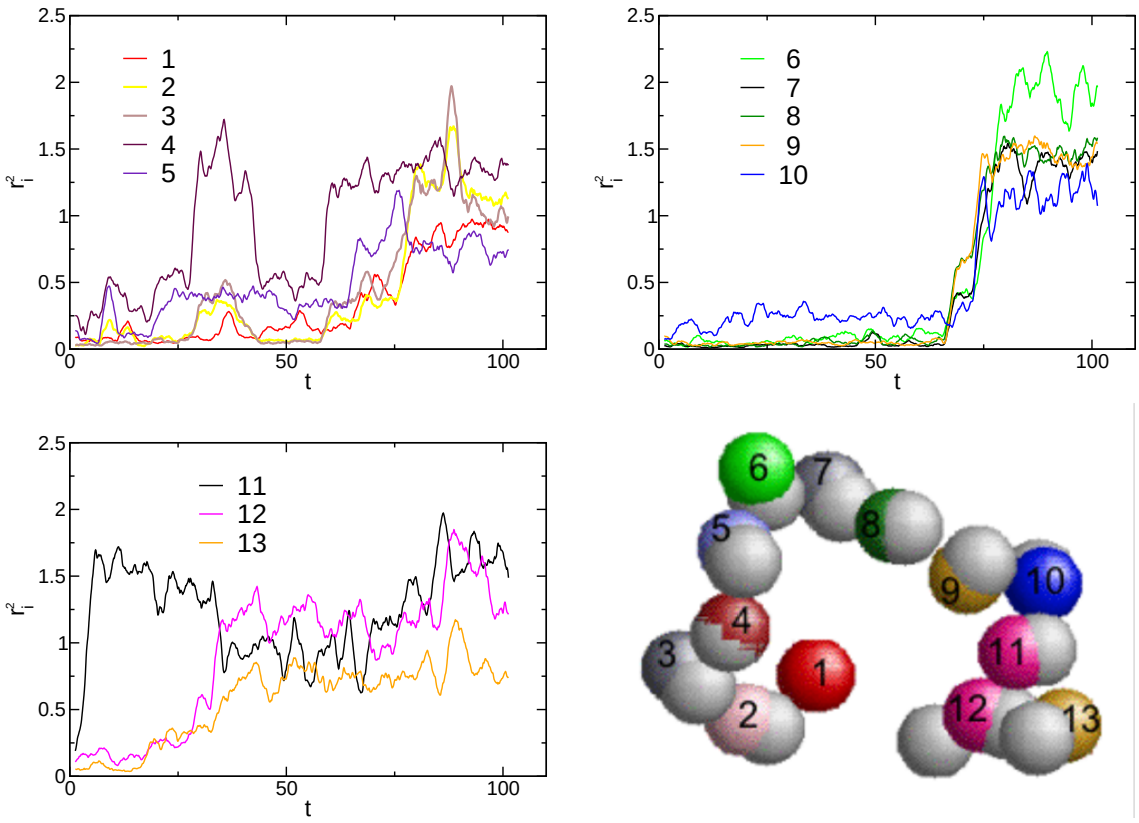

Figure 6.12: The square displacement $r_i^2(t)$ of particles in a large string. Those particles that are moving together are grouped in the same panel. We refer to these sub-units as microstrings. The position of particles in the string at the times t_0 and $t_0 + t_{\rm str}^{\rm max}$ are shown by spheres, where the numbered spheres represent positions at $t_{\rm str}^{\rm max}$ and the remaining gray spheres represent the positions of the corresponding particles at the reference time t_0.

$dr_{\rm ij}(t_{\rm str}^{\rm max}) < \delta \equiv 0.6$ is satisfied, where $dr_{\rm ij}(t_{\rm str}^{\rm max}) \equiv |\mathbf{r_j}(t_0 + t_{\rm str}^{\rm max}) - \mathbf{r_i}(t_0)|$. Then, the calculation of $dr_{\rm ij}(t < t_{\rm str}^{\rm max})$ for all pairs in a string and identification of the times when each pair first satisfies the criterion $dr_{\rm ij}(t) < \delta$ shows when the individual replacements in the string take place. Therefore, in this section we show $dr_{\rm ij}(t)$ for all

pairs in a typical string found at t_L, and then inspect the replacement mechanisms.

Fig. 6.13 depicts the plot of $dr_{\rm ij}(t)$ in the time window between t_0 and $t_0 + t_L$. A number of issues can be understood from the figure. Apparently, the replacements occur as sudden jumps, where for this string some of the jumps take place in groups (e.g. pairs 2, 3, and 4) at about the same time, while others (e.g. 5, 6 and 7) are well separated in time. Occassionally, we also observe unsuccessful replacement attempts (e.g., pair 7), where the replacing particle returns to its initial position prior to the jump, before the successful replacement eventually takes place at a later time $t < t_0 + t_{\rm str}^{\rm max}$. When this happens, we select the time of the final successful jump as the replacement time in the following analysis. By inspecting the duration $\Delta\tau$ (cf. Fig. 6.13) required for all pairs in a string to undergo successful replacement jumps, we can ascertain if the motion is coherent or not. If these jumps occur at once or within a short period of time, then we conclude that the motion is coherent, or simultaneous. Clearly, for the string analyzed, the motion is non-coherent since the replacements occur at widely separated times, and $\Delta\tau \approx 62$ is much longer than the dephasing time. Nevertheless, the string contains a sub-unit or microstring in which particles move simultaneously, as was recognized in the last subsection using a different approach.

When the motion within a string is non-coherent, we can further investigate whether or not the string-like motion involves sequential jumps of the constituting particles along the "backbone" of the string. This can be achieved by observing the time sequence of the replacement jumps in comparison to the pairs' order along the backbone of the string. However, due to partial coherent motion, as seen in pairs

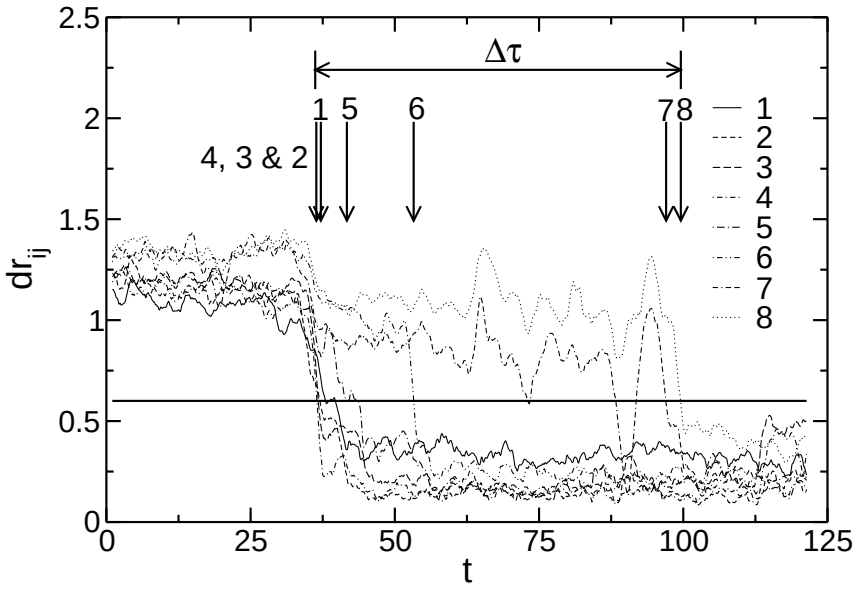

Figure 6.13: A plot of $dr_{\rm ij}(t) \equiv |\mathbf{r_j}(t_0+t) - \mathbf{r_i}(t_0)|$, which characterizes the time when the condition $dr_{\rm ij}(t_{\rm str}^{\rm max}) < \delta \equiv 0.6$ is first satisfied for any pair i, j in a particular string, i.e., the time when particle i is replaced by particle j. The pairs are labelled with a number describing their positions in the string from head to tail, where the pair at the head is labeled 1 and the pair at the tail is 8.

2, 3 and 4 of Fig. 6.13, the identification of sequential motion becomes complicated, since for these pairs the sequence becomes indistinguishable or the order irrelevant because the pairs jump together almost at once. In any case, to gain some insight, we define sequential motion for *strictly* ordered jumps. That means only those strings that show replacement jumps in a strict order from head to tail are considered as displaying sequential motion. Thus, the string analyzed in Fig. 6.13 has a sequence in which $4, 3,$ and 2 move before 1, and hence the motion in this string is not considered sequential. However, we observe other examples that exhibit sequential motion. In

the next section we perform a statistical analysis to quantify this and the other mechanisms.

In the above considerations, we based our analysis at the level of strings to determine the rearrangement mechanisms. For example, by measuring the time span $\Delta\tau$ of the replacement jumps in the whole string, we were able to examine whether the motion is coherent or not. However, with this approach we cannot distinguish those cases where the majority of the particles move simultaneously from those where all jumps are well separated in time. Consider, for example, a string that consists of n particles. If one of the n particles jumps at a much later time while all the others jump simultaneously, $\Delta\tau$ will be large simply because of the one particle with a delayed jump time. Hence, the motion will be interpreted as non-coherent although most involved particles move simultaneously. To capture this behavior, we re-examine the relative motion of particles in replacing pairs, but this time with respect to their current positions, i.e., we calculate $dr_{ij}^{c}(t) \equiv |\mathbf{r_j}(t_0 + t) - \mathbf{r_i}(t_0 + t)|$ for any pair i and j in any given string for which j replaces i. Notice that in our previous analysis the current position $\mathbf{r_j}(t_0 + t)$ of particle j is compared to the original position $\mathbf{r_i}(t_0)$ of particle i, but not to the current position. With this approach we will be able to determine the probability of coherent motion at a level of replacing pairs.

In Fig. 6.14 we show examples of $dr_{ij}^{c}(t)$ for representative pairs in two strings. This figure also includes the plots of $dr_{ij}(t)$ for the corresponding pairs, to mark the jump times described above. If the jump for a given pair occurs simultaneously, the plot of $dr_{ij}^{c}(t)$ remains flat, as shown in Fig. 6.14(a), since the particles do

not separate significantly during their motion. On the other hand, if the replacing particle waits some time before it jumps into the position vacated by a replaced particle, i.e., if the replacement process is delayed, this plot shows, immediately preceding the jump time of the replacing particle, a bump in the plot of $dr_{ij}^{c}(t)$ (cf. Fig. 6.14(b) as an example). Then, the height ($\Delta r_{\rm sep}$) and the width ($t_{\rm sep}$) of this bump (cf. Fig. 6.14(b)) characterize how far and how long the particles separate during the replacement process. From the time $t_{\rm sep}$ we can infer if a pair undergoes a coherent motion or not, while $\Delta r_{\rm sep}$ gives us additional information on the overall cage rearrangement. Clearly, the pair depicted by Fig. 6.14(a) exhibits a coherent jump, while the motion illustrated in Fig. 6.14(b) is non-coherent, since the jump of the replacing particle exhibits a significant delay.

6.4.2 Analysis of ensemble averaged quantities

Thus far, we have studied string-like motion by inspecting several representative examples. In this analysis, it has been demonstrated that the particle rearrangements involved in string-like motion result from a complex procedure involving different mechanisms. In order to determine the dominant mechanism, it is necessary to perform a statistical analysis. Therefore, we now calculate different probability distributions that quantify the average behavior.

We first show the probability distribution $P(\Delta\tau)$ of the time interval $\Delta\tau$ between the first and the last replacement jumps in a given string. In Fig. 6.15 we plot $P(\Delta\tau)$ obtained by averaging over all strings found in a time interval $t_{\rm str}^{\rm max}$, for

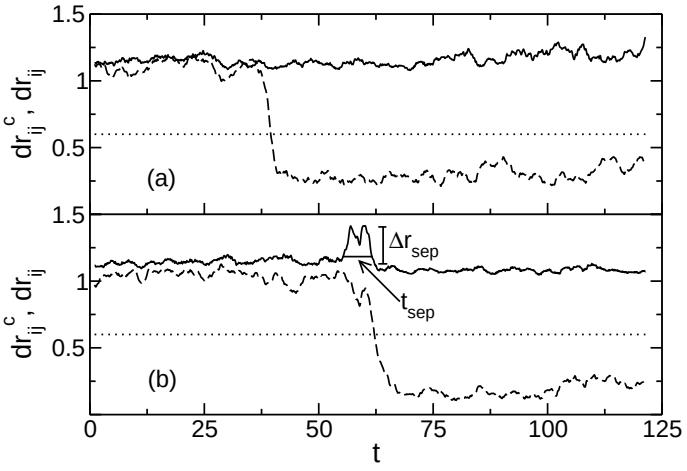

Figure 6.14: The relative motion $dr_{ij}^c(t)$ (solid lines) of replacing pairs that are moving (a) coherently and (b) non-coherently. The separation time and the separation distance between the replacing pairs are indicated by t_{sep} and Δr_{sep}, respectively. To mark the time when the replacement jumps took place, we plot $dr_{ij}(t)$ (dashed line) for each pair. The horizontal dotted lines mark the distance $dr_{ij} = 0.6$.

$T = 0.42$. This distribution quantifies the extent of coherent motion at a string level. As can be seen in the figure, $P(\Delta\tau)$ is a monotonically decreasing function of the time $\Delta\tau$.

To determine quantitatively the extent of coherent motion, we must assign a cutoff time interval below which the motion can be regarded as coherent. Ideally, we wish to define coherent motion as a process where the jumps occur at precisely the same time, but because this is an unlikely process we must choose a reasonably small time that captures this event. In particular, this choice must be larger than the vibrational time scale, since we are interested only in rearrangements that contribute to string-like motion, and these are beyond the individual vibrations. Therefore, by

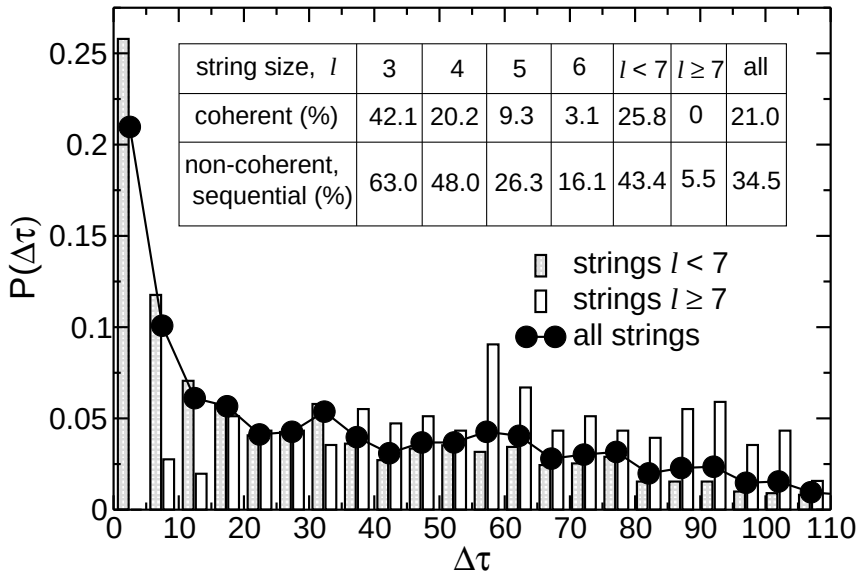

Figure 6.15: Probability distribution $P(\Delta\tau)$, where $\Delta\tau$ is the time between the first and the last replacement jumps in a string (cf. Fig. 6.13). The filled circles show $P(\Delta\tau)$ for all strings, while the shaded and unshaded bars represent $P(\Delta\tau)$ for strings of length $l < 7$ and $l \geq 7$, respectively. The x-axis of the two bars have been shifted from each other by $\Delta\tau = 1.0$ for clarity. In the inset we show the extent of coherent motion when we analyze different string lengths separately. For those strings in which particles that are not moving simultaneously, we calculate the probability of strings that exhibit sequential type of motion.

inspecting the MSD or intermediate scattering function of the bulk (cf. Fig. 3.4(a)), we define coherent or simultaneous motion as a process where all particles in a string undergo replacement jumps in a period $\Delta\tau < 5$. Based on this definition, we find that about 21% of the strings involve the simultaneous motion of all the particles in those strings.

If we study small ($l < 7$) and large ($l \geq 7$) strings separately, we find different

probability distributions. For the small strings, $P(\Delta\tau)$ is large for short time scales ($\Delta\tau < 5$), while for the large strings no monotonic decay, but instead a broad distribution of $P(\Delta\tau)$, is observed. In particular, in terms of the criterion used to quantify the coherent motion, we find that none of the large strings ($l \geq 7$) moves coherently when considering the motion of all particles in the string, whereas roughly 26% of the small strings ($l < 7$) do. This probability grows to roughly 45% for the smallest string size, $l = 3$ (See the inset of Fig. 6.15 for more details). These findings show that only small strings can move coherently as a unit whereas in large strings distinct sub-units of the strings move at different times, resulting in an overall large time interval $\Delta\tau$ for the entire string, and hence implying a non-coherent type of motion at the level of strings. Nevertheless, we will show later in this chapter that at the level of the individual sub-units, or microstrings, the motion is coherent.

For those strings that do not move coherently, i.e., 79% of the strings, we quantify the relevance of sequential type of motion by counting those strings in which particles undergo strictly ordered jumps along the backbone of the string during a rearrangement process. We find that 35% of these strings exhibit sequential motion (cf. inset of Fig. 6.15), while 65% of the strings that are found to be non-coherent exhibit a non-sequential, temporally random motion in which one or more particles disrupt the ordered sequence of the replacement jumps. In fact, out of all strings, coherent as well as non-coherent, this amounts to nearly 51%. Therefore, non-sequential type of motion appears to be an important element of the string-like motion. If we further break down our analysis for different sizes, we find that sequential motion is more prevalent than non-sequential motion for smaller strings.

For example, we find that about 63% of non-coherent strings of size $l = 3$ undergo ordered jumps along the backbone of the strings, while only 9.2% of strings with size $l = 7$ show strictly ordered jumps. Hence, as may be expected, strictly ordered replacements occur prevailingly for small non-coherent strings.

To gain further insight into the rearrangement mechanisms and to further establish the concept of microstrings, we will next discuss the statistical analysis done at the level of replacing pairs in strings. We first calculate the probability distribution $P(t_{\rm sep})$ that characterizes the separation times $t_{\rm sep}$ between replacing particles. Fig. 6.16(a) displays the probability distribution obtained by averaging over all replacing pairs in strings found at a time interval $t_{\rm str}^{\rm max}$ for $T = 0.42$. Similar to what we find from $P(\Delta\tau)$, $P(t_{\rm sep})$ decays monotonically. If we integrate the probability distribution up to a cutoff time $t_{\rm sep} \approx 5$ for estimating the probability of coherent jumps in pairs, we find a value of 0.56. Therefore, about 56% of the replacing pairs move simultaneously in the replacement process. This number is significantly larger than that obtained by analyzing $P(\Delta\tau)$ (21%). The difference shows the presence of a substantial number of strings that have been counted as non-coherent, while the majority of the particles in these strings actually move simultaneously in microstrings. Therefore, the two probability distributions complement each other and, only in combination, give us complete information on the extent of coherent motion in strings.

Additional information about the rearrangement of particles in strings can be extracted by examining Fig. 6.16(b), which shows the probability distribution $P(\Delta r_{\rm sep})$ of an excess separation distance $\Delta r_{\rm sep}$ between replacing pairs in a string

during the replacement process. Questions such as, "By how much do particles separate from each other during the replacement process as compared to the size of the cage radius r_c?" can be studied. Clearly, high probability is attained for small separations, with the tail of the probability extending to $\Delta r_{\rm sep} \approx 1$, i.e., one inter-particle distance. In fact, most pairs have $\Delta r_{\rm sep} < r_c$, where r_c can be estimated using three-time correlation functions [185]. As shown in the section below, this analysis yields $r_c \approx 0.45$ for the Dzugutov liquid at $T = 0.42$, i.e., the cage radius is slightly smaller than the half inter-particle distance, which is consistent with previous findings for supercooled liquids [178, 185]. Based on this estimate, an integration of $P(\Delta r_{\rm sep})$ in the range $\Delta r_{\rm sep} \leq r_c$ yields 0.81, suggesting that only 19% of the pairs separate by a distance larger than the cage radius during the replacement process. Since coherent motion leads to small pair separations, e.g. $\Delta r_{\rm sep} \leq r_c$, the high probability in this region is again consistent with the previous finding that large strings typically consist of several microstrings in which the particles move simultaneously.

6.5 Estimate of the cage radius, r_c

Following Refs. [178, 185], we calculate three-time correlation function characterizing the motion of individual particles in two time intervals in order to estimate the cage size. Specifically, we measure the displacements $\mathbf{r_{01}}$ and $\mathbf{r_{12}}$ of the particles during successive time intervals t_{01} and t_{12}, and then calculate the projection of $\mathbf{r_{12}}$ on the direction of $\mathbf{r_{01}}$, i.e.,

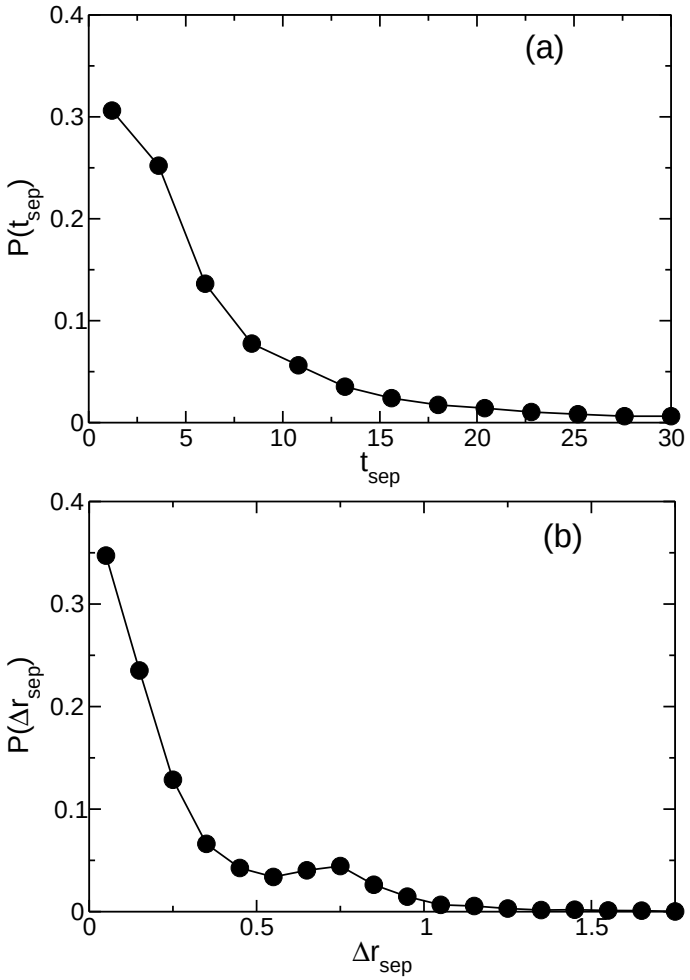

Figure 6.16: Probability distributions of (a) the separation time t_{sep} and (b) the separation distance Δr_{sep} of replacing pairs during the replacement process.

$$x_{12} = \frac{\mathbf{r_{01}}}{r_{01}} \cdot \mathbf{r_{12}}, \tag{6.2}$$

where $r_{01} \equiv |\mathbf{r_{01}}|$. Based on these data, we compute the conditional probability function $P(x_{12}|r_{01})$, which measures the probability to find a specific value x_{12} provided the particle has moved a distance r_{01} in the first time interval t_{01}. Information about the direction of subsequent steps of the motion can be extracted from the first

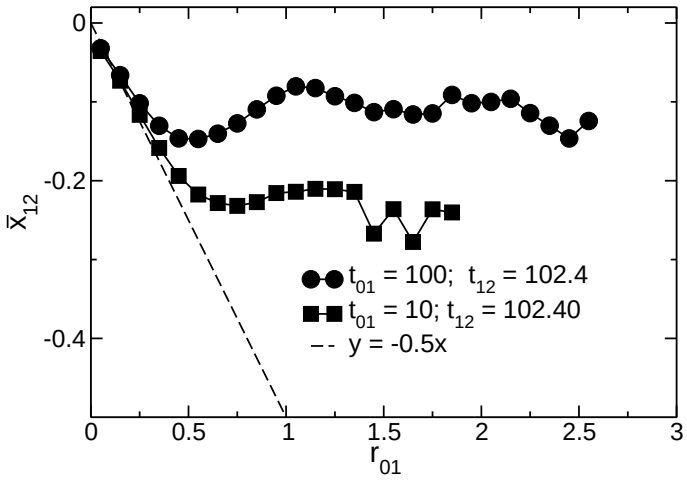

Figure 6.17: Three-time correlation function $\overline{x}_{12}(r_{01})$ characterizing the motion of individual particles in two successive time intervals.

moment of this distribution $\overline{x}_{12}(r_{01})$ [178, 185]. Specifically, $x_{12}(r_{01}) \equiv 0$ will result if the directions of the motions during t_{01} and t_{12} are uncorrelated. In contrast, if the subsequent motion for a given r_{01} is backward (forward) correlated, a negative (positive) value of $\overline{x}_{12}(r_{01})$ will be found. In particular, it has been shown that $\overline{x}_{12}(r_{01}) = -\frac{1}{2}r_{01}$ results from stochastic dynamics in a harmonic potential [186]. Choosing the intervals t_{01} and t_{12} in the caging regime, these effects can be used to estimate the cage size for supercooled liquids [178, 185]. Fig. 6.17 shows $\overline{x}_{12}(r_{01})$ for $t_{01} = 10, 100$, and $t_{12} = 102.4$ at $T = 0.42$. Obviously, $\overline{x}_{12}$ is negative for all values of r_{01}. Thus, as a signature of the cage effect, on average a particle moves opposite to the direction it has moved before. For $r_{01} < r_c \approx 0.45$, the curves nicely follow $\overline{x}_{12}(r_{01}) = -\frac{1}{2}r_{01}$ indicating that the particle is dragged back by a constant fraction of its previous displacement. On the other hand, the back dragging effect decreases for $r_{01} > r_c$. Therefore the cage radius can be estimated to be $r_c \approx 0.45$.

Chapter 7

Configurational entropy and the inherent structure properties of the Dzugutov liquid

So far we have studied the dynamics of supercooled liquids by investigating the nature of correlated motion at a microscopic level. Our observations from these studies open up several questions as to *why* the particles undergo the kind of rearrangement observed in the last chapters: Are there local forces or stresses that drive the particles to rearrange in a certain way? Are there local structures that facilitate these motions? How does the string-like motion take place in relative to local icosahedral arrangement? In the conclusion chapter, we will briefly assess our observations in terms of the theoretical and computational results found in the literature, and leave the above issues for future investigations. Some of these issues are currently under

investigation by other members of the group. In this chapter, we investigate the Dzugutov liquid at a macroscopic level in terms of the thermodynamic approach of the potential energy landscape formalism [46, 47, 48]. This formalism has become highly useful in describing the properties of glass-forming liquids at sufficiently low T.

Our main interest is to make a connection between the dynamical clusters or strings we have investigated and the cooperatively rearranging regions (CRR) of the Adam-Gibbs (AG) theory. Despite the wide applicability of the AG theory, the CRR are not well defined. Our interest is further inspired by recent results of Giovambattista et al. [167] who found a connection between the CRR and the dynamical clusters in the SPC/E model of water, providing a connection between the AG approach and spatially heterogeneous dynamics. Here we would like to assess the validity of their findings for the Dzugutov liquid, testing their observation for a different system. Furthermore, since we have found that strings are the elementary unit of cooperativity, we investigate the extent to which the strings represent the CRR. In order to accomplish our goal we need to calculate the configurational entropy of the system using the inherent structure (IS) formalism. This calculation will provide us with an opportunity to analyze the thermodynamic properties of the Dzugutov liquid. Therefore, the purpose of this chapter is twofold: (i) to study the thermodynamic properties of a glass-forming Dzugutov liquid through the IS formalism, and (ii) to explore the connection between the dynamic and thermodynamic properties through the phenomenological Adam-Gibbs theory.

The calculation of the configurational entropy involves the diagonalization of a

large Hessian matrix, to be described below. To accomplish this task in a reasonable amount of time, we have reduced our system size to $N = 2197$. Unfortunately, small systems have a notorious tendency of nucleation in simulations [5], counter to what is generally expected. To avoid including nucleating configurations in our analysis, we first simulate several independent samples at each T, and then remove those simulations that show a tendency towards nucleation before beginning our analysis. This has been done by carefully inspecting the thermodynamic as well as the dynamic properties of each simulation, where we compare these behaviors with the large system ($N = 17576$) and discard those configurations that have significantly different behavior from the large system. As a result, apart from its size, the properties of the system studied is within statistical error of the large system. Nevertheless, as T decreases the number of configurations that need to be discarded increases and our statistics becomes somehow affected. For example, the analysis for the $T = 0.43$ data may have some limitations with regard to statistics. As a result, we take extra care in drawing certain conclusions at this stage. We indicate any statistically questionable results as we proceed.

7.1 Potential energy landscape

The notion of the potential energy landscape (PEL) was first proposed by Goldstein [45] in his topographic view of the glass transition phenomenon. For an N-body system, the PEL refers to the system's potential energy U plotted as a function of $3N$ particle coordinates in $3N + 1$ dimensional space [45]. The dynamics of an

N-body system can thus be viewed as the motion of a point on the multidimensional potential energy surface. This surface consists of a large number of local minima, of varying depths, surrounded by potential barriers. At sufficiently low temperatures, the system resides near these local minima executing mostly vibrations around the minima, with infrequent transitions from one minimum to another [181].

The qualitative description of the potential energy landscape formulated by Goldstein was later formalized by Stillinger and Weber [46] using the concept of inherent structures (IS), defined as the local minimum configurations of the potential energy hypersurface. Each of these inherent structures is surrounded by a basin of attraction that is defined as the set of points that map to the same inherent structure upon a local minimization of the potential energy. Based on this operational definition, Stillinger and Weber proposed a mathematical formalism referred to as the inherent structure formalism to quantify the thermodynamic properties of the PEL. Since then, this approach has become an essential tool for understanding the dynamic and thermodynamic properties of systems that show glassy behavior, like glass-forming supercooled liquids [96, 97, 128, 130, 164, 182, 187], proteins [10] and disordered spin systems [188]. In the following section we present the mathematical formalism that describes the thermodynamics of the IS, which can also be found in Refs. [46, 95, 164, 184, 187]

7.2 Inherent structure thermodynamic formalism

The central idea to the IS formalism is the notion that, at sufficiently low T, the dynamics of a liquid can be separated into vibrations within a single basin and infrequent transitions between basins. This partitioning is motivated by the concept that at these temperatures the time scales corresponding to the intrabasin and interbasin motions that involve vibrations and structural relaxation by thermally activated crossing of potential energy barriers, respectively, differ by several orders of magnitude. Direct numerical evidence for the separation of the dynamics of a liquid into vibrations around and transitions between IS was provided by Schrøder et al. [130], where such separation become possible in the vicinity of the mode-coupling temperature T_{MCT}. The consequence of this partitioning is that the canonical partition function can be conveniently re-written as a sum over all local potential energy minima [46].

Generally, the canonical partition function of a system of N interacting particles is given by [125]

$$Q_N(V,T) = \frac{h^{3N}}{N!} \iint \exp\left[-\beta \mathscr{H}_N(\mathbf{r}^N, \mathbf{p}^N)\right] \mathrm{d}\mathbf{r}^N \mathrm{d}\mathbf{p}^N. \tag{7.1}$$

where $\mathscr{H}_N(\mathbf{r}^N, \mathbf{p}^N)$ is the Hamiltonian of the system that is expressed as

$$\mathscr{H}_N(\mathbf{r}^N, \mathbf{p}^N) = \frac{1}{2m} \sum_{i=1}^{N} |\mathbf{p}_i|^2 + U_N(\mathbf{r}^N). \tag{7.2}$$

Here $\mathbf{r}^N$ and $\mathbf{p}^N$ are shorthand representations for the positions and momenta of the N particles, and $U_N(\mathbf{r}^N)$ is the total potential energy of the system. For the above Hamiltonian, the integration over momenta can be can be carried out explicitly, and

$Q_N(V,T)$ can be reduced to

$$Q_N(V,T) = \frac{\Lambda^{-3N}}{N!} Z_N(V,T), \tag{7.3}$$

where $\Lambda = (2\pi\beta\hbar^2/m)^{1/2}$ is the de Broglie thermal wave length, and

$$Z_N(V,T) = \int \exp\left[-\beta U_N(\mathbf{r}^N)\right] d\mathbf{r}^N \tag{7.4}$$

is the configuration integral. In the IS formalism, the partitioning of the configuration space into an ensemble of nonoverlapping basins allows the partition function expressed above to be re-written as [46]

$$Q_N(V,T) = \Lambda^{-3N} \sum_{\text{basins}} \exp(-\beta e_{IS}) \int_{R_{\text{basin}}} \exp[-\beta(U_N - e_{IS})] d\mathbf{r}^N, \tag{7.5}$$

where R_{basin} is the set of points associated to a specific basin, and e_{IS} is the potential energy of the inherent structure corresponding to the basin. This last equation shows that the contribution to the partition function $Q_N(V,T)$ can be separated into two parts: the IS energy of all distinct basins, and the thermal excitation within the basins. If we introduce $\Omega(e_{IS})de_{IS}$ as the number density of states with IS energy between e_{IS} and $e_{IS} + de_{IS}$, then the above equation can be written as

$$Q_N(V,T) = \int de_{IS} \Omega(e_{IS}) \exp[-\beta e_{IS} - \beta f_{\text{basin}}(\beta, e_{IS})], \tag{7.6}$$

where

$$-\beta f_{\text{basin}}(\beta, e_{IS}) = \ln\left(\int_{R(e_{IS})} \exp[-\beta(U_N - e_{IS})] \frac{d\mathbf{r}^N}{\Lambda^{3N}}\right). \tag{7.7}$$

$f_{\text{basin}}(\beta, e_{IS})$ is interpreted as the basin free energy with IS energy e_{IS}. Eq. 7.6 is further simplified if we define the configurational entropy $S_{\text{conf}}(e_{IS})$ as

$$S_{\text{conf}}(e_{IS}) = k_B \ln\Omega(e_{IS}). \tag{7.8}$$

Then, Eq. 7.6 becomes

$$Q_N(V,T) = \int de_{IS} \exp[-\beta(e_{IS} + f_{\text{basin}}(\beta, e_{IS}) - TS_{\text{conf}}(e_{IS}))]. \tag{7.9}$$

In the thermodynamic limit the free energy A of the system can be obtained from this equation by employing a maximum integrand evaluation, which yields

$$A = \bar{e}_{IS} - TS_{\text{conf}}(\bar{e}_{IS}) + f_{\text{basin}}(\beta, \bar{e}_{IS}). \tag{7.10}$$

Here $\bar{e}_{IS}$ is the average IS energy for a given T that maximizes the integrand, and is the solution of the equation

$$1 + \frac{\partial}{\partial e_{IS}} f_{\text{basin}}(\beta, \bar{e}_{IS}) - T\frac{\partial}{\partial e_{IS}} S_{\text{conf}}(\bar{e}_{IS}) = 0. \tag{7.11}$$

The expression for the free energy (Eq. 7.10) can be interpreted as follows. The first two terms on the right hand side of Eq. 7.10 account for the average energy of the PEL minima visited and the degeneracy of the average IS energy $\bar{e}_{IS}$, respectively. The vibrational and the kinetic contributions are captured in the last term. Eq. 7.10 thus provides a formal expression for the separation of configurational and vibrational contributions.

7.3 Methods for the evaluation of configurational entropy

The configurational entropy can be evaluated using two different methods, referred to as the potential energy landscape (PEL) method and the thermodynamic integration (TI) method [164]. The PEL method is based on constructing the probability

distribution $P(e_{IS}, T)$ of the IS potential energy e_{IS} sampled at a temperature T. Examining Eq. 7.9, it can be easily recognized that the probability $P(e_{IS}, T)$ that the liquid populates a given inherent structure e_{IS} at temperature T can be expressed as

$$P(e_{IS}, T) = \frac{\exp[-\beta(e_{IS} + f_{\text{basin}}(\beta, e_{IS}) - TS_{\text{conf}}(e_{IS}))]}{Q_N(V, T)} \tag{7.12}$$

If $P(e_{IS}, T)$ and the free energy of the basin $f_{\text{basin}}(\beta, e_{IS})$ can be obtained from simulation, then Eq. 7.12 can be inverted to yield $S_{\text{conf}}(e_{IS})$ up to an unknown T-dependent constant $C(T) = k_B \ln Q_N(V, T)$ as [164, 181, 187]

$$S_{\text{conf}}(e_{IS}) = k_B \ln P(e_{IS}, T) + \frac{e_{IS}}{T} + \frac{f_{\text{basin}}}{T} + C(T). \tag{7.13}$$

In practice, if the basin free energy has a weak dependence on e_{IS}, the unknown T-dependent constant $C(T)$ is estimated by superimposing the $P(e_{IS}, T)$ curves at different temperatures and selecting the constant that provides a maximum overlap between curves with different T (see Ref. [164, 187] for more details).

The temperature dependence of S_{conf} is obtained by taking the equilibrium average of $S_{\text{conf}}(e_{IS})$ for each T, i.e., integrating $S_{\text{conf}}(e_{IS})$ for each T using the relation [164]

$$S_{\text{conf}}(T) = \int de_{IS} S_{\text{conf}}(e_{IS}) P(e_{IS}, T). \tag{7.14}$$

The TI method, on the other hand, uses the fact that, upon deep supercooling, the motion of particles in configuration space is separable into vibrations within a

basin and infrequent transitions between basins. As a result, the total entropy $S(T)$ can be expressed as the sum of the configurational entropy $S_{\rm conf}(T)$ that results from the multiplicity of local potential energy minima sampled by the liquid, and the vibrational entropy $S_{\rm vib}(T)$ of typical basins sampled at T. Thus,

$$S_{\rm conf}(T) = S(T) - S_{\rm vib}(T). \tag{7.15}$$

As will be described below in detail, the total entropy at a selected reference state point is obtained by integrating the pressure obtained from simulation along an isothermal path from the ideal gas limit value to the reference point. Once the total entropy is obtained at the reference T, the total entropy at any other T is determined by integrating the potential energy along an isochoric path. Our analysis is made using the thermodynamic integration method that will be described below in more detail. We first discuss the properties of the inherent structure.

7.4 Properties of the inherent structure

As a first step towards the evaluation of the configurational entropy we search for the inherent structures. This is accomplished by carrying out a conjugate gradient minimization [189] of the potential energy to find the inherent configuration from the configuration of the equilibrium liquid, which is obtained during the molecular dynamics simulation. This process is often referred to as "quenching". For a given temperature, several equilibrium configurations are then mapped onto the inherent configurations.

Once the inherent structures are identified a number of features can be analyzed. For example, in order to classify the spatial distribution of particles in a given potential energy minimum one measures quantities like the pair correlation function $g(r)$ or the static structure factor $S(q)$. Here we show the pair correlation functions of the inherent configuration for two different temperatures, $T = 0.43$ and $T = 1.0$ (see Fig. 7.1). As first recognized by Stillinger and Weber [46] the hidden structure obscured by the influence of vibration in the equilibrium liquid becomes more apparent in the inherent structure. This is also the case for our system where the quenching of the vibrations causes splitting of the second peak in the pair correlation that arise from icosahedral ordering to become sharper than for the equilibrium liquid. In fact, for our system the splitting of the second peak is already obvious for the equilibrium liquid at low T, but the difference is quite significant for $T = 1.0$ (see inset of Fig. 7.1 for g(r) of the equilibrium liquid).

Another important property obtained from the inherent structure is its potential energy value, $e_{\rm IS}$. As mentioned above briefly, $e_{\rm IS}$ has a central role in the determination of the configurational entropy $S_{\rm conf}$ using the PEL method. This is also true in the TI method (see below). The temperature dependence of the average energy of the local minima $\bar{e}_{IS}$ has been studied by several authors [95, 97, 98, 128, 165, 184, 190, 191]. For sufficiently high temperatures, $\bar{e}_{IS}$ is observed to be essentially constant. On cooling below the onset of caging, the IS average energy decreases, suggesting that the system populates deeper and deeper basins upon supercooling. Moreover, the T dependence of $\bar{e}_{IS}$ is often found to follow a $1/T$ law for fragile liquids [95, 128, 165, 184, 190]. This has been shown to

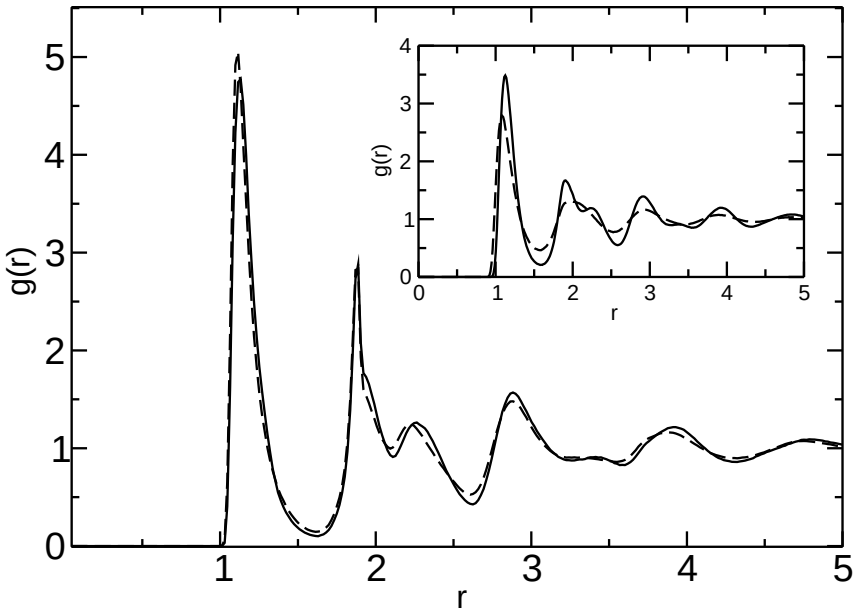

Figure 7.1: Inherent structure pair correlation function $g(r)$ for $T = 0.43$ (solid line) and $T = 1.0$ (dashed line). Inset: pair correlation function $g(r)$ for the equilibrium liquid for $T = 0.43$ (solid line) and $T = 1.0$ (dashed line).

be true provided $P(e_{IS}, \beta)$ is a Gaussian [162, 165, 192].

Motivated by these studies, we evaluate $\bar{e}_{IS}$ for the Dzugutov liquid. Fig. 7.2 shows the plot of $\bar{e}_{IS}$ as a function of T. As found in other systems [95, 97, 98, 128, 165, 184, 190, 191], $\bar{e}_{IS}$ is essentially temperature independent at high T, and then decreases to lower values with decreasing T. Notice that, similar to what is observed for these systems, the T at which $\bar{e}_{IS}$ starts to decrease is near the onset of caging ($T \sim 1$). At sufficiently low T, $\bar{e}_{IS}$ shows the expected $1/T$ temperature dependence, cf. inset of Fig. 7.2. For the lowest T studied ($T = 0.43$), we observe a slight deviation from the expected $1/T$ behavior, showing a tendency towards a constant value. Although this is an interesting phenomenon that was also observed

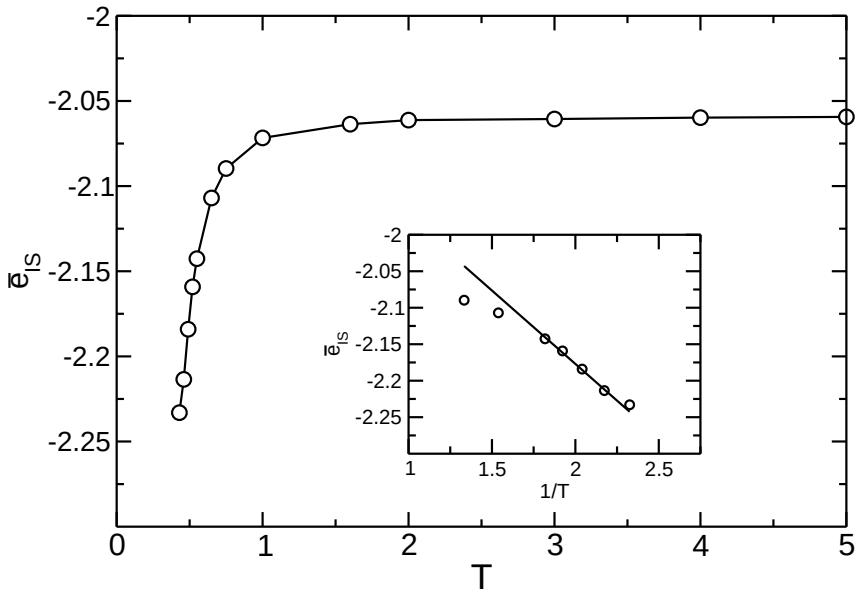

Figure 7.2: Average inherent structure energy per particle $\bar{e}_{IS}/N$ as a function of T. Inset: $\bar{e}_{IS}/N$ (circle) as a function of the inverse temperature for $T \leq 0.75$. The solid straight line is a guide to an eye.

for the strong liquid silica [97, 98], where $\bar{e}_{IS}$ appears to approach a constant value on cooling, we must point out that the deviation in our system may be due to insufficient statistics for the lowest T. Further exploration is required to conclusively determine this issue.

The shape of a basin can be studied by investigating the properties of the density of states, which is the histogram of the square root of the eigenvalues obtained from the diagonalization of the Hessian matrix **H**. The latter is calculated for the configurations of the inherent structures. Near a local minimum, the potential energy of a system of $3N$ particles can be approximated as a Taylor series expansion around the minimum, i.e.,

$$U(\mathbf{r}^N) = U(\mathbf{r}_\alpha^N) + \sum_{i=1}^{N}(\mathbf{r}_i - \mathbf{r}_{\alpha i})\frac{\partial U}{\partial \mathbf{r}_i} + \sum_{i,j=1}^{N}(\mathbf{r}_i - \mathbf{r}_{\alpha i})\frac{\partial^2 U}{\partial \mathbf{r}_i \partial \mathbf{r}_j}\bigg|_{\mathbf{r}_\alpha^N}(\mathbf{r}_j - \mathbf{r}_{\alpha j}) + \text{higher order terms.} \tag{7.16}$$

where $\mathbf{r}_\alpha^N$ is the $3N$-dimensional inherent configuration at the local minimum α. At the local minimum, which is the basin minimu, the first derivative vanishes and $U(\mathbf{r}_\alpha^N) = e_{IS}$. The above expression can thus be reduced to

$$U(\mathbf{r}^N) = e_{IS} + \sum_{i,j=1}^{N}(\mathbf{r}_i - \mathbf{r}_{\alpha i})\frac{\partial^2 U}{\partial \mathbf{r}_i \partial \mathbf{r}_j}\bigg|_{\mathbf{r}_\alpha^N}(\mathbf{r}_j - \mathbf{r}_{\alpha j}) + \text{higher order terms.} \tag{7.17}$$

The $3N$x$3N$ mass-weighted Hessian matrix elements H_{ij} are then defined as

$$H_{ij} = \frac{1}{\sqrt{m_i m_j}}\frac{\partial^2 U}{\partial \mathbf{r}_i d\mathbf{r}_j}\bigg|_{\mathbf{r}_\alpha^N} \tag{7.18}$$

where m_i and m_j are the masses of particle i and j. These values are unity for our system.

The diagonalization of this matrix yields $3N-3$ positive eigenvalues $\{h_i\}$, with the remaining three eigenvalues being zeros. The three zero eigenvalues account for the three independent translations of the entire system, one in each direction. The density of states $N(\omega)$ of the resulting eigenfrequencies ω_i $\left(= \sqrt{h_i}\right)$ are plotted for our system in Fig. 7.3, where data for different T are shown. As seen in the figure, the spectrum of w_i changes weakly with T. But one can clearly see that the position

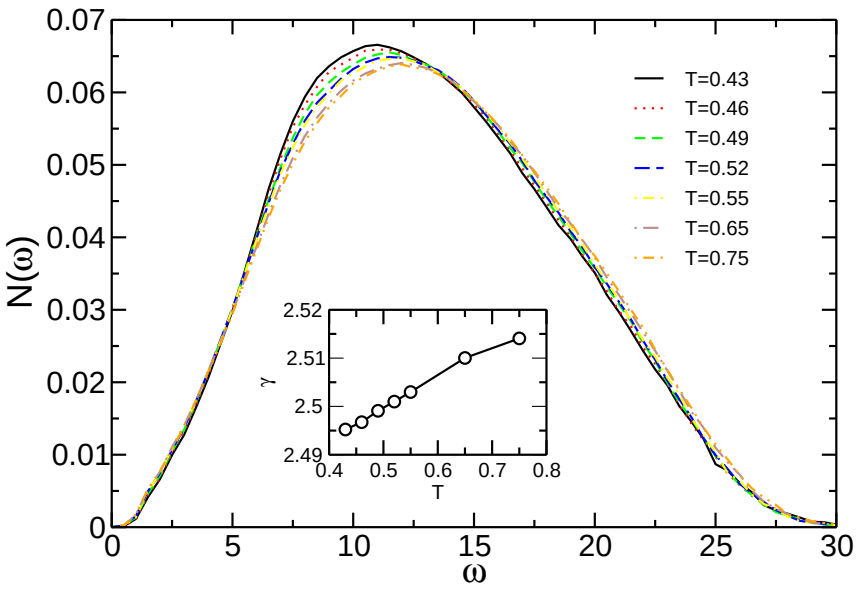

Figure 7.3: Density of states $N(w)$ at different temperatures. This quantity is the histogram of the square root of the eigenvalues of the Hessian matrix evaluated at the basin minimum. Inset: γ as a function of T.

of the maximum slightly shifts to smaller values of w with decreasing T. To closely inspect this change we calculate a quantity γ, which is the ensemble average of the sum of the logarithms of the frequencies of normal modes,

$$\gamma = \left\langle \frac{1}{3N-3} \sum_{i=1}^{3N-3} \ln w_i \right\rangle \tag{7.19}$$

This quantity can be considered as the first moment of the eigenfrequencies, and captures the average quadratic shape of a basin. The temperature dependence of γ is shown in the inset of Fig. 7.3. Albeit small, the value of γ decreases with decreasing T. A similar decrease is observed in the Lewis-Wahnström model of supercooled orthoterphenyl (OTP) [184]. But, it is in contrast to the observation

made for the SPC/E model of water [95] where γ *increases* with decreasing T. The fact that the average basin frequency becomes larger upon cooling (equivalently in deeper basins) in the water model implies that the basins become increasingly "sharp" with decreasing T or e_{IS} [95]. The basins for our system, on the other hand, become slightly broader upon cooling, since γ decreases with T.

7.5 Calculation of the configurational entropy

We calculate the configurational entropy using the thermodynamic integration method, as mentioned above. In this section we present the methods we used and the results we found in the calculation of $S_{\rm conf}(T)$. In the TI method, the total entropy (or free energy) at any temperature T is evaluated by performing thermodynamic integration first along an isothermal path, starting from an ideal gas limit where interactions are far less important, and then along an isochoric path. These procedures, together with the methods for calculating the vibrational entropy, will be described below.

7.5.1 Total entropy $S(R)$ at a reference state point R

For a system of N particles in equilibrium at constant V and T, the Helmholtz free energy A can be expressed as

$$A = -k_B T \ln Q_N(V,T), \tag{7.20}$$

where $Q_N(V,T)$ is the canonical partition function given by Eq. 7.1. For an ideal

gas, $U_N(\mathbf{r}^N) = 0$, and hence $Z_N(V,T) = V^N$. As a result, the partition function of an ideal gas can be expressed as

$$Q_N^{id}(V,T) = \frac{\Lambda^{-3N}}{N!} V^N. \tag{7.21}$$

Then the partition function of a system of interacting particles is conveniently written as

$$Q_N(V,T) = \frac{Q_N^{id}(V,T)}{V^N} Z_N(V,T). \tag{7.22}$$

From this last equation, it can be seen that the free energy A can be split into the ideal term A^{id} and the excess term A^{ex} as

$$A = A^{id} + A^{ex} \tag{7.23}$$

where

$$A^{id} = -k_B T \ln Q_N^{id}(V,T). \tag{7.24}$$

and

$$A^{ex} = -k_B T \ln \frac{Z_N(V,T)}{V^N}. \tag{7.25}$$

From Eq. 7.21 and 7.24, we can find an expression for the ideal free energy A^{id} as

$$\beta A^{id} = 3N \ln \Lambda + N \ln \rho - N. \tag{7.26}$$

The excess free energy A^{ex} contains all the contributions to A that arise from the interaction between particles. Since the free energy can be split into the ideal and excess terms, all thermodynamic properties that can be derived from the free energy can be split in the same manner. For example, the total entropy S can be written as $S = S^{id} + S^{ex}$, where S^{id} and S^{ex} are the ideal and excess terms of the entropy, respectively.

In order to determine the total entropy at a reference state point $R \equiv (T_R, V_R)$, where T_R and V_R are the reference temperature and volume, respectively, a thermodynamic integration is performed at T_R starting from a volume $V \to \infty$ (ideal gas limit) to the reference volume V_R. In particular, we integrate the excess pressure P_{ex} to obtain A^{ex}, from which the entropy S can be specified. Noting that $A = E - TS$, where E is the internal energy and S is entropy, it can be easily shown that

$$S(R) = S^{id}(R) + \frac{U(R)}{T} + \int_{\infty}^{V_R} \frac{P_{ex}}{T} \mathrm{d}V. \tag{7.27}$$

where $S^{id} = -\partial A^{id}/\partial T$ is expressed as

$$\frac{S^{id}}{k_B} = \frac{3}{2} N \ln\left(\frac{emV^{2/3}}{2\pi\hbar^2\beta}\right) - N \ln\left(\frac{N}{e}\right). \tag{7.28}$$

Here e is the Neper number. To evaluate this last equation, we use $\hbar = 0.063507\,\mathrm{kJ\,ps\,mol^{-1}}$, and assume the unit of energy to be $\epsilon = 1\,\mathrm{kJ\,mol^{-1}}$, the unit of length to be $\sigma = 1\,\mathrm{nm}$, and the unit of time to be $\tau = 1\,\mathrm{ps}$. Additionally, we choose the reference state point $R \equiv (k_B T_R = 5.0\,\epsilon,\, V_R = (2197/0.85)\,\sigma^3)$ such that the temperature of the system is high enough to be a gas. Using these values, $S^{id}(R)/k_B$ becomes

$$\frac{S^{id}}{k_B} = 23265.517. \tag{7.29}$$

Notice that $S^{id}(R)/k_B$ is dimensionless. If we divide Eq. 7.27 by k_B, all the other terms can also be written in dimesionless units as

$$\frac{S(R^*)}{k_B} = \frac{S^{id}(R^*)}{k_B} + \frac{U(R^*)}{T^*} + \int_{\infty}^{V_R^*} \frac{P_{ex}^*}{T^*} \mathrm{d}V^*. \tag{7.30}$$

where we made use of the fact that $k_B T = \epsilon T^*$, $U(T) = U(T^*)\epsilon$, $V = V^*\sigma^3$ and $P_{ex} = P_{ex}^*\epsilon/\sigma^3$. Here $R^* \equiv (T_R^* = 5.0,\, V_R^* = 2197/0.85)$ is the reference point in reduced units. In what follows we drop the symbol '$*$'.

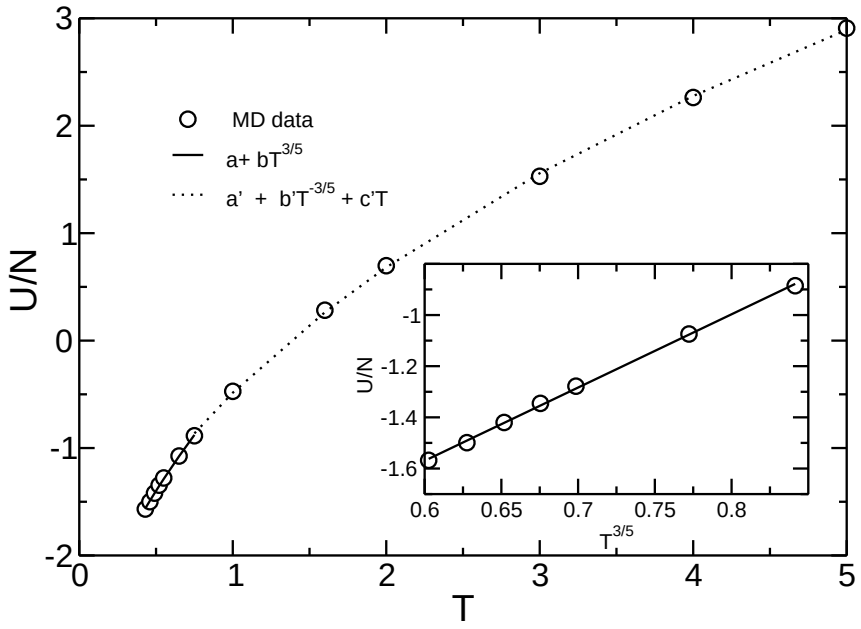

Figure 7.4: The potential energy per particle U/N (circles) of the equilibrium liquid plotted as a function of T. The temperature range $T \leq 0.75$ is fitted (solid line) using the functional form $a + bT^{3/5}$ with $a = -7225.63$ and $b = 6294.06$. The dotted line is a fitting to the data for the range $T \geq 0.75$ using the functional form $a' + b'T^{3/5} + c'T$, with $a' = -6771.84$, $b' = 6492.36$, and $c' = -785.32$. Inset: The data for $T \leq 0.75$ is separately plotted on $T^{3/5}$ axis to demonstrate that the Rosenfeld-Tarazona law [193] holds for low T.

The potential energy $U(R)$ at the reference point R is obtained from the simulation data plotted in Fig. 7.4, i.e., $U(T_R = 5.0, V_R = 2197/0.85)/N = 2.90872$. Therefore,

$$\frac{U(R)}{T} = 6390.4578/5.0. \tag{7.31}$$

The values of the excess pressure $P_{ex} = P - P_{id}$, where P is the total pressure obtained from the simulation at $T = 5.0$ and $P_{id} = \rho k_B T$, is plotted in Fig. 7.5 as a

function of V. P_{ex} has been fit using the virial expansion, i.e.,

$$P_{ex} = \sum_{i=1}^{4} a_i V^{-(i+1)}. \tag{7.32}$$

The a_i values that best fit the data are reported in the figure. To decrease the numerical integration error in Eq. 7.30, the integration over the excess pressure P_{ex} from the ideal gas limit to the reference state point is carried out as follows: we first subtract the virial term $B_2(T)T(N/V)^2$ from P_{ex}, and then integrate the difference over volume. The contribution arising from the first virial correction is then integrated analytically and added to the remaining calculation. The resulting expression is

$$\int_{\infty}^{V_R} \frac{P_{ex}}{T} \mathrm{d}V = \int_{\infty}^{V_R} \frac{1}{T} \left[P_{ex} - B_2(T)T \left(\frac{N}{V} \right)^2 \right] \mathrm{d}V - B_2(T) \frac{N^2}{V_R}. \tag{7.33}$$

Here $B_2(T)$ is the virial coffecient in reduced units. The same exercise described above can convert $B_2(T)k_BT(N/V)^2$ to an expression in reduced units noting that $B_2(T) = B_2(T^*)\sigma^3$ [125]. As a result $B_2(T)k_BT(N/V)^2 = B_2(T^*)T^*(N/V^*)^2\epsilon/\sigma^3$, and hence the expression utilized in Eq. 7.33 is in reduced units, where we dropped the symbol '$*$' as usual.

$B_2(T)$ can be determined from the fit of the P_{ex} vs V curve (Fig. 7.5), at the region where P_{ex} approximates the second term of the virial expansion, i.e., as $V \to \infty$, P_{ex} is expressed as,

$$P_{ex} = B_2(T)T \left(\frac{N}{V} \right)^2 = \frac{a_1}{V^2}. \tag{7.34}$$

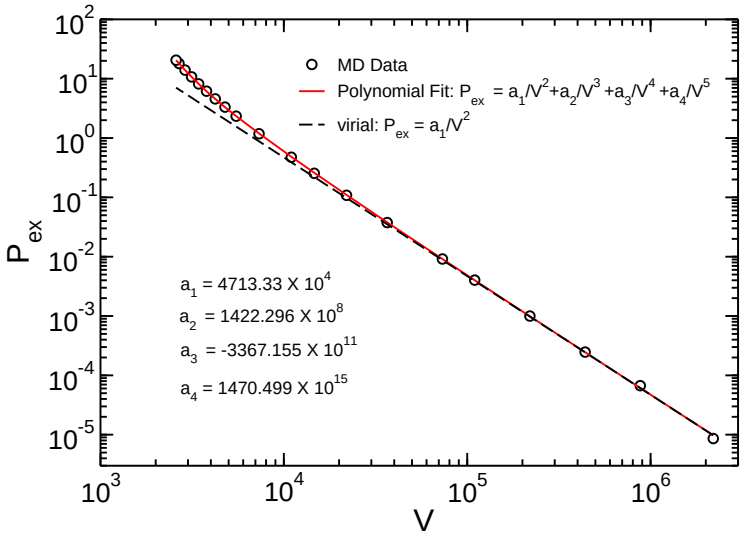

Figure 7.5: The excess pressure P_{ex} as a function of volume at $T = 5.0$. The open circles are the MD result. The dashed line is the fitting to the pressure using the first virial correction to the ideal gas law. The solid line is a polynomial fit

The best fit of P_{ex} to a_1/V^2 for $V \to \infty$ yields $a_1 = 4713.33\ X\ 10^4$. Noting that $a_1 = B_2(T)\,T\,N^2$, we find that $B_2(T) = 1.95298$. Using this value of $B_2(T)$, the first and the second terms in the right hand side of Eq. 7.33 yield $-12696.7/5$ and 3647.093, respectively. Therefore,

$$\int_{\infty}^{V_R} \frac{P_{ex}}{T} \mathrm{d}V = -6186.293. \tag{7.35}$$

The total entropy at the reference state point is thus determined by combining the values obtained in Eq. 7.29, 7.31, and 7.35. As a result, we obtain

$$\frac{S(R)}{k_B} = 23265.517 + 6390.4578/5.0 - 6186.293 = 18357.32 \tag{7.36}$$

7.5.2 Entropy $S(T)$ at different temperatures

Noting that $\mathrm{d}E = T\mathrm{d}S - P\mathrm{d}V$ for a constant number of particles N, where E is the internal energy of the system, the entropy at any T along the isochoric path we studied can be obtained by integrating the internal energy as

$$S(T, V_R) = S(T_R, V_R) + \int_{T_R}^{T} \frac{\mathrm{d}E}{T'} \tag{7.37}$$

The above equation may be re-written interms of the heat capacity C_V as

$$S(T, V_R) = S(T_R, V_R) + \int_{T_R}^{T} C_V(T') \frac{\mathrm{d}T'}{T'} \tag{7.38}$$

where

$$C_V(T) = \frac{3}{2} N k_B + \left(\frac{\partial U}{\partial T} \right)_V . \tag{7.39}$$

$C_V(T)$ is thus calculated by evaluating the derivative of the system's potential energy obtained from the simulation with respect to T. To perform this numerically, we first express U as a function of T. An equation that expresses the temperature dependence of U can be obtained by fitting the simulation data to a functional form that best describes the data.

Based on the free energy functional calculation for hard spheres, Rosenfeld and Tarazona [193] showed that the temperature dependence of the potential energy of liquids can be described by $U(T) \sim T^{3/5}$. In a number of simulations [164, 184, 187, 194, 195], this theoretical prediction has been found to hold for sufficiently low T. The potential energy for our system is also well represented by the form

$$U(T) = a + bT^{3/5} \tag{7.40}$$

for $T < 1.0$, see inset of Fig. 7.4. The best fit for the equation yields $a = -7225.63$, and $b = 6294.06$, in the temperature range $T \leq 0.75$, where the splitting of the dynamics into intra-basin and inter-basin motions is assumed to hold. However, the Rosenfeld-Tarazona law fails to hold for $T \geq 1.0$. For this region, motivated by the functional form suggested in Ref. [184], we fitted the simulation data using the form

$$U(T) = a' + b'T^{3/5} + c'T \tag{7.41}$$

As shown in Fig. 7.4, we find a good interpolation in this region. The best fit for the data is found for $a' = -6771.84$, $b' = 6492.36$, and $c' = -785.32$. It is interesting to note that with a different set of fitting parameters, this functional form fits well for the entire range of the simulation data. But, to be able to use the theoretically predicted functional form, we apply the latter form (Eq. 7.41) only for the T range where the $T^{3/5}$ law fails to hold. Therefore, in order to determine $S(T)$ for $T \leq 0.75$, we first carry out an isochoric thermodynamic integration in the range $T_R \leq T \leq T^*$, where $T^* = 0.75$, using the C_V derived from the potential energy expressed by Eq. 7.41. This procedure links the entropy of the system $S(T_R)$ at the reference temperature $T_R = 5.0$ to that at $T^* = 0.75$. Then, we use C_V derived from the potential energy expressed by Eq. 7.40 together with the value of S at T^* to calculate $S(T)$ at any $T < 0.75$.

The total entropy at T^* is thus obtained using an expression given by

$$\frac{S(T^*, V_R)}{k_B} = \frac{S(T_R, V_R)}{k_B} + \left(\frac{3}{2}N + c'\right) \ln\left(\frac{T^*}{T_R}\right) - \frac{3}{2}b' \left(T^{*-2/5} - T_R^{-2/5}\right). \quad (7.42)$$

Substituting the known values, the above equation becomes

$$\frac{S(T^*, V_R)}{k_B} = 18357.32 + 2510.18 \ln\left(\frac{T^*}{T_R}\right) - 9738.54 \left(T^{*-2/5} - T_R^{-2/5}\right). \quad (7.43)$$

from which we find that, setting $T^* = 0.75$ and $T_R = 5.0$,

$$\frac{S(T^*, V_R)}{k_B} = 7784.6643 \quad (7.44)$$

The total entropy at any $T \leq T^*$, a region where the $T^{3/5}$ law for potential energy holds, is found using the expression

$$\frac{S(T, V_R)}{k_B} = \frac{S(T^*, V_R)}{k_B} + \frac{3}{2}N \ln\left(\frac{T}{T^*}\right) - \frac{3}{2}b \left(T^{-2/5} - T^{*-2/5}\right), \quad (7.45)$$

which is obtained using Eq. 7.40 to express $U(T)$. With the known values, $S(T, V_R)$ becomes

$$\frac{S(T, V_R)}{k_B} = 7784.6643 + 3295.5 \ln\left(\frac{T}{T^*}\right) - 9441.05 \left(T^{-2/5} - T^{*-2/5}\right). \quad (7.46)$$

This last equation is used to evaluate the total entropy at any $T \leq 0.75$, i.e., a T region below the onset of caging where the splitting of the dynamics into

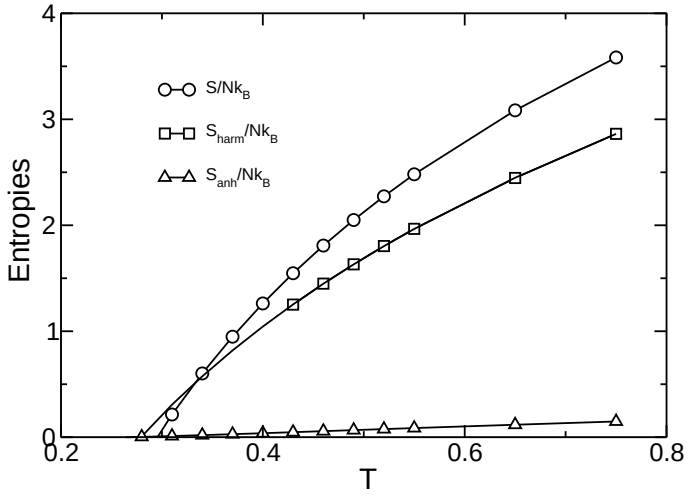

Figure 7.6: The total S, the harmonic $S_{\rm harm}$ and the anharmonic $S_{\rm anh}$ entropies as a function of T. S and $S_{\rm anh}$ are extended below $T < 0.43$, the lowest T simulated, using Eq. 7.46 and 7.52, respectively. On the other hand, $S_{\rm harm}$ is extended below $T < 0.43$ using a fit $a_{\rm harm} + b_{\rm harm}\ln(T/T^*)$, where $a_{\rm harm}$ and $b_{\rm harm}$ are free fitting parameters found to be 2.86227 and 2.89232, respectively, and $T^* = 0.75$.

vibrations within basins and transitions between basins is assumed to hold. Fig. 7.6 shows the plot of the total entropy $S(T)$ as a function of T.

7.5.3 Vibrational entropy $S_{\rm vib}$

The vibrational entropy $S_{\rm vib}$ is the sum of the harmonic $S_{\rm harm}$ and the anharmonic $S_{\rm anh}$ terms, i.e.,

$$S_{\rm vib} = S_{\rm harm} + S_{\rm anh} \tag{7.47}$$

In some systems, e.g., LJ [164], where the harmonic approximation holds, the vibrational entropy is approximated by a harmonic entropy, i.e., $S_{\rm vib} \approx S_{\rm harm}$. But

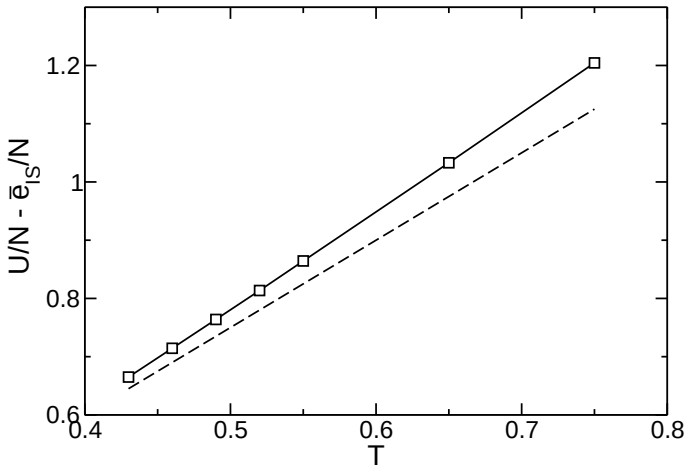

Figure 7.7: A harmonicity test. The difference between the average potential energy per particle U/N of the equilibrium liquid and the inherent structure $\bar{e}_{IS}/N$ are plotted as a function of T (open square). The dashed line is the harmonic approximation $U - \bar{e}_{IS} = \frac{3}{2}(N-1)k_BT$.

for our system, as in systems like the SPC/E model of water [96], silica [97], and OTP [184], the harmonic approximation does not hold, cf. Fig. 7.7. If the harmonic approximation were to hold, $U - \bar{e}_{IS}$ would be equivalent to $\frac{3}{2}(N-1)k_BT$. But, this is not the case for our system. Therefore, to obtain $S_{\rm vib}$ we need to calculate both the harmonic and the anharmonic contributions.

The harmonic contribution to the vibrational entropy resulting from the motion of particles within the basins is calculated using the relation given by

$$\frac{S_{\rm harm}}{k_B} = \frac{1}{N}\sum_{i=1}^{3N-3}\left[1 - \ln\left(\frac{\hbar\omega_i}{k_BT}\right)\right], \tag{7.48}$$

where $\{w_i\}$ are the square root of the eigenvalues of the Hessian matrix. As discussed above, we calculate $\{w_i\}$ from the normal mode spectrum of the liquid after quenching to the inherent structure from the equilibrium liquid at each T studied.

$S_{\rm harm}(T)/k_B$ is shown in Fig. 7.6.

The anharmonic contribution to $S_{\rm vib}$ is calculated by first evaluating the anharmonic contribution to the potential energy, $U_{\rm anh}$. Recalling the expression for the Taylor expansion of U about a local minimum, the anharmonic term corresponds to the higher order term in the expansion. Thus, $U_{\rm anh}$ can be approximated as

$$U_{\rm anh}(T) = U(T) - \bar{e}_{IS}(T) - U_{\rm harm}(T). \tag{7.49}$$

$U(T)$ and $\bar{e}_{IS}(T)$ are found from the simulation, while $U_{\rm harm}(T) = \frac{3}{2}(N-1)k_BT$. Then $S_{\rm anh}$ can be numerically calculated using the relation

$$S_{\rm anh}(T) = \int_0^T \frac{1}{T'}\frac{\partial U_{\rm harm}(T')}{\partial T'}dT'. \tag{7.50}$$

To perform the above numerical integration, the value of $U_{\rm anh}(T)$ found from the simulation is further fitted to a polynomial in T. The fit to $U_{\rm anh}$ is constrained so that $U_{\rm anh}$ and its derivative vanish at $T = 0$. This is consistent with the fact that $U_{\rm anh}$ is a correction to the harmonic approximation. We find that the functional form expressed as

$$U_{\rm anh}(T) = \sum_{k=2}^{5} c_k T^k \tag{7.51}$$

fits our data very well, with the fitting parameters c_k given by $c_2 = -0.175197$, $c_3 = 1.2112$, $c_4 = -1.57974$, and $c_5 = 0.705082$. Using Eq. 7.51, $S_{\rm anh}$ can be expressed as

$$S_{\rm anh}(T) = \sum_{k=2}^{5} \frac{kc_k}{k-1} T^{k-1}. \tag{7.52}$$

In Fig. 7.8 we show the plot of $U_{\rm anh}$ and $S_{\rm anh}$ obtained from these calculations. Additionally, $S_{\rm anh}$ is plotted together with S and $S_{\rm harm}$ in Fig. 7.6 to emphasize the extent of its contribution in the calculation of $S_{\rm conf}$. In general, as already mentioned earlier, $S_{\rm conf}(T)$ is obtained from the relation

$$S_{\rm conf}(T) = S(T) - S_{\rm harm}(T) - S_{\rm anh}. \tag{7.53}$$

Fig. 7.9 shows the plot of $S_{\rm conf}$ as a function of temperature.

7.6 Cooperatively rearranging regions and dynamical clusters

One of the goals of this work is to relate dynamic and thermodynamic properties of the Dzugutov supercooled liquid through the use of the Adam-Gibbs relation. The Adam-Gibbs theory predicts the temperature dependence of viscosity η (also diffusion coefficient D or relaxation time τ) to be expressed as [12]

$$\eta = \eta_o \exp\left[\frac{A}{TS_{\rm conf}}\right]. \tag{7.54}$$

We have shown above that $S_{\rm conf}$ can be calculated from the thermodynamic properties of the PEL of the system. On the other hand, quantities like D, η, or τ can

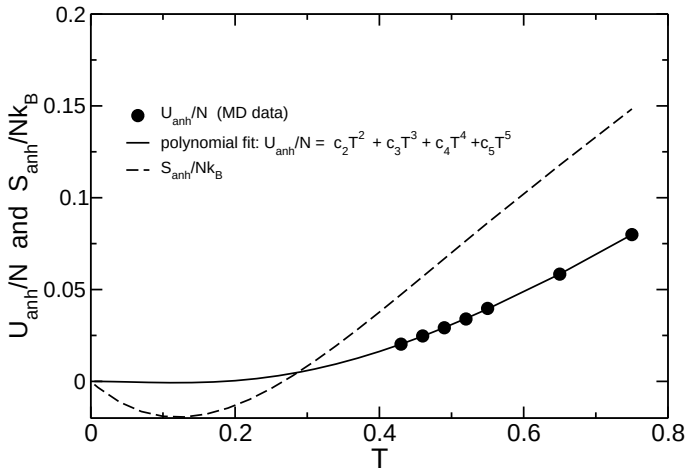

Figure 7.8: The anharmonic potential $U_{\rm anh}$ and entropy $S_{\rm anh}$ as a function of T. The filled circles represent the values of $U_{\rm anh}$ obtained from the simulation data through the relation given by Eq. 7.49. This data is fitted using Eq. 7.51, and extrapolated below the lowest T simulated, i.e., below $T = 0.43$ (solid line). The dashed line represents $S_{\rm anh}$ that is evaluated using Eq. 7.52. This equation is further used to extend $S_{\rm anh}$ below $T = 0.43$.

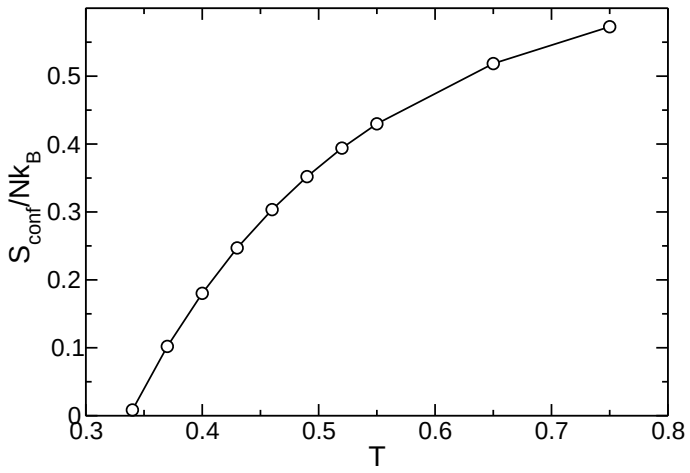

Figure 7.9: Temperature dependence of configurational entropy $S_{\rm conf}$ calculated as the difference between the total and vibrational entropy.

be calculated from dynamic studies. The Adam-Gibbs relation proposes a connection between the dynamic and thermodynamic properties. Here we are particularly interested in the connection between the cooperatively rearranging regions (CRR) envisaged by the Adam-Gibbs theory and the dynamical strings and clusters of mobile particles that we have investigated in the previous chapters.

By calculating the number average of the clusters S_n formed by the mobile particles, which is defined as

$$S_n = \frac{\sum_n nP(n)}{\sum_n P(n)} \tag{7.55}$$

where n is the number of particles in a cluster, and $P(n)$ is the probability of finding a cluster of size n, Giovambattista et al. found a linear relationship between S_n^* and $1/S_{\text{conf}}$ in the SPC/E model of water. In particular, they found that

$$S_n^* - 1 \propto \frac{1}{S_{\text{conf}}} \tag{7.56}$$

Here $S_n^* \equiv S_n(t_{\text{clu}}^{\max})$ is the average cluster S_n at the time $t_{\text{clu}}^{\max}$ when S_n is maximum. Recalling that the minimum size z^* of the CRR is inversely proportional to S_{conf}, their finding implies that S^* can be interpreted as the size of the CRR. In our work we would like to investigate the above result for the Dzugutov liquid, to examine the universality of the observation. Additionally, since strings are the elementary units of cooperativity, we wish to explore the extent to which the string size represents the CRR.

To facilitate comparison with previous work, we use the number average to define clusters or strings. The number averaged cluster size S_n and string size L_n for the Dzugutov liquid are plotted in Fig. 7.10. L_n is defined in the same way

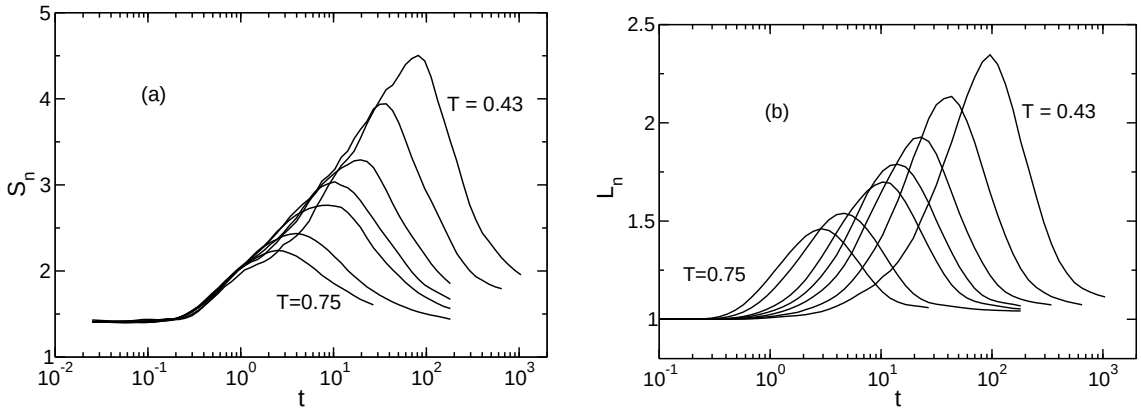

Figure 7.10: (a) The number averaged cluster size S_n and (b) the number averaged string size L_n plotted as a function of time. Note the strings are averaged over all strings including $l = 1$.

as S_n, where the cluster size is replaced by the string size. For each T, S_n^* and L_n^* are extracted from these plots, and plotted against $1/S_{\rm conf}$, cf. Fig. 7.11. As shown in the figure the linear relationship between S_n^* and $1/S_{\rm conf}$ reasonably holds, supporting the observation in simulated water [167]. Nevertheless, a similar analysis for the strings reveals that a linear relationship can also be found between L_n^* and $1/S_{\rm conf}$, making strings another candidate for the CRR. The implication of this will be explored in future work.

Another important point that we want to explore is the validity of the Adam-Gibbs theory for the Dzugutov liquid. The validity of this theory is usually tested by plotting $\log(D)$ against $1/TS_{\rm conf}$. Implicit in this relation is that $D \propto 1/\eta$. The Adam-Gibbs relation is then considered to be obeyed by data that follows a straight line on such a plot. A wide range of systems have been documented to obey this form of the Adam-Gibbs relation (see, e.g., Ref. [96, 164, 184]). To test the validity

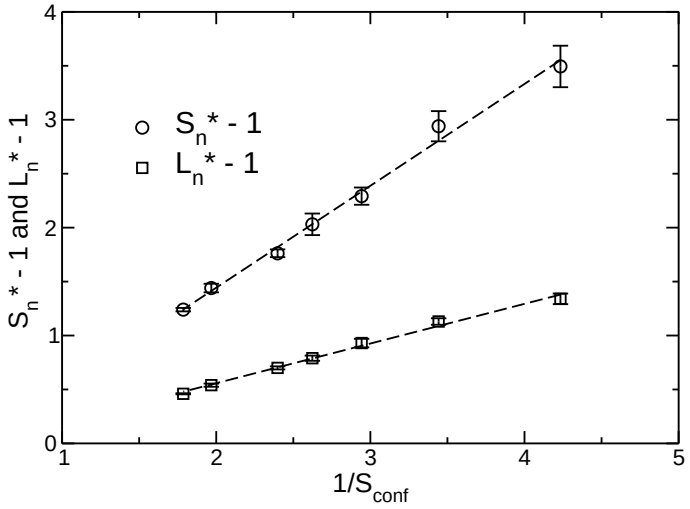

Figure 7.11: A plot of $S_n^* - 1$ and $L_n^* - 1$ against $1/S_{\rm conf}$. S_n^* is the number averaged cluster S_n at the time $t_{\rm clu}^{\rm max}$ when S_n is maximum. L_n^* is the number averaged string L_n at the time $t_{\rm str}^{\rm max}$ when L_n is maximum.

of this relation for our system we plot $\log(D)$ against $1/TS_{\rm conf}$ in Fig. 7.12(a). As indicated by the error bars, except for the lowest T, the expression does hold within statistical error. However, a careful look at the data points, even excluding the lowest T which clearly deviates from the line, indicates that there is systematic curvature. A significant curvature has also been observed for the BKS model of silica [98] when $\log(D)$ is plotted against $1/TS_{\rm conf}$, implying that the relation $D \propto \exp[-A/TS_{\rm conf}]$, where A is a constant, fails for silica.

However, it is not clear that this is the proper expression relating D to $S_{\rm conf}$. From the Stokes-Einstein (SE) relation, $D \propto T/\eta$, so that $\eta \propto T/D$. Substituting this expression for η into the original AG expression (Eq. 7.54) yields $D/T \propto \exp(-A/TS_{\rm conf})$. Thus if the so-called AG relation for D holds, D/T, and

not D, is the relevant quantity to consider. This test was performed for BKS silica [98], and this revised expression was found to be a significantly better fit than the previous expression over a range of densities. Performing the same test on our system, we observed little to no improvement, as illusrtated in Fig. 7.12.

On the other hand, the SE relation does not necessarily hold for low T when SHD is present [32], leaving us without an obvious connection between D and $S_{\rm conf}$. In fact, all experiments on the decoupling of D and η (or the rotational diffusion coefficient D_R, which is related to η through the Stokes-Einstein-Debye expression at all T, even in the glass [196]), shows D to be enhanced over the value predicted from SE for a given viscosity. This enhancement of diffusion is consistent with the deviation at low T of the diffusion constants we calculate, which are also enhanced over the values expected if both SE and AG were to hold. Thus when SE fails to hold, as it should when SHD is prominent, we have no reason to expect either $\log(D/T)$ or $\log(D)$ to be linearly proportional to $1/TS_{\rm conf}$.

In fact, the Dzugutov system has more pronounced dynamical heterogeneity than other systems studied for which the AG relation is found to hold. For example, the stretching exponent β found from the KWW fit (Eq. 2.4) for our system is $\beta \sim 0.5$ as $T \to T_{\rm MCT}$, indicating fragility and strong heterogeneity. Other systems that satisfy the AG relation (e.g. water [96], silica [97], LJ [164]), either in terms of D or D/T, have substantially larger values of β (i.e., closer to one). Also, the average cluster size is larger for the same distance from $T_{\rm MCT}$ as compared with systems like the polymer melt, binary LJ, and BKS silica. Consequently, these other studies of the AG relation may all have been conducted in temperature regimes where

SHD was not sufficiently pronounced to cause a breakdown of SE. In contrast, the strong SHD in our system at our lowest temperatures would be consistent with a breakdown in SE, and consequently with a breakdown in the expressions for D vs S_{conf} traditionally used. To clearly understand this issue beyond what we have argued above, it is necessary to independently calculate η. Further work is thus underway to calculate η for our system to obtain a direct relationship between η and D for a rigorous test of the AG theory.

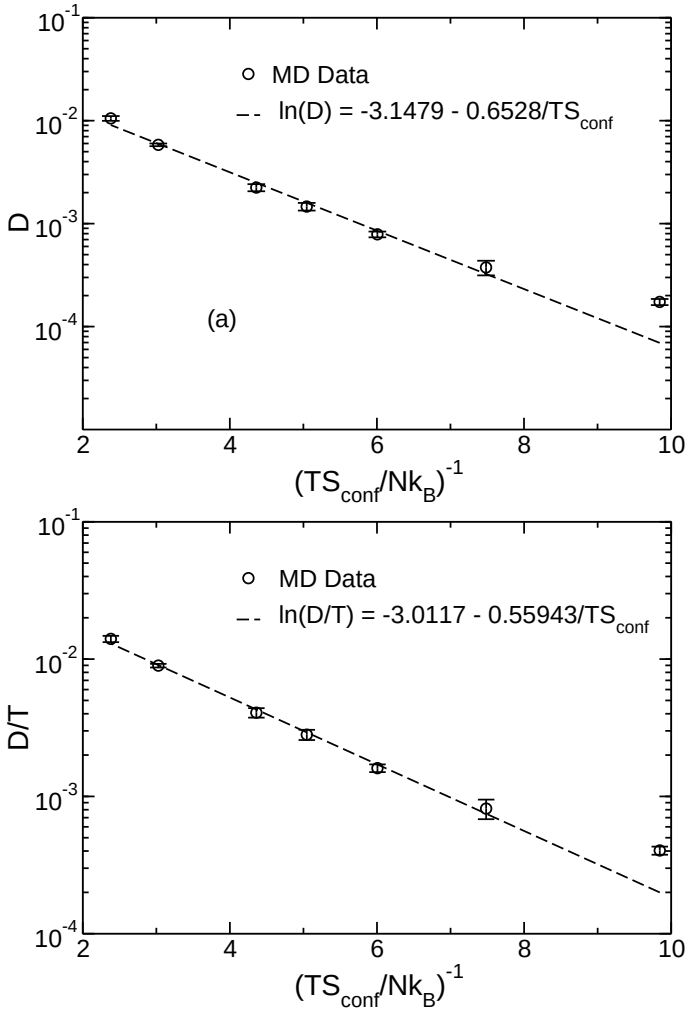

Figure 7.12: Test of the validity of the Adam-Gibbs theory for the Dzugutov liquid. (a) The diffusion coefficient D plotted against $1/S_{\rm conf}$ on a semi-logarithmic axis. (b) D/T plotted against $1/S_{\rm conf}$ on a semi-logarithmic axis.

Chapter 8

Conclusion

In this study we found that highly mobile particles aggregate together to form clusters of different sizes and shapes. The temperature dependence of the mean cluster size S is found to show a similar behavior as in the LJ system in that S is found to increase upon cooling towards T_{MCT}. We found that the mean cluster size "grows" and "shrinks" with time. The time $t_{\mathrm{clu}}^{\mathrm{max}}$ at which S is maximum is found to be in the late-β/early-α relaxation regime, which corresponds to the time when the mean square displacement crosses over from the plateau or "caging" regime to the subdiffusive regime, in the case of the polymer melt, or to the diffusive regime, in the case of the Dzugutov liquid. Because this crossover marks the time scale when the particles are most likely to escape their cages, this correspondence strongly suggests that clustering is required for the cage-breaking process. On the other

hand, since $t_{\rm clu}^{\rm max}$ is much smaller than the α-relaxation time τ_α, which marks the time scale for the structural relaxation of the system, our studies suggest that a collection of cooperative molecular rearrangements give rise to the eventual primary relaxation of the system consistent with the prediction of MCT. Our results of the clustering phenomena are also consistent with confocal microscopy experiments on dense colloidal suspensions, which confirmed the dynamic nature of the clustering of mobile particles [21].

In all the systems we studied, as well as the LJ system [76] studied before, the cluster size distribution $P(n)$ at the peak times, and for the lowest T studied, is found to follow a power law behavior $P(n) \sim n^{-\tau}$ with an exponent $\tau < 2$, whereas classical percolation theory implies $\tau > 2$ [156]. This discrepancy was thought to be partly due to the finite system size of the polymer melts, which has relatively smaller size that may restrict n to only small clusters, and hence skew the estimated exponent in an uncontrolled way. But after finding a similar exponent for the large system of the Dzugutov liquid, we now believe that the value of τ may simply be different for that expected from percolation theory for *static* clusters of randomly distributed particles, since the clusters we study are intrinsically *dynamic* (i.e. a dynamic criterion is used to define the particles that make up the clusters, although a static "snapshot" is used to analyze the clusters). However, we would like to point out that even for static clusters there is experimental evidence for $\tau = 1.4 \pm 0.15$ [197].

A closer inspection of the clusters revealed that groups of mobile particles within a cluster follow one another in quasi one-dimensional paths, forming strings of different sizes, suggesting that strings are the elementary unit of cooperativity.

The analysis of these strings showed that the transient nature and temperature dependence of the mean string length L is similar to that of S. The time $t_{\mathrm{clu}}^{\mathrm{max}}$ where $L(t)$ is maximum occurs at a time close to $t_{\mathrm{clu}}^{\mathrm{max}}$, and L increases with decreasing T. However, in contrast to what was found for the clusters, the probability distribution of the string length decays exponentially. This was found to be the case for both the polymer melt and the Dzugutov liquid, as is found in the LJ system.

To gain insight on the nature of cooperativity at the microscopic level, we investigated the detailed motion of particles in strings. Understanding the detailed nature of the string-like motion has relevance in new, emerging as well as established theories. Thus, we conducted a detailed analysis of the formation of strings in the Dzugutov liquid by investigating the mechanisms that lead to the emergence of these dynamical objects. For this purpose, we identified strings at a time when the string-like motion is most pronounced as measured by their mean value, and traced the trajectories of the particles in the strings during this period of time. By inspecting the rearrangement of particles within strings we found that the formationof strings is a complex process that involves several mechanisms at the time scale shorter than the late-β/early-α relaxation regimes. They are formed as a result of coherent motion, where several particles move together as a unit, and non-coherent motion, where individual particles or segments of the string known as microstrings move at different times, but along the same path. Statistically, we found that coherent motion is prevalent in small strings, and over all roughly only less than a quarter of the strings move coherently. The non-coherent strings are found to exhibit particle motion that is both sequential, i.e., motion in which particles move in a strict order

of head to tail along the backbone of the strings, but at widespread time intervals, and non-sequential or random motion. Thus, for most non-coherent strings, we find that individual particles or microstrings move in a non-sequential manner. In particular, sequential motion is usually not observed for long strings. However, simultaneous (coherent) motion is observed within small sub-units of the strings, referred to as microstrings.

The present findings for the mechanism of string-like motion are consistent with the outcome of a study on the particle rearrangements resulting from the transitions between successively visited inherent structures of a binary LJ liquid [182]. There, it was observed that long strings identified after sequences of transitions do not result from a coherent motion of all particles during a single transition, but instead the particles replace each other at different times either in single-particle type motion or in small coherent microstrings.

Altogether, the following picture appears to emerge: On very short time scales, small groups of particles (microstrings) move together and, hence, the length scale of cooperative motion is small. At longer times, the interplay of these individual motions leads to the formation of larger and larger strings, which in turn aggregate into clusters. Within these dynamical objects, particles assist each other to escape from their respective cages and the length scale of cooperative motion becomes maximum at times in the late-β/early-α relaxation regime. At even later times, the cooperativity diminishes due to the independent diffusion of the particles. Along these lines, it may be suggested that – in analogy to the formation of the strings due to the concerted motion of smaller sub-units – the growth of the clusters may

be a consequence of the interplay of various strings. In other words, cooperativity on various time scales and length scales may be the basis of spatially heterogenous dynamics.

One may ask why the small length scale associated with the cooperativity of particle motion observed at short time scales increases and become organized into larger, quasi-one dimensional objects in the late-β/early-α relaxation regime. Very recently Chandler, Garrahan and co-workers [60, 61] proposed that dynamics in a supercooled liquid can be understood via two simple ingredients, namely, the existence of spatially heterogeneous dynamics and the facilitation of dynamics in the vicinity of regions exhibiting high particle mobility. The mechanism for the formation of large strings resembles the concept of dynamic facilitation. In particular, one may speculate that the local excitations envisaged by Garrahan and Chandler are associated with the coherent motion of particles within microstrings, thereby facilitating the creation of neighboring excitations that extend throughout the string.

Another possible reason for string-like motion lies in the local static structure of a supercooled liquid. In a numerical study of a 2D monodisperse system of particles, Reichardt and Reichardt [198] found fluctuating topological defects that form a string-like structure. Although this observation has to be further investigated for a 3D system such as ours, weak one-dimensional fissures may possibly develop in the liquid, providing a path for string-like motion. Dzugutov et al. [103] established a relation between dynamical heterogeneity and structural heterogeneity for the studied model liquid. In particular, it was demonstrated that clusters of icosahedrally coordinated particles exist in the Dzugutov liquid where, on average, the particles

inside and outside of an icosahedral environment show reduced and enhanced mobilities, respectively. Hence, one may speculate that string-like motion, at least in the Dzugutov liquid, is most pronounced in non-icosahedral environments, e.g., in channels between clusters of icosahedrally coordinated particles. Indeed, preliminary results [199] suggest that string-like motion occurs primarily along the edges of such clusters. Further work along these lines is in progress.

In general, several theories attempt to explain the slowing down of dynamics. So far a comprehensive theory that explains all the salient features of glass-forming liquids is not yet available. Some theories, however, have been highly useful in explaining certain aspects of the glass transition. For example, the initial slowing down of dynamics at T well above T_g can be described to a large extent via MCT, which predicts diverging relaxation time at T_{MCT}. The phenomenological view point of the Adam-Gibbs theory, which predicts the increasing range of cooperatively rearranging regions as a cause for slowing dynamics, is another example. The topological view point of Stillinger and Weber based on the inherent structure formalism is also a very useful tool for understanding the thermodynamic properties of glass-forming liquids, while the new emerging non-topographic view point of Chandler and Garrahan that is based on the concept of dynamic facilitation is emerging to explain some aspects of the correlated motion. We thus believe that our detailed analysis in this thesis, where the nature of correlated motion has been investigated in detail, may contribute to the development of new theories that are more comprehensive, or to the reworking of for the well-established theories like the AG theory. In this regard, our own ongoing effort to connect the dynamic and thermodynamic properties of

glass-forming liquids through the AG theory is one good example.

Appendix A

Molecular dynamics simulation code

This section documents the molecular dynamics code used to simulate the Dzugutov system. The program consists of several c-files and h-files that should be compiled together. It is a force-decomposition parallel code using MPI and is designed to simulate the system using linked list cells with embedded neighbor list. The program is modified from a simulation code prepared for a LJ system, originally written by Francis Starr.

```
/*
 * lj.c:
 *
 * This part contains the "main" of the program.
 *
 *
 */

#include <time.h>
```

```
#include <stdio.h>
#include <math.h>
#include "lj.h"

#include "mpi.h"

void main(int argc, char **argv)
{
  double **x, **v, **f, *floc, *ftot;
  cell *cells;
  list *nlist;
  double mass;
  double temperature, T, pressure, P, density;
  double potential, kinetic, virial;
  double potlocal, virlocal;
  double L, volume, dt;
  double cutoff, cutoff2, minCutoff, skin, permit;
  int t, tmax, write_time, restart, Tconstant;
  char inputfile[50], datafile[50], restartfile[50], configfile[50];
  int *timeList;
  int simu;
  char timefile[50];
  unsigned short seed[3];
  int ncell, ncell3;
  int rank, np;
  int i, j, k;
  int time_index = 0, block = 0, latest_time, output_time;
  int Number_of_times, ascii_flag;
  double time;


  MPI_Init(&argc, &argv);
  time = MPI_Wtime();
  MPI_Comm_rank(MPI_COMM_WORLD, &rank);
  MPI_Comm_size(MPI_COMM_WORLD, &np);


  /* get input file name or set default */
  /*
  if (argc>1)
    strcpy(inputfile, argv[1]);
  else
  */

for (simu=100; simu<200; simu++){ /* for producing independent
                                     configurations*/
sprintf(inputfile, "simu2197_%d/lj.inp",simu);
printf(inputfile);
     /*strcpy(inputfile, "lj.inp");*/

   read_input(&restart, &tmax, &write_time, &Number_of_times,
                  &Tconstant, &T, &pressure, &density, &dt, seed,
                  &ascii_flag, inputfile, timefile, datafile,
                  restartfile, configfile);
```

```
   timeList = (int *) calloc(Number_of_times, sizeof(int));

   read_timefile(timeList, Number_of_times, timefile);

   /* allocate memory */
   x = (double**) calloc(N, sizeof(double *));
   v = (double**) calloc(N, sizeof(double *));
   f = (double**) calloc(N, sizeof(double *));
   floc = (double*) calloc(N*NDIM, sizeof(double));
   ftot = (double*) calloc(N*NDIM, sizeof(double));
   for (i=0; i<N; i++) {
       x[i] = (double*) calloc(NDIM, sizeof(double));
       v[i] = (double*) calloc(NDIM, sizeof(double));
       f[i] = (double*) calloc(NDIM, sizeof(double));
   }

   init_simulation(&mass, &cutoff, &minCutoff, &cutoff2,
                                              &skin, &permit);
   volume = ((double)N)*mass/(N_A*density);

   if (restart) {
       read_restartfile(&t, &temperature, &pressure, &density,
                        &time_index, &latest_time, &block,
                        x, v, &L, mass, restartfile);

     printf("time_index\t%d\n", time_index);
     printf("latest_time\t%d\n", latest_time);
     printf("time\t%d\n",t);

/* Task_0: use only for new T restart */

     latest_time = 0;
     block = 0;
     t = 0;
     time_index = 0;

/* end of Task_0 */

     output_time = timeList[time_index] + latest_time;
     printf("output_time\t%d\n", output_time);

     kinetic = 0.0;
     for (i=0; i<N; i++)
         kinetic += mass*(v[i][0]*v[i][0] + v[i][1]*v[i][1]
                                          + v[i][2]*v[i][2]);
       kinetic *= 0.5;
   }
   else {
       init_lattice(x, v, density, mass, T, seed, &kinetic, &L);
       t=0;
       latest_time = 0;
       output_time = timeList[0];
   }
```

```
t++;

/* Calculate number of cells and cell size */
ncell = (int) (L/(sqrt(skin)));
printf("\nSplit Computational cell into %d cells
                               per dimension\n", ncell);
if (ncell < 3) {
    printf("\nCell too small!  Must split into at
                            least 3 per dimension\n");
    exit(1);
}

 ncell3 = ncell*ncell*ncell;

/* allocate cells and give them capacity to hold
                                     many mocecules */
cells = (cell *) calloc(ncell3, sizeof (cell));
for (i=0; i<ncell3; i++) {
     cells[i].last = 0;
     cells[i].molecules = (int *) calloc(10*N/ncell3,
                                                sizeof(int));
     for (j=0; j<10*N/ncell3; j++)
       cells[i].molecules[j] = 0;
}


/* allocate neighbor list space */
nlist = (list*) calloc(N, sizeof(list));
for (i=0; i<N; i++) {
    nlist[i].last = 0;
    /* since memory is cheap, over allocate to avoid
                                      ever overloading */
    nlist[i].neighbors =  (int*)
      calloc((int)(40.0/3.0*PI*skin*sqrt(skin)*density/mass),
                                                  sizeof(int));
}

init_cells(cells, ncell, ncell3);
load(x, cells, L, ncell, ncell3);
neighbor_list(nlist, x, cells, L, skin, permit, ncell3,
                                                   rank, np);

/* must start with the forces for velocity verlet */

/* begin = time(NULL); */

force(x, floc, cells, nlist, L, ncell3, cutoff, minCutoff,
                      cutoff2, &potlocal, &virlocal, rank, np);

/* MPI_Barrier(MPI_COMM_WORLD); */
MPI_Reduce(&potlocal, &potential, 1, MPI_DOUBLE, MPI_SUM, 0,
                MPI_COMM_WORLD);
MPI_Reduce(&virlocal, &virial, 1, MPI_DOUBLE, MPI_SUM, 0,
                MPI_COMM_WORLD);
```

```
for (i=0; i<N*NDIM; i++)
      ftot[i] = 0.0;
/* MPI_Barrier(MPI_COMM_WORLD); */
MPI_Allreduce(floc, ftot, N*NDIM, MPI_DOUBLE, MPI_SUM,
                  MPI_COMM_WORLD);
for (i=0; i<N; i++)
    for (k=0; k<NDIM; k++)
        f[i][k] = ftot[i*NDIM+k];


for (t; t<=tmax; t++) {

     verlet1(x, v, f, mass, L, dt);

     load(x, cells, L, ncell, ncell3);

     neighbor_list(nlist, x, cells, L, skin, permit, ncell3,
                                                    rank, np);

     force(x, floc, cells, nlist, L, ncell3, cutoff, minCutoff,
                         cutoff2, &potlocal, &virlocal, rank, np);

     /* elapse_time = (end - begin)/(double)CLOCKS_PER_SEC;
      printf("elapsed_time = %lf\n", elapse_time); */


      /* MPI_Barrier(MPI_COMM_WORLD);*/
     MPI_Reduce(&potlocal, &potential, 1, MPI_DOUBLE, MPI_SUM, 0,
                  MPI_COMM_WORLD);
     MPI_Reduce(&virlocal, &virial, 1, MPI_DOUBLE, MPI_SUM, 0,
                  MPI_COMM_WORLD);
     for (i=0; i<N*NDIM; i++)
    ftot[i] = 0.0;
      /* MPI_Barrier(MPI_COMM_WORLD); */
      MPI_Allreduce(floc, ftot, N*NDIM, MPI_DOUBLE, MPI_SUM,
                       MPI_COMM_WORLD);
     for (i=0; i<N; i++)
    for (k=0; k<NDIM; k++)
f[i][k] = ftot[i*NDIM+k];


      verlet2(v, f, mass, dt, &kinetic, Tconstant, T);

     /* write_restartfile(t, temperature, pressure, density,
             time_index, latest_time, x, v, restartfile, restart); */

    /* begin  = clock();*/
    /* if (t >= 2000){ */
        if (rank==0) {

         if (t%write_time==0) {
          write_data(t, dt, kinetic, potential, virial,
                        &temperature, &pressure, volume, datafile,
                         restart);
```

```
            }

            if (t==output_time) {

            if(ascii_flag){
             /* sprintf(configfile, "xyz_t=%d.%d.dat", block,
                                                 output_time); */
              sprintf(configfile, "simu2197_%d/xyz_t=%d.%d.dat",
                                   simu, block,timeList[time_index]);
            }

            /* write_data(t, dt, kinetic, potential, virial,
                          &temperature,&pressure, volume, datafile,
                                                      restart);*/

             write_configuration(t, ascii_flag, x, v, configfile);

             time_index++;

             if(time_index == Number_of_times){
                 time_index = 0;
                 latest_time = t;
                 block++;
             }


             /* printf("LATESTTIME\t%d\n", latest_time); */

             output_time = timeList[time_index] + latest_time;
         /* } */

         write_restartfile(t, temperature, pressure, density,
                                   time_index, latest_time, block,
                                         x, v, restartfile, restart);

             }
     }


  }


   if (rank == 0) {
     printf("Time taken per processor: %lf\n",
                                        MPI_Wtime()-time);
     printf("\nALL DONE!\n");
   }
    } /* for (simu=0....)*/
    MPI_Finalize();
}


=======================================================
```

```
/*
 *
 * init.c
 *
 *
 */


#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "define.h"
#include "extern.h"

void init_simulation(double *mass, double *cutoff,
                     double *minCutoff, double *cutoff2,
                     double *skin, double *permit)
{
  double sigma, epsilon;

  /* set masses in amu */
  *mass = 1.0;

  /* set Lennard-Jones paramaters */
  sigma    = 1.0;
  epsilon  = 1.0;

  /* *cutoff  =  2.5*sigma; */
  *cutoff  =  1.94*sigma;
  *cutoff2 = (*cutoff)*(*cutoff);
  *minCutoff = 1.87*sigma;
  *skin    = 1.1*(*cutoff);
  *permit  = 0.25*(*skin-*cutoff)*(*skin-*cutoff);
  *skin    *= *skin;

}

/* Initialization of cubic lattice. */
void init_lattice(double **x, double **v, double density,
                  double mass, double temperature,
                  unsigned short seed[3], double *kinetic,
                                                double *L)
{
  double spacing, L2;
  double v0;
  double vavg[NDIM];
  int i, j, k, ijk, N3;


  double randNorm;

  *L       = pow((double)(N)*mass/(N_A*density), 1.0/3.0);
  N3       = ceil(pow((double)N, 1.0/3.0));
  spacing  = *L/(double)N3;
```

```
  L2       = -(*L+spacing)/2.0;
  v0       = sqrt(3.0*k_B*temperature/mass);
  *kinetic = 0.0;

  randNorm = pow((double)2,(double)31);

  printf("System Size L =  %f\n", *L);

  vavg[0] = vavg[1] = vavg[2] = 0.0;

  ijk = 0;
  for (i=1; i<=N3; i++)
    for (j=1; j<=N3; j++) {
      for (k=1; k<=N3; k++) {
/* molecules placed on a cubic lattice */
x[ijk][0] = (double)i*spacing + L2;
x[ijk][1] = (double)j*spacing + L2;
x[ijk][2] = (double)k*spacing + L2;

v[ijk][0] = (2.0*erand48(seed)-1.0)*v0;
v[ijk][1] = (2.0*erand48(seed)-1.0)*v0;
v[ijk][2] = (2.0*erand48(seed)-1.0)*v0;


/* accumulate average velocities to insure no
                                          net momentum */
vavg[0] += v[ijk][0];
vavg[1] += v[ijk][1];
vavg[2] += v[ijk][2];

ijk++;
      }
    }

  for(k=0; k<NDIM; k++)
    for(i=0; i<N; i++) {
      v[i][k] -= vavg[k]/(double)N;
      x[i][k] += v[i][k]*0.001;
    }


  for (i=0; i<N; i++)
    *kinetic += mass*(v[i][0]*v[i][0] + v[i][1]*v[i][1]
                                      + v[i][2]*v[i][2]);
  *kinetic *= 0.5;

  printf("inital temp = %lf\n",
              *kinetic*2.0/((double)(NDIM*N-NDIM)*k_B));

}


=========================================================
```

```
/*
 * cell.c
 *
 *
 */

/*#include <mpi.h>*/
#include <math.h>
#include <stdio.h>
#include "cell.h"
#include "extern.h"

void
init_cells_slow (cell * cells, int ncell, int ncell3)
{
  int i, j, k, c, cn, x, y, z, n;
  int ncell2;

  ncell2 = ncell * ncell;

  /* loop over cells */
  for (i = 0; i < ncell; i++)
    for (j = 0; j < ncell; j++)
      for (k = 0; k < ncell; k++)
{

  /* identify current cell */
  c = i * ncell2 + j * ncell + k;

  /* always interact within cell */
  cells[c].neighbors[0] = c;

  /* collect near neighbors of cell c */
  n = 1;
  x = 1;
  for (y = -1; y <= 1; y++)
    for (z = -1; z <= 1; z++)
      {
cn = ((i + x + ncell) % ncell) * ncell2
                    + ((j + y + ncell) % ncell) * ncell
                    + (k + z + ncell) % ncell;
cells[c].neighbors[n] = cn;
n++;
      }

  x = 0;
  y = 1;
  for (z = -1; z <= 1; z++)
    {
      cn = ((i + x + ncell) % ncell) * ncell2
                  + ((j + y + ncell) % ncell) * ncell
                  + (k + z + ncell) % ncell;
      cells[c].neighbors[n] = cn;
      n++;
```

```
    }

  y = 0;
  z = 1;
  cn = ((i + x + ncell) % ncell) * ncell2
               + ((j + y + ncell) % ncell) * ncell
               + (k + z + ncell) % ncell;
  cells[c].neighbors[n] = cn;
  n++;
}

}

void
init_cells (cell * cells, int ncell, int ncell3)
{
  register int i; /* counters */
  register int j;
  register int k;
  register int c;
  register int itemp1;  /* temp store */
  register int itemp2;
  register int itemp3;
  register int itemp4;
  register int itemp5;
  register int itemp6;
  register int itemp7;
  register int itemp8;
  register int itemp9;
  register int ncell2;

  ncell2 = ncell * ncell;

  /* loop over cells */
  for (i = 0; i < ncell; i++)
    for (j = 0; j < ncell; j++)
      for (k = 0; k < ncell; k++)
{

  /* identify current cell */
  c = i * ncell2 + j * ncell + k;

  /* dependent on c */
  /* next 6 statements can run simultaneously */
  /* always interact within cell */
  cells[c].neighbors[0] = c;

  /* collect near neighbors of cell c */
  itemp1 = ((i + 1 + ncell) % ncell) * ncell2;
  itemp2 = ((j - 1 + ncell) % ncell) * ncell;
  itemp3 = (k - 1 + ncell) % ncell;
  itemp4 = (k + ncell) % ncell;
  itemp5 = (k + 1 + ncell) % ncell;
```

```
  /* dependency on itemp3, itemp4, itemp5 */
  /* itemp1 should have been computed by this point */
  itemp7 = itemp1 + itemp3;
  itemp8 = itemp1 + itemp4;
  itemp9 = itemp1 + itemp5;

  /* these 3 statements can run simultaneously */
  cells[c].neighbors[1] = itemp7 + itemp2;
  cells[c].neighbors[2] = itemp8 + itemp2;
  cells[c].neighbors[3] = itemp9 + itemp2;

  /* dependency on itemp2 */
  itemp2 = ((j + ncell) % ncell) * ncell;
  /* next 3 statements can run simultaneously */
  cells[c].neighbors[4] = itemp7 + itemp2;
  cells[c].neighbors[5] = itemp8 + itemp2;
  cells[c].neighbors[6] = itemp9 + itemp2;

  /* dependency on itemp6 */
  itemp6 = ((j + 1 + ncell) % ncell) * ncell;
  /* next 3 statements can run simultaneously */
  cells[c].neighbors[7] = itemp7 + itemp6;
  cells[c].neighbors[8] = itemp8 + itemp6;
  cells[c].neighbors[9] = itemp9 + itemp6;

  /* dependency on itemp1 */
  itemp1 = ((i + ncell) % ncell) * ncell2;

  /* next 3 statements can run simultaneously */
  itemp7 = itemp1 + itemp3;
  itemp8 = itemp1 + itemp4;
  itemp9 = itemp1 + itemp5;

  /* next 4 statements can run simultaneously */
  cells[c].neighbors[10] = itemp7 + itemp6;
  cells[c].neighbors[11] = itemp8 + itemp6;
  cells[c].neighbors[12] = itemp9 + itemp6;

  cells[c].neighbors[13] = itemp9 + itemp2;
}

}


void
load (double **x, cell * cells, double L, int ncell, int ncell3)
{
  register int i; /* counters */
  register int j;
  register int c;
  register int ncell2; /* ncell^2 */
  double L2; /* L / 2 */
  double size;
```

```
  ncell2 = ncell * ncell;
  L2 = L / 2.0;
  size = L / (double) ncell;

  /* first empty the cells */
  for (i = 0; i < ncell3; i++)
    {
      for (j = 0; j < cells[i].last; j++)
cells[i].molecules[j] = 0;
      cells[i].last = 0;
    }

  /* now put particles into cells */
  for (i = 0; i < N; i++)
    {
      c = (int) ((x[i][0] + L2) / size) * ncell2
         + (int) ((x[i][1] + L2) / size) * ncell
         + (int) ((x[i][2] + L2) / size);
      cells[c].molecules[cells[c].last] = i;
      cells[c].last++;
    }

}

=======================================================

/*
 *
 * nlist.c
 *
 *
 */

#include <math.h>
#include "nlist.h"
#include "cell.h"
#include "extern.h"

#define anint(x) ((x >= 0.5) ? (1.0) : (x <= -0.5) ? (-1.0) : (0.0))
#define magic 6755399441055744.0

void neighbor_list(list *nlist, double **x, cell *cells, double L,
   double skin, double permit, int ncell3, int rank,
                                                              int np)
{
  int i, j, k, c, ii, jj, n, nbr;
  double *ri, *rj, r[NDIM], r2, invL;
  static double **xold;
  static int firsttime = 1;

  if (firsttime) {
    firsttime = 0;
    xold = (double**) calloc(N, sizeof(double*));
    for (i=0; i<N; i++)
```

```
      xold[i] = (double*) calloc(NDIM, sizeof(double));
  }

  invL = 1.0/L;

  /* test if we need to build neighbor list */
  for (i=0; i<N; i++) {
    r2 = 0.0;
    for (k=0; k<NDIM; k++) {
      r[k] = x[i][k]-xold[i][k];
      /*r[k] -= L*(r[k]*invL+magic-magic);*/
      r[k]  -= L*anint(r[k]*invL);
      r2   += r[k]*r[k];
    }
    if(r2 > permit)
      goto L100;
  }

  return;

  /* build neighbor list */
 L100:

  for (i=0; i<N; i++) {
    xold[i][0] = x[i][0];
    xold[i][1] = x[i][1];
    xold[i][2] = x[i][2];
  }

  /* empty old list */
  for (i=0; i<N; i++)
    nlist[i].last = 0;

  for (c=rank*ncell3/np; c<(rank+1)*ncell3/np; c++) {
    if (c<ncell3) {   /* extra careful... */

      /* find neighbors within a cell */
      for (i=0; i<cells[c].last-1; i++) {
ii = cells[c].molecules[i];
ri = x[ii];

for (j=i+1; j<cells[c].last; j++) {
  jj = cells[c].molecules[j];
  rj = x[jj];

  r2 = 0.0;

  for (k=0; k<NDIM; k++) {
    r[k]  = ri[k] - rj[k];
    r[k] -= L*anint(r[k]*invL);
    /*r[k] -= L*(r[k]/L+magic-magic);*/
    r2   += r[k]*r[k];
  }
```

```
  if (r2 <= skin) {
    nlist[ii].neighbors[nlist[ii].last] = jj;
    nlist[ii].last++;
  }

}
      }

      /* now find neighbors in bordering cells */
      for(n=1; n<NBRS; n++) {
nbr = cells[c].neighbors[n];

for (i=0; i<cells[c].last; i++) {
  ii = cells[c].molecules[i];
  ri = x[ii];

  for (j=0; j<cells[nbr].last; j++) {
    jj = cells[nbr].molecules[j];
    rj = x[jj];

    r2 = 0.0;
    for (k=0; k<NDIM; k++) {
      r[k]  = ri[k] - rj[k];
      r[k] -= L*anint(r[k]*invL);
      /*r[k] -= L*(r[k]/L+magic-magic);*/
      r2   += r[k]*r[k];
    }

    if (r2 <= skin) {
      nlist[ii].neighbors[nlist[ii].last] = jj;
      nlist[ii].last++;
    }
  }
}
      }   /* end loops over neighbor cells */
    }
  }  /* end loop over cells */

}


=======================================================

/*
 *
 * force.c
 *
 *
 */

#include <math.h>
#include <stdio.h>
#include "cell.h"
#include "nlist.h"
```

```
#include "extern.h"

#define anint(x) ((x >= 0.5) ? (1.0) : (x <= -0.5) ? (-1.0) : (0.0))
#define magic 6755399441055744.0

void pforce(double **x, double *f, cell *cells, list *nlist, double L,
    int ncell3, double cutoff, double minCutoff, double cutoff2,
            double *potential, double *virial, int c);

void force(double **x, double *f, cell *cells, list *nlist, double L,
   int ncell3, double cutoff, double minCutoff, double cutoff2,
           double *potential, double *virial, int rank, int np)
{
  int i, c;
  double *flocal;
  double pot, vir;

  *potential = *virial = 0.0;
  for (i=0; i<N*NDIM; i++)
    f[i] = 0.0;

  flocal = (double*) calloc(N*NDIM, sizeof(double));
  pot = vir = 0.0;

  for (c=rank*ncell3/np; c<(rank+1)*ncell3/np; c++) {
    if (c<ncell3)
      pforce(x, flocal, cells, nlist, L, ncell3, cutoff, minCutoff,
             cutoff2, &pot, &vir, c);
  }
  *potential += pot;
  *virial    += vir;
  for (i=0; i<N*NDIM; i++)
    f[i] += flocal[i];
  free(flocal);
}

void pforce(double **x, double *f, cell *cells, list *nlist, double L,
    int ncell3, double cutoff, double minCutoff,
            double cutoff2, double *potential, double *virial, int c)
{
  register double *ri, *rj, *fi, *fj, invr, invr2, invr8, invr16, invr17;
  double pbc[NDIM], r, r2;
  double rij[NDIM], vir[NDIM];
  register double pot;
  register int i, ii, j, jj, k;
  static int firsttime = 1;
  static double invL;

  register double item0, item1, item2, item3, item4, item5;
  register double term0, term1, term2, term3;
  double pot_term1, pot_term2;

 if (firsttime) {
    firsttime = 0;
```

```
    invL = 1.0/L;
  }

  /* first handle interactions within the cell */
  for (i=0; i<cells[c].last; i++) {
    ii = cells[c].molecules[i];
    ri = x[ii];
    fi = f+ii*NDIM;

    /* loop over neighbor list of ii */
    for (j=0; j<nlist[ii].last; j++) {
      jj = nlist[ii].neighbors[j];
      rj = x[jj];
      fj = f+jj*NDIM;

      r2 = 0.0;

      for (k=0; k<NDIM; k++) {
rij[k]  = ri[k] - rj[k];
pbc[k]  = L*anint(rij[k]*invL);
/*pbc[k]  = L*(rij[k]*invL+magic-magic);*/
rij[k] -= pbc[k];
r2   += rij[k]*rij[k];
      }

      if (r2 <= cutoff2) {

        term2 = 0.;
        pot_term1 = 0.;
vir[0] = vir[1] = vir[2] = 0.0;
        r      = sqrt(r2);

        while (r < minCutoff) {
            invr   = 1.0/r;
            invr2  = 1.0/r2;
            invr8  = invr2*invr2*invr2*invr2;
            invr16 = invr8*invr8;
            invr17 = invr*invr16;
            invr16 -= 1.28;
            invr17 *= 93.12;

            item0 = r - 1.87;
            item1 = 1.1/item0;
            item2 = 6.402/(item0*item0);

            term0 = invr17 + invr16*item2;
            term1 = exp(item1);
            term2 = term1*term0;

            pot_term1 = 5.82*invr16*term1;
            break;
         }

        item3 = r - 1.94;
```

```
        item4 = 0.27/item3;
        item4 = exp(item4);
        item5 = 0.3456/(item3*item3);

        term3 = item4*item5;

        pot_term2 = 1.28*item4;

        /* Lennard-Jones contribution
        *potential += sigma12p*invr12 - sigma6p*invr6 +
            lj_pshift  + (r-cutoff)*lj_fshift;
        pot = (sigma12f*invr12 - sigma6f*invr6) * invr2 - lj_fshift*invr;
*/

        *potential += pot_term1 + pot_term2;

        pot = (term2 + term3);

        pot = pot/r;

for (k=0; k<NDIM; k++) {
  fi[k] += pot * rij[k];
  fj[k] -= pot * rij[k];
  vir[k]   += pot * rij[k];
}

for (k=0; k<NDIM; k++)
  *virial += vir[k]*rij[k];

      }
    }
  }

}
```

```
============================================================
```

```
/*
 *
 * move.c
 *
 *
 */

#include <math.h>
#include "define.h"
#include "extern.h"

#define tau 1.0
#define anint(x) ((x >= 0.5) ? (1.0) : (x <= -0.5) ? (-1.0) : (0.0))

/* velocity verlet algorithm */
```

```
void verlet1(double **x, double **v, double **f, double mass,
             double L, double dt)
{
  int i, k;
  double dt1, dt2, invL;

  dt1 = 0.5*dt;
  dt2 = 0.5*dt*dt;
  invL = 1.0/L;


  for (i=0; i<N; i++){

    for (k=0; k<NDIM; k++) {

x[i][k] += v[i][k]*dt + f[i][k]/mass*dt2;
v[i][k] += dt1*f[i][k]/mass;
x[i][k] -= L*anint(x[i][k]*invL);
      }


}
/* printf("sum:\t%lf\n",sqrt(sum/N));//test */

}

void verlet2(double **v, double **f, double mass, double dt,
     double *kinetic, int Tconstant, double T)
{
  int i, k;
  double dt1, lambda, vcm[NDIM];

  dt1 = 0.5*dt;

  /* compute Berendsen scaling factor for velocities */
  if (Tconstant) {
    *kinetic *= 2.0/((double)(NDIM*N-NDIM)*k_B);
    lambda = sqrt(1.0 + dt/tau*(T/(*kinetic)-1.0));
  }
  else lambda = 1.0;

  *kinetic = 0.0;

  for(i=0; i<N; i++)
    for (k=0; k<NDIM; k++) {
      v[i][k] += f[i][k]/mass*dt1;
      v[i][k] *= lambda;
      *kinetic   += mass*v[i][k]*v[i][k];
    }
  *kinetic *= 0.5;

  /* take out center of mass velocity (because Berendsen
                                     does not conserve) */
  vcm[0] = vcm[1] = vcm[2] = 0.0;
```

```
  for(i=0; i<N; i++)
    for(k=0;k<NDIM; k++)
      vcm[k] += v[i][k];
  for(i=0; i<N; i++)
    for(k=0; k< NDIM; k++)
      v[i][k] -= vcm[k]/((double)N);
}


=============================================================


/*
 *
 * io.c
 *
 *
 */


#include <stdio.h>
#include <math.h>
#include "define.h"
#include "extern.h"

void read_input(int *restart, int *tmax, int *write_time,
                int *Number_of_times, int *Tconstant,
                double *temperature, double *pressure,
double *density, double *dt, unsigned short seed[3],
                int *ascii_flag, char *inputfile, char *timefile,
char *datafile, char *restartfile, char *configfile)
{
  FILE *fp;

  /* check for inputfile */
  if ((fp=fopen(inputfile, "r"))==NULL) {
    printf("\nError opening file %s;  Program aborted!\n\n", inputfile);
    exit(1);
  }

  /* read input parameters */
  else {
    printf("\nReading Data from input file %s\n", inputfile);
    fscanf(fp, "%d %d %d %d %d %d", restart, tmax, write_time,
                                      Number_of_times, Tconstant, &N);
    fscanf(fp, "%lf %lf %lf %lf", temperature, pressure, density, dt);
    fscanf(fp, "%hu %hu %hu", &seed[0], &seed[1], &seed[2]);
    fscanf(fp, "%d", ascii_flag);
    fscanf(fp, "%s %s %s %s", timefile, datafile, restartfile,
                                                  configfile);

    /* echo back input to confirm */
    if (*Tconstant)
      printf("\nNVT Ensemble\n");
```

```
    else
      printf("\nNVE Ensemble\n");
    printf("\nRestart: %d\tMax Time: %d\n\nMolecules:\t%d\n",
                                              *restart, *tmax, N);
    printf("Temperature:\t%lf\nPressure:\t%lf\nDensity:\t%lf\n",
                                *temperature, *pressure, *density);
    printf("\nseeds:\t%d\t%d\t%d\n", seed[0], seed[1], seed[2]);
    printf("\nDataFile:\t%s\nRestartFile:\t%s\nConfigurationFile:\t%s\n\n",
                                        datafile, restartfile, configfile);
  }

  fclose(fp);
}

void write_data(int t, double dt, double kinetic,
double potential, double virial, double *temperature,
double *pressure, double volume, char *datafile,
                                                        int restart)
{
  FILE *fp;
  static int firsttime = 1;

  if (restart || !firsttime) {
    if((fp=fopen(datafile, "a"))==NULL) {
      printf("\nError opening file %s;  Program aborted!\n\n", datafile);
      exit(1);
    }
  }
  else if (firsttime) {
    if((fp=fopen(datafile, "w"))==NULL) {
      printf("\nError opening file %s;  Program aborted!\n\n", datafile);
      exit(1);
    }
    firsttime = 0;
  }

  virial     /= 3.0;
  *temperature = 2.0*kinetic/((double)(NDIM*N-NDIM)*k_B);
  *pressure    = ((double)N*k_B*(*temperature) + virial)/volume;

  /* write data */

  fprintf(fp, "%e\t%lf\t%lf\t%lf\t%lf\t%lf\n", (double)t*dt,
  kinetic/(double)N, potential/(double)N,
  (kinetic+potential)/(double)N, virial/(double)N, *temperature,
  *pressure);

  printf("\t%d\t%lf\n", t, *temperature);

  fclose(fp);

}
```

```
/* fread/fwrite does not work for writing entire array.  why?? */

void read_restartfile(int *t, double *temperature, double *pressure,
 double *density, int *time_index, int *latest_time,
                 int *block, double **x, double **v, double *L,
                 double mass, char *restartfile)
{
  double *xr, *vr;
  int i, k;
  FILE *fp;

  /* check for configuration */
  if ((fp=fopen(restartfile, "r"))==NULL) {
    printf("\nError opening file %s;  Program aborted!\n\n",
                                                 restartfile);
    exit(1);
  }

  xr = (double*) calloc(N*NDIM, sizeof(double));
  vr = (double*) calloc(N*NDIM, sizeof(double));

  fread(t, sizeof(int), 1, fp);
  fread(temperature, sizeof(double), 1, fp);
  fread(pressure, sizeof(double), 1, fp);
  fread(density, sizeof(double), 1, fp);
  fread(time_index, sizeof(int), 1, fp);
  fread(latest_time, sizeof(int), 1, fp);
  fread(block, sizeof(int), 1, fp);
  fread(xr, sizeof(double), N*NDIM, fp);
  fread(vr, sizeof(double), N*NDIM, fp);
  fclose(fp);

  *L       = pow((double)(N)*mass/(N_A*(*density)), 1.0/3.0);
  printf("System Size L =  %f\n", *L);

  for (i=0; i<N; i++)
    for (k=0; k<NDIM; k++) {
x[i][k] = xr[NDIM*i+k];
v[i][k] = vr[NDIM*i+k];
      }

  free(xr);
  free(vr);
}


void write_configuration(int t, int ascii_flag, double **x, double **v,
                         char *configfile)
{
  static double *xw, *vw;
  int i, k;
  static int firsttime=1;
  FILE *fp;
```

```
  /*
  if((fp=fopen(restartfile, "w"))==NULL) {
    printf("\nError opening file %s;  Program aborted!\n\n",
                                                 restartfile);
    exit(1);
  }

  */

  if((fp=fopen(configfile, "a"))==NULL) {
    printf("\nError opening file %s;  Program aborted!\n\n",
                                                  configfile);
    exit(1);
  }


   if (firsttime) {
    firsttime = 0;
    xw = (double*) calloc(N*NDIM, sizeof(double));
    vw = (double*) calloc(N*NDIM, sizeof(double));
  }

  if(ascii_flag){
    for (i=0; i<N; i++){
        for (k=0; k<NDIM; k++)
            fprintf(fp, "%lf\t", x[i][k]);
        fprintf(fp, "\n");
    }
  }

  else {
    for (i=0; i<N; i++)
        for (k=0; k<NDIM; k++) {
            xw[NDIM*i+k] = x[i][k];
            vw[NDIM*i+k] = v[i][k];
        }

    fwrite(&t, sizeof(int), 1, fp);
    fwrite(xw, sizeof(double), N*NDIM, fp);
    fwrite(vw, sizeof(double), N*NDIM, fp);
    }

  fclose(fp);
}


void write_restartfile(int t, double temperature, double pressure,
                       double density, int time_index, int latest_time,
                       int block, double **x, double **v,
                       char *restartfile, int restart)
{
  static double *xw, *vw;
  int i, k;
  static int firsttime=1;
```

```
  FILE *fp;

  if((fp=fopen(restartfile, "w"))==NULL) {
    printf("\nError opening file %s;  Program aborted!\n\n",
                                                 restartfile);
    exit(1);
  }

   if (firsttime) {
    firsttime = 0;
    xw = (double*) calloc(N*NDIM, sizeof(double));
    vw = (double*) calloc(N*NDIM, sizeof(double));
  }


  for (i=0; i<N; i++)
    for (k=0; k<NDIM; k++) {
      xw[NDIM*i+k] = x[i][k];
      vw[NDIM*i+k] = v[i][k];
    }


  fwrite(&t, sizeof(int), 1, fp);
  fwrite(&temperature, sizeof(double), 1, fp);
  fwrite(&pressure, sizeof(double), 1, fp);
  fwrite(&density, sizeof(double), 1, fp);
  fwrite(&time_index, sizeof(int), 1, fp);
  fwrite(&latest_time, sizeof(int), 1, fp);
  fwrite(&block, sizeof(int), 1, fp);
  fwrite(xw, sizeof(double), N*NDIM, fp);
  fwrite(vw, sizeof(double), N*NDIM, fp);

  fclose(fp);
}


void read_timefile(int *timeList, int N_t, char *timefile) {
  int i;
  FILE *fp;
  if((fp = fopen(timefile, "r"))==NULL) {
    printf("\nError opening file %s;  Program aborted!\n\n",
                                                   timefile);
    exit(1);
  }

  for(i = 0; i < N_t; i++)
    fscanf(fp, "%d\n", &timeList[i]);

    printf("timeList = %d\n", timeList[10]);
    printf("N_t = %d\n", N_t);

 fclose(fp);

}
```

```
============================================================

/*
 *
 * lj.h
 *
 *
 */

#include "cell.h"
#include "nlist.h"

/* global variables */
int N;

void read_input(int *restart,  int *tmax, int *write_time,
                int* Number_of_times, int *Tconstant,
                double *temperature, double *pressure,
double *density, double *dt, unsigned short seed[3],
                int *ascii_flag, char *inputfile, char *timefile,
char *datafile, char *restartfile, char *configfile);

void read_timefile(int *timeList, int N_t, char *timefile);

void init_simulation(double *mass, double *cutoff, double *cutoff2,
     double *minCutoff, double *skin, double *permit);

void read_restartfile(int *t, double *temperature, double *pressure,
                      double *density, int *time_index, int *latest_time,
                      int *block, double **x, double **v, double *L,
                      double mass, char *restartfile);

void init_lattice(double **x, double **v, double density, double mass,
  double temperature, unsigned short seed[3],
  double *kinetic, double *L);

void write_data(int t, double dt, double kinetic,
double potential, double virial, double *temperature,
double *pressure, double volume, char *datafile,
                                                    int restart);

void write_configuration(int t, int ascii_flag, double **x, double **v,
                                                   char *configfile);

void write_restartfile(int t, double temperature, double pressure,
                       double density, int time_index, int latest_time,
                       int block, double **x, double **v,
                                  char *restartfile, int restart);


void init_cells(cell* cells, int ncell, int ncell3);
```

```
void load(double **x, cell *cells, double L, int ncell, int ncell3);

void neighbor_list(list *nlist, double **x, cell *cells, double L,
                   double skin, double permit, int ncell3, int rank,
                                                           int np);

void force(double **x, double *f, cell *cells, list *nlist, double L,
   int ncell3, double cutoff, double minCutoff, double cutoff2,
           double *potential, double *virial, int rank, int np);

void verlet1(double **x, double **v, double **f, double mass, double L,
     double dt);

void verlet2(double **v, double **f, double mass, double dt,
     double *kinetic, int Tconstant, double T);


clock_t clock(void);
```

```
============================================================
```

```
/*
 *
 * cell.h
 *
 *
 */

#include "define.h"

typedef struct _cell {
  int *molecules;     /* array of molecule numbers in the cell */
  int last;           /* array element containing the last molecule */
  int neighbors[NBRS];  /* nearest neighbor identities */
} cell;
```

```
============================================================
```

```
/*
 *
 * define.h
 *
 *
 */

#define ATOMS 1
#define NDIM  3
#define NBRS  14

/* random constants */
#define PI 3.14159265358979323846264 3
```

```
#define k_B 1.0
#define N_A 1.0

/* for real units (fs, nm, kJ/mol)
#define k_B 0.0083144
#define N_A 602.205
*/
```

```
/*
 *
 * extern.h
 *
 * Global Variables:  see lj.h for further explanation
 */

extern int N;

extern double sigma6p;
extern double sigma12p;
extern double sigma6f;
extern double sigma12f;
extern double lj_pshift;
extern double lj_fshift;
```

```
/*
 *
 * nlist.h
 *
 */

#include "define.h"

typedef struct _list {
  int *neighbors;    /* array of neighboring atoms/molecules */
  int last;          /* last molecule on the list */
} list;
```

Appendix B

Analysis code

B.1 A program for calculating mean string and cluster sizes

```
/*
 *  dzugMeanClusterString.cpp
 *
 *  This program calculates the weight averaged mean cluster
 *  or mean string sizes. It can be easily modified to
 *  calculate the number averaged mean cluster or string sizes.
 *
 *
 */

#include <fstream>
#include <assert.h>
#include <stdio.h>
#include <math.h>

#include <vector>

using namespace std;

#define N 17576
#define DIM 3
#define Temperature 0.43

#define Percentage 0.05

const int tMax = 81;
const int nof_times = 1;

const int blockMax = 10;

const double density = 0.85;

struct Atom{
```

```
    double *x;
    double *x0;
    double *xold;
    double *dr;
    double rSquare;
};

struct Cluster{
    int size;
    int *member;
};

struct String{
    int size;
    int *member;
};


struct Mobile{
     int ID;
     double *x;
     double *x0;
     double rSquare;
};

void allocate (Atom*, Cluster*, String*, Mobile*);

void msdFunction(Atom*, double, double&);

void r2Sort(const Atom*,double*, Mobile*, int);
void clusterFunction(Atom*, Cluster*, Mobile*, double,
                     double*, int *, const int, double**);

void stringofallMobile(const Atom*, String*, Mobile*, const double,
                       int*, double*, double**, const int);

void allocate (Atom *theAtom, Cluster *theCluster,
               String *theString, Mobile *theMobile){
    int i, k;
    for (i = 0; i < N; i++){
        theAtom[i].rSquare = 0.;
        theAtom[i].x = new double[DIM];
        theAtom[i].x0 = new double[DIM];
        theAtom[i].xold = new double[DIM];
        theAtom[i].dr = new double[DIM];
        for (k = 0; k < DIM; k++){
            theAtom[i].x[k] = theAtom[i].x0[k] = 0.;
            theAtom[i].dr[k] = theAtom[i].xold[k] = 0.;
        }
    }

    for (i =0; i < N; i++){
        theCluster[i].size = 0;
        theCluster[i].member = new int[(int)(Percentage*N)];
```

```
    }

    for (i = 0; i < (int)(Percentage*N); i++){
        theString[i].size = 0;
        theString[i].member = new int[(int)(Percentage*N)];
    }


    for (i = 0; i < (int)(Percentage*N); i++){
        theMobile[i].ID
        theMobile[i].rSquare = 0.;
        theMobile[i].x = new double[DIM];
        theMobile[i].x0 = new double[DIM];
        for (k = 0; k < DIM; k++)
            theMobile[i].x[k] = theMobile[i].x0[k] = 0.;

    }

    return;
}

void msdFunction(Atom *theAtom, double L, double& sum_r2){
    int i, k;
    double r2, dx, dx2;

    sum_r2 = 0.;

    for (i = 0; i < N; i++){

        r2 = 0.;

        for (k = 0; k < DIM; k++){

            dx = theAtom[i].x[k] - theAtom[i].xold[k];
            dx -= (floor(dx/L + 0.5))*L;
            assert(fabs(dx) <= L/2.);

            theAtom[i].dr[k] += dx;
            dx2 = theAtom[i].dr[k]*theAtom[i].dr[k];

            r2 += dx2;
            theAtom[i].xold[k] = theAtom[i].x[k];
        }

        theAtom[i].rSquare = r2;
        sum_r2 += r2;
    }

    return;
}


void r2Sort(const Atom *theAtom, double *temp_rSquare,
                        Mobile *theMobile, int LowerIndexFast){
```

```
    int i, j, k, rMinIndex;
    int mobCount = 0;
    double rMin, rMinSquare;

    for (i = 0; i < N; i++)
        temp_rSquare[i] = theAtom[i].rSquare;


    for (i = 0; i < N; i++){

        rMin = temp_rSquare[i];
        rMinIndex = i;

        for (j = i+1; j < N; j++){

            if (temp_rSquare[j] < rMin){
                rMin = temp_rSquare[j];
                rMinIndex = j;
            }
        }

        temp_rSquare[rMinIndex] = temp_rSquare[i];
        temp_rSquare[i] = rMin;
    }

    for (i = 0; i < N; i++){
        if (theAtom[i].rSquare >= temp_rSquare[LowerIndexFast]){
            theMobile[mobCount].ID = i;
            theMobile[mobCount].x = theAtom[i].x;
            theMobile[mobCount].x0 = theAtom[i].x0;
            theMobile[mobCount].rSquare = theAtom[i].rSquare;
            mobCount++;
        }
    }

        cout << "Sorting Finished !!!!!!!" <<endl;
        return ;
}


void clusterFunction(Atom *theAtom, Cluster *theCluster, Mobile *theMobile,
                     double L, double *clusterSize, int *bigClusterIndex,
                     const int t, double **probCluster){
    int i, j, k, n;
    int newIndex, oldIndex;
    double dxij, dxij2, deltaR2;
    int nMobile = (int)(Percentage*N);
    int index[nMobile];

    int probCount[nMobile];
    int sum_probCount = 0;
```

```
for (i = 0; i < nMobile; i++){
    index[i] = i;
    theCluster[i].size = 0;
    probCount[i] = 0;
}



for (i = 0; i < nMobile; i++){
    for (j = 0; j < nMobile; j++){

        dxij2 = 0;

        for (k = 0; k < 3; k++){
            dxij = theMobile[i].x[k] - theMobile[j].x[k];
            dxij -= (floor(dxij/L + 0.5))*L;

            assert(fabs(dxij) <= L/2.);

            dxij2 += dxij*dxij;
        }


        deltaR2 = dxij2;

        if (deltaR2 <= rof_1stNshellSquare){
            if (index[i] <= index[j]){
                newIndex = index[i];
                oldIndex = index[j];
            }
            else{
                newIndex = index[j];
                oldIndex = index[i];
            }
        } /* end if deltaR2 ...*/

        for (k = 0; k < nMobile; k++){
            if (index[k] == oldIndex)
                index[k] = newIndex;
        }

    }/* j loop */

} /* i loop*/


for (i = 0; i < nMobile; i++){
    k = index[i];
    theCluster[k].size++;
    theCluster[k].member[theCluster[k].size - 1] = theMobile[i].ID;
}

int infinity = 1000000;
```

```
    /*
        //================================================================
  // A part that searches percolating clusters.

        int m;
        //int infinity = 1000000;
        int NumbZeros = 15;
        double binSize = 0.1;
        int NoBins = (int)floor(L /binSize);


        int max = 2;
        int index_maxCluster;
        int span[NoBins][DIM];
        int pcount[DIM];
        int empty_count;

        for(m = 0; m < NoBins; m++)
  for (k = 0; k < DIM; k++)
    span[m][k] = 0;

        for (i = 0; i < nMobile; i++){
            if (theCluster[i].size > max){
                max = theCluster[i].size;
                index_maxCluster = i;
            }
        }

        for (int mm = 0; mm < theCluster[index_maxCluster].size; mm++){
            m = (int)floor((theAtom[theCluster[index_maxCluster].
                            member[mm]].x[k] + L/2.)/binSize);
            span[m][k] += 1;
        }


for (k = 0; k < DIM; k++){
pcount[k] = 0;
empty_count = 0;
for (m = 0; m < NoBins; m++){

if (span[m][k] > 0)
empty_count = 0;
else
empty_count++;
if((empty_count > NumbZeros) &&
(pcount[k] < empty_count))
pcount[k] = empty_count;

}

if (pcount[k] ==0)
                    theCluster[index_maxCluster].size == infinity;
}
```

```
   //============================================================
*/

   int sum_n2 = 0;
   int sum_n = 0;

   for (i = 0; i < nMobile; i++){
       if (theCluster[i].size != infinity){
           n = theCluster[i].size;
           probCount[n]++;
           sum_n2 += n*n;
           sum_n += n;
       }
   }

   for (i = 1; i < nMobile; i++)
       sum_probCount += probCount[i];

   clusterSize[t] += (float)sum_n2/(float)sum_n;
   cout << clusterSize[t] << endl;

   for (i = 1; i < nMobile; i++)
       probCluster[t][i] += (double)probCount[i]/(double)sum_probCount;




   /* Task(1): printing members of big cluster

   ofstream fout;
   fout.open("cMembers");

   for (i = 0; i < theCluster[maxIndex].size; i++)
       fout << theCluster[maxIndex].member[i]<< endl;

   fout.close();

  end of Task(1) */


   /* Task(2): printing all clusters at time t

   char strType[40][3] = {"He", "Li","Mn", "Be", "B", "C",
                          "ca", "N", "O", "F", "Na","Zn", "Mg",
                          "Si","P", "S", "I", "K", "Cu","H", "Al",
                          "ca", "cl","Ni", '\0'};

  int typeCount = 0;

   char outputFile[50];
   sprintf(outputFile, "theClusters/xyz_t=%d",t);
```

```
    fout.open(outputFile);

    fout <<totalNumber<<endl;
    fout << "clusters with size > "<<theLowerSize<<"at time "<<t<<endl;

     for (i = 0; i < nMobile; i++){
        if (theCluster[i].size > theLowerSize){

            for (int mm = 0; mm < theCluster[i].size; mm++){
                fout << "Mg" << "\t";
                for (k = 0; k < DIM; k++)
                    fout << theAtom[theCluster[i].member[mm]].x[k]<<"\t";
                fout << endl;
            }
            typeCount++;
        }
     }

     fout.close();

     end of Task(2) */

    return;
}


void stringofallMobile(const Atom *theAtom, String *theString,
                       Mobile *theMobile, const double L,
                       int *atomsInStrings, double *meanString,
                       double **probString, const int t){

        int i, j, k, n;
        int newIndex, oldIndex;
        double deltaXij0, deltaXi0j, deltaXij02, deltaXi0j2;
        int nMobile =  (int)(Percentage*N);
        int index[nMobile];
        int nStrings_ofSize[nMobile];

        int sum_nStrings_ofSize_n = 0;
        int sum_nonTrivialStrings_ofSize_n  = 0;


        for (i = 0; i < nMobile; i++){
            index[i] = i;
            theString[i].size = 0;
        }


        for (i = 0; i < nMobile; i++){
            for (j = 0; j < nMobile; j++){


                deltaXij02 = 0;
                deltaXi0j2 = 0;
```

```
            for (k = 0; k < DIM; k++){
                deltaXij0 = theMobile[i].x[k] - theMobile[j].x0[k];
                deltaXi0j = theMobile[j].x[k] - theMobile[i].x0[k];

                deltaXij0 -= (floor(deltaXij0/L + 0.5))*L;
                deltaXi0j -= (floor(deltaXi0j/L + 0.5))*L;

                assert(fabs(deltaXij0) <= L/2.);
                assert(fabs(deltaXi0j) <= L/2.);

                deltaXij02 += deltaXij0*deltaXij0;
                deltaXi0j2 += deltaXi0j*deltaXi0j;
            }

           if((deltaXij02 < 0.36) || (deltaXi0j2 < 0.36)){


                if (index[i] <= index[j]){
                    newIndex = index[i];
                    oldIndex = index[j];
                }
                else{
                    newIndex = index[j];
                    oldIndex = index[i];
                }
            }  /*end if deltaXij2 ...*/


            for (k = 0; k < nMobile; k++){
                if (index[k] == oldIndex)
                    index[k] = newIndex;
            }


        } /* j loop */

    } /* i loop */


    for (i = 0; i < nMobile; i++){
        k = index[i];
        theString[k].size++;
        theString[k].member[theString[k].size - 1] = theMobile[i].ID;
    }


    for (i = 0; i < nMobile; i++)
nStrings_ofSize[i] = 0;

    int count = 0;
    int min = 0; //Note: change this number from 0 to 2
                 //when calculating mean for non-trivial strings.
    int strIndex = -1;
```

```
        for (i = 0; i < nMobile; i++){
            if (theString[i].size > min){
                count += theString[i].size;
                nStrings_ofSize[theString[i].size]++;
            }
        }

       *atomsInStrings = count;

/* mean string calculation */
int nNn, sum_nNn = 0;
int n2Nn, sum_n2Nn = 0;

for (n = 0; n < nMobile; n++){
    if (nStrings_ofSize[n] > 0){
                nNn = n*nStrings_ofSize[n];
                n2Nn = n*nNn;
                sum_nNn += nNn;
                sum_n2Nn += n2Nn;
             }
        }

        if (sum_nNn > 0)
            *meanString = (double)sum_n2Nn/(double)sum_nNn;
        else
            *meanString = 0.;


        //int min_for_nonTrivialStrings = 2;
        //for (n = min_for_nonTrivialStrings; n < nMobile; n++);
        //    sum_nonTrivialStrings_ofSize_n += nStrings_ofSize[n];

        //*atomsInStrings = sum_nonTrivialStrings_ofSize_n;

        for (n = min; n < nMobile; n++)
            sum_nStrings_ofSize_n += nStrings_ofSize[n];

        for (n = min; n < nMobile; n++){
            if (sum_nStrings_ofSize_n > 0)
                probString[t][n] +=
                  (double)nStrings_ofSize[n]/(double)sum_nStrings_ofSize_n;
        }

        return;

}


void main (int argc, char *argv[])
{


    int i, j, k, t;
    int startBlock = 0;
```

```
int time, start_time, present_time;
int timeIndex;
int repeatCount = 0, blockCount = 0;
int timeArray[tMax] = {0};
int tInterval[nof_times*tMax];
double L, sum_r2, temp_rSquare[N] = {0.};
double min_rSquareFast;
int bigClusterIndex;

int mobile_atomsInStrings;

double msd[nof_times*tMax] = {0.};
double mean_clusterSize[nof_times*tMax] = {0.};

double meanString, sum_meanString[nof_times*tMax] = {0.};
double sum_mobile_in_strings[nof_times*tMax] = {0.};


double **probCluster;
double **probString;

probCluster = new double*[nof_times*tMax];
probString = new double*[nof_times*tMax];

for (t = 0; t < nof_times*tMax; t++){
        probCluster[t] = new double[(int)(2*Percentage*N)];
        probString[t] = new double[2*(int)(Percentage*N)];
}



char inputFile[50], outputFile[50];

FILE *fp;


Atom *theAtom = new Atom[N];
Cluster *theCluster = new Cluster[(int)(Percentage*N)];
String *theString = new String[(int)(Percentage*N)];
Mobile *theMobile = new Mobile[(int)(Percentage*N)];

ofstream fout;
ifstream fin;

if (argc !=2){
    cerr << "\n Error: output file is not specified \n";
    exit(1);
}

fin.open("log_time.dat");

if (fin.fail()){
    cerr << "\n Error: Can not open log_time.dat \n";
    exit(1);
```

```
}

for (t = 0; t < tMax; t++)
    fin >> timeArray[t];

fin.close();

allocate (theAtom, theCluster, theString, theMobile);

L = pow((double)N/density,1.0/3.0);
int LowerIndexFast = N - (int)(Percentage*N);
int upperIndexSlow = (int)(Percentage*N);

cout<<"LowerIndexFast = "<<LowerIndexFast<<endl;
cout << "upperIndexSlow = "<<upperIndexSlow<<endl;

for (int block = startBlock; block < blockMax; block++) {

    time = repeatCount*timeArray[tMax - 1];

    blockCount++;
    repeatCount++;
    start_time = -1;

    if (block == startBlock)
        start_time = timeArray[0];

    for (t = 0; t < tMax; t++){

        sprintf(inputFile, "gunzip -c T=%.2fprod/xyz_t=%d.%d.dat.gz",
                           Temperature, block, timeArray[t]);

        if ((fp=popen(inputFile, "r"))==NULL) {
            printf("\nError opening file %s;
                     Program aborted!\n\n", inputFile);
            exit(1);
        }

        ifstream fin1(fileno(fp));

        cout <<inputFile<<endl;

        if (!fin1){
          cout <<" couldn't read the file  " << inputFile <<endl;
          exit(1);
        }

        for (i = 0; i < N; i++)
            for(k = 0; k < DIM; k++)
                fin1 >> theAtom[i].x[k];

        fin1.close();
        pclose(fp);
```

```
        if (timeArray[t] == start_time){
            for (i = 0; i < N; i++)
                for(k = 0; k < DIM; k++){
                   theAtom[i].x0[k] = theAtom[i].x[k];
                   theAtom[i].xold[k] = theAtom[i].x[k];
                   theAtom[i].dr[k] = 0.;
                }
        }


        msdFunction(theAtom, L, sum_r2);

        present_time = time + timeArray[t];
        timeIndex = (repeatCount -1)*tMax + t;

        tInterval[timeIndex] = present_time - timeArray[0];
        msd[timeIndex] += sum_r2/(double)N;

       if (present_time != start_time){

            r2Sort(theAtom, temp_rSquare, theMobile, LowerIndexFast);

            // Mean cluster calculation
            clusterFunction(theAtom, theCluster, theMobile, L,
                              mean_clusterSize, &bigClusterIndex,
                              timeIndex,  probCluster);


            // Mean string calculation

            //stringofallMobile(theAtom, theString,
            //                theMobile, L, &mobile_atomsInStrings,
            //                &meanString, probString, timeIndex);

            //sum_mobile_in_strings[timeIndex] += mobile_atomsInStrings;
            //sum_meanString[timeIndex] += meanString;
      }

    } /* time loop */

    if (repeatCount == nof_times){
        repeatCount = 0;
        startBlock += nof_times;
    }


} /* block loop */



//fout.open("msd.dat");
```

```
strcpy(outputFile, argv[1]);
fout.open(outputFile);

for (t = 1; t < nof_times*tMax; t++)
    fout << tInterval[t]<<"\t"
         <<msd[t]/(double)(blockCount/nof_times)<<endl;

fout.close();


fout.open("mean_clusterSize_T=0.43.dat");

for (t = 1; t < nof_times*tMax; t++)
    fout << tInterval[t]
         <<"\t"<<mean_clusterSize[t]/(double)(blockCount/nof_times)
         <<endl;

fout.close();

/*
fout.open("mean_String_n_grOReq_3_T=0.42.dat");

for (t = 1; t < nof_times*tMax; t++)
    fout << tInterval[t]
         <<"\t"<<sum_meanString[t]/(double)(blockCount/nof_times)
         <<endl;

fout.close();


fout.open("theFractions_T=0.42.dat");
for (t = 1; t< nof_times*tMax; t++)
    fout <<tInterval[t]<<"\t"
        <<sum_mobile_in_strings[t]/(Percentage*(double)N)
         /(double)(blockCount/nof_times) <<endl;

fout.close();


fout.open("probString_n_grOReq_3_T=0.42.dat");

for (t = 1; t < nof_times*tMax; t++)
    for (int n = 1; n < (int)(Percentage*N); n++)
        fout << tInterval[t] <<"\t"<< n <<"\t"
             << probString[t][n]/(double)(blockCount/nof_times)
             <<endl;

fout.close();


fout.open("probCluster_T=0.42.dat");


for (t = 1; t < nof_times*tMax; t++)
```

```
        for (int n = 1; n < (int)(Percentage*N); n++)
            fout << tInterval[t] <<"\t"<< n <<"\t"
                 << probCluster[t][n]/(double)(blockCount/nof_times)
                 <<endl;

    fout.close();
    */

    return (0);

}
```

B.2 A program utilized for performing normal mode analysis

```
/*
 *  normalMode.c
 *
 *
 *  This program performs normal mode analysis for the inherent structures by
 *  first calculating the elements of the Hessian matrix for the Dzugutov
 *  potential, and then diagonalizing the matrix using function calls of the
 *  standard functions tred2() and tqli() taken from "Numerical Recipes in C,
 *  Cambridge University Press, Cambridge, 1988."
 *
 *
 */


#include "nrutil.h"
#include <stdio.h>
#include <math.h>

#include <stdlib.h>

#include <stddef.h>

#include <time.h>

#define NR_END 1
#define FREE_ARG char*
```

```
#define NRANSI



#define DIM 3
#define N 2197
#define density 0.85

#define T 0.55
#define time 10239

#define anint(x) ((x >= 0.5) ? (1.0) : (x <= -0.5) ? (-1.0) : (0.0))


const double minCutoff = 1.87;   /* a = 1.87 */
const double cutoffsq = 1.94*1.94;   /* b = 1.94 */


/* const double cutoffsq = 2.5*2.5; // LJ */


void derivatives_of_pot_ij(double rij, double rijsq, double *duij,
                           double *d2uij);

/*void derivatives_of_pot_ij_LJ(double rij, double rijsq, double *duij,
                           double *d2uij); */

void tred2(double **a, int n, double d[], double e[]);
void tqli(double d[], double e[], int n, double **z);

double *dvector(long nl, long nh);
double **dmatrix(long nrl, long nrh, long ncl, long nch);
void nrerror(char error_text[]);

void free_dvector(double *v, long nl, long nh);
void free_dmatrix(double **m, long nrl, long nrh, long ncl, long nch);

int main(int argc, char **argv)
{

    char inputFile[80];
    char outputFile[80];
    char nucleationTest[80];
    int i, j, k, l, ll;
    int Ierror;
    int NM = (int)DIM*N;
    int mm;
    int simu;
    int isNucleating;
    double a;
    double L, invL;
    double *pbc;
    double rij, rijsq;
    double *xij, *xijsq;
```

```
    double d2uij, duij;
    double **x, **uxxii;

    double *Omega, *Em;
    double **Zeta;

   /* double junk; */

    FILE *fp0, *fp1, *fp2;



for (simu=100; simu<140; simu++){

    clock_t timeElapsed;

    sprintf(nucleationTest, "simu2197_%d/nucleating", simu);

    printf("\n %s \n", nucleationTest);

    if((fp0 = fopen(nucleationTest, "r")) == NULL){
        printf("\nError opening file %s;
               Program aborted!\n\n", nucleationTest);
        exit(1);
    }

    fscanf(fp0, "%d", &isNucleating);

    fclose(fp0);
    if (isNucleating)
        printf("\n!!!!!! Escape !!!!!!\t%d\t%s", isNucleating,
               nucleationTest);


    if (!isNucleating){

       pbc = dvector(1,3);
       xij = dvector(1,3);
       xijsq = dvector(1,3);

       Omega = dvector(1,NM);
       Em = dvector(1,NM);

       x = dmatrix(1, NM, 1, 3);
       uxxii = dmatrix(1, NM, 1, NM);
       Zeta = dmatrix(1, NM, 1, NM);

       printf("before\n");

       for (i = 1; i <= 5; i++){
          for (k = 1; k <= 5; k++)
             printf("%lf\t", uxxii[i][k]);
          printf("\n");
```

```
 }



sprintf(inputFile, "simu2197_%d/T=%.2fmin/IS_xyz_t=0.%d.dat",
        simu, T, time);
if((fp1 = fopen(inputFile, "r")) == NULL){
    printf("\nError opening file %s;
            Program aborted!\n\n", inputFile);
    exit(1);
}

printf("\n %s \n", inputFile);


for (i = 1; i <= N; i++)
    fscanf(fp1, "%lf %lf %lf", &x[i][1], &x[i][2], &x[i][3]);

for (i = 1; i <= 5; i++){
    for (k = 1; k <= 3; k++)
        printf("%lf\t",x[i][k]);
    printf("\n");
}

fclose(fp1);

L = pow((N/density), 1/3.);
invL = 1/L;

/* case 1 and 2 */
for(i = 1; i <=N; i++){
    l = (i-1)*DIM + 1;
    for (j = 1; j <= N; j++){
        if (j != i){
            rijsq = 0.;

           for (k = 1; k <= DIM; k++){
                xij[k] = x[i][k] - x[j][k];
                pbc[k]  = L*anint(xij[k]*invL);
                xij[k] -= pbc[k];
                xijsq[k] = xij[k]*xij[k];
                rijsq += xijsq[k];
            }

            if (rijsq < cutoffsq){
                rij = sqrt(rijsq);

                derivatives_of_pot_ij(rij, rijsq, &duij, &d2uij);

                for (ll = l; ll < l + DIM; ll++){
                    for (k = ll - l + 1; k <= DIM; k++){
                        if (((i-1)*DIM+k) == ll){
                            a = xijsq[k]/rijsq;
                            uxxii[(i-1)*DIM+k][ll] += d2uij*a +
```

```
                                                (1/rij)*duij*(1-a);
                        }
                        else{
                            a = xij[ll-l + 1]*xij[k]/rijsq;
                            uxxii[(i-1)*DIM+k][ll]  += d2uij*a
                                                    - (1/rij)*duij*a;
                            uxxii[ll][(i-1)*DIM+k] =
                                     uxxii[(i-1)*DIM+k][ll];
                        }
                    } /* k loop */
                }/*ll loop */
            }/* if (rijsq < cutoff)*/

        } /* if (j != 1) */
    } /* j loop */
} /*i loop */



 for (i = 1; i < N; i++){
    for (j = i + 1; j <= N; j++){
        rijsq = 0.;

        for (k = 1; k <= DIM; k++){

            xij[k] = x[i][k] - x[j][k];
            pbc[k]  = L*anint(xij[k]*invL);
            xij[k] -= pbc[k];
            xijsq[k] = xij[k]*xij[k];
            rijsq += xijsq[k];
        }

        if (rijsq < cutoffsq){
            rij = sqrt(rijsq);
           derivatives_of_pot_ij(rij, rijsq, &duij, &d2uij);

            for (ll = (i-1)*DIM+1; ll < (i-1)*DIM+1 + DIM; ll++){

                for (k = 1; k <= DIM; k++){
                    if (((ll - ((i-1)*DIM+1)) + 1) == k){ /* case 3 */
                        a = xijsq[k]/rijsq;
                        uxxii[(j-1)*DIM + k][ll] = -(d2uij*a
                                           + (1/rij)*duij*(1-a));
                        uxxii[ll][(j-1)*DIM + k] = uxxii[(j-1)*DIM
                                                + k][ll];

                    }
                    else { /* case 4 */
                        mm = (ll - ((i-1)*DIM+1)) + 1;
                        a = xij[mm]*xij[k]/rijsq;
                        uxxii[(j-1)*DIM + k][ll] = -d2uij*a
                                              + (1/rij)*duij*a;
                        uxxii[ll][(j-1)*DIM + k] = uxxii[(j-1)*DIM
                                                    + k][ll];
```

```
                    }

                } /* k loop */

            }/* ll loop */



        } /*if (rijsq < cutoff)*/
    } /* j loop */
} /* i loop */

printf("after\n");

for (i = 1; i <= 7; i++){
    for (j = 1; j <= 5; j++)
        printf("%lf\t", uxxii[i][j]);
    printf("\n");
}

printf ("\n before starting tred2 \n");
    printf("\n%d", NM);
    printf("\n%lf\n", uxxii[NM][NM]);

tred2(uxxii, NM, Omega, Em);
printf("\n finished tred2 \n");

tqli(Omega, Em, NM, uxxii);
printf("\n finished tqli \n");

sprintf(outputFile, "simu2197_%d/T=%.2fnorm/Omega_t=0.%d.dat",
                                    simu, T, time);
if((fp2 = fopen(outputFile, "w")) == NULL){
    printf("\nError opening file %s;
              Program aborted!\n\n", outputFile);
    exit(1);
}

for (i = 1; i <= DIM*N; i++)
       fprintf(fp2, "%lf\n", Omega[i]);
fclose(fp2);

free_dvector(pbc, 1, 3);
free_dvector(xij, 1, 3);
free_dvector(xijsq, 1, 3);
free_dvector(Omega, 1, NM);
free_dvector(Em, 1, NM);

free_dmatrix(x, 1, NM,1, 3);
free_dmatrix(uxxii, 1, NM,1, NM);
free_dmatrix(Zeta, 1, NM,1, NM);
```

```
    } /* if (!isNucleating) */

    timeElapsed = clock();
    printf("\n time elapsed in seconds: \t%ld\n",
                          timeElapsed/CLOCKS_PER_SEC);
 } /* for (simu ...) */

    return(0);

}


void derivatives_of_pot_ij(double r, double r2,
                double *first_deriv_pot, double *second_deriv_pot)
{
        register double invr, invr2, invr8, invr16, invr17;
        register double item0, item1, item2, item3, item4, item5;
        register double term0, term1, term2 = 0., term3;
        double second_deriv_pot_1 = 0., second_deriv_pot_2;



        while (r < minCutoff) {
            invr   = 1.0/r;
            invr2  = 1.0/r2;
            invr8  = invr2*invr2*invr2*invr2;
            invr16 = invr8*invr8;
            invr17 = invr*invr16;
            invr16 -= 1.28;
            invr17 *= 93.12;

            item0 = r - 1.87;
            item1 = 1.1/item0;
            item2 = 6.402/(item0*item0);

            term0 = invr17 + invr16*item2;
            term1 = exp(item1);
            term2 = term1*term0;


            /* Task_0:  for second derivative calculation */
            second_deriv_pot_1 = term1*(invr17*(17*invr
                                + 2.2/(item0*item0))
      + (item2*invr16/item0)*(item1 + 2));

      /* end Task_0 */
            break;
         }

        item3 = r - 1.94;
        item4 = 0.27/item3;
        item4 = exp(item4);
        item5 = 0.3456/(item3*item3);
```

```
        term3 = item4*item5;


        *first_deriv_pot = -(term2 + term3);  /* du/dr */

        second_deriv_pot_2 = (term3/item3)*(0.27/item3 + 2);
        *second_deriv_pot = second_deriv_pot_1 + second_deriv_pot_2;
    }

void tred2(double **a, int n, double d[], double e[])
{
int l,k,j,i;
        double scale,hh,h,g,f;

        for (i=n;i>=2;i--) {
l=i-1;

h=scale=0.0;
if (l > 1) {
for (k=1;k<=l;k++)
scale += fabs(a[i][k]);
if (scale == 0.0)
e[i]=a[i][l];
else {
for (k=1;k<=l;k++) {
a[i][k] /= scale;
h += a[i][k]*a[i][k];
}
f=a[i][l];
g=(f >= 0.0 ? -sqrt(h) : sqrt(h));
e[i]=scale*g;
h -= f*g;
a[i][l]=f-g;
f=0.0;
for (j=1;j<=l;j++) {
a[j][i]=a[i][j]/h;
g=0.0;
for (k=1;k<=j;k++)
g += a[j][k]*a[i][k];
for (k=j+1;k<=l;k++)
g += a[k][j]*a[i][k];
e[j]=g/h;
f += e[j]*a[i][j];
}
hh=f/(h+h);
for (j=1;j<=l;j++) {
f=a[i][j];
e[j]=g=e[j]-hh*f;
for (k=1;k<=j;k++)
   a[j][k] -= (f*e[k]+g*a[i][k]);
}
}
} else {
e[i]=a[i][l];
```

```
                        }
d[i]=h;

}
/* Next statement can be omitted if eigenvectors not wanted */
d[1]=0.0;
e[1]=0.0;

        /* Contents of this loop can be omitted if eigenvectors not
wanted except for statement d[i]=a[i][i]; */
for (i=1;i<=n;i++) {
/*
                l=i-1;
                printf("\ni = %d\n", i);
if (d[i]) {
for (j=1;j<=l;j++) {
g=0.0;
for (k=1;k<=l;k++)
g += a[i][k]*a[k][j];
for (k=1;k<=l;k++)
a[k][j] -= g*a[k][i];
}
}
                */

d[i]=a[i][i];
/*
a[i][i]=1.0;
for (j=1;j<=l;j++) a[j][i]=a[i][j]=0.0;
                printf("\nclosing Eigenvectors\n");
                */
}

}

double *dvector(long nl, long nh)
/* allocate a double vector with subscript range v[nl..nh] */
{
double *v;

v=(double *)malloc((size_t) ((nh-nl+1+NR_END)*sizeof(double)));
if (!v) nrerror("allocation failure in dvector()");
return v-nl+NR_END;
}

double **dmatrix(long nrl, long nrh, long ncl, long nch)
/* allocate a double matrix with subscript range m[nrl..nrh][ncl..nch] */
{
long i, nrow=nrh-nrl+1,ncol=nch-ncl+1;
double **m;

/* allocate pointers to rows */
m=(double **) malloc((size_t)((nrow+NR_END)*sizeof(double*)));
if (!m) nrerror("allocation failure 1 in matrix()");
```

```
m += NR_END;
m -= nrl;

/* allocate rows and set pointers to them */
m[nrl]=(double *) malloc((size_t)((nrow*ncol+NR_END)*sizeof(double)));
if (!m[nrl]) nrerror("allocation failure 2 in matrix()");
m[nrl] += NR_END;
m[nrl] -= ncl;

for(i=nrl+1;i<=nrh;i++) m[i]=m[i-1]+ncol;

/* return pointer to array of pointers to rows */
return m;
}

void nrerror(char error_text[])
/* Numerical Recipes standard error handler */
{
fprintf(stderr,"Numerical Recipes run-time error...\n");
fprintf(stderr,"%s\n",error_text);
fprintf(stderr,"...now exiting to system...\n");
exit(1);
}

void free_dvector(double *v, long nl, long nh)
/* free a double vector allocated with dvector() */
{
    free((FREE_ARG) (v+nl-NR_END));
}

void free_dmatrix(double **m, long nrl, long nrh, long ncl, long nch)
/* free a double matrix allocated by dmatrix() */
{
    free((FREE_ARG) (m[nrl]+ncl-NR_END));
    free((FREE_ARG) (m+nrl-NR_END));
}

void tqli(double d[], double e[], int n, double **z)
{
        double pythag(double a, double b);
int m,l,iter,i,k;
        double s,r,p,g,f,dd,c,b;

for (i=2;i<=n;i++) e[i-1]=e[i];
e[n]=0.0;
for (l=1;l<=n;l++) {
                /*printf("\n Inside tqli l = %d", l);*/
iter=0;
do {
for (m=l;m<=n-1;m++) {
dd=fabs(d[m])+fabs(d[m+1]);
                                if ((double)(fabs(e[m])+dd) == dd) break;
}
if (m != l) {
```

```
if (iter++ == 300000)
                                    nrerror("Too many iterations in tqli");
g=(d[l+1]-d[l])/(2.0*e[l]);
r=pythag(g,1.0);
g=d[m]-d[l]+e[l]/(g+SIGN(r,g));
s=c=1.0;
p=0.0;
for (i=m-1;i>=l;i--) {
f=s*e[i];
b=c*e[i];
e[i+1]=(r=pythag(f,g));
if (r == 0.0) {
d[i+1] -= p;
e[m]=0.0;
break;
}
s=f/r;
c=g/r;
g=d[i+1]-p;
r=(d[i]-g)*s+2.0*c*b;
d[i+1]=g+(p=s*r);
g=c*r-b;
                                        /*Next loop can be omitted if
                                         eigenvectors are not wanted*/
                                        /*
for (k=1;k<=n;k++) {
f=z[k][i+1];
z[k][i+1]=s*z[k][i]+c*f;
z[k][i]=c*z[k][i]-s*f;
}
                                        */
}
if (r == 0.00 && i >= l) continue;
d[l] -= p;
e[l]=g;
e[m]=0.0;
}
} while (m != l);
}
}

#undef NRANSI

double pythag(double a, double b)
{
        double absa,absb;
absa=fabs(a);
absb=fabs(b);
if (absa > absb) return absa*sqrt(1.0+SQR(absb/absa));
else return (absb == 0.0 ? 0.0 : absb*sqrt(1.0+SQR(absa/absb)));
}
#undef NRANSI
```

BIBLIOGRAPHY

[1] J. Zarzycki, *Glasses and the Vitreous State* (Cambridge University Press, Cambridge, 1990).

[2] C. A. Angell, Science, **267** 1942 (1995).

[3] K. D. Rector, J. R. Engholm, C. W. Rella, J. R. Hill, D. D. Dlott, and M. D. Fayer, J. Phys. Chem. A **103** 2381 (1999); H. Frauenfelder, S. G. Sligar, P. G. Wolynes, Science **254** 1598 (1991).

[4] U. Balucani and M. Zoppi, *Dynamics of the Liquid State* (Clarendon Press, Oxford, 1994).

[5] P. G. Debenedetti, *Metastable Liquids: Concepts and Principle* (Princeton University press, Princeton, New Jersey, 1996).

[6] The intersection of the liquid and vitreous portions of the volume versus temperature curve provides one definition of T_g. An other definition of T_g is the temperature at which the shear viscosity reaches 10^{13} Poise, or, equivalently, the relaxation time τ becomes of the order of 100 sec [5].

[7] The supercooled liquid state is a metastable state, but it can be regarded as equilibrium liquid state as long as no crystal nuclei are present.

[8] M. D. Ediger, C. A. Angell, and S. R. Nagel, J. Phys. Chem., **100**, 13200 (1996).

[9] N. A. Morgan, and F. J. Spera, Geochim. Cosmochim. Ac., **65**, 4019 (2001).

[10] H. Frauenfelder, S. G. Sligar, P. G. Wolynes, Science **254**, 1598 (1991); G. U. Nienhaus, J. D. Müller, B. H. McMahon, and H. Frauenfelder, Physica D **107**, 297 (1997); V. I. Abkevich, A. M. Gutin, and E. I. Shakhnovich, J. Chem. Phys. **101**, 6052 (1994); J. G. Saven, J. Wang, and P. G. Wolynes, J. Chem. Phys. **101** 11037 (1994); J. Wang, J. Onuchic, and P. G. Wolynes, Phys. Rev. Lett. **76** 4861 (1996); K. A. Dill and H. S. Chan, Nature Struc. Biol. **4**, 10 (1997).

[11] M. D. Ediger, Science **287**, 604 (2000).

[12] G. Adam and J. H. Gibbs, J. Chem. Phys. **43**, 139 (1965).

[13] J. H. Gibbs, E. A. DiMarzio, J. Chem. Phys. **28**, 373 (1958); J. H. Gibbs, J. Chem. Phys. **25**, 185 (1956); E. A. DiMarzio and J. H. Gibbs, J. Chem. Phys. **28**, 807 (1958).

[14] D. Turnbull and M. H. Cohen, J. Chem. Phys. **29**, 1049 (1958); J. Chem. Phys. **34**, 120 (1961); J. Chem. Phys. **52**, 3038 (1970).

[15] M. H. Cohen and G. S. Grest, Phys. Rev. B **20**, 1077 (1979); Phys. Rev. Lett. **45**, 1271 (1980).

[16] G. S. Grest and M. H. Cohen, Phys. Rev. B **21**, 4113 (1980); Adv. Chem. Phys. **49**, 455 (1981).

[17] E. Leutheusser, Phys. Rev. A **29**, 2765 (1984); U. Bengtzelius, W. Götze, and A. Sjölander, J. Phys. C **17**, 5915 (1984); W. Götze, L. Sjögren, Rep. Progr. Phys. **55**, 241 (1992); W. Götze, L. Sjögren, Chem. Phys. **212**, 47 (1996); W. Götze, L. Sjögren, Transport Theory Stat. Phys. **24**, 801 (1995).

[18] D. Kivelson and G. Tarjus, J. Chem. Phys. **109** 5481 (1998); J. Non-Cryst. Solids **235**, 86 (1998); D. Kivelson, G. Tarjus, X, Zhao, S. A. Kivelson, Phys. Rev. E **53**, 751 (1996).

[19] P. W. Anderson, Science **267**, 1615 (1995).

[20] W. K. Kegel and A. V. Blaaderen, Science **287**, 290 (2000).

[21] E. Weeks, J. C. Crocker, A. C. Levitt, A. Schofield, and D. A. Weitz, Science **287**, 627 (2000).

[22] J. Leisen, K. Schmidt-Rohr and H.W. Spiess, Physica A **201**, 3020 (1993); R. Böhmer, G. Hinze, G. Diezemann, B. Geil, and H. Sillescu, Europhys. Lett. **36**, 55 (1996); S.C. Kuebler, A. Heuer, and H.W. Spiess, Phys. Rev. E **56**, 741

(1997); R. Böhmer, G. Diezemann, G. Hinze, H. Sillescu, J. Chem. Phys. **108**, 890 (1998); G. Hinze, Phys. Rev. E **57**, 2010 (1998); I. Chang, F. Fujara, B. Geil, G. Heuberger, T. Mangel, and H. Sillescu, J. Non-Cryst. Solids **172**, 248 (1994); A. Heuer, M. Wilhelm, H. Zimmerman, and H. W. Spiess, **75**, 2851 (1995).

[23] C. A. Angell, K. L. Ngai, G. B. McKenna, P. F. McMillan, and S. W. Martin, J. Appl. Phys. **88**, 3113, (2000).

[24] J. T. Bendler and M. F. Shlesinger, J. Stat. Phys. **53**, 531 (1998).

[25] H. Bässler Phys. Rev. Lett. **58**, 767 (1987).

[26] R. Richert and H. Bässler, J. Phys.:Condens. Matter **2**, 2273 (1990).

[27] R. H. Colby, Phys. Rev. E **61**, 1783 (2000).

[28] M. H. Cohen and G. Grest, Adv. Chem. Phys. **48**, 370 (1981); Phys. Rev. B **20**, 1077 (1979).

[29] H. Vogel, Phys. Z. **22**, 645 (1921); G. Tammann and G. Hesse, Z. Anorg. Allg. Chem. **156**, 245 (1926); G. S. Fulcher, J. Am. Chem. Soc. **8**, 339 (1925).

[30] M. L. Williams, R. F. Landel, J. D. Ferry, J. Am. Ceram. Soc.**77**, 3701 (1953).

[31] M. T. Cicerone and M. D. Ediger, J. Chem. Phys. **104**, 7210 (1996).

[32] M. D. Ediger, Annu. Rev. Phys. Chem. **51**, 99 (2000)

[33] F. Fujara, B. Geil, H. Sillescu, and G. Fleischer, Z. Phys. B Condens. Matter **88**, 195 (1992).

[34] L. D. Landau and E. M. Lifshitz, *Fluid Mechanics* (Pergamon, New York, 1987).

[35] F. H. Stillinger and J. A. Hodgdon, Phys. Rev. E **50**, 2064 (1994).

[36] M. T. Cicerone and M. D. Ediger, J. Phys. Chem. **97**, 10489 (1993).

[37] M. T. Cicerone, F. R. Blackburn, and M. D. Ediger, J. Chem. Phys. **102**, 471 (1995).

[38] P. G. Debenedetti and F. H. Stillinger, Nature **410**, 259 (2001).

[39] J. Wong and C. A. Angell, *Glass Structure by Spectroscopy* (M. Dekker, New York, 1974).

[40] R. Kohlrausch, Ann. Rev. Phys. Chem. **91**, 179 (1874); G. Williams and D. C. Watts, Trans. Farad. Soc. **66**, 80 (1970).

[41] The sharp decrease in entropy is a consequence of the scarcity of acccessible configurations that affect both the equilibrium and transport properties of the supercooled liquids.

[42] W. Kauzmann, Chem. Rev. **43**, 219 (1948).

[43] C. A. Angell and J. C. Tucker, J. Phys. Chem. **78**, 278 (1974).

[44] F. H. Stillinger, J. Chem. Phys. **88**, 7818 (1988).

[45] M. Goldstein, J. Chem. Phys. **51**, 3728 (1969).

[46] F. H. Stillinger and T. A. Weber, Phys. Rev. A **25**, 978 (1982); Phys. Rev. A **28**, 2408 (1983); J. Phys. Chem. **87**, 2833 (1983); Science **225**, 983 (1984); J. Chem. Phys. **80**, 4434 (1984).

[47] F. H. Stillinger and R. A. LaViolette, J. Chem. Phys. **83**, 6413 (1985).

[48] F. H. Stillinger, Science **267**, 1935 (1995).

[49] The remaining part of the excess volume is the one that can not be randomly distributed because of the severe penality imposed by packing constraints when the system is cooled to a very low T. This part of the excess volume is uniformly distributed around fixed positions about which molecules undergo vibrational motions during thermal expansions. The excess volume is thus expressed as $v - v_o = v_f + \Delta v_c$, where Δv_c is that part of the excess volume that can not be randomly distributed.

[50] K. Binder and A. P. Young, Rev. Mod. Phys. **58**, 801 (1986).

[51] T. R. Kirkpatrick, D. Thirumalai, and P. G. Wolynes, Phys. Rev. A **40**, 1045 (1989).

[52] The original equation of motion for $F(q,t)$ is the same as that of a damped harmonic oscillator, but with additional complication of a retarded friction that is proportional to a memory kernel function $\Gamma(q,t)$. The so-called idealized version of the MCT simplifies this equation and approximates the memory

function as a quadratic form of the correlation function $F(q,t)$ as $\Gamma(q,t) = \sum_{\mathbf{k}+\mathbf{k}'=\mathbf{q}} V(\mathbf{q};\mathbf{k},\mathbf{k}')F(k,t)F(k',t)$, where the vertices $V(\mathbf{q};\mathbf{k},\mathbf{k}')$ are computed from the structure factor $S(q)$.

[53] W. van Megan and S. M. Underwood, Phys. Rev. E **47**, 248 (1993); W. van Megan and S. M. Underwood, Phys. Rev. E **49**, 4206 (1994); W. van Megan and S. M. Underwood, and P. N. Pusey, Phys. Rev. Lett. **67**, 1586 (1991).

[54] Mean-field spin glasses are known to exhibit a dynamic transition at a temperature which is identified with T_{MCT}. But, the connection between glass-forming liquids and spin glasses is yet to be resolved.

[55] M. Fuchs, W. Gotze, S. Hildebrand, and A. Latz, J. Phys.:Condens. Matter **4**, 7709 (1992); W. Götze and L. Sjorgen, J. Non-Cryst. Solids **131**, 161 (1991); L. Sjorgen and W. Götze, J. Non-Cryst. Solids **131**, 153 (1991).

[56] F. H. Stillinger, Nature **410**, 259 (2000).

[57] K. L. Ngai, Comments Solid State Phys. **9**, 127 (1979); J. Colmenero, A. Alegria, I. Telleria, K. L. Ngai, Phys. Rev. B **47**, 5041 (1993).

[58] R. A. Denny, D. R. Reichman, and J. -P. Bouchaud, Phys. Rev. Lett. **90** 025503 (2003); C. Monthus and J. -P. Bouchaud, J. Phys. A **29**, 3847 (1996).

[59] G. J. Fan and H. J. Fecht, J. Chem. Phys. **116**, 5002 (2002).

[60] J. P. Garrahan and D. Chandler, Phys. Rev. Lett. **89**, 035704 (2002); J. P. Garrahan and D. Chandler, Proc. Natl. Acad.Sci. USA **100** 9710 (2003).

[61] L. Berthier and J. P. Garrahan, J. Chem. Phys. **119**, 4367 (2003); L. Berthier and J. P. Garrahan, Phys. Rev. E **68**, 041201 (2003).

[62] R. Richert, J. Phys.: Condens. Matter **14**, R703 (2002).

[63] R. Richert, J. Chem. Phys. **113**, 8404 (2000).

[64] E. V. Russel and N. E. Israelof, Nature **408**, 695 (2000).

[65] L. A. Deschenes and D. A. Vanden Bout, Science **292**, 255 (2001).

[66] H. Sillescu, J. Non-Cryst. Solids **243**, 81 (1999).

[67] S. C. Glotzer, J. Non-Cryst. Solids **274**, 342 (2000).

[68] U. Tracht, M. Wilhelm, A. Heuer, H. Feng, K. Schmidt-Rohr, and H. W. Spiess, Phys. Rev. Lett. **81**, 2727 (1998).

[69] S. A. Reinsberg, X. H. Qiu, M. Wilhelm, H. W. Spiess, and M. D. Ediger, J. Chem. Phys. **114**, 7299 (2001).

[70] R. D. Mountain, *Supercooled Liquids, ACS Symposium Series 676*, edited by J. T. Fourkas, D. Kivelson, U. Mohanty, and K. A. Nelson (American Chemical Society, Washington, DC, 1997).

[71] R. M. Ernst, S. R. Nagel, and G. S. Grest, Phys. Rev. B **43**, 8070 (1991).

[72] C. Dasgupta, A. V. Indrani, S. Ramaswamy, and M. K. Phani, Euro, Phys. Lett. **15**, 307 (1991).

[73] L defined by first calculating the transverse current correlation function $C_T(q,t)$, which is the perpedicular component of the current-current correlation function $\mathbf{J_q}(t_0)\mathbf{J_q}(t+t_0)$, and obtaining the position of maximum power spectrum $w_m(q)$ of the transverse current fluctuation $C_T(q,w)$ for different q. Then L is obtained from from the dispersion curve of the transverse mode by extrapolatin $w_m(q)$ to zero [70].

[74] W. Kob, C. Donati, S. J. Plimpton, P. H. Poole, and S. C. Glotzer, Phys. Rev. Lett. **79**, 2827 (1997).

[75] C. Donati, J. F. Douglas, W. Kob, P. H. Poole, S. J. Plimpton, and S. C. Glotzer, Phys. Rev. Lett. **80**, 2338 (1998).

[76] C. Donati, S. C. Glotzer, P. H. Poole, W. Kob, and S. J. Plimpton, Phys. Rev. E **60**, 3107 (1999).

[77] S. C. Glotzer and C. Donati, J. Phys.: Condens. Matter **11**, A285 (1999).

[78] C. Donati, S. C. Glotzer, and P. H. Poole, Phys. Rev. Lett., **82**, 5064 (1999).

[79] C. Bennemann, C. Donati, J. Baschnagel and S. C. Glotzer, Nature **399**, 246 (1999).

[80] P. H. Poole, C. Donati, and S. C. Glotzer, Physica A **261**, 51 (1998).

[81] S. C. Glotzer, C. Donati, and P. H. Poole, *Computer Simulation Studies in Condensed Matter Physics XI*, D.P. Landau, ed. (Springer-Verlag, 1999).

[82] P. Allegrini, J. F. Douglas and S. C. Glotzer, Phys. Rev. E **60**, 5714 (1999).

[83] S. C. Glotzer, V. N. Novikov, T. B. Schrøder, J. Chem. Phys. **112**, 509 (2000); N. Lacevic, F. W. Starr, T. B. Schrøder, V. N. Novikov, and S. C. Glotzer, Phys. Rev. E **66**, 030101 (2002); N. Lacevic, F. W. Starr, T. B. Schrøder, and S. C. Glotzer, J. Chem. Phys. **119**, 7372 (2003).

[84] G. Johnson, A. Mel'cuk, H. Gould, W. Klein, R. D. Mountain, Phys. Rev. E **57**, 5707 (1998).

[85] R. Ahluwalia and S. P. Das, Phys. Rev. E **57**, 5771 (1998).

[86] W. Kob and H. C. Andersen, Phys. Rev. E **51**, 4626 (1995).

[87] T. Muranaka and Y. Hiwatari, Phys. Rev. E **51**, R2735 (1995).

[88] R. Yamamoto and A. Onuki, Phys. Rev. E **58**, 3515 (1998).

[89] H. Fynewever and P. Harrowell, J. Phys. Condens. Matter **12**, 6323 (2000).

[90] J. D. Weeks, D. Chandler, H. C. Andersen, J. Chem. Phys. **54**, 5237 (1971).

[91] B. Doliwa and A. Heuer, Phys. Rev. E **61**, 6898 (2000).

[92] H. Yin and B. Chakraborty, Phys. Rev. Lett. **86**, 2058 (2001).

[93] M. Sellitto, J. Phys.:Condens. Matter **12**, 6477 (2000); E. Follana and F. Ritort, Phys. Rev. B **54**, 930 (1996); M. Schulz and S. Trimper, Phys. Rev. E **57**, 6398 (1998).

[94] C. Donati, S. Franz, S. C. Glotzer, G. Parisi, J. of Non-Cryst. Solids **307**, 215 (2002).

[95] F. W. Starr, S. Sastry, E. La Nave, A. Scala, H. E. Stanley, and F. Sciortino, Phys. Rev. E **63**, 041201 (2001).

[96] A. Scala, F. W. Starr, E. La Nave, F. Sciortino, and H. E. Stanley, Nature **406**, 166 (2000).

[97] I. S-. Voivod, P. H. Poole, and F. Scortino, Nature **412**, 514 (2001).

[98] I. S-. Voivod, F. Sciortino, and P. H. Poole, cond-mat/0309481.

[99] Y. Hiwatari and J. Habasaki, J. Phys. Condens. Matter **12**, 6405 (2000).

[100] C. Bennemann, W. Paul, K. Binder, and B. Dünweg, Phys. Rev. E **57**, 843 (1998).

[101] M. Dzugutov, Phys. Rev. A **46**, R2984 (1992).

[102] F. Zetterling, M. Dzugutov, and S. Simdyankin, J. Non-Cryst. Solids, **293**, 39 (2001).

[103] M. Dzugutov, S. I. Simdyankin, F. H. M. Zetterling, Phys. Rev. Lett. **89**, 195701 (2002).

[104] A. R. Leach, *Molecular Modelling Principles and Applications*, 2nd ed. (Prentice Hall, 2001)

[105] M. P. Allenand and D. J. Tildesley, *Computer Simulation of Liquids* (Clarendon Press, Oxford, 1987)

[106] L. Verlet, Phys. Rev. **159**, 98 (1967); Phys. Rev. **165**, 201 (1967).

[107] R. W. Hockney, *Methods in Computational Physics* **9** (Plasma Physics, Academic Press, New York, 1970)

[108] D. Beeman, J. Comput. Phys. **20**, 130 (1976).

[109] S. Nosé Mol. Phys. **52**, 255 (1984); W. G. Hoover, Phys. Rev. A **31**, 1695 (1985).

[110] H. C. Anderson, J. Chem. Phys. **72**, 2384 (1980).

[111] M. Parrinello and A. Rahman, Phys. Rev. Lett. **45**, 1196 (1980).

[112] H. J. C. Berendsen, J. P. M. Postma, W. F. Van Gunsteren, A. Di Nola, and J. R. Haak, J. Chem. Phys. **81**, 3684 (1984).

[113] K. Kremer and G. S. Grest, J. Chem. Phys. **92**, 5057 (1990). A. Kopf, B. Dünweg, and W. Paul, J. Chem. Phys. **107**, 6945 (1997).

[114] C. Bennemann, W. Paul, K. Binder, and B. Dünweg, Phys. Rev. E **57**, 843 (1998); C. Bennemann, W. Paul, J. Baschnagel and K. Binder, J. Phys.: Condens. Matter **11**, 2179 (1999); J. Baschnagel, C. Bennemann,W. Paul, K. Binder, J. Phys.: Condens. Matter **12**, 6365 (2000); C. Bennemann, J. Baschnagel, and W. Paul, Eur. Phys. J. B. **10**, 323 (1999).

[115] The simulation was further extended to higher temperature, i.e., $T = 1.0$, and hence some of our analyses involve a data for $T = 1.0$.

[116] M. Dzugutov, K. -E. Larsson, and I. Ebbsjö, Phys. Rev. A **38**, 3609 (1988).

[117] A. Harrison, *Pseudopotential in the Theory of Metals* (Benjamin, New York, 1966).

[118] J. Roth and A. R. Denton, Phys. Rev. E **61**, 6845 (2000).

[119] S. Sachdev and D. Nelson, Phys. Rev. B **32**, 1480 (1985).

[120] J. Hafner, *From Hamiltonian to Phase Diagram* (Springer-Verlag, Berlin, 1987).

[121] D. Nelson, Phys. Rev. B **28**, 5515 (1983).

[122] M. Dzugutov, Phys. Rev. Lett. **70**, 2924 (1993).

[123] S. Simdyankin, M. Dzugutov, S. N. Taraskin, and S. R. Elliott, Phys. Rev. B **63**, 184301 (2001).

[124] H. E. Stanley *Introduction to Phase Transitions and Critical Phenomena* (Oxford University Press, New York, 1971).

[125] J. P. Hansen and I. R. McDonald, *Theory of Simple Liquids* (Academic Press, London, 1986).

[126] M. Aichele, Y. Gebremichael, F. W. Starr, J. Baschnagel, and S. C. Glotzer, J. Chem. Phys. **119**, 5290 (2003).

[127] C. P. Lindsey and G. D. Patterson, J. Chem. Phys. **73**, 3384 (1980).

[128] S. Sastry, P. G. Debenedetti, and F. H. Stillinger, Nature **393**, 554 (1998).

[129] T. B. Schrøder and J. C. Dyre, J. Non-Cryst. Solids **331**, 235 (1998); F. W. Starr, F. Sciortino, and H. E. Stanley, Phys. Rev. E **60**, 6757 (1999); H. Miyagawa, Y. Hiwatari, B. Bernu, and J. P. Hansen, J. Chem. Phys. **88**, 3879 (1988).

[130] T. B. Schrøder, S. Sastry, J. C. Dyre, and S. C. Glotzer, J. Chem. Phys. **112**, 9834 (2000).

[131] J. -N. Roux, J. -L. Barrat, J. -P. and Hansen, J. Phys.: Condens. Matter **1**, 7171 (1989).

[132] G. Wahnström, Phys. Rev. A **44**, 3752 (1991).

[133] F. Mezei, W. Knaak, and B. Farrago, Phys. Rev. Lett. **58**, 571 (1987).

[134] Y. Gebremichael, T. B. Schrøder, F. W. Starr, and S. C. Glotzer, Phys. Rev. E **64**, 051503 (2001).

[135] W. Kob, C. Donati, S.J. Plimpton, P.H. Poole, and S.C. Glotzer, Phys. Rev. Lett. **79**, 2827 (1997); C. Donati, J.F. Douglas, W. Kob, P.H. Poole, S.J. Plimpton, and S.C. Glotzer, Phys. Rev. Lett. **80**, 2338 (1998); C. Donati, S. Glotzer, P. Poole, W. Kob, and S. Plimpton, Phys. Rev. E **60**, 3107 (1999); S. C. Glotzer and C. Donati, J. Phys.: Condens. Matter **11**, A285 (1999); C. Donati, S.C. Glotzer, and P.H. Poole, Phys. Rev. Lett., **82**, 5064 (1999); C. Bennemann, C. Donati, J. Baschnagel and S. C. Glotzer, Nature **399**, 246 (1999); P. H. Poole, C. Donati, and S. C. Glotzer, Physica A**261**, 51 (1998); S. C. Glotzer, C. Donati, and P.H. Poole, *Computer Simulation Studies in*

Condensed Matter Physics XI, D.P. Landau, ed. (Springer-Verlag, 1999); P. Allegrini, J. F. Douglas and S.C. Glotzer, Phys. Rev. E **60**, 5714 (1999); A. Heuer and K. Okun, J. Chem. Phys. **106**, 6176 (1997); B. Doliwa and A. Heuer, Phys. Rev. Lett. **80**, 4915 (1998); B. Doliwa and A. Heuer, J. Phys.:Condens. Matter **11**, A277 (1999); Y. Hiwatari and T. Muranaka, J. Non-Cryst. Solids **235**, 19 (1998); D. N. Perera and P. Harrowell, J. Non-Cryst. Solids **235**, 314 (1998); R. Yamamoto and A. Onuki, Phys. Rev. E **58**, 3515 (1998); D. N. Perera and P. Harrowell, J. Chem. Phys. **111**, 5441 (1999).

[136] E. Hemple, G. Hemple, A. Hensel, C. Schick, and E. Donth, J. Phys. Chem. B **104**, 2460 (2000).

[137] A. H. Marcus, J. Schofield and S. A. Rice, Phys. Rev. E **60**, 5725 (1999).

[138] J. Horbach, W. Kob, and K. Binder, Phil. Mag. B **77**, 297 (1998).

[139] J. Colmenero, F. Alvarez, and A. Arbe, Phys. Rev. E **65**, 041804 (2002).

[140] S. Kämmerer, W. Kob, and R. Schilling, Phys. Rev. E **58**, 2131 (1998).

[141] F. Sciortino, P. Gallo, P. Tartaglia, S. -H. Chen, Phys. Rev. E **54**, 6331 (1996).

[142] S. Mossa, R. Di Leonardo, G. Ruocco, and M. Sampoli, Phys. Rev. E **62**, 612 (2000).

[143] R. Zorn, Phys. Rev. B **55**, 6249 (1997).

[144] M. Aichele and J. Baschnagel, Eur. Phys. J. E **5**, 229 (2001).

[145] M. Aichele, S. Khefif, H. Meyer, J. Baschnagel, and M. Fuchs, (in preparation).

[146] V. Krakoviack, J. P. Hansen, A. A, and Louis, Europhys. Lett. **58**, 53 (2002).

[147] M. Guenza, Phys. Rev. Lett. **88**, 025901 (2002).

[148] S. -H. Chong and M. Fuchs, Phys. Rev. Lett. **88**, 185702 (2002).

[149] M. Fuchs, W. Götze, and M. R. Mayr, Phys. Rev. E **58**, 3384 (1998).

[150] M. Doi and S. F. Edwards, *Theory of Polymer Dynamics* (Wiley, New York, 1987).

[151] K. Kremer and G. S. Grest, in *Monte Carlo and Molecular Dynamics Simulations in Polymer Science*, edited by K. Binder (Oxford University Press, New York, 1995).

[152] Note that, according to this procedure, any other choice for $t^*_{\alpha_2}$ would result in smaller fractions than the time interval when α_2 is maximum. This is because at time intervals earlier or later than the time when α_2 is maximum, the deviation of the van Hove correlation function from the Gaussian distribution becomes smaller, resulting in a small number of particles in the tail of the distribution. Therefore, ϕ is maximum at $t^*_{\alpha_2}$ when α_2 is maximum.

[153] Although one uses configurations both at $t = 0$ and at t to define monomer mobility, our results are the same regardless of which configuration is used for constructing clusters.

[154] D. A. McQuarrie, *Statistical Mechanics* (Harper and Row, New York, 1976).

[155] The normalization of S_w by S_0 factors out the contribution from random clustering, and allows us to quantify the degree of correlation irrespective of the choice of fraction ϕ.

[156] D. Stauffer, *Introduction to Percolation Theory* (Taylor and Francis, London and Philadelphia, 1985).

[157] At much longer times, beyond the scope of these simulations, the diffusive regime is reached.

[158] W. Kob, J. Phys.: Condens. Matter **11**, A1 (1999).

[159] W. Götze, J. Phys.: Condens. Matter **11**, A1 (1999).

[160] E. J. Donth, J. Non-Cryst. Solids **53**, 325 (1982); **131**, 204 (1991); *Relaxation and Thermodynamics in Polymers: Glass Transition* (Akademie Verlag, Berlin, 1992); C.T. Moynihan and J. Schroeder, J. Non-Cryst. Solids **160**, 52 (1993); **161**, 148 (1993); U. Mohanty, Adv. Chem. Phys. **89**, 89 (1995); R. E. Robertson, J. Poly. Sci., Poly. Symp. **63**, 173 (1978); M. D. Ediger, J. Non-Cryst. Solids **235**, 10 (1998).

[161] D. Sappelt and J. Jäckle, J. Phys. A: Math. Gen. **26**, 7325 (1993); C. Donati and J. Jäckle, J. Phys.: Condens. Matter **8** 2733 (1996).

[162] R. J. Speedy, J. Phys. Chem. B **103**, 4060 (1999).

[163] M. Mézard and G. Parisi, J. Phys.: Condens. Matter **11**, A157 (1999).

[164] S. Sastry, Phys. Rev. Lett. **85**, 590 (2000); Nature, **409**, 164 (2001); J. Phys. Condens. Matter **12**, 6515 (2000).

[165] A. Heuer, Phys. Rev. Lett. **78**, 4051 (1997); S. Buechner and A. Heuer, Phys. Rev. E. **60**, 6507 (1999).

[166] Ref. [76] found that $S_{\max}$ is well-described by the form $S_{\max} \sim (T - T_{\rm MCT})^{-\gamma}$, with $T_{\rm MCT} = 0.45$. Our data does not show a good power law fit for $T_{\rm MCT} = 0.45$; $S_{\max}$ can be fit with a power law form of $(T - T_p)^{-\gamma}$ with $T_p = 0.28$. We point out that Ref. [76] studied a much longer system. The relatively small size of our system might contribute to the difference we find between T_p and $T_{\rm MCT}$.

[167] N. Giovambattista, S. V. Buldgreu, F. W. Starr, and H. E. Stanley, Phys. Rev. Lett. **89**, 125501 (2003).

[168] G. Strobl, *The Physics of Polymers: Concepts for Understanding Their Structures and Behavior* (Springer, Berlin-Heidelberg, 1997).

[169] S. C. Greer, Adv. Chem. Phys. **94**, 261 (1996).

[170] Note that we include "strings" of size one in analogy with percolation theory, where clusters of size one are customarily included in the evaluation of mean cluster size, and with equilibrium polymerization, where polymers of size one are customarily included in the calculation of mean polymer size.

[171] R. Bellissent, L. Descotes, P. Pfeuty, J. Phys.: Condens. Matter **6**, A211 (1994).

[172] Y. Rouault and A. Milchev, Phys. Rev. E **51**, 5905 (1995).

[173] J. P. Wittmer, A. Milchev, and M. E. Cates, J. Chem. Phys. **109**, 834 (1998).

[174] Y. Gebremichael, M. Vogel, and S. C. Glotzer, J. Chem. Phys. **120**, 4415 (2004).

[175] Y. Gebremichael, M. Vogel, and S. C. Glotzer, Mol. Simul.**30**, 281 (2004).

[176] M. Russina and F. Mezei, Phys. Rev. Lett. **84**, 3630 (2000); M. Russina and F. Mezei, Physica B **276**, 437 (2000).

[177] B. Cui, B. Lin, and S. A. Rice, J. Chem. Phys. **114**, 9142 (2001).

[178] E. Weeks and D. A. Weitz, Phys. Rev. Lett. **89**, 095704 (2002).

[179] G. H. Fredrickson and H. C. Andersen, Phys. Rev. Lett. **53**, 1244 (1984); G. H. Fredrickson and H. C. Andersen, J. Chem. Phys. **83**, 5822 (1985).

[180] T. F. Middleton and D. J. Wales, Phys. Rev. B **64**, 0204205 (2001).

[181] S. Büchner and A. Heuer, Phys. Rev. Lett. **84**, 2168 (2000); S. Büchner and A. Heuer, Phys. Rev. E **60**, 6507 (1999); B. Doliwa and A. Heuer, Phys. Rev. E **67**, 030501 (2003).

[182] M. Vogel, B. Doliwa, A. Heuer, and S. C. Glotzer, J. Chem. Phys. (in press).

[183] R. J. Speedy, Mol. Phys. **95**, 169 (1998).

[184] S. Mossa, E. La Nave, H. E. Stanley, C. Donati, F. Sciortino, and P. Tartaglia, Phys. Rev. E **65**, 041205 (2002).

[185] B. Doliwa and A. Heuer, Phys. Rev. Lett. **80**, 4915 (1998); B. Doliwa and A. Heuer, J. Phys.:Condens. Matter **11**, A277 (1999).

[186] A. Heuer, M. Kunow, M. Vogel, and R. D. Banhatti, Phys. Rev. B **66**, 224201 (2002).

[187] F. Sciortino, W. Kob, and P. Tartaglia, Phys. Rev. Lett. **83**, 6331 (1999).

[188] F. H. Stillinger, Physica D **107**, 383 (1987); D. A. Beaton, C. DeCoste, D. L. Hunter, and A. Coniglio, Int. J. of Mod. Phys. C **11**, 41 (2000); A. Crisanti and F. Ritort, Europhys. Lett. **52**, 640 (2000).

[189] The conjugate gradient code we used for minimization was originally developed by S. Sastry and T. Schrøder, and later modifed by M. Bergroth.

[190] E. La Nave, F. Sciortino, P. Tartaglia, C. De Michele, and S. Mossa, J. Phys.: Condens. Matter **15**, S1085 (2003).

[191] S. Sastry, Physica A **315**, 267 (2002).

[192] R. J. Speedy, P. G. Debenedetti, Mol. Phys. **88**, 1293 (1998).

[193] Y. Rosenfeld and P. Tarazona, Mol. Phys. **95**, 141 (1998).

[194] A. Scala, F.W. Starr, E. La Nave, H.E. Stanley and F. Sciortino, Phys. Rev. E **62**, 8016 (2000).

[195] B. Coluzzi, G. Parisi, P. Verrocchio, J. Chem. Phys. **112**, 2933 (2000).

[196] D. D. Hall, D. D. Deppe, K. E. Hamilton, A. Dhinojwala, J. M. Torkelson, J. Non-Cryst. Solids **235**, 48 (1998).

[197] G. K. von Schulthess, G. B. Benedek, and R. W. De Blois, Macromolecules **13**, 939 (1980).

[198] C. Reichhardt and C. J. Reichhardt, Phys. Rev. Lett. **90**, 095504 (2003).

[199] M. Bergroth, A. S. Keyes, M. Vogel, and Y. Gebremichael, and S. C. Glotzer, (unpublished results).