

Supplementary Material for ‘Interplay between percolation and glassiness in the random Lorentz gas’

Giulio Biroli,¹ Patrick Charbonneau,^{2,3} Eric I. Corwin,⁴ Yi Hu,^{2,*}
Harukuni Ikeda,⁵ Grzegorz Szamel,⁶ and Francesco Zamponi¹

¹*Laboratoire de Physique de l’Ecole Normale Supérieure, ENS, Université PSL,
CNRS, Sorbonne Université, Université de Paris, F-75005 Paris, France*

²*Department of Chemistry, Duke University, Durham, North Carolina 27708, USA*

³*Department of Physics, Duke University, Durham, North Carolina 27708, USA*

⁴*Department of Physics and Material Science Institute,
University of Oregon, Eugene, Oregon 97403, USA*

⁵*Graduate School of Arts and Sciences, The University of Tokyo 153-8902, Japan*

⁶*Department of Chemistry, Colorado State University, Fort Collins, CO 80523, USA*

(Dated: February 5, 2021)

I. NOTATION

In order to investigate the interplay between the percolation and glassiness in the RLG, we first need to reconcile the two set of notations. The central quantity for both is the number density of obstacles, ρ , which allows to define a unitless volume fraction of obstacles $\Phi = \rho V_d \sigma^d$, where V_d is the d -dimensional volume of a unit sphere and σ is the obstacle radius. For the RLG, the obstacle radius is commonly set to $\sigma = 1$ while the tracer radius σ_{tracer} is infinitesimal, and hence naturally we can define $\Phi = \rho V_d$. Without loss of generality, and by analogy to the Mari-Kurchan model [1–3], we can equivalently choose $\sigma = \sigma_{\text{tracer}} = 1/2$, which naturally defines $\varphi = \Phi 2^{-d}$. For high-dimensional scaling convenience, we further define the rescaled packing fraction

$$\hat{\varphi} = \Phi/d = 2^d \varphi/d. \quad (\text{S1})$$

Similarly, the cage size, Δ , defined as the infinite-time limit of mean squared displacement (MSD) of the tracer, can be rescaled as $\hat{\Delta} = d\Delta$. For reference, Table S1 provides the correspondence between notations commonly used in the scientific literature about the RLG.

TABLE S1: Common notations for packing fraction and cage size

Quantity	Equivalence
ρ [3]	n [4]
Φ [3]	
φ [3, 5]	
$\hat{\varphi}$ [2, 3, 5]	
Δ [2]	$\delta r^2/\sigma^2$ [4]
$\hat{\Delta}$	Δ [5], Ad^2 [6]

II. MEAN-FIELD THEORY DERIVATION

As mentioned in the main text, the RLG cage size in the $d \rightarrow \infty$ limit can be obtained by applying the replica technique to a cavity computation. We here provide details about this derivation. The setup consists of N hard spherical obstacles of radius σ placed at positions R_i within a volume V centered around the origin. Crucially, obstacle positions are chosen uniformly at random with the constraint $|R_i| > \sigma$, in such a way that the origin is guaranteed to belong to one of the voids, and this is the void whose statistical properties we want to investigate. The motivation for

*Electronic address: yi.hu@duke.edu

doing so comes from the so-called *cavity reconstruction* process, described in details in the Supplementary Information of Ref. 2 and references therein. In a few words, the cavity reconstruction process formally shows that in order to sample voids with their equilibrium weight (i.e., proportionally to their volume), it is equivalent to draw obstacles uniformly at random and choose uniformly at random a particular point in the void space as the origin, or to fix a point as the origin and draw obstacles in such a way that this point belongs to a void, as we do here.

The free volume available to a tracer placed at the origin is thus

$$Z = \int dx \prod_{i=1}^N \theta(|x - R_i| > \sigma), \quad (\text{S2})$$

where $\theta(x)$ denotes the Heaviside function. When the spatial dimension $d \rightarrow \infty$, the integral is over an infinite number of coordinates, and the resulting free volume scales exponentially in d , $Z \sim \exp(dz)$, with z of order one. Although Z is not self-averaging, z is, and hence we need to study the average of $z = \log Z/d$ to obtain a proper description of the typical cages. This evaluation is here achieved by using the replica method. More specifically, we evaluate the replicated partition function in the thermodynamic, $N \rightarrow \infty$, limit at fixed obstacle density $\rho = N/V$ is

$$\overline{Z^n} = \int d\bar{x} \left[\frac{\int_{|R|>\sigma} dR \prod_{a=1}^n \theta(|x_a - R| > \sigma)}{\int_{|R|>\sigma} dR} \right]^N, \quad (\text{S3})$$

in order to obtain the free energy

$$F = -\overline{\log Z} = -\lim_{n \rightarrow 0} \partial_n \overline{Z^n}. \quad (\text{S4})$$

For this system, we expect two phases:

- In the *liquid* phase, the (replicated) tracers are not confined close to the origin. Each replica thus decorrelates over the whole volume and

$$\overline{Z^n} \sim V^n \left(\frac{V - (n+1)V_\sigma}{V - V_\sigma} \right)^N \sim V^n e^{-n\rho V_\sigma}, \quad F_{\text{liq}} = -\log V + \rho V_\sigma, \quad (\text{S5})$$

where V_σ is the volume of a sphere of diameter σ .

- In the *glass* phase, with high probability the origin is contained within a cage. Many other cages exist in the volume, but a tracer starting at the origin remains confined within that cage. Note that the cage at the origin is metastable, because faraway cages thermodynamically dominate the measure, thus motivating our choice of a cavity computation.

In the glass phase, we can write, after introducing a fictitious coordinate $x_0 = 0$,

$$\begin{aligned} \overline{Z^n} &= \int d\bar{x} \left\{ \frac{\int dR [\prod_{a=0}^n \theta(|x_a - R| > \sigma)]}{V - V_\sigma} \right\}^N \\ &= \int d\bar{x} \left\{ 1 + \frac{\int dR [\prod_{a=0}^n \theta(|x_a - R| > \sigma) - 1] + V_\sigma}{V - V_\sigma} \right\}^N \\ &= \int d\bar{x} \left\{ 1 + \frac{-\int dR \theta(\min_{a \in [0, n]} |x_a - R| < \sigma) + V_\sigma}{V - V_\sigma} \right\}^N \\ &= e^{\rho V_\sigma} \int d\bar{x} e^{-\rho \int dR \theta(\min_{a \in [0, n]} |x_a - R| < \sigma)} \\ &= e^{\rho V_\sigma} C_{n+1, d} \int d\hat{q} e^{\frac{d-n-1}{2} \log \det \hat{q} + \rho \bar{f}_{n+1}(\{0, \bar{x}\})}, \end{aligned} \quad (\text{S6})$$

where overlap variables have been changed to rotationally invariant quantities, $\hat{q}_{ab} = x_a \cdot x_b$ as in Ref. 5, Eq.(2.96), and

$$\bar{f}_n(\bar{x}) = - \int dR \theta\left(\min_{a \in [1, n]} |x_a - R| < \sigma\right), \quad (\text{S7})$$

as in Ref. 7, Eq.(30).

Following the approach of Refs. 5, 8 for evaluating $\overline{Z^n}$ by saddle point integration for $d \rightarrow \infty$, we obtain

$$\begin{aligned} \log \overline{Z^n} &= \text{cnst} + \frac{d}{2} \log \det \hat{q} + d\hat{\varphi}\hat{g}_n \\ &= \text{cnst} + \frac{d}{2} \log \left[\left(n\hat{\Delta}_r - (n-1)\frac{\hat{\Delta}}{2} \right) \left(\frac{\hat{\Delta}}{2} \right)^{n-1} \right] + d\hat{\varphi} \left[\bar{f}_{n+1}(\hat{\Delta}_r, \hat{\Delta}) + 1 \right], \end{aligned} \quad (\text{S8})$$

where irrelevant constants have been dropped and the rescaled squared displacement and density, $\hat{\Delta}$ and $\hat{\varphi}$, respectively, are defined as in Section I. This expression shows that the free volume scales in such a way that $\log Z \propto d$, as conjectured at the beginning of this section.

By taking the replica symmetric solution, $\hat{\Delta}_r = \hat{\Delta}$, the expression can then be reduced to a one-dimensional integral [5], such that

$$f_{n+1}(\hat{\Delta}) = \int_{-\infty}^{\infty} dh e^h (q(\hat{\Delta}/2, h)^{n+1} - 1), \quad (\text{S9})$$

where $q(\hat{\Delta}, h) = [1 + \text{erf}(\frac{h+\hat{\Delta}/2}{\sqrt{2\hat{\Delta}}})]/2$. Note that in the original hard sphere derivation, in which all particles oscillate, the large variance term has the form $\hat{\Delta} = (\hat{\Delta}_{\text{tracer}} + \hat{\Delta}_{\text{obstacle}})/2$. By contrast, obstacles are fixed in the RLG, and hence $\hat{\Delta} = \hat{\Delta}_{\text{tracer}}/2$.

Under the replica symmetric assumption, the free energy is then

$$\log \overline{Z^n} = \text{cnst} + \frac{dn}{2} \log \hat{\Delta} + d\hat{\varphi} \left\{ \int_{-\infty}^{\infty} dh e^h \left[q(\hat{\Delta}/2, h)^{n+1} - 1 \right] + 1 \right\}. \quad (\text{S10})$$

Solving for $\frac{\partial \log \overline{Z^n}}{\partial \hat{\Delta}} = 0$ provides the cage size that optimizes the free energy,

$$\frac{n}{2\hat{\Delta}} = -\hat{\varphi} \int_{-\infty}^{\infty} dh e^h (n+1) q(\hat{\Delta}/2, h)^n \frac{\partial q(\hat{\Delta}/2, h)}{\partial \hat{\Delta}}. \quad (\text{S11})$$

Noting that $\int_{-\infty}^{\infty} dh e^h \frac{\partial q}{\partial \hat{\Delta}} = 0$ and taking the limit $n \rightarrow 0$, the cage size and the obstacle density are then related by

$$\frac{1}{2\hat{\varphi}} = -\hat{\Delta} \int_{-\infty}^{\infty} dh e^h \log q(\hat{\Delta}/2, h) \frac{\partial q(\hat{\Delta}/2, h)}{\partial \hat{\Delta}}. \quad (\text{S12})$$

III. VOID PERCOLATION THRESHOLD COMPUTATION

This section details the algorithm used for detecting the void percolation threshold of the RLG. We first place N obstacles uniformly at random within a d -dimensional box under periodic boundary conditions. Conventional cubic boxes, \mathbb{Z}^d , are used in $d \leq 6$, while the Wigner-Seitz cell of the checkerboard, D_d , lattice, the E_8 lattice and Λ_9 lattice are used in $4 \leq d \leq 7, 8$ and 9 , respectively. A Voronoi tessellation of the obstacles allows us to map the void percolation problem onto that of the bond percolation of edges in that tessellation. Each edge is weighted by the circumscribed radius of the facet in the Delaunay triangulation that is dual to this edge, which defines the minimum radius of the obstacles that can block this edge. Because the number of Voronoi vertices and edges grows exponentially with dimension, memory use must be carefully handled. First, we build the tessellation point by point [9]. Specifically, for each obstacle, p_i , we calculate the convex hull [10] of the inverse coordinates of the other obstacles, after translating p_i at the origin. The vertices of this convex hull then correspond to the neighbors of p_i in the Voronoi tessellation [11]. Second, edges with a sufficiently small weight remain blocked in a percolating network and are dropped on-the-fly, while building the tessellation. Orders of magnitude in memory use are therefore gained, thus enabling the analysis of sufficiently large systems, even in the highest dimension considered.

The percolation threshold is then determined by an algorithm akin to that used for the continuum-space percolation of obstacles [12, 13]. This approach is applied on a disjoint-set forest data structure. A disjoint-set consists of a number of nodes, each of which corresponding to a Voronoi vertex. Each node maintains a parent pointer and the displacement vector to its parent, tracing back to a unique root node in the set. Each disjoint-set thus corresponds to a cavity in the system. Voronoi edges are first sorted in descending order, and then the neighboring vertices of each edge are iteratively considered. If the two vertices, X_1 and X_2 , do not yet belong to a same cavity, they are merged; otherwise, percolation is checked by:

1. Calculating the displacement vector between X_1 and X_2 (under minimal image convention) $\mathbf{r}_0 = X_1 - X_2$;
2. Calculating the displacement vector from X_1 and X_2 to the root, \mathbf{r}_1 and \mathbf{r}_2 , respectively;
3. Comparing if $\mathbf{r}_1 - \mathbf{r}_2 \neq \mathbf{r}_0$.

If the displacements calculated from the two methods differ (necessarily, by integers), then the cavity must span across the periodic boundary and form a cycle. Percolation is deemed to take place when there exist such cycles in all dimensions, which reduces sample-to-sample variations compared to other percolation criteria [3]. From the standard percolation universality class [14], we know that the percolation threshold in a finite system of N obstacles converges to the thermodynamic, $N \rightarrow \infty$, limit, with asymptotic scaling

$$\Phi_p(N) - \Phi_p(\infty) \sim N^{-1/d\nu}, \quad (\text{S13})$$

where ν is the correlation length exponent, $\nu = 4/3, 0.8774, 0.6852, 0.5723$ for $d = 2$ to 5 [15] and $1/2$ for $d \geq 6$. Our percolation threshold detection algorithm increases the range of accessible system sizes by orders of magnitude, which makes this fitting robust in all dimensions considered in this work. Formally, the neighbors of an obstacle obey the Poisson distribution without bias if the system size (inscribed radius of the periodic box) is greater than the maximum neighbor-distance of obstacles. In high dimension, this condition requires an increasingly large number of obstacles. Empirically we find the asymptotic scaling is still recovered in smaller systems, within the range of numerical uncertainty, when the next-nearest periodic image of neighboring obstacles are included in the construction of Voronoi tessellation. The resulting percolation threshold are listed in Table S2. Note that our results reveal a systematic bias in the numerical treatment of Ref. [3] for $d \geq 4$, because it included pre-asymptotic system sizes in the fit.

TABLE S2: Numerical estimates of the void percolation threshold

d	Φ_p	$\hat{\varphi}_p$
2	1.1276(9)	0.5638(5)
3	3.510(2)	1.1698(8)
4	6.248(2)	1.5621(5)
5	9.170(8)	1.834(2)
6	12.24(2)	2.040(4)
7	15.46(5)	2.209(7)
8	18.64(8)	2.330(9)
9	22.1(4)	2.46(4)

IV. NUMERICAL CAVITY RECONSTRUCTION SCHEME

At high density, the RLG model is amenable to numerical cavity reconstruction, which allows to compute efficiently the properties of the localized regime within a spherical shell of radius r_{\max} . The number of obstacles N to be placed within that shell is first picked at random from the Poisson distribution

$$p(N) = \frac{N_0^N e^{-N_0}}{N!}, \quad (\text{S14})$$

where $N_0 = d\hat{\varphi}(r_{\max}^d - \sigma^d)$ is the average number of obstacles for the system size and density considered. These N obstacles are then placed uniformly at random within a hypersphere shell of inner radius $\sigma = 1$ and outer radius $r_{\max} > \sigma$. Because N is chosen in accordance to the fluctuation of the Poisson random field in a finite volume, this construction guarantees that the probability of obtaining a cavity containing the origin, \mathbb{C} , exactly tracks the distribution of cavities at that same $\hat{\varphi}$ in an infinitely large system. The properties of this cavity can then be sampled using either static or dynamical algorithms.

A. Static Sampling

For a purely geometric sampling of the cavity properties, a Delaunay triangulation (into d -simplicial cells) of the obstacles within that cavity is built using CGAL's dD Triangulation library [16]. The cavity is then constructed by

a graph search with cells as vertices and facets as edges. Starting from the cell that contains the origin, an edge (facet) is connected if the circumcenter of two cells are in same side, or the circumcenter are on opposite sides and the circumradius of the facet is greater than σ . All visited cells are added to the cavity. The cavity is valid if the displacement of any sites in the cavity to the origin is less than $r_{\max} - \sigma$. Care must be taken in choosing r_{\max} , such that this condition is met. Like in the Leath algorithm for lattice percolation [17], cavities are evenly sampled in a site base, that is, the probability of generating a cavity of volume V_{cavity} is proportional to $V_{\text{cavity}}P(V_{\text{cavity}})$, where $P(V_{\text{cavity}})$ is the probability of having a cavity of volume V_{cavity} in the thermodynamic limit.

Sastry *et al.* proved that the visited cells constructed this way contain and only contain the void space that belongs to the same cavity [18]. They also introduced an exact algorithm to determine the cavity volume through a recursive division of d -simplices. Because the exact decomposition of a cavity into simple primitives is quite involved in general dimension, we consider instead a random sampling algorithm. The basic idea is to generate uniformly distributed random points (samples) within the cavity and to use these samples to approximate the cavity volume and other physical quantities. The high level description of the principal algorithm is as follows:

Algorithm 1 Sampling a cavity

```

for  $C_i$  in visited cells do
   $V_i \leftarrow \text{SIMPLEXVOLUME}(C_i)$ 
  Increment  $V_{\text{cells}}$ 
end for
for  $j = 1$  to  $N_{\text{samples}}$  do
  Randomly choose a simplex  $C_k$  in  $\{C_i\}$  with probability  $V_k/V_{\text{cells}}$ 
  Place a random sample  $S \leftarrow \text{SAMPLESIMPLEX}(C_k)$ 
  if  $S$  in the void space then
    Add  $S$  to the void sample list and increment  $N_{\text{voids}}$ 
  end if
end for

```

Note that the volume of a d -simplex defined by the vertices $\{p_0, p_1, \dots, p_d\}$ is

$$V_{\text{simplex}} = \left| \frac{1}{d!} \det(p_1 - p_0, p_2 - p_0, \dots, p_d - p_0) \right|. \quad (\text{S15})$$

Obtaining uniform samples in a d -simplex is equivalent to generating $d + 1$ random spacing with unit sum [19, p. 568]. To generate $d + 1$ random spacings, x_0, \dots, x_d , one first generates d independent and uniformly distributed random variables y_1, \dots, y_d in $[0, 1)$ and sort them in place, in addition to $y_0 = 0$ and $y_{d+1} = 1$, then $x_i = y_{i+1} - y_i$, and the random sample $S = \sum_{i=0}^d x_i p_i$.

Determining whether S is in the void space requires a nearest-neighbor query of the obstacles. Although the nearest obstacle of S is most likely to be one of the vertices of C_i , outliers are possible. To accelerate the computation, one may pre-compute the point-to-simplex distances of these possible obstacles other than the simplex vertices, and store those with distance less than σ as candidate nearest neighbors.

As the obstacle density increases, the fraction and size of the voids become increasingly small, which makes this sampling approach inefficient. We then instead find the vertices of the cavity, build the triangulation over these vertices, and then run the cavity sampling algorithm in the new triangulation. Note that a simplex generated this way may lie completely in occupied space, or even contain the voids of other cavities. The later case should be rare and in fact is not observed in practice. One should nonetheless test for this case and drop occupied or invalid simplices from thus sampling. With this simple optimization, the fraction of void samples ($N_{\text{voids}}/N_{\text{samples}}$) typically varies from a half to nearly one.

From the N_{voids} samples out of N_{samples} within the cavity, we approximate the cavity volume

$$V_{\text{cavity}} = V_{\text{cells}} \frac{N_{\text{voids}}}{N_{\text{samples}}}. \quad (\text{S16})$$

where V_{cells} is the total volume of visited cells. From the set of samples within the void space, $\{S_i\}$, we can also approximate the infinite-time mean squared displacement of a tracer in this cavity as

$$\Delta_{\text{cavity}} = \langle (S_i - S_j)^2 \rangle = 2(\langle S_i^2 \rangle - \langle S_i \rangle^2). \quad (\text{S17})$$

The self van Hove function, $G_s(r, t)$, which is defined as the probability of finding a tracer at displacement r at time t , can be computed in the $t \rightarrow \infty$ limit, when every site is equally probable, for a single cavity as

$$G_{s,\text{cavity}}(r) = G_{s,\text{cavity}}(r, t \rightarrow \infty) \sim \sum_{i \neq j} \delta(|S_i - S_j| - r) \quad (\text{S18})$$

and is normalized as $\int_0^\infty G_s(r)dr = 1$. Note that the summation is over sites $i \neq j$, in order to eliminate the artificial peak at $r = 0$ due to the discretization scheme. Finally, the expected V , Δ and $G_s(r)$ are the arithmetic mean over all randomly generated cavities.

B. Dynamical Sampling

For the tracer dynamics, we implemented a high-dimensional generalization of the simulation scheme of Höfling *et al.* [4, 20].

For the cage escape analysis, obstacles are first generated according to the numerical cavity reconstruction scheme, which allows the vicinity of $\hat{\varphi}_d$ in dimensions as high as $d = 20$ to be reached. A tracer is then placed at the origin and ballistic dynamics is run. The simulation terminates when t_{\max} is reached or when the tracer escapes the shell, i.e., $r(t) > r_{\max} - \sigma$, whichever comes first. The maximal valid tracer squared displacement $\hat{\Delta}_{\max} = d(r_{\max} - \sigma)^2$ defines the simulation shell thickness. For a given t , multiple time intervals are sampled and averaged to obtain the dynamical cage size $\Delta(t)$ for a specific realization of disorder. The mode cage size at time t is defined as the maximum likelihood value in the distribution of time dependent cage sizes, $\hat{\Delta}_{\text{mode}}(t) = \arg \max P(\hat{\Delta}(t))$ (Fig. S1 (a)). The escape event time, t_{esc} at Δ_{esc} is calculated as the first-passage time of the tracer square displacement from the origin being Δ_{esc} .

By construction, the cavity reconstruction scheme exhibits no finite-size correction as long as the cage is closed. The obstacles are indeed then generated according to the Poisson distribution, as in an infinite-size system. For imperfectly closed cages, however, finite-size corrections arise because the tracer escapes that cage at different times, depending on the shell thickness. While the MSD is sensitive to the rare samples that exhibit large displacement, we find that finite-size corrections to $\hat{\Delta}_{\text{mode}}$ only become larger than statistical noise for fairly small system sizes, e.g., $\hat{\Delta}_{\max} \leq 8$ in $\hat{\varphi} = 2.7$ and $d = 16$. A significant fraction of tracers can then escape on a time comparable to reaching the plateau height, and hence $\hat{\Delta}_{\text{mode}}(t)$ shrinks with time. In all other cases, the mode reaches a plateau that persists for multiple time decades. We thus extract the plateau value of $\hat{\Delta}_{\text{mode}}$ by taking the average of $\hat{\Delta}_{\text{mode}}(\hat{t})$ from $\hat{t} = 50$ to 10^5 . The magnitude of the finite-size effect is comparable with the statistical noise of extracting the modal cage size from different realization of sample cages. It is worth note that, because in the highest approachable dimension, $d = 20$, a smaller simulation shell size, $\hat{\Delta}_{\max} = 14$, is used than in $d \leq 16$, the difference between the plateau heights, which scales like $1/d$, then becomes statistically indistinguishable (Fig. 4(a) in the main text).

To assure better accuracy, we randomly choose 4000 samples out of the total, evaluating the modal cage size and repeat multiple times. This bootstrap sampling gives the expected $\hat{\Delta}_{\text{mode}}$ and the confidence interval of the approximation shown in Fig. 4(b) in the main text. Results are then fitted with linear form with zero intercept. We chose this intercept in order to obtain clearer results for Fig. 4(c). If the intercept is fitted as well, its value deviates at most $\pm 5\%$ from the origin, which is well within the accuracy of the fitted data.

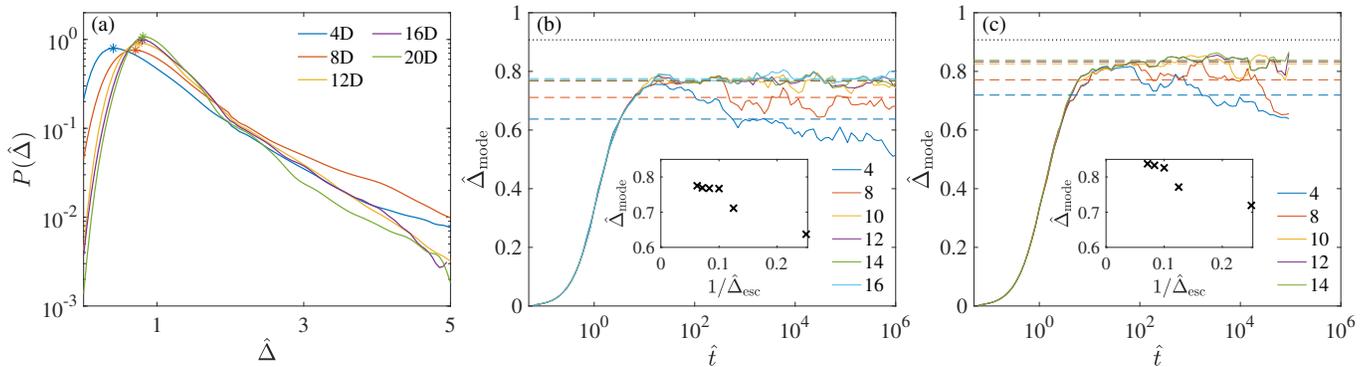


FIG. S1: (a) Dynamical cage size distribution in $\hat{\varphi} = 2.7$ at $\hat{t} = 2^{16}$ in various dimensions. The modal cage size $\hat{\Delta}_{\text{mode}}$ is denoted by asterisks. (b, c) $\hat{\Delta}_{\text{mode}}(t)$ in $\hat{\varphi} = 2.7$ and (b) $d = 16$ and (c) $d = 20$ obtained for systems with different shell thickness $\hat{\Delta}_{\max}$. Colored dashed lines denote the plateau value of $\hat{\Delta}_{\text{mode}}$ and also plotted in insets. The black dotted line denotes the mean-field prediction.

-
- [1] R. Mari, F. Krzakala, and J. Kurchan, *Phys. Rev. Lett.* **103**, 025701 (2009).
 - [2] P. Charbonneau, Y. Jin, G. Parisi, and F. Zamponi, *Proc. Natl. Acad. Sci. U.S.A* **111**, 15025 (2014).
 - [3] Y. Jin and P. Charbonneau, *Phys. Rev. E* **91**, 042313 (2015).
 - [4] F. Höfling, T. Franosch, and E. Frey, *Phys. Rev. Lett.* **96**, 165901 (2006).
 - [5] G. Parisi, P. Urbani, and F. Zamponi, *Theory of simple glasses: Exact Solutions in Infinite Dimensions* (Cambridge University Press, Cambridge CB2 8BS, United Kingdom, 2020).
 - [6] M. Mangeat and F. Zamponi, *Phys. Rev. E* **93**, 012609 (2016).
 - [7] J. Kurchan, G. Parisi, and F. Zamponi, *J. Stat. Mech. Theory Exp.* **2012**, P10012 (2012).
 - [8] G. Parisi and F. Zamponi, *Rev. Mod. Phys.* **82**, 789 (2010).
 - [9] B. Charbonneau, P. Charbonneau, and G. Tarjus, *J. Chem. Phys.* **138**, 12A515 (2013).
 - [10] A. V. Tomilov, *Header-only single-class implementation of the quickhull algorithm for convex hulls finding in arbitrary dimension space* (2016), based on [21, 22], URL <https://github.com/tomilov/quickhull>.
 - [11] J.-D. Boissonnat and C. Delage, in *European Symposium on Algorithms* (Springer, 2005), pp. 367–378.
 - [12] M. E. J. Newman and R. M. Ziff, *Phys. Rev. E* **64**, 016706 (2001).
 - [13] S. Mertens and C. Moore, *Phys. Rev. E* **86**, 061109 (2012).
 - [14] D. Stauffer and A. Aharony, *Introduction To Percolation Theory* (Taylor & Francis, 1994).
 - [15] Z. Koza and J. Poła, *J. Stat. Mech. Theory Exp.* **2016**, 103206 (2016).
 - [16] O. Devillers, S. Hornus, and C. Jamin, in *CGAL User and Reference Manual* (CGAL Editorial Board, 2019), 4.14 ed., URL <https://doc.cgal.org/4.14/Manual/packages.html#PkgTriangulations>.
 - [17] P. L. Leath, *Phys. Rev. B* **14**, 5046 (1976).
 - [18] S. Sastry, D. S. Corti, P. G. Debenedetti, and F. H. Stillinger, *Phys. Rev. E* **56**, 5524 (1997).
 - [19] L. Devroye, *Non-uniform random variate generation* (Springer-Verlag, New York, New York 10010, USA, 1986).
 - [20] F. Höfling, T. Munk, E. Frey, and T. Franosch, *J. Chem. Phys.* **128**, 164517 (2008).
 - [21] C. B. Barber, D. P. Dobkin, D. P. Dobkin, and H. Huhdanpaa, *ACM Trans. Math. Softw.* **22**, 469 (1996).
 - [22] K. Mehlhorn, S. Näher, M. Seel, R. Seidel, T. Schilz, S. Schirra, and C. Uhrig, *Comput. Geom* **12**, 85 (1999).