Structure of Basins of Attraction of Soft sphere packings

by

Praharsh Suryadevara

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Physics

New York University

December, 2025

Stefano Martiniani

To strive, to seek, to find, and not to yield.

— Alfred Lord Tennyson, *Ulysses*

Acknowledgements

First of all, I would like to thank my advisor Stefano. Stefano has shown me how to think broadly about problems in science with little regard for the artificial distinctions between fields. I was lucky enough to be part of his group from the earliest days in Minnesota until now. I have also definitely picked up a personality trait or two since the very first semester I interacted with him.

I am extremely indebted to Mathias Kasiulis. It has been helpful to have a sounding board for my stream of consciousness on science, jokes, life, and everything under the sun. It was incredibly helpful to have a collaborator who believed deeply in the science I did, even in situations I had no belief that my own work would pan out. Not to mention the thesis would not have been possible without detailed chapter-by-chapter constructive comments from him. I am both better as a person and a scientist because he was in my life.

I am grateful to my committee members Matthew Kleban, Dries Sels, and Glen Hocky. I'm particularly grateful to Alexander Grosberg, whose thought provoking questions have contributed a lot to this thesis. I am extremely grateful to Shenglong Wang and the NYU HPC team, for responding to me at all the odd hours I ran my simulations.

Amongst friends, I cannot thank Shivang enough. I could not have asked for a better labmate, officemate, roommate, and friend. I am so incredibly grateful that my PhD led me to becoming friends with you. I'm grateful to the rest of my group: Aaron, Satyam, Asit, Flaviano, Guanming, JiYeon, Maya, Philipp, Tom, Mia, Akshada, and Greg; the rest of CSMR; and both NYU and the UMN physics departments for surrounding me with cool science in a supportive environment.

I am grateful to my UMN cohort: Sushmita, Mohit, Khilav, Joe, Jenna, Konstantin, Aset, Richard, Juan, and Axel, for their support and friendship through the early days of my PhD.

I am grateful to Chirag, Amrita, Govind and Mooli for giving me a social life beyond work in NYC. I am extremely grateful to Jaya, Anushka, and Mahesh for making me a better person and being there for me in many ways large and small in my PhD, and more generally for more than a decade since I was in my teens.

There are more friends I'm grateful for than I can include here. That does not lessen the fact that every one of them has contributed meaningfully to my life in a way that led to the writing of this thesis.

I cannot thank my family enough. I've been very lucky to have supportive and understanding parents who've been a constant source of inspiration. I'm grateful to my grandparents and my extended family for making sure I was always loved throughout my life. I'm grateful to Ravi Mama, Kalyanakka, and Chinnu Atta for making me feel at home in the US, and the Dippakay-alapadu gang for making sure I had a never-ending supply of pala-kovas I did not know what to do with.

Finally, I'm thankful to Ben and Jerry's for all the pints of ice cream that kept me going, and running for making sure I was able to burn through the ice cream to still keep going on my legs.

Abstract

The energy landscape picture is a central tool to study many-body systems. In particular, the energy landscapes of glass-forming liquids, jammed packings, constraint satisfaction problems, or neural networks contain a plethora of minima corresponding to competing states. Due to their complexity, these landscapes resist analytical treatment and must be studied numerically. We focus on jammed soft spheres, a paradigmatic model of glasses and granulars, to expose the limitations of standard numerical methods commonly used since the 1980s in the study of liquids, glasses and jammed systems.

Using this numerical approach, we provide unequivocal evidence that optimizers widely used in computational studies destroy all semblance of the true landscape geometry, even in moderately low dimensions. Employing a range of geometric indicators, both low- and high-dimensional, we show that earlier claims on the fractality of basins of attraction of minima originated from the use of inadequate mapping strategies. In reality, the basins of attraction of soft sphere packings are smooth structures with well-defined length scales

Contents

Dedication **iii**

Acknowledgments **iv**

Abstract **vi**

List of Figures **xii**

List of Tables **xx**

1 Prelude **1**

1.1 A dynamical systems perspective 3

1.1.1 On structure 4

1.1.2 On volume 7

1.2 Basins on an energy landscape 9

1.2.1 On methods 12

1.2.2 On structure 13

1.2.3 On volume 13

1.3 Thesis Organization 15

2 Model and Methods **18**

2.1 Introduction 18

2.2 Model . . . 19
2.3 Optimizers . . . 21
2.3.1 Gradient Descent . . . 21
2.3.2 Momentum-Based Methods: FIRE . . . 22
2.3.3 Quasi-Newton Methods: L-BFGS . . . 24
2.3.3.1 The BFGS Update . . . 24
2.3.3.2 Limited-Memory BFGS . . . 26
2.3.3.3 Step Size Control and Modifications . . . 27
2.4 Implicit ODE Solvers: BDF Methods . . . 28
2.4.1 Stiffness and the Failure of Explicit Methods . . . 28
2.4.2 Implicit Methods . . . 29
2.4.3 Backward Differentiation Formulas . . . 31
2.4.4 CVODE . . . 31
2.4.5 Cost Tradeoff . . . 33
2.5 Benchmarking ODE Solvers . . . 33
2.5.1 Performance on Other Systems . . . 36
2.6 Discussion . . . 37

3 Accuracy and Basic Observables **39**
3.1 Introduction . . . 39
3.2 Method Description . . . 41
3.3 Distribution of Energies at Minima . . . 42
3.4 Jamming point . . . 44
3.5 Accuracy of Optimizers . . . 46
3.6 Visualizing Basin Scrambling . . . 48
3.6.1 Procedure for Generating 2D Energy Landscape Slices . . . 48

3.6.2 Landscape slices . . . 49
3.6.3 Dependence on Packing Fraction . . . 51
3.7 Discussion . . . 54

4 Geometry of basins **57**
4.1 Introduction . . . 57
4.2 Line Cuts . . . 59
4.2.1 Method . . . 59
4.2.2 Results and Discussion . . . 60
4.3 Box-Counting Dimension of Basin Boundaries . . . 61
4.3.1 Measurement Procedure . . . 62
4.3.2 Results . . . 62
4.4 Survival Probability . . . 65
4.4.1 Method . . . 65
4.4.2 Fractality . . . 66
4.4.3 Method Dependence of Survival Curves . . . 69
4.4.4 Survival Probability Across Packing Fractions . . . 69
4.4.4.1 Fitting Methodology . . . 70
4.4.4.2 Fit Quality Comparison . . . 71
4.5 Basin Arrangement . . . 72
4.6 Discussion . . . 75

5 Sampling uniformly in a high-dimensional basin **77**
5.1 Introduction . . . 77
5.2 Drawing uniform samples from a high-dimensional basin . . . 78
5.2.1 The problem with a single chain . . . 79
5.2.2 Parallel Tempering . . . 80

5.2.3 Estimating a normalizing constant: MBAR . . . 83
5.3 Procedure . . . 84
5.3.1 Finding $k_{\max}$ . . . 84
5.3.2 Implementation details of parallel tempering . . . 85
5.3.3 Inner sphere sampling . . . 87
5.3.4 Estimating volume . . . 88
5.3.5 Computing the radial density of states . . . 90
5.4 Validation on hypercubes . . . 92

6 Volume and Radial Structure of Basins 94
6.1 Introduction . . . 94
6.2 Basin volumes . . . 95
6.3 Density of States . . . 96
6.4 Accuracy vs Distance . . . 97
6.5 Basin Shape Comparison . . . 99
6.6 Half-Survival Radius . . . 99
6.6.1 Methods . . . 99
6.6.2 Results . . . 102
6.7 Discussion . . . 103

7 Conclusions and Future directions 105
7.1 Primary Contributions . . . 105
7.2 Where can we go from here . . . 107
7.2.1 Jamming and glasses . . . 107
7.2.2 Basin volume methodology . . . 107
7.2.2.1 Caveats . . . 108
7.2.3 Other fields . . . 108

7.3 Concluding remarks . . . 109

A Convergence Criterion **111**

B Minima Matching Procedure **113**

B.1 Rattler Identification . . . 113

B.1.1 Geometric Criterion . . . 113

B.1.2 Convex Hull Test in Two Dimensions . . . 114

B.1.3 Iterative Identification . . . 115

B.1.4 Jamming Stability Check . . . 116

B.2 Structure Comparison . . . 116

B.2.1 Pre-checks . . . 116

B.2.2 Translational Alignment . . . 117

B.2.3 Distance Computation with Periodic Boundaries . . . 117

B.2.4 Matching Criterion . . . 118

C Solver Parameters **119**

C.0.1 Choices of solver parameters . . . 119

C.0.1.1 CVODE Parameters . . . 119

C.0.1.2 FIRE, LBFGS, and Gradient Descent parameters . . . 122

D Video Description **124**

Bibliography **125**

List of Figures

1.1 A schematic rugged energy landscape with a multitude of energy minima, maxima, and saddles. Arrows denote some of the possible relaxation pathways. Figure credit: Chiara Cammarota, Simons Collaboration on Cracking the Glass Problem [13]. 2

1.2 Illustration of the uncertainty exponent algorithm. Points are sampled uniformly in phase space and classified as uncertain if a disk of radius ε centered at the point intersects a basin boundary. Solid circles indicate uncertain points (disk crosses boundary); the dashed circle shows a certain point (disk remains within basin A). 6

1.3 Monte Carlo estimation of a circular basin's "volume" and basin stability. Points are sampled uniformly in the square; the fraction falling inside the circle (blue) estimates the ratio of areas. 8

2.1 Jammed packings of $N = 512$ bidisperse Hertzian disks in two dimensions. Left: at $\phi = 0.845$, blue particles form the mechanically stable backbone while orange particles are rattlers—particles with insufficient contacts to be part of the rigid structure. Right: at $\phi = 0.9$, transparency reveals particle overlaps as darker regions where disks interpenetrate; All blue particles have overlaps, although overlaps are more clearly visible at $\phi = 0.9$. Both packings are confined to periodic square boxes and interact via the repulsive contact potential (Eq. 2.1). 19

2.2 The newton optimization trajectory on an anisotropic quadratic energy landscape $E(\mathbf{x}) = x_1^2 + 10\,x_2^2$. Blue: Newton's method converges to the minimum $\mathbf{x}^*$ in one step. Orange: gradient flow follows the steepest descent ODE, curving along the elongated contours. 25

2.3 Steepest descent on an ill-conditioned quadratic $E(\mathbf{x}) = x_1^2 + 100\,x_2^2$. The trajectory from $\mathbf{x}_0$ to the minimum $\mathbf{x}^*$ quickly decays along the short direction (high curvature, $\lambda_{\max}$) then slowly traverses the long direction (low curvature, $\lambda_{\min}$). Forward Euler's step size is constrained by the shortest direction $\lambda_{\max}$, yet most computational effort is spent moving along the long direction. 29

2.4 Distribution of condition numbers $\lambda_{\max}/\lambda_{\min}$ for $N = 512$ Hertzian packings in $2d$ at $\phi = 0.845$. The 491 jammed minima were obtained via L-BFGS minimization from random initial conditions. The mean condition number is $(4.2 \pm 0.3) \times 10^4$ (standard error), with a maximum of $8.7 \times 10^5 \sim O(10^6)$. 30

2.5 **Benchmarking ODE solvers.** (a)–(e) Plots of the Mean Trajectory Distance versus Average Computation Time for different numbers of particles $N = 8$, 16, 32, 64, and 128, respectively. Error bars represent the standard error of the mean (SEM) for both the Mean Trajectory Distance and Average Computation Time. Points highlighted by filled triangles correspond to the maximum tolerance values used among all calculations. 35

2.6 **Benchmarking ODE solvers on other systems.** (a) Interacting pairwise potential with harmonic interactions. (b) 1D XY model (Kuramoto). (c) Frustrated XY model on a triangular lattice. Error bars represent the standard error of the mean (SEM) for both the Mean Trajectory Distance and Average Computation Time. For the XY model [(b) and (c)] we utilize Jacobian-free GMRES solvers provided by the libraries for all implicit methods (CVODE, QNDF, FBDF) for speed. 37

3.1 **Energies at minima.** (a) Differences between mean energies of minima for each method and the mean energies obtained with CVODE against N, in log-linear scales. The CVODE line is represented to indicate standard error on the CVODE mean. (b) Relative error on the energies with respect to CVODE against $1/N$, in log-log scales. The dashed black line indicates $1/N^{1/6}$. (c) Empirical distribution of energies E at minima for all three methods for $N = 2048$. (d) Corresponding empirical distribution of reduced energies $e \equiv (E - \langle E \rangle)/\sigma_E$, in dashed lines. Throughout the figure, we encode CVODE by green, FIRE by orange, and L-BFGS by blue. Error bars on the mean are obtained by bootstrapping over 1000 subsamples. 43

3.2 **Jamming Point.** (a) P_J against ϕ for $N = 16$ across methods. Student-T 95% confidence intervals are smaller than symbols. (b) ϕ_J against N across methods. Solid lines are critical power-law fits. Inset: relative error on ϕ_J compared to CVODE against N. Dashed black line: $N^{1/3}$ scaling (guide for the eyes). 45

3.3 **Jamming transition across sizes.** Fraction of jammed states as a function of packing fraction, across system sizes (color within panels), using (a) CVODE, (b) FIRE, and (c) L-BFGS. Error bars are Clopper-Pearson 95% confidence intervals. . 46

3.4 **Optimizers: fast but inaccurate.** (a) Average accuracy of algorithms, computed over 10^4 random points. Error bars are Clopper-Pearson 95% confidence intervals [131]. Dashed lines are exponential fits, long-dashed line stretched exponential fits. (b) Corresponding average computation times. Error bars are Student-T 95% confidence intervals. (c)–(f) 800×800-pixel slices of configuration space for $N = 128$ particles for (c) CVODE, (d) GD, (e) FIRE, and (f) L-BFGS. 47

3.5 **Slicing the energy landscape.** 1350×2400 pixels in a random $2d$ plane in the configuration space of $N = 16$ disks. At each pixel, we use CVODE (top) and FIRE (bottom), to identify which basin of attraction it belongs to. Each basin is uniquely encoded by one color across both panels. 50

3.6 **Small-system slices.** Random $2d$ slice of the energy landscape of $N = 8$ particles at $\phi = 0.9$ using (a) CVODE, (b) FIRE, and (c) L-BFGS. Minima are matched across slices then color-coded. 51

3.7 **Gallery of energy landscape slices.** Slices of the landscape of $N = 128$ particles at $\phi = 0.9$, obtained with the same parameters as in Fig. 3.4 but with different random seeds. 52

3.8 **Optimizers: fast but inaccurate at every ϕ.** (a) Average accuracy of algorithms compared to low-tolerance CVODE, computed over 10^4 random uniform initial conditions. Error bars are 95% confidence intervals, obtained using a Clopper-Pearson estimator. Dashed lines are exponential fits, and the long-dashed line is a stretched exponential fit. (b) Corresponding scalings of the average computation times, with error bars obtained from a Student-T 95% confidence interval. (c)–(e) 800×800-pixel slices for $N = 128$ particles for (c) CVODE, (d) FIRE, and (e) L-BFGS. 53

3.9 **Optimizer accuracy dependence on volume fraction.** Accuracy of (a) L-BFGS and (b) FIRE optimizers compared to high-precision CVODE solutions as a function of volume fraction ϕ. Data points show the fraction of random initial conditions that yield identical minima to the CVODE reference, ignoring fluid states, for system sizes $N = 16$ (blue circles) and $N = 32$ (orange circles). Solid lines represent linear regression fit guidelines. Error bars are Clopper-Pearson 95% confidence intervals. 54

3.10 **Density and landscape slices.** Slices of the energy landscape for $N = 128$ and at ϕ equal to (a) 0.828, (b) 0.830, (c) 0.835, (d) 0.84, (e) 0.845, (f) 0.85, (g) 0.86, (h) 0.9. Individual basins are encoded by colors, while liquid states are shown as black pixels. 55

4.1 **Linear intersects of basins.** (a) Intersection lengths distributions obtained with L-BFGS (blue triangles) and CVODE (green squares) over 10 lines of 10^6 pixels for $N = 16$, in log scales. A dashed line indicates $1/\ell$ behavior. (b) CDF of the distribution of $\log \ell$ obtained with CVODE by zooming 100× on each basin boundary found from panel (a). The dashed black line is a truncated Gaussian fit. Inset: Corresponding histogram of the pdf, the dashed line shows a kernel regression. . 60

4.2 **Box-counting dimension.** (a) Gaussian kernel density estimate of the PDF of the box counting dimension d_B from top 10 largest basins for $N = 128$ and $\phi = 0.9$, and (b) corresponding average box counting dimension estimated from the top n_B largest basins. In (c), (d), we show the same plots for $N = 128$ and $\phi = 0.86$. . 63

4.3 **Survival probability for the Sierpinski carpet.** The survival probability $P_{\text{in}}(R)$ as a function of perturbation radius R, averaged over 100 reference points sampled uniformly from inside the carpet. The dashed line shows the predicted scaling $P_{\text{in}}(R) \sim R^{d_f - D}$ with $d_f - D = \log 8/\log 3 - 2 \approx -0.107$. Error bars are Clopper-Pearson 95% confidence intervals. 67

4.4 **Survival** (a) Survival from a random point in the landscape of $N = 1024$ disks, in log-log scale. Solid black lines are stretched-exponential fits for CVODE and FIRE. Dashed gray line: saturating power-law fit to L-BFGS. Error bars are BCa bootstrap 95% confidence intervals [141]. (b) Same curve at $\phi = 0.85$. Gray lines are power-law fits for FIRE and L-BFGS. The black line is a stretched-exponential fit to CVODE. 68

4.5 **Survival probability approaching jamming.** Survival probability as a function of perturbation radius R for several packing fractions approaching the jamming transition, shown in log-log scale. Panels show results for (a) CVODE, (b) FIRE, and (c) L-BFGS. Error bars are BCa bootstrap 95% confidence intervals [141]. 70

4.6 **Fit quality comparison.** (a) Proportion P_{pl} of leave-one-out ensembles for which the power-law fit outperforms the stretched exponential, as a function of packing fraction. Panels (b)–(f) show the fitted parameters for each model and method: (b) R_{se}, (c) α, (d) A, (e) R_{pl}, and (f) β. 73

4.7 **Distance between neighboring minima.** Average normalized distance $\overline{d_{\mathrm{min}}}$ between an initial minimum and minima reached after perturbations of size R, for $N = 1024$ particles. Inset: relative error in $\overline{d_{\mathrm{min}}}$ for optimizers compared to CVODE, shown on a logarithmic scale. 74

5.1 Volume concentration in high-dimensional hypercubes. *Top*: Visualizations of hypercubes in $d = 2, 3, 4$, and 5 dimensions, with color indicating the radial distribution of volume. As dimensionality increases, volume shifts from the center (dark) toward the corners (light). *Bottom*: The relative weight $w(r)$ of each radial shell—the fraction of total volume contained in a thin spherical shell at distance r from the center—for the hypercube (orange) compared to a hyperball (purple). In low dimensions, the hypercube's volume peaks near its inscribed sphere, but as d grows, the distribution shifts to larger radii where the corners dominate, illustrating why a random walker in high dimensions becomes trapped in corners far from the central core. Figure adapted from Ref. [142]. 81

5.2 Validation of the volume estimation method on unit hypercubes. The estimated free energy $\langle F_0 \rangle$, which should equal zero for a unit hypercube, is plotted against dimension d for three minimizers: CVODE (blue), FIRE (orange), and L-BFGS (green). Error bars represent the standard error ($n = 10$, except $n = 3$ for $d = 512$). CVODE and FIRE remain consistent with zero across all dimensions, while L-BFGS begins to deviate at high d. 93

6.1 **Free energies.** Intensive free energies F_0/N across methods, against N, each averaged over the same 5 basins for each N. 96

6.2 **Density of states.** Example densities of states (DOS) for one basin per N, comparing CVODE (left) and FIRE (right). 97

6.3 **Accuracy vs distance.** Averaged accuracy over $O(10^5)$ FIRE samples used, against their distance to the minimum, in semi-log scale. Dashed lines are exponential fits $y = C \exp(-\lambda x)$ of each curve. Inset: best decay rate λ against N in log-log, with a dashed power-law $\lambda \sim N^{0.6}$. 98

6.4 **Basin shape.** $\log_{10}$ of the ratio between the CVODE DOS and that of a hyperball in $N(d-1)$ dimensions. Dashed lines are exponential fits. Inset: decay rate μ against N in log-log, with a dashed power-law $\mu \sim N^{0.9}$. 100

6.5 **Half-survival radius.** Half-survival radius $R_{1/2}$ inside a single basin against distance r to its minimum for $N = 128$. Lines indicate proposed fits: exponential (gray) for FIRE and L-BFGS, stretched-exponential (black) for CVODE. 102

C.1 **CVODE Accuracy and Time vs Tolerance.** For each N, we vary `rtol` and report accuracy and time, averaged across 10^4 random initial conditions. Error bars represent Clopper-Pearson confidence intervals for accuracy and standard error for time. Highlighted points correspond to points used in the main text, corresponding to $> 98\%$ accuracy. The point at the lowest rtol is taken to have accuracy 1.0 121

List of Tables

4.1 **Box-counting dimensions.** Table of d_B values (with bootstrapped 95% confidence intervals) estimated from the 10 largest basins in a slice, for each method. 64

C.1 **Tolerances for CVODE.** Table of CVODE tolerances we choose. For small systems, while a higher `rtol` is optimal for performance, the resulting numerical noise can prevent the gradient from satisfying the 10^{-10} convergence threshold close to a minimum. Here iterative means that an iterative Newton-Krylov scheme is used to solve the Newton's equation instead of using the dense Hessian. 120

C.2 FIRE algorithm parameters 123

C.3 L-BFGS algorithm parameters 123

C.4 Gradient Descent algorithm parameters 123

1 | Prelude

All stable processes we shall predict. All unstable processes we shall control.

— John von Neumann

A broad class of systems across the sciences are well described by properties a high-dimensional energy (or cost) function with an intractable number of minima, ranging from the condensed matter systems such as glasses[1–3], to biomolecular processes such as the folding of proteins[1, 4], to modern machine learning architectures. In each of these domains, the energy landscape paradigm provides a unifying language. Glasses are understood in terms of relaxation slowed amongst a number of metastable states on a rough landscape (Fig. 1.1), proteins fold by descending in a funneled landscape towards the native states, neural networks are trained by descending on a loss landscape whose structure and geometry determine both learning dynamics and generalization [5–12].

The simplest relaxation process to define in an energy or cost function is steepest descent. In the context of energy functions in statistical physics, this choice of dynamics is particularly meaningful, as it describes the dynamics followed by a system after a sudden quench to zero temperature. This perspective admits a dynamical-systems formulation, where steepest descent defines a deterministic mapping from initial conditions to asymptotic minima. Within this framework, one may define the *basin of attraction* associated with a minimum as the set of all initial configurations whose trajectories under steepest descent converge to that minima. The statisti-

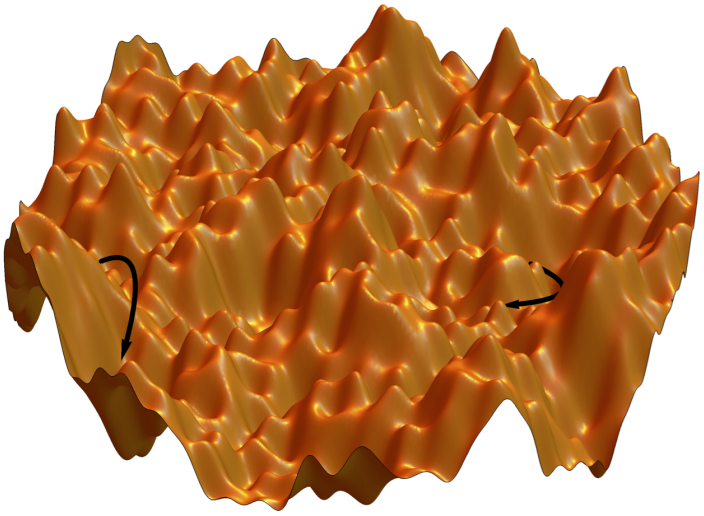

Figure 1.1: A schematic rugged energy landscape with a multitude of energy minima, maxima, and saddles. Arrows denote some of the possible relaxation pathways. Figure credit: Chiara Cammarota, Simons Collaboration on Cracking the Glass Problem [13].

cal and geometric properties of these high dimensional basins play a fundamental role in understanding a large variety of systems. For example the volumes of these basins have been evoked to understand models of associative memory[14, 15], as a way to estimate configurational entropy in jammed systems[16–18] and glasses and as a measure of stability beyond the traditional linear stability paradigm [19]. Moreover the volume of the sync basin on a ring of kuramoto oscillators has been a topic of rich debate in the dynamical systems community [20–22].

In this thesis we focus on a common model of soft sphere packings [16–18, 23–36] an archetypal model of glasses and granulars. We emphasize that this model, and more broadly glassy landscapes in general have deeper connections to other fields, notably constraint satisfaction [37–39], machine learning [10, 40–47] and ecology[48–52]. The approaches we use in this thesis

also extend beyond glassy landscapes to any problem that involve a study of a high-dimensional energy-like function with many minima, from biological molecules [1, 53] to string theory [54, 55].

1.1 A dynamical systems perspective

We start with a standard analysis that arises across multiple fields, concerning a differential equation of the form

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{f}(\boldsymbol{x}) \tag{1.1}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^N$ is the state vector, a complete specification of the system at time t—and $\boldsymbol{f} : \mathbb{R}^N \to \mathbb{R}^N$ is a vector field that governs the dynamics. The space $\mathbb{R}^N$ of all possible states is called the *phase space*, and the vector field $\boldsymbol{f}$ defines a *flow*: each point $\boldsymbol{x}$ in phase space has an associated velocity $\boldsymbol{f}(\boldsymbol{x})$ that determines how the system evolves from that state.

For many systems, trajectories in phase space often converge toward invariant regions, called attractors, regions, once entered, the system cannot leave. More formally, an invariant region A is called an *attractor* if all trajectories starting sufficiently close to it are drawn back: formally, for initial conditions $\boldsymbol{x}(0) = \boldsymbol{x}_0 + \boldsymbol{\epsilon}$ with $\boldsymbol{x}_0 \in A$ and $\boldsymbol{\epsilon}$ small, we have $\lim_{t\to\infty} d(\boldsymbol{x}(t), A) = 0$, where d denotes a suitable distance metric. For example, when A is a single point, the L_2 norm suffices.

To determine stability, we analyze how small perturbations $\boldsymbol{\epsilon}$ from a fixed point evolve. The traditional linear stability analysis proceeds as follows. We expand the perturbation in the differential equation, giving us

$$\frac{\mathrm{d}\boldsymbol{\epsilon}}{\mathrm{d}t} = J(\boldsymbol{x}^*)\boldsymbol{\epsilon}, \quad J_{ij}(\boldsymbol{x}^*) = \frac{\partial}{\partial x_j} f_i(x^*) \tag{1.2}$$

where $J(\boldsymbol{x})$ is the Jacobian matrix at $\boldsymbol{x}$. Let $\boldsymbol{v}_i$ be eigenvectors of $J(\boldsymbol{x}^*)$ with eigenvalues λ_i. Expand the perturbation as

$$\boldsymbol{\epsilon} = \sum_i \boldsymbol{v}_i x_i' = V\boldsymbol{x}' \tag{1.3}$$

where $V = [\boldsymbol{v}_1\, \boldsymbol{v}_2\, \cdots]$ and $\boldsymbol{x'} = (x_1', x_2', \ldots)^T$. The linearized equation becomes

$$\frac{\mathrm{d}x_i'}{\mathrm{d}t} = \lambda_i x_i' \tag{1.4}$$

with decoupled solutions: $x_i'(t) = x_i'(0)e^{\lambda_i t}$. In the case that all the real parts of λ_i are negative, we have an attractor. The slowest timescale, $|1/\mathrm{Re}(\lambda_{\min})|$, where $\lambda_{\min}$ is the eigenvalue with the smallest (in magnitude) real part, determines the rate of convergence to the attractor.

This approach constitutes a *local* stability analysis—local because it is valid only for infinitesimally small perturbations $\boldsymbol{\epsilon}$ from the fixed point, where the linear approximation in Eq. 1.2 holds. Traditional textbook analyses [56, 57] focus heavily on such linear stability analyses, which are widely employed across scientific disciplines.[1].

However, the question of *global* stability remains: how does the system behave under *finite* perturbations, where linearization breaks down? A natural framework for addressing this question is the *basin of attraction* $B(A)$—the set of all initial conditions in phase space that asymptotically converge to attractor A as $t \to \infty$. In this thesis, we focus on two complementary properties of basins: their geometric structure (Chapter 4) and their volume (Chapter 6). These concepts have deep roots in the dynamical systems literature. We first review how basins are characterized in the dynamical systems literature with a historical perspective on developments, while noting that the approaches we use are different in this thesis.

1.1.1 On structure

The preceding analysis raises a practical question: in real systems, initial conditions are never known exactly. Given some measurement uncertainty, how confident can we be about which attractor the system will reach? A seminal approach to solve this problem, developed by Grebogi, Ott, Yorke, and collaborators [58–61], characterizes the fractality of basin boundaries to measure

[1]Eq. 1.4 can be interpreted as N independent overdamped harmonic oscillators, where the damping term fully dominates any acceleration, following our usual strategy of starting by treating everything as a harmonic oscillator.

final state uncertainty. To illustrate their viewpoint, consider a 2D state space partitioned into two different basins as shown in Fig. 1.2. Given an initial condition known only to within some uncertainty radius ϵ, how confident can we be about which attractor the system will reach? If the initial condition lies far from any boundary, the outcome is certain regardless of ϵ. However, if a basin boundary passes through the uncertainty disk, the final state becomes ambiguous as small perturbations could lead to qualitatively different outcomes. This motivates a quantitative measure of final state uncertainty through the following procedure

1. Sample N points at random in the phase space.
2. For each point, Determine whether a basin boundary intersects in a disk of size ϵ.
3. Classify the points where intersections are found as uncertain and compute the fraction of "uncertain" states $f(\epsilon)$

The scaling behaviour of $f(\epsilon)$ reveals the structure of boundaries. typically [61]

$$f(\epsilon) \sim \epsilon^{\alpha}, \tag{1.5}$$

Here α is called the *uncertainty exponent*. Intuitively, $f(\epsilon)$ measures the fraction of phase space where small uncertainties in initial conditions lead to unpredictable outcomes, points near basin boundaries where perturbations of size ϵ could push the trajectory into a different basin.

The uncertainty exponent is related to the fractal dimension of basin boundaries through [61]

$$\alpha = D - d, \tag{1.6}$$

where D is the dimension of phase space and d is the *box-counting dimension* of the basin boundaries. The box-counting dimension measures how the boundary fills space: tile space with boxes of side length ϵ and count the minimum number $N(\epsilon)$ needed to cover the boundary. The box-

counting dimension d is defined by

$$d = -\lim_{\epsilon \to 0} \frac{\log\left(N(\epsilon)\right)}{\log\left(\epsilon\right)} \tag{1.7}$$

For smooth boundaries, $d = D - 1$ and $\alpha = 1$. Fractal boundaries have $d > D - 1$, yielding $\alpha < 1$ i.e they fill more of phase space than a smooth boundary.

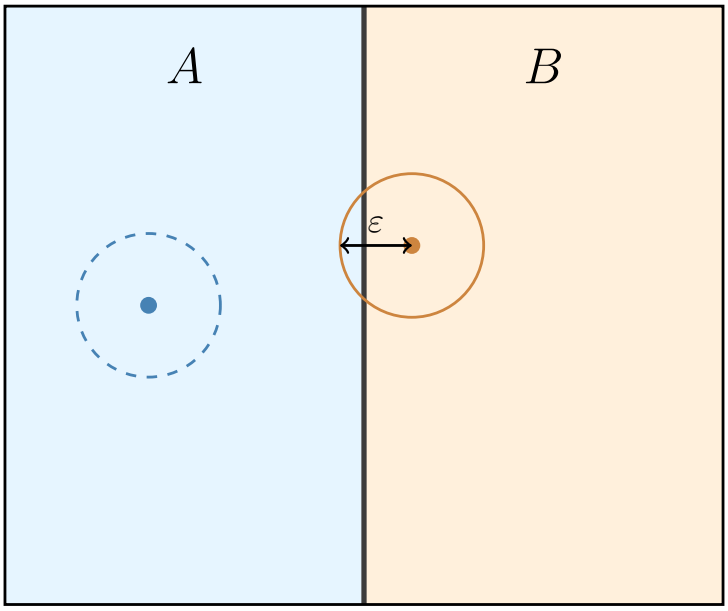

Figure 1.2: Illustration of the uncertainty exponent algorithm. Points are sampled uniformly in phase space and classified as uncertain if a disk of radius ε centered at the point intersects a basin boundary. Solid circles indicate uncertain points (disk crosses boundary); the dashed circle shows a certain point (disk remains within basin A).

This geometric approach faces severe computational challenges in high dimensions because the cost to determine whether a disk intersects a boundary scales as $O(N^D)$, making direct application prohibitively expensive as D increases. While optimizations exist, the vast majority of systems studied by this method in the dynamical systems community have state spaces of fewer than ten dimensions, primarily damped oscillators [62] and ecological competition models [63, 64]. In high dimensions, we address this scaling by using a related measure described in Chapter 4.

1.1.2 On volume

Fractal boundaries are one measure of stability, however a basin with smooth boundaries as stable if it has occupies a comparative small volume in phase space. The importance of basin volumes for predicting convergence was first identified by Wiley, Strogatz, and Girvan [20] in the context of Kuramoto oscillators on a ring, where the synchronized state competes with "twisted" states characterized by a winding number q. Initial investigations using naive sampling suggested that the synchronization basin scales as e^{-kq^2} [20], whereas a semi-analytical treatment in hypercubic shells proposed an $e^{-k|q|}$ scaling [65]. This discrepancy was resolved when Zhang and Strogatz demonstrated that the volume of the synchronization basins concentrates in "tentacles" extending through phase space [19], a geometric feature that independently confirms similar results observed in jammed sphere packings [27]. The originally proposed e^{-kq^2} scaling was subsequently verified analytically by Groisman [66]. a measure based on the basinvolume, *basin stability* was formally defined in [19] as the relative volume of an attractor's basin compared to that of the full phase space. In lines of work by Kurths and collaborators, basin stability has since been applied to characterize the robustness of power grids [67, 68], analyze climate tipping cascades [69], and investigate dynamics in neural networks [70]. The standard numerical procedure utilized in these works relies on a full sampling of the phase space using a naive monte carlo method proceeding as follows:

1. Sample N initial conditions from a specified region of interest, Q, in the D-dimensional phase space according to a distribution ρ (typically uniform random).

2. For each sampled initial condition $\boldsymbol{x}_0$, numerically integrate the system dynamics forward in time until reaching the asymptotic state $\boldsymbol{x}^*$ (after time t^*).

3. Classify each trajectory into one of k attractors based on the final state.

4. Estimate the basin stability estimate for each attractor i as $\hat{S}_B(\mathcal{A}_i) = N_i/N$, where N_i is the

number of initial conditions converging to attractor i. This ratio represents the estimated volume fraction of that basin in the region of interest.''

5. We can also estimate the volume of the basin of attraction $\hat{v}(\mathcal{A}_i) = V_{\text{tot}}\hat{S}_B(\mathcal{A}_i)$ where V_{tot} is the the total volume of phase space

We note that this method is the same as the "first" example encountered in introductory Monte Carlo courses: estimating π by sampling uniformly in a square and counting the fraction of points falling inside an inscribed circle (Fig. 1.3). For a circular basin of radius r inscribed in a phase space of side $2r$, the basin stability estimate is $\hat{S}_B(\mathcal{A}_{\text{circle}}) = \pi/4 \approx 0.785$, and the estimated "volume" is $\hat{v}_{\text{circle}} = 4r^2 \cdot (\pi/4) = \pi r^2$.

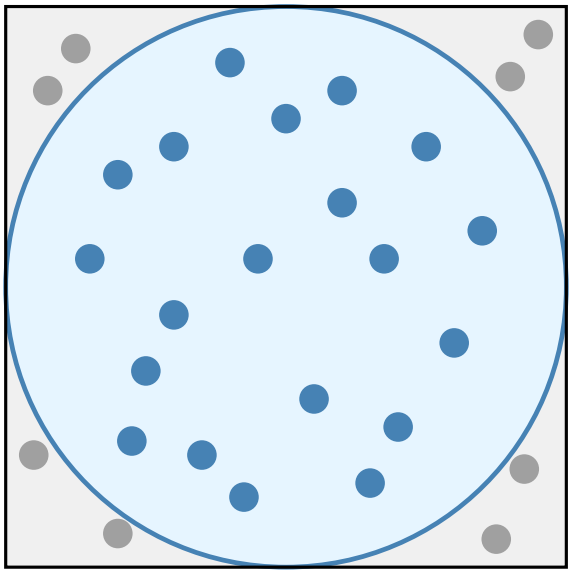

Figure 1.3: Monte Carlo estimation of a circular basin's "volume" and basin stability. Points are sampled uniformly in the square; the fraction falling inside the circle (blue) estimates the ratio of areas.

It is important to note, however, that this approach is computationally efficient only when the basin occupies a significant fraction of the sampling region. We show this limitation, by looking at estimating the volume of the volume fraction occupied by a hypersphere inscribed in a hypercube vanishes rapidly with dimension. In D dimensions, a hypersphere of radius r

inscribed in a hypercube of side $2r$ has volume ratio

$$S_B^{(D)} = \frac{\pi^{D/2}}{2^D \, \Gamma(D/2+1)}, \tag{1.8}$$

which decays approximately exponentially with dimension D. This ratio, which equals $\pi/4 \approx 0.785$ in two dimensions, drops to $\pi/6 \approx 0.524$ in three dimensions, $\sim 10^{-2}$ in ten dimensions, and becomes astronomically small thereafter (e.g., $\sim 10^{-70}$ for $D = 100$). so if we were to randomly sample inside our hypercube, we would never land within our hypersphere at large D

This limitation renders naive sampling infeasible for calculating volumes when basins occupy an extremely small fraction of phase space. For Kuramoto oscillators on a ring, verifying the e^{-kq^2} scaling at large winding number q becomes exceedingly difficult as these basins occupy a vanishing fraction of phase space [20, 21]. The same problem appears in the context of soft sphere packings, the focus of this thesis: the number of attractors grows exponentially with dimension [71], and every basin occupies such a small portion of phase space that the probability of landing in any minimum twice under uniform sampling approaches zero. An alternative approach to calculating basin volumes was developed by Frenkel and collaborators [16–18, 31] in this context. We use this approach in this thesis to calculate basin volumes; we discuss it in Sec. 1.2.3 and present it in detail with examples in Chapter 5.

1.2 Basins on an energy landscape

The inherent structure formalism, introduced by Stillinger and Weber [72], provides the foundational framework for connecting energy landscape structure to thermodynamics: steepest descent maps every configuration to a local minimum, partitioning configuration space into basins. To connect the ideas of structure discussed in Section 1.1.1 and volumes in Section 1.2.3 to the statistical physics of soft sphere packings, we formally specialize the dynamical systems framework of

Section 1.1 to the potential energy of a classical many-body system. Consider a classical system of N particles in a volume V in dimension d, described by the position vector $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N)$ in the configuration space Ω (called a phase space in the context of Section 1.1). The specific form of Ω depends on the boundary conditions: for example, for open boundaries $\Omega = \mathbb{R}^{Nd}$, while for periodic boundaries in a box of side L, $\Omega = (L\mathbb{T})^{Nd}$ where $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ is the unit torus. The potential energy is a function $E : \Omega \to \mathbb{R}$ that has a well-defined minimum. The topography of this surface is typically rugged[72, 73], characterized by a multiplicity of local minima. Although we consider only classical classical many-body systems here, high dimensional energy or energy-like functions with the same properties as $E(\boldsymbol{X})$ come up in a variety of fields such as ecology[74] or machine learning[75].

In the language of Eq. (1.1), the flow $\boldsymbol{f}(\boldsymbol{X})$ is determined by the negative gradient of the potential energy surface. The steepest descent equation

$$\frac{\mathrm{d}\boldsymbol{X}}{\mathrm{d}t} = -\boldsymbol{\nabla} E(\boldsymbol{X}) \tag{1.9}$$

determines a unique mapping: as $t \to \infty$, any configuration $\boldsymbol{X}(t)$ starting at $\boldsymbol{X}(0)$ converges to a specific local minimum, or *inherent structure*, denoted by $\boldsymbol{X}_{IS}$. It is also the fastest "quench" that does not allow for equilibration or barrier crossing. Here, t is a fictitious time coordinate representing the descent. The fixed points of this dynamical system are the stationary points of the potential energy surface where $\boldsymbol{\nabla} E(\boldsymbol{X}) = \boldsymbol{0}$. Specifically, we are interested in the stable fixed points (local minima), which correspond to the attractors discussed in Section 1.1. We can therefore formally define the *basin of attraction* associated with a specific minimum α, denoted as $\mathcal{B}_\alpha$, as the set of all initial configurations in the phase space that map to $\boldsymbol{X}_{IS}^{\alpha}$ under the gradient flow relaxation path:

$$\mathcal{B}_\alpha = \left\{ \boldsymbol{X} \in \Omega \mid \lim_{t\to\infty} \boldsymbol{X}(t) = \boldsymbol{X}_{IS}^{\alpha} \right\} . \tag{1.10}$$

The gradient flow partitions the entire configuration space, up to a set of measure zero, into disjoint basins of attraction:

$$\Omega = \left(\bigcup_{\alpha} \mathcal{B}_\alpha \right) \cup \mathcal{N}. \tag{1.11}$$

The zero-measure set $\mathcal{N}$ consists of basin boundaries (separatrices) and unstable fixed points such as saddles and maxima.

To illustrate the utility of this formulation, we briefly show how the basin picture provides a framework to obtain physics. (1.11) means that the total volume of configuration space V_{tot} becomes

$$V_{\text{tot}} = \sum_{\alpha=1}^{N_m} v_\alpha. \tag{1.12}$$

given that the mean volume is $\langle v \rangle = \frac{1}{N_m} v_\alpha$, we get the number of minima as

$$N_m = \frac{V_{\text{tot}}}{\langle v \rangle}, \tag{1.13}$$

We can define a Boltzmann-like entropy discussed in the packing context [17, 76],

$$S_B = \ln(N_m) - \ln(N!). \tag{1.14}$$

Alternatively, we define a Gibbs-like configurational entropy based on the probability of occupancy $p_\alpha = v_\alpha / V_{\text{tot}}$ [17]:

$$S_G = -\sum_{\alpha=1}^{N_m} p_\alpha \ln(p_\alpha) - \ln(N!). \tag{1.15}$$

Under the Edwards hypothesis [76], which assumes all valid packings are equiprobable, the Gibbs and Boltzmann entropies become equivalent. This equivalence was used to test the hypothesis for soft sphere packings in $2d$ [31].

1.2.1 On methods

While the steepest descent equation (Eq. 1.9) provides a mathematically precise definition of basins, solving this ODE numerically is computationally demanding. The computational expense of accurate ODE integration led to the widespread adoption of optimizer proxies—algorithms designed for rapid energy minimization rather than faithful trajectory integration. We note that while Stillinger's original work on liquids used ODE solvers well suited to the problem [72, 77], later work in the 1980s started a shift to optimizer proxies [78].

The problem with this shift is that most optimizers do not solve the steepest descent ODE, and optimization generates trajectories with different properties than solving the ODE. Despite these differences, modern jamming and glassy literature conflates solving steepest descent with solving an optimization problem, neglecting the differences that arise [36, 79–81]. Even when using methods that do solve the steepest descent ODE, the optimization framework persists, leading to a situation where for soft sphere packings of any significant size, researchers have explictly mentioned being in regimes where the inherent structure was sensitive to solver parameters [79], and finding the inherent structure has also been described as impossible above system sizes of 64 particles in $2d$ in 2025 [80].

We reject this framing in this thesis. In Chapter 2, we review the classes of numerical methods used in the literature, discuss the numerical aspects relevant to solving this problem correctly, and present a tool that finds the correct inherent structure at system sizes the community cares about, upto at least $N = 1024$ in $2d$ [2]. In Chapter 3, we examine the consequences of optimizer use for basin assignment; in Chapter 4, we examine the consequences for basin geometry; and in Chapter 6, we examine the consequences for basin volumes.

[2] While this thesis shows results for $N = 1024$, we have gone upto 4096

1.2.2 On structure

In Section 1.1.1, we discussed how uncertainty in initial conditions can lead to unpredictable final states, and how this phenomenon connects to the fractality of basin boundaries. In that framework we showed how the dynamical systems community tackled how to quantify arbitrarily small perturbations to initial conditions redirecting trajectories to different attractors.

Recent work has documented that numerical choices and slight perturbations significantly affect which minimum is obtained [79, 81]. These observations have been interpreted as suggesting that the landscape itself is "chaotic" [81], such that small perturbations to initial conditions lead to qualitatively different final states. However, no quantification of this phenomenon exists in the literature. Fractality has been discussed in several contexts in the potential energy landscape literature: the fractal organization of basins [33, 52, 82], the fractal structure of relaxation paths [29, 35, 83–85], and fractal basin boundaries [23, 28]. We emphasize that these distinct notions of fractality must be clearly separated.

Inspired by chaotic dynamical systems [56, 59, 86–88], it has further been suggested that the basins of attraction themselves may be fractal in neural networks [89, 90], constraint satisfaction problems [91, 92] and energy landscapes at large [23, 28, 36, 93–96].

In Chapter 4, we address claims of chaotic behavior and fractal boundaries, and show where the observed sensitivity to initial conditions originates. We develop a tool analogous to the uncertainty exponent of Grebogi, Ott, Yorke and collaborators [59, 61] that remains computationally tractable in high dimensions and can quantify both sensitivity and fractality to address this claim.

1.2.3 On volume

In Section 1.1.2, we showed that naive Monte Carlo sampling becomes infeasible for soft sphere packings, where exponentially many minima ensure that each basin occupies a vanishing fraction of configuration space. To calculate basin volumes for jammed systems, researchers developed

an approach that addresses the infeasibility of the naive sampling method described in 1.1.2. This line of work began with Xu [16] and was further improved by Asenjo [17] and Martiniani [18, 30].

The volume can be written as:

$$v_\alpha = \int_\Omega d\boldsymbol{X}\, \mathbb{1}_{\mathcal{B}_\alpha}(\boldsymbol{X}) \tag{1.16}$$

where $\mathbb{1}_{\mathcal{B}_\alpha}(\boldsymbol{X})$ is an indicator function that equals 1 inside the basin and 0 outside. Finding the volume is thus equivalent to estimating the normalization constant of a uniform distribution with unknown support $\mathcal{B}_\alpha$.

In the jamming literature, this problem is recast in statistical mechanical terms. The volume (normalization constant) is treated as the partition function of an infinite potential well $U_\alpha(\boldsymbol{X})$ that equals 0 inside the basin and ∞ outside, at temperature $1/\beta$. The indicator function becomes $\mathbb{1}_{\mathcal{B}_\alpha}(\boldsymbol{X}) = \exp\left(-\beta U_\alpha(\boldsymbol{X})\right)$, and the negative log of the volume becomes a free energy. This reformulation allows computational chemistry methods developed for free-energy calculations to be applied to calculate basin volumes.

The algorithm works as follows. We attach a harmonic oscillator with spring constant k to the attractor $\boldsymbol{X}_{IS}^\alpha$ within the well, giving the partition function:

$$Z_\alpha(k) = \int_\Omega d\boldsymbol{X}\, \exp\left(-U_\alpha(\boldsymbol{X}) - H_\alpha(\boldsymbol{X})\right) \tag{1.17}$$

where $H_\alpha(\boldsymbol{X}) = k\left|\boldsymbol{X} - \boldsymbol{X}_{IS}^\alpha\right|^2/2$. The free energy $F_\alpha(k) = -\ln(Z_\alpha(k))$ equals the negative log volume. In the limit $k \to \infty$, the partition function and free energy are known analytically, while at $k = 0$, $F_\alpha(0) = -\ln(v_\alpha)$. An estimator for the negative log volume is:

$$\hat{F}_\alpha(0) = F(k \to \infty) + \Delta\hat{F}(k = 0, k \to \infty) \tag{1.18}$$

where $\Delta\hat{F}(k=0, k\to\infty)$ estimates the free energy difference

$$\Delta F(k=0, k\to\infty) = F(k=0) - F(k\to\infty)\ . \tag{1.19}$$

Various free energy difference estimators exist, including thermodynamic integration (TI) [97], MBAR [98], and the stepping stone estimator [99], which use random samples generated at several k values, typically obtained through MCMC sampling. The rate-limiting step is determining whether each point lies inside or outside the basin, which requires solving the steepest descent ODE at every MCMC step.

Importantly, the basin volumes calculated in these works do not correspond to the true basins of the steepest descent ODE due to the computational cost of solving it exactly. Instead, optimizer proxies such as FIRE[100], L-BFGS[101], and Conjugate Gradient Descent [102–105] are used for speed. This follows standard practice in potential energy landscape studies since the late 1980s [106]. A central theme of this thesis is to examine the consequences of these choices in the study of soft sphere packings

1.3 Thesis Organization

This thesis is structured as follows:

Chapter 2 establishes the methodological foundation by reviewing numerical aspects of energy landscapes and methods for energy minimization and steepest descent integration used in studies of jamming and glasses. Beyond serving the thesis, this chapter provides a comprehensive overview of tools and nuances for descending the landscape that is aimed at researchers studying potential energy landscapes in physics and chemistry. We believe the viewpoint it provides does not currently exist in the field. The chapter includes benchmarking that yields a reliable method for solving the steepest descent ODE (CVODE), which we employ in subsequent chapters.

Chapter 3 demonstrates that commonly used optimizers fail catastrophically at basin identification, with accuracy decaying exponentially with system size N. For systems of $N \geq 64$ particles, optimizer accuracy drops to near zero, and we visualize this failure through configuration-space slices showing basin "scrambling." We further show that optimizer errors systematically bias estimates of minimum energies and the jamming transition density.

Chapter 4 characterizes the geometric structure of basins using multiple complementary approaches: line intersections, box-counting dimension, and survival probability under perturbations. We demonstrate that the apparent fractality can emerge as an artifact of optimizer use rather than an intrinsic property of the energy landscape.

Chapter 5 describes the algorithmic approach for measuring basin volumes in high dimensions, where naive Monte Carlo sampling becomes infeasible. We introduce parallel tempering with harmonic spring biasing to sample different radial regions within a basin, and explain how MBAR (Multistate Bennett Acceptance Ratio) enables estimation of basin volumes from these samples.

Chapter 6 applies the volume measurement methodology to reveal systematic biases from optimizer use. We show that optimizers systematically overestimate basin volumes by incorrectly assigning distant points to basins, with accuracy decaying exponentially with distance from the minimum. These inaccuracies appear precisely in the "tentacle" regions where most basin volume concentrates.

We conclude with Chapter 7, which discusses open questions and potential avenues for future research, including extensions to other energy landscapes and systems.

A significant portion of this thesis concerns numerical methods and their consequences for understanding physics. Recognizing that readers may have different interests, we organize the chapters into two categories. **Methods chapters** (Chapter 2 and Chapter 5) focus on numerical techniques for basin identification and volume calculation, including solver comparisons, convergence criteria, and sampling algorithms. **Physics chapters** (Chapter 3, Chapter 4, Chapter 6)

present results on basin properties.

Readers primarily interested in physical results may proceed directly to Chapter 3 after this introduction, consulting Chapter 2 only for context on the Hertzian disk model and solver comparisons. The geometry results in Chapter 4 can be read independently of the volume chapters. For volume results, Chapter 6 provides self-contained context, though readers seeking a deeper understanding of the sampling methodology should first consult Chapter 5. Readers interested in reproducing or extending this work should read the relevant methods chapter in full.

2 | Model and Methods

> Stiff equations are equations where certain implicit methods, in particular BDF, perform better, usually tremendously better, than explicit ones.
>
> — Ernst Hairer & Gerhard Wanner, *Solving Ordinary Differential Equations II*

2.1 Introduction

This chapter provides the methodological foundation for the thesis. The numerical details described here were crucial to the results that follow, and we believe that a clear understanding of these tools would benefit researchers studying energy landscapes.

We begin by reviewing the classes of numerical methods commonly used in energy landscape studies, highlighting modifications made in the field: gradient descent (Section 2.3.1), momentum-based methods such as FIRE (Section 2.3.2), and quasi-Newton methods including L-BFGS (Section 2.3.3). We then introduce implicit ODE solvers based on backward differentiation formulas (Section 2.4), explaining why they are essential for the stiff problems encountered in jammed packings. After describing the Hertzian disk model that serves as our testbed (Section 2.2), we present systematic benchmarks comparing ODE solvers (Section 2.5), justifying our choice of CVODE as the reference method throughout this thesis.

2.2 Model

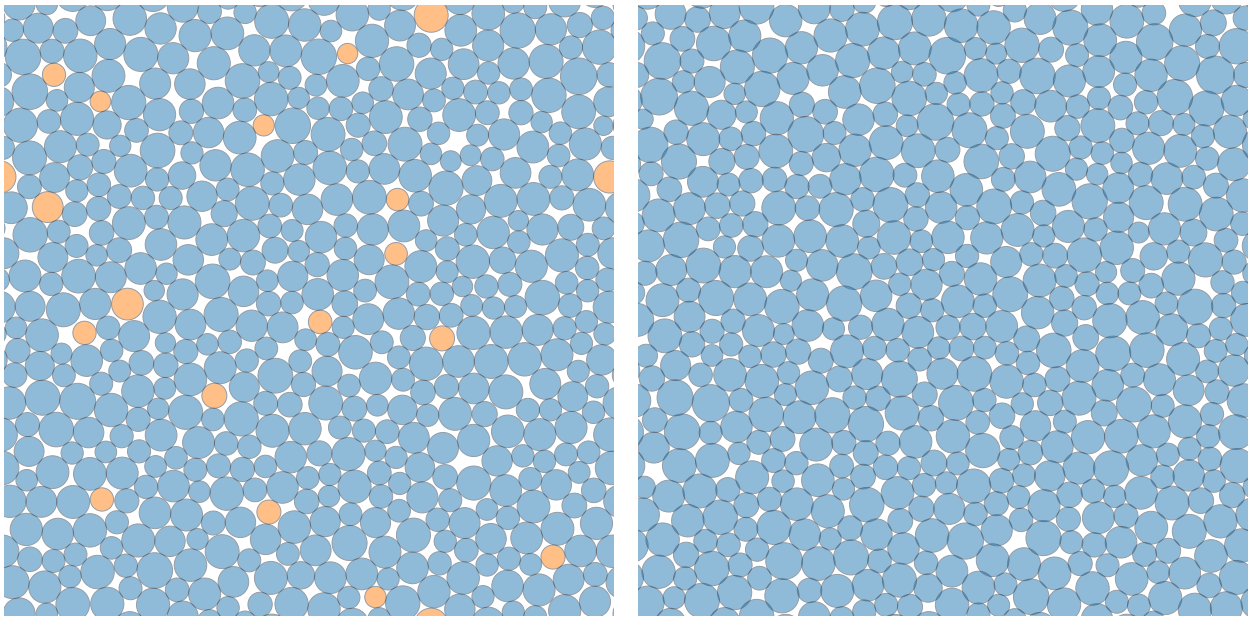

Figure 2.1: Jammed packings of $N = 512$ bidisperse Hertzian disks in two dimensions. Left: at $\phi = 0.845$, blue particles form the mechanically stable backbone while orange particles are rattlers—particles with insufficient contacts to be part of the rigid structure. Right: at $\phi = 0.9$, transparency reveals particle overlaps as darker regions where disks interpenetrate; All blue particles have overlaps, although overlaps are more clearly visible at $\phi = 0.9$. Both packings are confined to periodic square boxes and interact via the repulsive contact potential (Eq. 2.1).

Throughout this thesis, we consider two-dimensional polydisperse collections of N particles confined to periodic square boxes of sidelength L. The particles interact via a Hertzian repulsive potential,

$$V_{ij}(r_{ij}) = \left(1 - \frac{r_{ij}}{R_i + R_j}\right)^{5/2} \mathbb{1}\left(r \leq R_i + R_j\right), \tag{2.1}$$

where R_i is the radius of particle i, r_{ij} is the metric distance between the centers of particles i and j, and $\mathbb{1}$ is an indicator function that restricts the interaction to overlapping particles. The total energy is then $E = \sum_{i<j} V_{ij}$.

The exponent $5/2$ originates from Hertzian contact mechanics for three-dimensional elastic

spheres [107]. We adopt it in $2d$ because the harmonic potential, with the power set to 2 instead of $5/2$, more commonly used for soft disks in $2d$, has a discontinuous Hessian at the onset of contact. This may produce artifacts that do not generalize to other potentials.

The radii are drawn from a bidisperse distribution to prevent crystallization: half the particles have radii sampled from a positive-normal distribution with mean $\mu_s = 1.0$ and standard deviation $\sigma_s = 0.05$, while the other half are drawn with $\mu_\ell = 1.4$ and $\sigma_\ell = 0.07$. This choice ensures that particles do not crystallize and that distinct minima of the energy are not connected by permutation symmetry [16, 108].

Due to periodic boundary conditions, the energy is invariant under global translations in $d = 2$ directions, reducing the number of independent degrees of freedom from Nd to $(N-1)d$. With these conventions, the packing fraction is defined as

$$\phi = \frac{\pi \sum_{i=1}^{N} R_i^2}{L^2}. \tag{2.2}$$

This system undergoes a jamming transition at $\phi_J \approx 0.84$ [109]. As ϕ decreases towards ϕ_J, an increasing fraction of generic configurations can be relaxed to states with $E = 0$, corresponding to unjammed or liquid configurations in which all particle overlaps can be removed at no energy cost. For $\phi > \phi_J$, most minima of the energy are jammed: the particles form a mechanically stable backbone in which displacing any backbone particle necessarily increases the energy. A small number of particles, called rattlers, remain free to move in certain directions without affecting the energy as shown in Figure 2.1; these are identified and removed when comparing minima across methods (see Section B).

2.3 Optimizers

2.3.1 Gradient Descent

The steepest descent ODE (Eq. 1.9) may be solved numerically by discretizing time. The simplest approach is the forward Euler scheme,

$$\boldsymbol{X}(t + \mathrm{d}t) \approx \boldsymbol{X}(t) - \mathrm{d}t\, \boldsymbol{\nabla}_{\boldsymbol{X}} E(\boldsymbol{X}(t)), \tag{2.3}$$

which updates the configuration by taking a step of size $\mathrm{d}t$ in the direction of steepest descent. This algorithm is widely known as "gradient descent" in the optimization literature, though in the jamming and glass communities it is often referred to simply as "steepest descent". The accuracy of forward Euler depends on the choice of time step $\mathrm{d}t$: smaller steps yield trajectories closer to the true ODE solution, but at a greater computational cost. A fixed time step that works well in one region of configuration space may be unnecessarily small elsewhere or catastrophically large near sharp features of the landscape. Adaptive time-stepping addresses this by adjusting $\mathrm{d}t$ dynamically. While many schemes exist, our goal is to understand their usage in the jamming community. We therefore employ the scheme used for soft-sphere packings in Ref. [79], where the step size is controlled by requiring that successive gradients remain nearly aligned:

$$\hat{\boldsymbol{g}}(t) \cdot \hat{\boldsymbol{g}}(t + \mathrm{d}t) > 1 - \epsilon, \tag{2.4}$$

where $\hat{\boldsymbol{g}}(t) = \boldsymbol{\nabla}_{\boldsymbol{X}} E(\boldsymbol{X}(t))/|\boldsymbol{\nabla}_{\boldsymbol{X}} E(\boldsymbol{X}(t))|$ is a unit vector pointing in the direction of the gradient. The parameter ϵ sets the tolerance: if the angle between successive gradients exceeds the threshold, the step is rejected and $\mathrm{d}t$ is reduced. If the condition has been satisfied for some number of successive gradients, we increase dt. The idea being that if the trajectory "curves" too much, we need to be careful and reduce step size, and if the trajectory isn't curving at all,

we could try increasing the step size for performance reasons.. This adaptive scheme improves the time-accuracy trade-off compared to fixed-step gradient descent [79, 110], but as we show in Section 3.5, it remains computationally expensive for large systems. We discuss why it's computationally expensive, based on the forward Euler discretization, in Section 2.4.1.

2.3.2 Momentum-Based Methods: FIRE

Momentum-based methods accelerate convergence by allowing the system to build up "inertia" as it moves downhill, helping it traverse flat regions and narrow valleys more efficiently. The most widely used such method in the jamming and glass communities is the Fast Inertial Relaxation Engine (FIRE) [100], which has been employed in numerous studies of jammed packings [18, 30, 31, 33, 36].

To understand what we mean by adding momentum, let us take the Newton's equation with damping

$$m\ddot{\boldsymbol{X}}(t) + \gamma\dot{\boldsymbol{X}}(t) = -\boldsymbol{\nabla}_X E(\boldsymbol{X}(t)). \tag{2.5}$$

where m is the mass, γ is a damping coefficient, and the gradient term is the force from the potential. Gradient descent can be understood as the limiting case we take $m \to 0$, having no inertia, but a finite damping term γ, which we can reabsorb into the energy $E(\boldsymbol{X}(t))$. FIRE and other momentum based methods such as Nesterov's accelerated gradient descent [111, 112] broadly used in optimization reintroduce the momentum term with tweaks.

FIRE tweaks the momentum term the following way,

$$m\ddot{\boldsymbol{X}}(t) + \gamma(t)\left(\dot{\boldsymbol{X}}(t) + |\dot{\boldsymbol{X}}(t)|\frac{\boldsymbol{\nabla}_X E(\boldsymbol{X}(t))}{|\boldsymbol{\nabla}_X E(\boldsymbol{X}(t))|}\right) = -\boldsymbol{\nabla}_X E(\boldsymbol{X}(t)), \tag{2.6}$$

where $\gamma(t)$ is now a time-dependent damping coefficient, and the additional term proportional to $\gamma(t)$ continuously reorients the velocity toward the steepest descent direction, but with a delay controlled by the mixing parameter $\alpha = \gamma\,\mathrm{d}t/m$. We note however that the additional reorien-

tiation term removes the limiting case of our Newton's equation going to steepest descent, for example under the limit $m \to 0$ we get

$$\dot{\boldsymbol{X}}(t) = -\boldsymbol{\nabla}_{\boldsymbol{X}} E(\boldsymbol{X}(t)) \left(\frac{1}{\gamma(t)} + \frac{|\dot{\boldsymbol{X}}(t)|}{|\boldsymbol{\nabla}_{\boldsymbol{X}} E(\boldsymbol{X}(t))|} \right). \tag{2.7}$$

This equation clearly shows that the extra "push" along the gradient survives in the limit we take to remove momentum, although we note that once we parametrize by path length, the equation is the same as steepest descent. We highlight this discussion in particular because the usage of FIRE is extremely common in the study of jamming and glasses and it has come up in conversation that FIRE is a more physical protocol[34] to generate packings than other optimizers. We note however that in that limit $\alpha \to \infty$, which affects the numerical procedure used to solve FIRE below.

At each step of trying to solve Newton's equations (2.5), the velocity is updated according to

$$\boldsymbol{v} \leftarrow (1 - \alpha)\boldsymbol{v} - \alpha \, |\boldsymbol{v}| \, \hat{\boldsymbol{g}}, \tag{2.8}$$

where $\hat{\boldsymbol{g}} = \boldsymbol{\nabla}_{\boldsymbol{X}} E / |\boldsymbol{\nabla}_{\boldsymbol{X}} E|$ is the unit gradient. This mixes the current velocity with a vector pointing along the steepest descent direction but with the same speed.

FIRE adapts its parameters based on the instantaneous power $P = -\boldsymbol{v} \cdot \boldsymbol{\nabla}_{\boldsymbol{X}} E$. When $P > 0$ (the system is moving downhill), the time step is gradually increased and α is decreased, allowing momentum to build up. When $P \leq 0$ (the system has overshot and is moving uphill), the time step is reduced, α is reset, and the velocity is zeroed. This adaptive scheme allows FIRE to take aggressive steps in smooth regions while remaining stable near minima.

We employ a strictly downhill variant of FIRE [17] that incorporates two modifications. First, whenever $P \leq 0$, the algorithm rejects the step and reverts to the previous configuration, ensuring that the energy decreases monotonically. Second, the step size is limited to a maximum displacement, preventing the trajectory from jumping across basin boundaries in a single step. These

modifications make FIRE more careful but do not change the fundamental issue: the dynamics of Eq. 2.6 differ from steepest descent.

2.3.3 Quasi-Newton Methods: L-BFGS

Gradient descent can be painfully slow in landscapes with anisotropic curvature. In an elongated valley, the gradient points mostly across the valley rather than along it, causing the algorithm to zigzag toward the minimum. Newton's method addresses this by incorporating curvature information. A second-order Taylor expansion around $\boldsymbol{X}_k$ gives

$$E(\boldsymbol{X}_k + \delta\boldsymbol{X}) \approx E(\boldsymbol{X}_k) + \boldsymbol{g}_k \cdot \delta\boldsymbol{X} + \frac{1}{2}\delta\boldsymbol{X}^T \cdot \overline{\overline{\boldsymbol{H}}}_k \cdot \delta\boldsymbol{X}, \tag{2.9}$$

where $\boldsymbol{g}_k = \boldsymbol{\nabla}_X E(\boldsymbol{X}_k)$ is the gradient and $\overline{\overline{\boldsymbol{H}}}_k$ is the Hessian. If we're close to a minimum we can set the gradient of this quadratic approximation to zero yielding the Newton step:

$$\delta\boldsymbol{X}_{\mathrm{N}} = -\overline{\overline{\boldsymbol{H}}}_k^{-1} \cdot \boldsymbol{g}_k. \tag{2.10}$$

For a purely quadratic landscape, this step reaches the minimum exactly. Figure 2.2 illustrates this distinction: gradient flow curves along the energy surface following $d\mathbf{x}/dt = -\nabla E$, while Newton's method jumps directly to the minimum by inverting the Hessian.

The practical obstacle is that computing and inverting the Hessian scales as $O(N^2)$ in storage and $O(N^3)$ in computation, which is prohibitive for the particle systems we consider.

2.3.3.1 The BFGS Update

Quasi-Newton methods approximate the inverse Hessian using only gradient evaluations. The Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [113–116] treats each optimization step as a measurement of the local curvature.

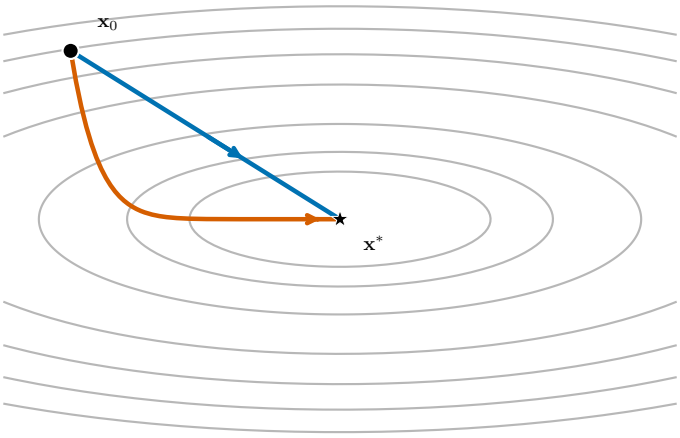

Figure 2.2: The newton optimization trajectory on an anisotropic quadratic energy landscape $E(\mathbf{x}) = x_1^2 + 10\,x_2^2$. Blue: Newton's method converges to the minimum $\mathbf{x}^*$ in one step. Orange: gradient flow follows the steepest descent ODE, curving along the elongated contours.

Suppose you take a step $\boldsymbol{s}_{k-1} = \boldsymbol{X}_k - \boldsymbol{X}_{k-1}$ and observe how the gradient changes: $\boldsymbol{y}_{k-1} = \boldsymbol{g}_k - \boldsymbol{g}_{k-1}$. Forf an exactly quadratic energy, these are related by the *secant condition*:

$$\overline{\overline{\boldsymbol{H}}} \cdot \boldsymbol{s}_{k-1} = \boldsymbol{y}_{k-1}. \tag{2.11}$$

This is a measurement: by probing the landscape along direction $\boldsymbol{s}_{k-1}$, you learn how the Hessian acts on that direction. One step provides one projective measurement that constrains our $N \times N$ Hessian.

BFGS maintains an estimate $\overline{\overline{\boldsymbol{B}}}_k$ of the inverse Hessian, updated after each step to incorporate the new measurement. The update must satisfy three requirements: (i) the secant condition in inverse form, $\overline{\overline{\boldsymbol{B}}}_k \cdot \boldsymbol{y}_{k-1} = \boldsymbol{s}_{k-1}$; (ii) symmetry; and (iii) positive definiteness, ensuring the search direction points downhill. Among all matrices satisfying these constraints, BFGS chooses the one closest to the previous estimate $\overline{\overline{\boldsymbol{B}}}_{k-1}$—the minimal modification needed to incorporate the new

measurement. The result is:

$$\overline{\overline{\boldsymbol{B}}}_k = \left(\overline{\overline{\boldsymbol{I}}} - \rho_{k-1}\boldsymbol{s}_{k-1}\boldsymbol{y}_{k-1}^T\right)\overline{\overline{\boldsymbol{B}}}_{k-1}\left(\overline{\overline{\boldsymbol{I}}} - \rho_{k-1}\boldsymbol{y}_{k-1}\boldsymbol{s}_{k-1}^T\right) + \rho_{k-1}\boldsymbol{s}_{k-1}\boldsymbol{s}_{k-1}^T, \tag{2.12}$$

where $\rho_{k-1} = 1/(\boldsymbol{y}_{k-1} \cdot \boldsymbol{s}_{k-1})$. This is a rank-two update: it modifies $\overline{\overline{\boldsymbol{B}}}_{k-1}$ only in the two-dimensional subspace spanned by $\boldsymbol{s}_{k-1}$ and $\boldsymbol{y}_{k-1}$, the directions probed by the latest step. Curvature information in all other directions is inherited unchanged.

While BFGS avoids computing the true Hessian, storing the full matrix $\overline{\overline{\boldsymbol{B}}}_k$ still requires $O(N^2)$ memory, which becomes impractical for large systems.

2.3.3.2 Limited-Memory BFGS

The limited-memory BFGS (L-BFGS) algorithm [101, 117] eliminates the need to store $\overline{\overline{\boldsymbol{B}}}_k$ explicitly. The key observation is that we never need the matrix itself—only its action on the current gradient, $\overline{\overline{\boldsymbol{B}}}_k \cdot \boldsymbol{g}_k$. By storing the M most recent measurement pairs $\{(\boldsymbol{s}_j, \boldsymbol{y}_j)\}_{j=k-M}^{k-1}$, we can reconstruct this matrix-vector product without forming the matrix. This reduces memory from $O(N^2)$ to $O(MN)$; typical values of M range from 3 to 20, and we use $M = 4$ throughout.

The algorithm proceeds via a two-loop recursion. To compute the search direction $\boldsymbol{p}_k = -\overline{\overline{\boldsymbol{B}}}_k \cdot \boldsymbol{g}_k$, we start with the current gradient $\boldsymbol{q} = \boldsymbol{g}_k$ and work backward through the stored pairs, progressively removing the contribution of each measurement:

$$\text{for } j = k-1, k-2, \ldots, k-M: \quad \alpha_j = \rho_j\,(\boldsymbol{s}_j \cdot \boldsymbol{q}), \quad \boldsymbol{q} \leftarrow \boldsymbol{q} - \alpha_j\,\boldsymbol{y}_j. \tag{2.13}$$

The coefficients α_j are stored for later use. After this loop, $\boldsymbol{q}$ represents the gradient transformed by the "oldest" part of the inverse Hessian estimate. We then apply an initial scaling:

$$\boldsymbol{r} = H_0\,\boldsymbol{q}, \quad \text{where} \quad H_0 = \frac{\boldsymbol{s}_{k-1} \cdot \boldsymbol{y}_{k-1}}{\boldsymbol{y}_{k-1} \cdot \boldsymbol{y}_{k-1}}. \tag{2.14}$$

This choice of H_0 sets the overall scale of the step based on the most recent curvature measurement. Finally, the second loop works forward, reincorporating each measurement:

$$\text{for } j = k - M, k - M + 1, \ldots, k - 1: \quad \beta_j = \rho_j \, (\boldsymbol{y}_j \cdot \boldsymbol{r}), \quad \boldsymbol{r} \leftarrow \boldsymbol{r} + (\alpha_j - \beta_j) \, \boldsymbol{s}_j. \tag{2.15}$$

The search direction is $\boldsymbol{p}_k = -\boldsymbol{r}$. This procedure computes the exact same direction that full BFGS would produce if started from $\overline{\overline{\boldsymbol{B}}}_0 = H_0 \, \overline{\overline{\boldsymbol{I}}}$ and updated with only the M most recent pairs.

2.3.3.3 Step Size Control and Modifications

The L-BFGS direction $\boldsymbol{p}_k$ provides a descent direction, but the step length must still be chosen. Standard implementations use a line search satisfying the Wolfe conditions [118], which can accept large steps when the curvature estimate is accurate. For basin identification, however, large steps risk jumping across basin boundaries.

We employ the"safe" version recommended in Ref. [28]. First, the step is constrained to a maximum length $\Delta_{\max}$: if $\|\boldsymbol{p}_k\| > \Delta_{\max}$, the step is rescaled. Second, a backtracking line search ensures the energy does not rise significantly: if $E(\boldsymbol{X}_k + \boldsymbol{p}_k) > E(\boldsymbol{X}_k) + \epsilon_f$ for a small tolerance ϵ_f, the step is halved repeatedly until the condition is satisfied. Third, if the search direction points uphill ($\boldsymbol{p}_k \cdot \boldsymbol{g}_k > 0$), which can occur when the curvature approximation becomes poor, the direction is reversed. It is important to note that these modifications are only heuristics that work well [28]. For example direction reversal is not really justified in the sense that the descent direction just aligns with the gradient, but nothing more.

These modifications make L-BFGS more conservative but do not resolve the fundamental issue: L-BFGS is designed to reach *a* minimum quickly, not to follow the gradient flow.

2.4 Implicit ODE Solvers: BDF Methods

The steepest descent equation (Eq. 1.9) is an ODE, and we can solve it directly using numerical integration rather than relying on optimizers. This approach has a significant advantage: by controlling the numerical error in the ODE solution, we can systematically improve the accuracy of basin identification. The challenge is choosing an integration scheme suited to the structure of the problem.

2.4.1 Stiffness and the Failure of Explicit Methods

The steepest descent ODE on high dimensional energy functions can be *stiff*[1]: the Hessian eigenvalues span many orders of magnitude. Physically, this means the system has both fast modes (steep directions with large curvature) and slow modes (soft directions with small curvature). The steepest descent dynamics cause fast modes to decay rapidly while slow modes persist, and the trajectory to the minimum is typically dominated by motion along the slow directions.

To understand why stiffness is problematic for explicit integrators, consider forward Euler applied to a simple test case: a one-dimensional linear ODE $\dot{x} = -\lambda x$ with $\lambda > 0$. The exact solution decays exponentially as $x(t) = x_0 e^{-\lambda t}$. Forward Euler gives the update

$$x_{k+1} = x_k - \mathrm{d}t\, \lambda\, x_k = (1 - \lambda\, \mathrm{d}t)\, x_k. \tag{2.16}$$

After n steps, $x_n = (1 - \lambda\, \mathrm{d}t)^n x_0$. For this to decay rather than blow up, we need $|1 - \lambda\, \mathrm{d}t| < 1$, which requires

$$\mathrm{d}t < \frac{2}{\lambda}. \tag{2.17}$$

[1] Correspondingly the energy function is ill conditioned for optimization

If the step size exceeds this threshold, the numerical solution oscillates with growing amplitude and diverges to infinity. For a multidimensional system, the same analysis applies to each eigenmode of the Hessian. The gradient can be decomposed into components along the eigenvectors of $\overline{\overline{H}}$, and each component evolves independently as $\dot{a}_i = -\lambda_i a_i$. The stability constraint becomes $\mathrm{d}t < 2/\lambda_{\max}$, where $\lambda_{\max}$ is the largest eigenvalue.

This constraint has nothing to do with accuracy, it prevents the numerical solution from blowing up. Even if we only care about tracking the slow dynamics, we must take tiny steps to keep the fast modes from exploding. Near jamming, where $\lambda_{\max}/\lambda_{\min}$ can be $O(10^6)$ (Fig. 2.4), this means taking millions of steps when only thousands would suffice for convergence. The result is that explicit methods spend enormous computational effort taking many small steps, most of which contribute little to progress toward the minimum.

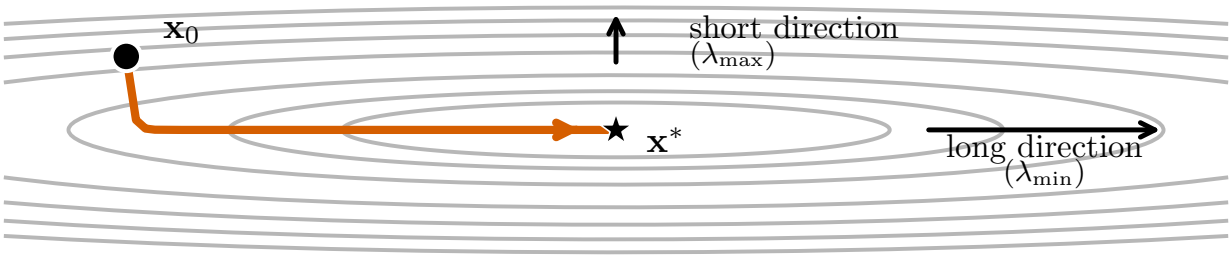

Figure 2.3: Steepest descent on an ill-conditioned quadratic $E(\mathbf{x}) = x_1^2 + 100\,x_2^2$. The trajectory from $\mathbf{x}_0$ to the minimum $\mathbf{x}^*$ quickly decays along the short direction (high curvature, $\lambda_{\max}$) then slowly traverses the long direction (low curvature, $\lambda_{\min}$). Forward Euler's step size is constrained by the shortest direction $\lambda_{\max}$, yet most computational effort is spent moving along the long direction.

2.4.2 Implicit Methods

Implicit methods avoid this constraint by evaluating the right-hand side of the ODE at the *future* time rather than the current time. The simplest example is backward Euler:

$$X_{k+1} = X_k - \mathrm{d}t\,\nabla_X E(X_{k+1}). \tag{2.18}$$

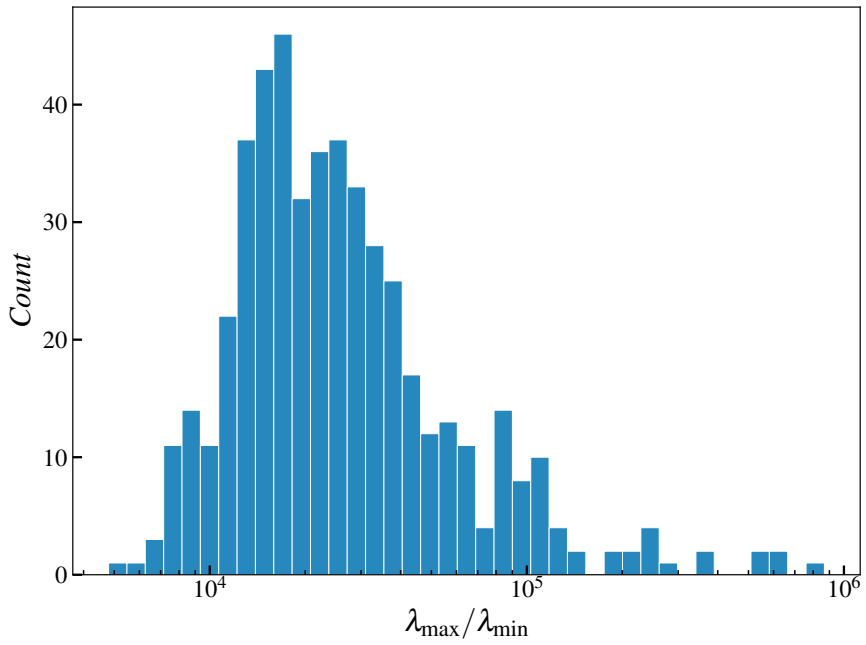

Figure 2.4: Distribution of condition numbers $\lambda_{\max}/\lambda_{\min}$ for $N = 512$ Hertzian packings in $2d$ at $\phi = 0.845$. The 491 jammed minima were obtained via L-BFGS minimization from random initial conditions. The mean condition number is $(4.2 \pm 0.3) \times 10^4$ (standard error), with a maximum of $8.7 \times 10^5 \sim O(10^6)$.

Unlike forward Euler, where $\boldsymbol{X}_{k+1}$ is computed explicitly from known quantities, here $\boldsymbol{X}_{k+1}$ appears on both sides. This is an implicit equation: given $\boldsymbol{X}_k$, we must solve a nonlinear system to find $\boldsymbol{X}_{k+1}$, typically via Newton iteration.

To see why this helps, consider again the test equation $\dot{x} = -\lambda x$. Backward Euler gives

$$x_{k+1} = x_k - \mathrm{d}t\,\lambda\,x_{k+1} \quad \Rightarrow \quad x_{k+1} = \frac{x_k}{1 + \lambda\,\mathrm{d}t}. \tag{2.19}$$

The amplification factor is $1/(1+\lambda\,\mathrm{d}t)$, which is less than 1 for *any* positive λ and $\mathrm{d}t$. The method is unconditionally stable: no matter how large $\lambda_{\max}$ is, the numerical solution never blows up. Fast modes are damped regardless of whether we resolve them temporally. The step size can thus be chosen based on accuracy requirements alone, allowing much larger steps than explicit methods when the dynamics are dominated by slow modes.

Backward Euler is first-order accurate: the local truncation error scales as $O(\mathrm{d}t^2)$. For better

efficiency, one neeeds higher-order methods that achieve the same accuracy with fewer, larger steps.

2.4.3 Backward Differentiation Formulas

The backward differentiation formulas (BDF)[119] are a family of implicit multistep methods that achieve higher-order accuracy by using information from several previous steps. The idea is to fit a polynomial through the points $\boldsymbol{X}_{k+1}, \boldsymbol{X}_k, \boldsymbol{X}_{k-1}, \ldots, \boldsymbol{X}_{k-q+1}$, differentiate this polynomial to approximate $\dot{\boldsymbol{X}}_{k+1}$, and require the result to satisfy the ODE:

$$\sum_{j=0}^{q} a_j \, \boldsymbol{X}_{k+1-j} = \mathrm{d}t \, \boldsymbol{f}(\boldsymbol{X}_{k+1}), \tag{2.20}$$

where $\boldsymbol{f}(\boldsymbol{X}) = -\boldsymbol{\nabla}_{\boldsymbol{X}} E(\boldsymbol{X})$ and the coefficients a_j are chosen to maximize the order of accuracy. The q-step BDF method is order-q accurate: local truncation error scales as $O(\mathrm{d}t^{q+1})$.

For $q = 1$, the BDF formula reduces to backward Euler. Higher-order BDF methods ($q = 2, \ldots, 5$) retain the favorable properties needed for stiff problems. BDF of order 6 and higher can exhibit growing oscillations for certain stiff problems and are rarely used in practice. For most stiff problems, BDF orders 1–5 provide an excellent tradeoff between accuracy and the number of steps required.

2.4.4 CVODE

We use CVODE [120], an adaptive BDF solver from the SUNDIALS suite developed at Lawrence Livermore National Laboratory. CVODE is a production-grade ODE solver incorporating decades of research into robust numerical integration. We highlight a few key aspects here; the full implementation includes many additional optimizations for efficiency and reliability.

Adaptive step size. CVODE estimates the local truncation error at each step by comparing solutions of different orders. It then adjusts dt to keep the estimated error below user-specified tolerances: when the solution varies slowly, it takes large steps; when the solution changes rapidly, it automatically refines the step size. This adaptivity is essential for efficiency, as the optimal step size can vary by orders of magnitude along a single trajectory.

Adaptive order. CVODE dynamically selects the BDF order between 1 and 5 based on the local behavior of the solution. Near the start of integration or after rapid changes, it uses lower-order methods which are more robust; in smooth regions, it increases the order for efficiency. The order selection is based on estimates of the error that would result from using neighboring orders.

Newton iteration. At each step, CVODE must solve the implicit system (2.20) for $\boldsymbol{X}_{k+1}$. It does this using Newton iteration, which requires the Jacobian $\partial \boldsymbol{f}/\partial \boldsymbol{X}$. Rather than recomputing the Jacobian at every step (which would be expensive), CVODE maintains an approximate Jacobian that is updated only when Newton convergence slows. This amortizes the cost of Jacobian evaluation over many steps.

Error tolerances. CVODE controls accuracy through two user-specified parameters: the relative tolerance ϵ_{rel} and the absolute tolerance ϵ_{abs}. At each step, the solver estimates the local error $\boldsymbol{e}_k$ and requires that each component satisfy

$$|e_{k,i}| \leq \epsilon_{\text{rel}} \, |X_{k+1,i}| + \epsilon_{\text{abs}}. \tag{2.21}$$

The relative tolerance controls accuracy when the solution is large: an error of 1 in a component of magnitude 10^6 is acceptable if $\epsilon_{\text{rel}} = 10^{-6}$. The absolute tolerance controls accuracy when the solution is small: without it, the relative criterion would demand arbitrarily small errors as

$X_{k+1,i} \to 0$.

2.4.5 Cost Tradeoff

Implicit methods are more expensive per step than explicit methods. Each step requires solving a nonlinear system, which involves multiple gradient evaluations and, periodically, Jacobian computations. For problems that are not stiff, this overhead makes implicit methods uncompetitive.

For stiff problems, however, the tradeoff changes entirely. An explicit method might require 10^6 tiny steps to maintain stability, while an implicit method achieves the same accuracy with 10^3 larger steps. Even if each implicit step costs 10 times more in gradient evaluations, the implicit method is faster by a factor of 100 overall. As we demonstrate in Section 2.5, CVODE provides the best time-error tradeoff for solving the steepest descent ODE in jammed packings.

2.5 Benchmarking ODE Solvers

In Section 2.4.1, we explained stiffness using essentially a linear stability analysis of forward Euler and backward Euler, note that our gradient is nonlinear. Here we adopt the working definition as quoted in the beginning of this chapter[119]: *a stiff equation is one where implicit methods perform much better than explicit ones.*

We investigate which ODE solver offers the best possible time-for-error trade-off. Due to the stiffness of the problem, we only focus on adaptive implicit solvers for $N > 16$ and compare a range of production-grade ODE solvers on the energy landscape of Hertzian disk packings, reporting a benchmark in Fig. 2.5 The benchmark consists of work-precision diagrams computed for $N = 8, 16, 32, 64, 128$, with the same size distribution as in the main text. For each system size, we spawn 100 random starting points $\boldsymbol{X}_0$, drawn unifor mly in the periodic square box $[0; L]^2$, then solve the steepest descent dynamics, Eq. 1.9, using a wide range of ODE solvers available in the Julia *DifferentialEquations.jl* package [121], including *lsoda*, a Fortran ODE solver library that

switches between implicit and explicit methods based on a stiffness criterion. For $N = 32$, 64, and 128, we restrict ourselves to more performant methods to avoid prohibitive computation times.

The error in adaptive ODE solvers is managed via a relative local error tolerance, `rtol`, and an absolute error tolerance `atol`, that ensure at each step that the trajectory does not stray away from the true ODE solution more than a specified amount (using a relative and an absolute distance, respectively). The absolute error tolerance `atol` sets the error bound when the coordinates are close to the origin, where the relative indicator `rtol` is ill-defined. Here, each ODE solver is run independently for a variety of values of the relative tolerance parameter `rtol`, and for a large enough integration time that the distance of the final point to the minimum is less than 10^{-2}, in this case $t_{\text{stop}} = 10000$. We ensure that the relative performance of ODE solvers is insensitive to the stopping time.

We then measure for each computed trajectory: (i) the time the computation took (in seconds), and (ii) the maximal distance a particle deviates from the true steepest descent path over the whole trajectory,

$$d_{\max}^{\text{solver}}(\boldsymbol{X}_0, \texttt{rtol}) \equiv \max_{0 \le t \le t_{\text{stop}}} \left\| \boldsymbol{X}_{\text{solver}}(t; \boldsymbol{X}_0, \texttt{rtol}) - \boldsymbol{X}_{\text{ref}}(t; \boldsymbol{X}_0) \right\|, \tag{2.22}$$

where $\boldsymbol{X}_{\text{solver}}(t; \boldsymbol{X}_0, \texttt{rtol})$ is the trajectory found by the solver from the initial position $\boldsymbol{X}_0$ and at a given value of `rtol`, and $\boldsymbol{X}_{\text{ref}}$ is the steepest descent trajectory. We then construct a Mean Trajectory Distance, defined as the average over random initializations $\boldsymbol{X}_0$ of $d_{\max}^{\text{solver}}$. In practice, we estimate the Mean Trajectory Distance using 100 evenly spaced points along the trajectory. The reference trajectory $\boldsymbol{X}_{\text{ref}}$ is obtained using *CVODE_BDF* with stringent tolerances ($\texttt{rtol} = \texttt{atol} = 10^{-12}$).

The results show that among the wide range of parameters we use, *CVODE_BDF* (see Section 2.4) outperforms other ODE solvers, with *QNDF* and *FBDF* also showing competitive performance as system size increases. We also note that at $N = 8$ and $N = 16$, we can clearly see

that explicit solvers outperform implicit ones by orders of magnitude confirming our working definition of stiffness for the steepest descent ODE on hertzians. We emphasize that all ODE solvers converge to the same trajectory as `rtol` is decreased, and our calculations are always performed in the regime where our identified basin is independent of precise solver parameters. We therefore adopt CVODE as our reference ODE solver in the rest of this thesis.

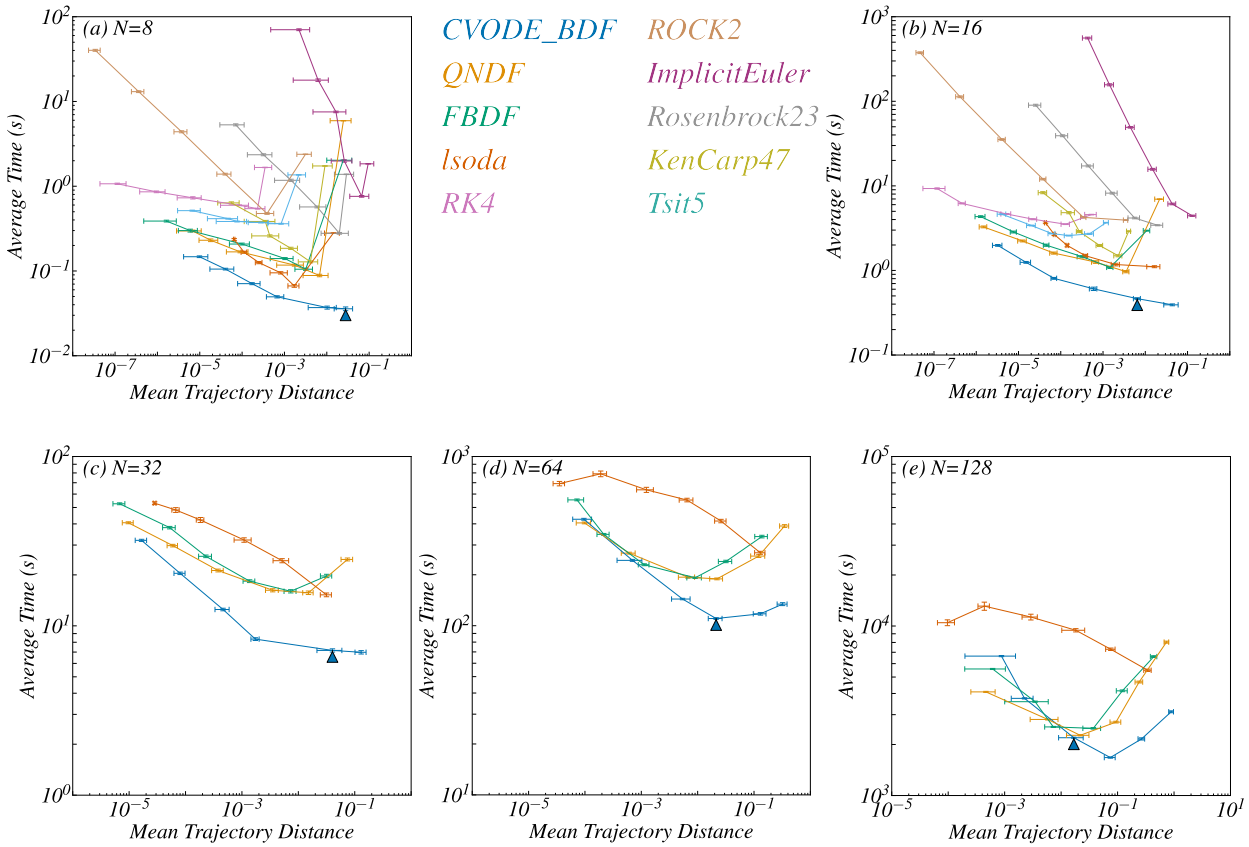

Figure 2.5: Benchmarking ODE solvers. (a)–(e) Plots of the Mean Trajectory Distance versus Average Computation Time for different numbers of particles $N = 8$, 16, 32, 64, and 128, respectively. Error bars represent the standard error of the mean (SEM) for both the Mean Trajectory Distance and Average Computation Time. Points highlighted by filled triangles correspond to the maximum tolerance values used among all calculations.

2.5.1 Performance on Other Systems

To rule out the possibility that CVODE is optimal only for our choice of interaction potential, we perform the same benchmark as in Section 2.5 on three other systems, spawning 100 random starting points uniformly in configuration space for each of them.

In Fig. 2.6(a), we consider a set of $N = 8$ soft spheres interacting via a harmonic potential,

$$V_{ij}(r_{ij}) = \varepsilon \left(1 - \frac{r_{ij}}{R_i + R_j}\right)^2 \mathbb{1}\left(r \leq R_i + R_j\right), \tag{2.23}$$

rather than a Hertzian one. We show that CVODE outperforms other solvers in a way similar to that described in the case of a Hertzian potential, Fig. 2.5. As discussed in 2.2, As discussed in Section 2.2, the harmonic potential exhibits a discontinuous Hessian at the onset of contact; nevertheless, CVODE retains its superior performance.

In Figs. 2.6(b)–(c), we consider variations on the XY model [122], with the energy function

$$E = -\sum_{i=1}^{N} \sum_{j \in \partial i} J_{ij} \hat{\mathbf{s}}_i \cdot \hat{\mathbf{s}}_j, \tag{2.24}$$

where $\hat{\mathbf{s}}_i = (\cos\theta_i, \sin\theta_i)$ is a two-dimensional unit vector parametrized by an angle $\theta_i \in [0; 2\pi)$, J_{ij} is the interaction constant between spins i and j ($J_{ij} > 0$ is aligning, $J_{ij} < 0$ is anti-aligning), and ∂i is the neighborhood of spin i, which depends on the chosen geometry for the problem. Note that the dynamical system defined by the steepest descent equation for Eq. 2.24 with aligning interactions is also known as the Kuramoto model, whose basins of attraction have been previously investigated [20–22, 123]. In Fig. 2.6(b), we choose a one-dimensional geometry with nearest-neighbor interactions only, and we set $J_{ij} = +1$ between neighbors, defining a usual ferromagnetic XY model. In Fig. 2.6(c), we choose a two-dimensional triangular lattice with $J_{ij} = -1$ interactions between nearest neighbors, thus defining a fully-frustrated XY model [124–126], a classical model of deterministic spin glass. In both cases, Tsit5 [121, 127], a variant of RK4, per-

forms comparably to CVODE. This implies that the steepest descent equation in the case of the XY model, even in a glassy regime, is not stiff enough that there is a significant performance gap between implicit and explicit methods. Note that we checked that *lsoda*, the solver used in Refs. [21, 22], performs significantly worse than the solvers we show in this benchmark, so that we do not show it in Fig. 2.6.

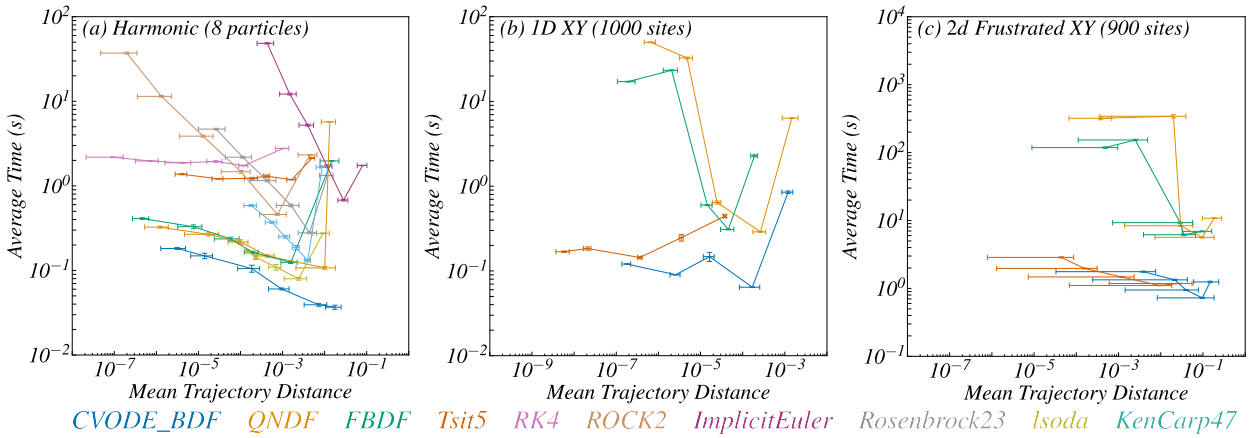

Figure 2.6: Benchmarking ODE solvers on other systems. (a) Interacting pairwise potential with harmonic interactions. (b) 1D XY model (Kuramoto). (c) Frustrated XY model on a triangular lattice. Error bars represent the standard error of the mean (SEM) for both the Mean Trajectory Distance and Average Computation Time. For the XY model [(b) and (c)] we utilize Jacobian-free GMRES solvers provided by the libraries for all implicit methods (CVODE, QNDF, FBDF) for speed.

2.6 Discussion

This chapter reviewed the numerical methods used in energy landscape studies and established the tools we use throughout this thesis. The central lesson is that the steepest descent equation is an ODE, and optimizing performance while preserving physics requires treating as such.

The optimizers commonly used in the energy landscape studies: Gradient Descent, FIRE, and L-BFGS were designed for different purposes. Gradient descent uses a forward Euler discretization of the steepest descent ODE and suffers from it's corresponding penalties if instead used to solve the ODE. FIRE introduces a velocity reorientation term, and the commonly used numerical

procedure does not solve steepest descent. L-BFGS approximates the inverse Hessian to converge quickly to *a* minimum, not to follow any particular trajectory. These methods can be used reach to minima efficiently, but they do not solve the ODE that defines basins of attraction.

An important aspect of the steepest descent ODE on jammed packings that is not commonly talked about, is that it is *stiff*: Both in the linear stability sense, Hessian eigenvalues span many orders of magnitude, and through the working definition, *stiff equations are those where implicit methods perform much better than explicit ones.*

Explicit integrators face a stability constraint unrelated to accuracy on stiff problems, whereas Implicit methods such as BDF avoid this constraint entirely, allowing step sizes to be chosen based on accuracy alone. Among the solvers we tested, CVODE provides the best time-accuracy tradeoff for this problem.

We emphasize that the performance results may not be universal. For the XY model at the system sizes shown, even in its frustrated (glassy) variant, explicit and implicit methods perform comparably, implying that the equation is not stiff enough for implicit methods to vastly outperform explicit ones. However researchers studying other energy landscapes should benchmark their specific system rather than assuming that conclusions from one domain transfer to another. But, as we show through the rest of the thesis, we strongly discourage using optimizers as a proxy to get physics requiring steepest descent paths.

The methods presented here—particularly the distinction between optimization and ODE integration, and the role of stiffness in solver selection—provide a foundation for the results that follow. In subsequent chapters, we use CVODE as our reference method for basin identification.

3 | Accuracy and Basic Observables

> Where the clear stream of reason has not lost its way
>
> Into the dreary desert sand of dead habit.
>
> — Rabindranath Tagore, *Gitanjali*

3.1 Introduction

One line of study of potential energy landscapes hinges on identifying basins of attraction, which are formally defined through the steepest descent equation: an ordinary differential equation (ODE) that traces the path from any initial configuration to a local minimum. Since the late 1980s[1] [128], researchers have framed this problem as one of energy minimization or a quench, and in practice have relied on numerical optimizers, algorithms designed to rapidly drive a system toward low-energy states. For many applications, such as exhaustively enumerating minima from sampled configurations, optimizers are well-suited to the task. However, the distinction between minimizing energy and solving the steepest descent ODE is often blurred or ignored, and the question remains when the fidelity to the underlying dynamics is too essential to ignore.

Framing basin identification as a minimization problem is fundamentally problematic because

[1] The earliest paper we could find was a 1985 paper by Stillinger and Weber using the Newton's method [128]. In a 1986 paper LaViolette and Stillinger [129], trying to describe the geometry of basin, by solving for a portion of the steepest descent path with forward Euler, says explicitly they would rather do a minimization instead if they just wanted the inherent structure/corresponding minimum

most optimizers do not solve the steepest descent ODE at all[2]. The one exception is gradient descent also commonly referred to simply as "steepest descent" in the jamming and glass communities, which corresponds to a forward Euler discretization of the ODE. Other widely used methods solve entirely different equations: quasi-Newton methods like L-BFGS exploit Hessian information to accelerate convergence but are only guaranteed to recover the correct basin for strictly convex landscapes; momentum-based methods like FIRE introduce inertia, solving a second-order ODE whose trajectories diverge from steepest descent paths. Previous work benchmarking these methods for small systems [28] has suggested reasonable accuracy. Even gradient descent, however, faces a serious numerical challenge: forward Euler is inherently unstable on stiff problems like the ones we consider, and achieving accurate basin identification requires prohibitively small time steps. Commonly introduced adaptive time-stepping strategies in the jamming community, such as those controlling step size via the cosine similarity between successive gradients, improve the time-accuracy trade-off but remain computationally expensive for large systems.

In this chapter, we demonstrate that the accuracy of these widely used optimizers collapses catastrophically as system size grows. We study two-dimensional polydisperse collections of Hertzian disks above the jamming transition, and benchmark algorithms against accurate solutions of the steepest descent ODE obtained using the CVODE solver, an adaptive implicit method offering the best time-for-error trade-off among the ODE solvers we test. We show that the accuracy of FIRE and L-BFGS decays exponentially with particle number N, dropping to near zero for $N \gtrsim 64$. For systems of 128 particles, not a single point in representative two-dimensional slices of configuration space is mapped to the correct basin by these optimizers. Even adaptive gradient descent, while degrading more slowly, fails to scale without prohibitive computational cost.

These findings have the following implications: the vast majority of past studies that relied

[2]The minimization picture treats the basin of attraction as a bowl. This makes sense for crystals they have a large convex region [129]. Starting from any point within the bowl, any descent path, one that always moves toward lower energy, will ultimately converge to the same local minimum at the bottom.

on optimizers for basin identification in systems larger than a handful of particles have likely misattributed every single basin. Furthermore, optimizer errors bias the distribution of minimum energies and even estimates of the jamming transition. The chapter proceeds as follows. We first briefly describe the methods we compare (Section 3.2), referring readers to Chapter 2 to explain the details of why we do what we do here. We then examine the effect of optimizer choice on the distribution of minimum energies (Section 3.3) and on estimates of the jamming transition (Section 3.4), present quantitative comparisons of accuracy and computational cost across methods (Section 3.5), and visualize the unscrambling of the energy landscape through two-dimensional slices (Section 3.6). We also demonstrate that the accuracy of solvers gets worse with packing fraction (Section 3.6.3). The central message is that convergent basin identification is achievable at practically relevant system sizes, but only by solving the steepest descent ODE with appropriate numerical methods.

3.2 Method Description

While we invite readers to consult Chapter 2 for detailed descriptions, we give brief summaries of what we mean when we say CVODE/FIRE/L-BFGS:

1. **CVODE:** basin assigned with numerical solution to the steepest descent ODE, at an error threshold that the basin obtained is not error sensitive, with the solver with the best time/error tradeoff.

2. **FIRE:** basin assigned with the FIRE optimizer with commonly used parameters/changes

3. **L-BFGS:** basin assigned with the L-BFGS optimizer with commonly used parameters/changes

3.3 Distribution of Energies at Minima

We first examine the effect of the minimization method on the distribution of energies at minima. For each method, and for $N \in \{8, 32, 128, 512, 2048\}$, we use the same 10^6 initial conditions for optimization, uniformly drawn at random in configuration space, at $\phi = 0.9$. We then collect the energies at the minima, and study their distributions. In Fig. 3.1(a), we plot the differences between the mean total energies obtained with FIRE/L-BFGS and CVODE, as a function of N. We report a growing bias towards lower energies in both FIRE and L-BFGS as N grows, meaning that the choice of method does affect the distribution of energies. It is worth noting that FIRE and L-BFGS do not create any new minima compared to CVODE, so that this difference is solely due to how often a given minimum is found—or, in other words, to the distribution of basin volumes being altered. To investigate the effect of this phenomenon further, we plot the relative errors on mean energy per particle against $1/N$ in Fig. 3.1(b). We show that the relative error in fact decays with N, meaning that the bias of Fig. 3.1(a) is subextensive. However, as indicated by a dashed line, the error vanishes algebraically slowly and with a small power, roughly as $N^{-1/6}$ for both FIRE and L-BFGS. While with our choice of potential the numerical value of the error remains rather small because the prefactor of the algebraic decay is small, another potential could display equally slow convergence with N but with larger prefactors. Finally, we assess the effects of the minimization method on the full distributions of energies. In Fig. 3.1(c), we plot the empirical distributions obtained for total energies at minima for $N = 2048$ particles. The distributions obtained through FIRE and L-BFGS display a systematic bias towards lower energies throughout. However, as shown in Fig. 3.1(d), the *shape* of the distribution is only very weakly modified, as empirical distributions of the reduced energies at minima $e \equiv (E - \langle E \rangle)/\sigma_E$, with σ_E the empirical standard deviation, overlap almost perfectly.

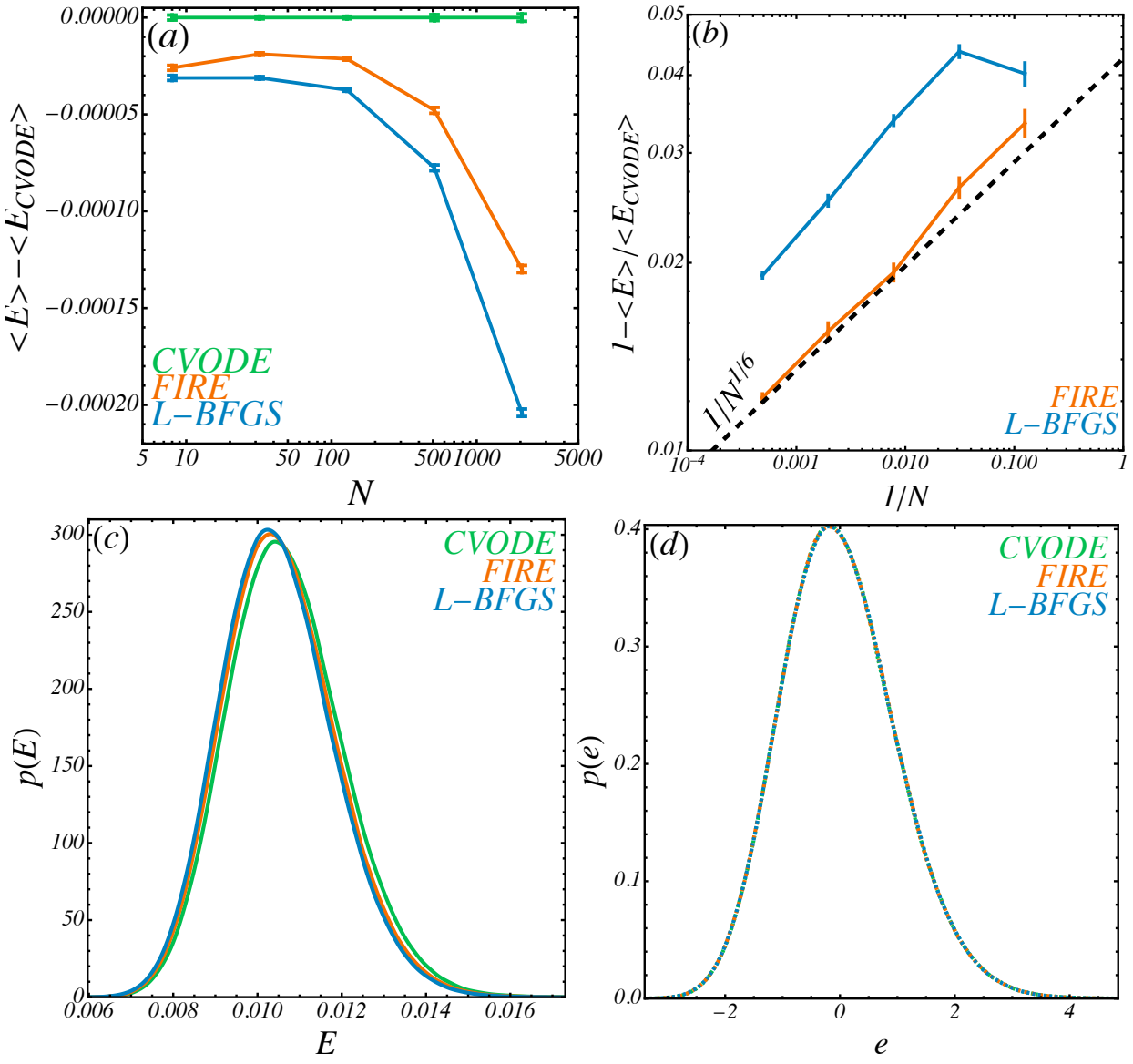

Figure 3.1: Energies at minima. (a) Differences between mean energies of minima for each method and the mean energies obtained with CVODE against N, in log-linear scales. The CVODE line is represented to indicate standard error on the CVODE mean. (b) Relative error on the energies with respect to CVODE against $1/N$, in log-log scales. The dashed black line indicates $1/N^{1/6}$. (c) Empirical distribution of energies E at minima for all three methods for $N = 2048$. (d) Corresponding empirical distribution of reduced energies $e \equiv (E - \langle E \rangle)/\sigma_E$, in dashed lines. Throughout the figure, we encode CVODE by green, FIRE by orange, and L-BFGS by blue. Error bars on the mean are obtained by bootstrapping over 1000 subsamples.

3.4 Jamming point

We study the effect of optimizers on the location of the jamming point. We consider a range of system sizes, $N = 8, 16, 32, 64, 128, 256, 512, 1024$, and, for each N, a range of packing fractions $\phi \in [0.80; 0.86]$. For each (N, ϕ) combination, we draw a set of 10^3 random initial conditions uniformly in $[0; L]^{Nd}$, and perform relaxations from them using FIRE, L-BFGS, and CVODE to generate a list of minima. We then measure the fraction P_J of these minima that is jammed, as per the criterion given in Sec. B, and report the fraction of them that falls into a minimum with all methods.

The results are shown in Fig. 3.2(a) for $N = 16$ and 10^5 random points. We show that there is a noticeable shift in the curve when switching to optimizers. Across (a) CVODE, (b) FIRE, and (c) L-BFGS, P_J displays a finite-size transition from 0 at low ϕ to 1 at high ϕ, with a transition that gets sharper as N increases, as expected at jamming [109]. We show that using FIRE or L-BFGS leads to a systematic bias of the curve $P_J(\phi)$ towards lower values (fewer jammed states, more liquid states), meaning that ϕ_J is overestimated. However, this bias is most pronounced at small N, so that previous reports of ϕ_J values relying on large N or on N scalings were likely not noticeably affected by their choice of optimizer.

We perform this measurement across N, using 10^3 points per N and density (see Fig. 3.3 for full curves) and measure the packing fraction ϕ_J with $P_J = 1/2$. To evaluate the jamming density, we perform a fit to a function of the form $P(\phi) = \sigma\,(ax + b)^p$ where $\sigma(x)$ is the sigmoid function, and we estimate a jamming packing fraction ϕ_J as the packing fraction at which $P_J = 1/2$.

The result is shown in Fig. 3.2(b). Curves roughly follow power laws of the form $\phi_J(\infty) - \phi_J(N) \propto 1/N^\theta$ with $\phi_J \approx 0.842$ and $\theta \approx 0.67$ as previously reported [109, 130]. To assess the effect of the choice of method on critical properties, we collect $\phi_J(N)$ values and fit them, for each method, to a power law of the form $|\phi_J(N) - \phi_J^\infty| = CN^{-\theta}$ with fitting parameters ϕ_J^∞ the asymptotic ϕ_J, θ a critical exponent, and C a proportionality constant. We find that CVODE

yields $\phi^{\infty}_{J(\mathrm{CVODE})} \approx 0.8415$, whereas FIRE and L-BFGS yield $\phi^{\infty}_{J(\mathrm{FIRE})} \approx 0.8423$, $\phi^{\infty}_{J(\mathrm{LBFGS})} \approx 0.8422$, but slightly different exponents: $\theta_{\mathrm{CVODE}} \approx 0.673$, $\theta_{\mathrm{FIRE}} \approx 0.668$, and $\theta_{\mathrm{LBFGS}} \approx 0.674$. These values are compatible with results from similar scaling analysis [109, 130].

Like for energies, a deviation between optimizers and CVODE remains noticeable at large N. In the inset, we show the evolution of the relative difference of ϕ_J between optimizers and CVODE, showing a trend slower than $N^{-1/3}$. We note that the probability of observing a jammed state is always higher with CVODE at all densities, as compared to using optimizer methods. In short, optimizers introduce a bias on sampled minima that biases finite-size estimates of the jamming density, with a slowly decaying error.

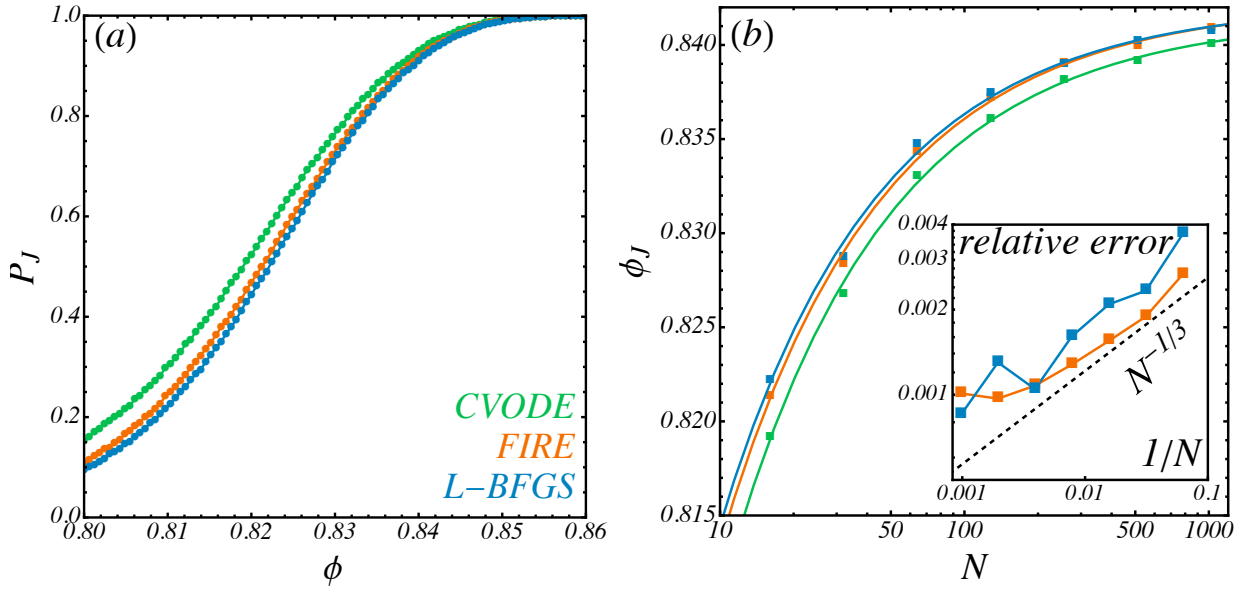

Figure 3.2: Jamming Point. (a) P_J against ϕ for $N = 16$ across methods. Student-T 95% confidence intervals are smaller than symbols. (b) ϕ_J against N across methods. Solid lines are critical power-law fits. Inset: relative error on ϕ_J compared to CVODE against N. Dashed black line: $N^{1/3}$ scaling (guide for the eyes).

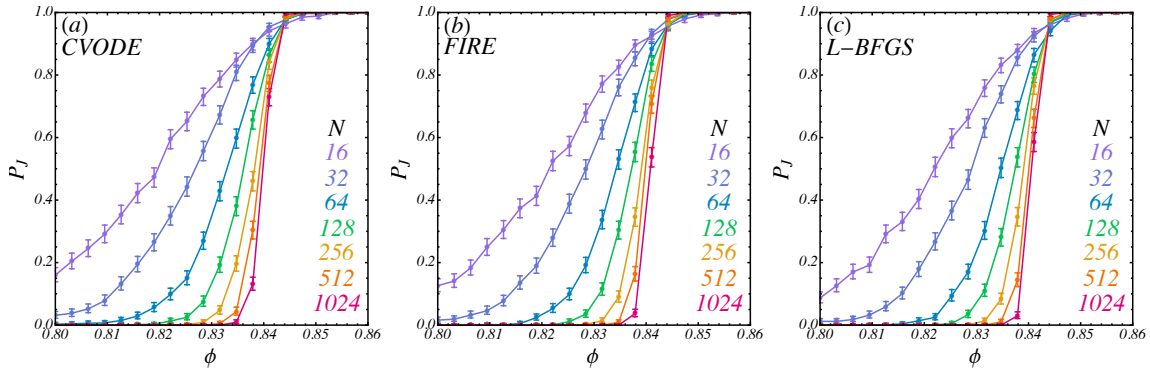

Figure 3.3: Jamming transition across sizes. Fraction of jammed states as a function of packing fraction, across system sizes (color within panels), using (a) CVODE, (b) FIRE, and (c) L-BFGS. Error bars are Clopper-Pearson 95% confidence intervals.

3.5 Accuracy of Optimizers

Armed with CVODE as a fast, accurate method for solving the steepest descent ODE, we now measure how much error is introduced by using optimizers to identify basins. For each set (N, ϕ), we tighten the tolerance of CVODE until the minima associated with a collection of random points in configuration space stop changing, establishing a ground truth for basin identification. In Fig. 3.4, we consider independent random points drawn from the configuration spaces of collections of Hertzian disks with $N \in [8; 1024]$ at $\phi = 0.9$. Panel (a) shows the accuracy, the fraction of points that are mapped to the correct minimum, against N across methods, with the ground truth given by low-tolerance CVODE ($\texttt{rtol} = 10^{-14}$). The CVODE values are obtained using a looser tolerance, $\texttt{rtol} = 10^{-13}$, demonstrating that it retains high accuracy. While FIRE and L-BFGS are relatively accurate for $(N-1)d \lesssim 10$ [28], their accuracies fall exponentially with N (dashed lines), dropping near zero for $N \gtrsim 64$. Thus, for systems with $N \gtrsim 64$, optimizers practically never map a point in configurational space to the correct basin of attraction.

We also test the adaptive gradient descent algorithm described in Section 2.3.1 with two choices of the parameter ϵ that controls the stepsize via Eq. 2.4. If ϵ is fixed (GD), accuracy also falls (like a stretched exponential), and approaches zero for $N > 128$ for the ϵ used in Ref. [79].

We also show that ϵ may be tuned at each N to achieve 98% accuracy (GD98)—at a significant time cost.

Panel (b) shows the corresponding wall times against N. While CVODE is slower than FIRE and L-BFGS ($O(N^{5/2})$ vs. $O(N^{3/2})$), it achieves computation times much smaller than GD98 (and comparable to GD) with better accuracy. Note the timescales involved in CVODE, which reach minutes per minimization in the systems we consider: generating the slice in Fig. 3.4(c) required 3 weeks of CPU time, compared to 3 hours for Fig. 3.4(e).

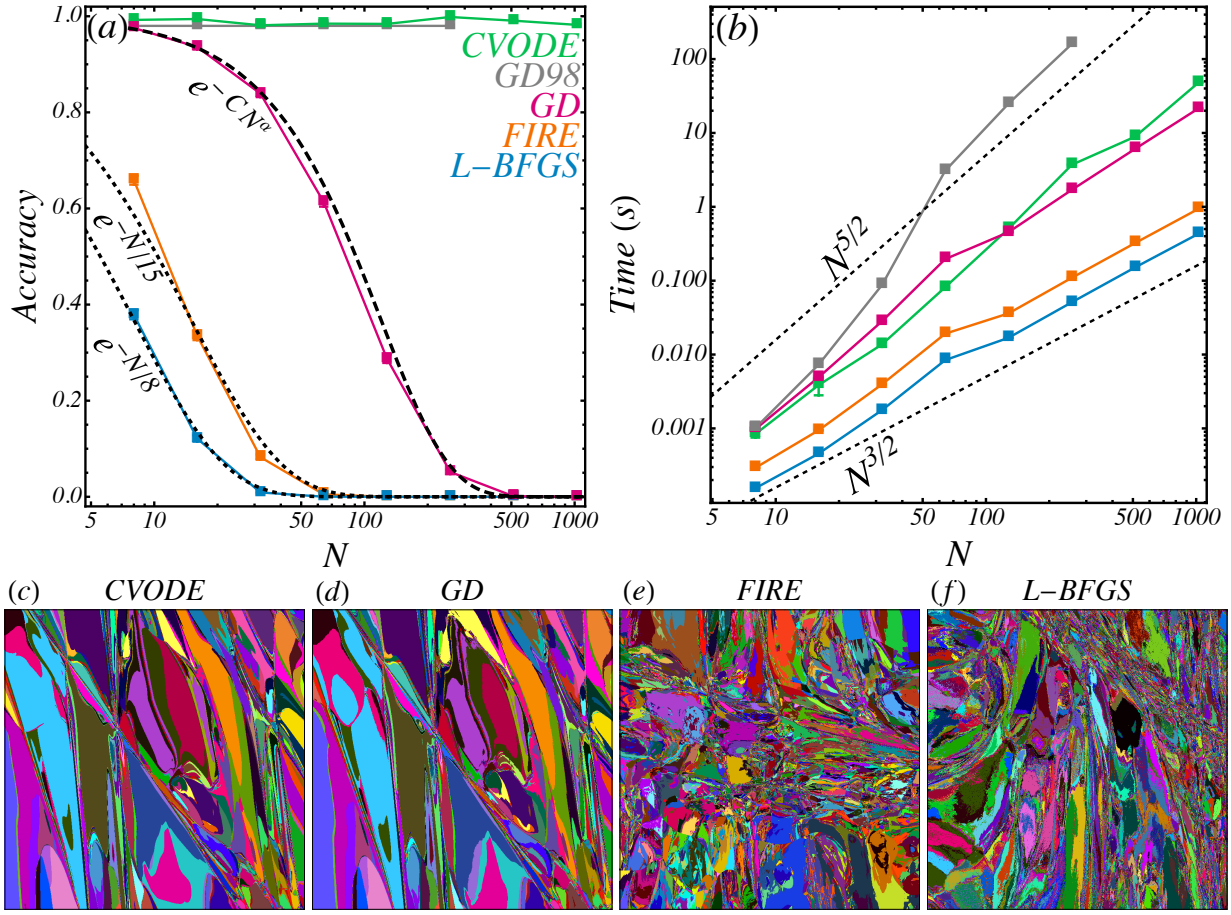

Figure 3.4: Optimizers: fast but inaccurate. (a) Average accuracy of algorithms, computed over 10^4 random points. Error bars are Clopper-Pearson 95% confidence intervals [131]. Dashed lines are exponential fits, long-dashed line stretched exponential fits. (b) Corresponding average computation times. Error bars are Student-T 95% confidence intervals. (c)–(f) 800 × 800-pixel slices of configuration space for $N = 128$ particles for (c) CVODE, (d) GD, (e) FIRE, and (f) L-BFGS.

3.6 Visualizing Basin Scrambling

3.6.1 Procedure for Generating 2D Energy Landscape Slices

To produce two-dimensional slices of the energy landscape color-coded by basin, as shown in Fig. 3.5, we follow the procedure outlined below:

1. **Sample the center point:** A single configuration $\boldsymbol{X}_0$ is sampled uniformly at random from the configuration space $[0, L]^{Nd}$.

2. **Define orthogonal directions:** Two orthogonal unit vectors, $\hat{\boldsymbol{n}}_1$ and $\hat{\boldsymbol{n}}_2$, are randomly sampled such that $\hat{\boldsymbol{n}}_1 \cdot \hat{\boldsymbol{n}}_2 = 0$.

3. **Construct the grid:** A two-dimensional grid of configurations is constructed, centered on $\boldsymbol{X}_0$. Each grid point is located at:

$$\boldsymbol{X}_{pq} = \boldsymbol{X}_0 + p\, P_x\, \hat{\boldsymbol{n}}_1 + q\, P_y\, \hat{\boldsymbol{n}}_2 \tag{3.1}$$

   where $(p, q) \in \mathbb{Z}^2$, with p spanning N_1 integers and q spanning N_2 integers (both ranges centered on zero). The parameters P_x and P_y denote the spatial resolution along each axis.

4. **Assign basin labels:** For each configuration $\boldsymbol{X}_{pq}$, the corresponding energy minimum (basin) is identified via energy minimization or solving the ODE.

5. **Generate the image:** The results are rendered as an $N_1 \times N_2$ pixel image, where each pixel is colored according to its basin membership. Colors are assigned using the Glasbey colormap [132] to ensure perceptual distinctiveness between basins.

3.6.2 Landscape slices

In Fig. 3.5, we produce pictures of 1920×1080 pixels that correspond to a rectangle in configuration space with side-lengths $12\mu_s \times 6.75\mu_s$ (in units of the mean radius μ_s of the small disks). For Fig. 3.4, and all $N = 128$ cuts in this document, we produce 800×800 pictures corresponding to squares in configuration space with side-lengths $0.44\mu_s \times 0.44\mu_s$. This configuration-space width is chosen so that the $2d$ density of basins on the slices would be fairly similar to that of Fig. 3.5. They are produced using 800×800 pictures corresponding to squares in configuration space with side-lengths $10\mu_s \times 10\mu_s$. The quantitative accuracy measurements are complemented by direct visualization of basins through two-dimensional slices of configuration space. In Fig. 3.5 we show the same random $2d$ slice, where each pixel on a grid is used as an initial condition for steepest descent and each basin is represented by a unique color, comparing CVODE and FIRE. Minima are matched across the two slices based on the metric distance between their locations (after rattler removal). Even in moderate dimension, $(N-1)d = 30$, FIRE scrambles the basins, creates discontinuities in their shapes, and alters their sizes. The apparent roughness of basins in $2d$ cuts was previously argued to be a feature of basins and an indicator of the fractal nature of their geometry [23, 28, 89]. These features are in fact artifacts of inaccurate noiseless relaxations that map points to the wrong basins of attraction.[3]

In Figs. 3.4(c)–(f), we show slices at $N = 128$, where the accuracies of FIRE and L-BFGS are essentially zero. These slices reveal a much starker contrast between methods: FIRE and L-BFGS turn the whole landscape into an unrecognizable collection of confetti-like, largely disconnected basins. In fact, not a single pixel of the slices obtained with these optimizers falls into the correct basin. The GD slice still looks reasonably smooth but displays large regions of incorrectly tagged basins.

To demonstrate that these findings are robust, Fig. 3.6 shows slices for a smaller system ($N = 8$)

[3] This is reminiscent of fractals generated by the iterative Newton-Raphson method on equations that yield roots of unity [133].

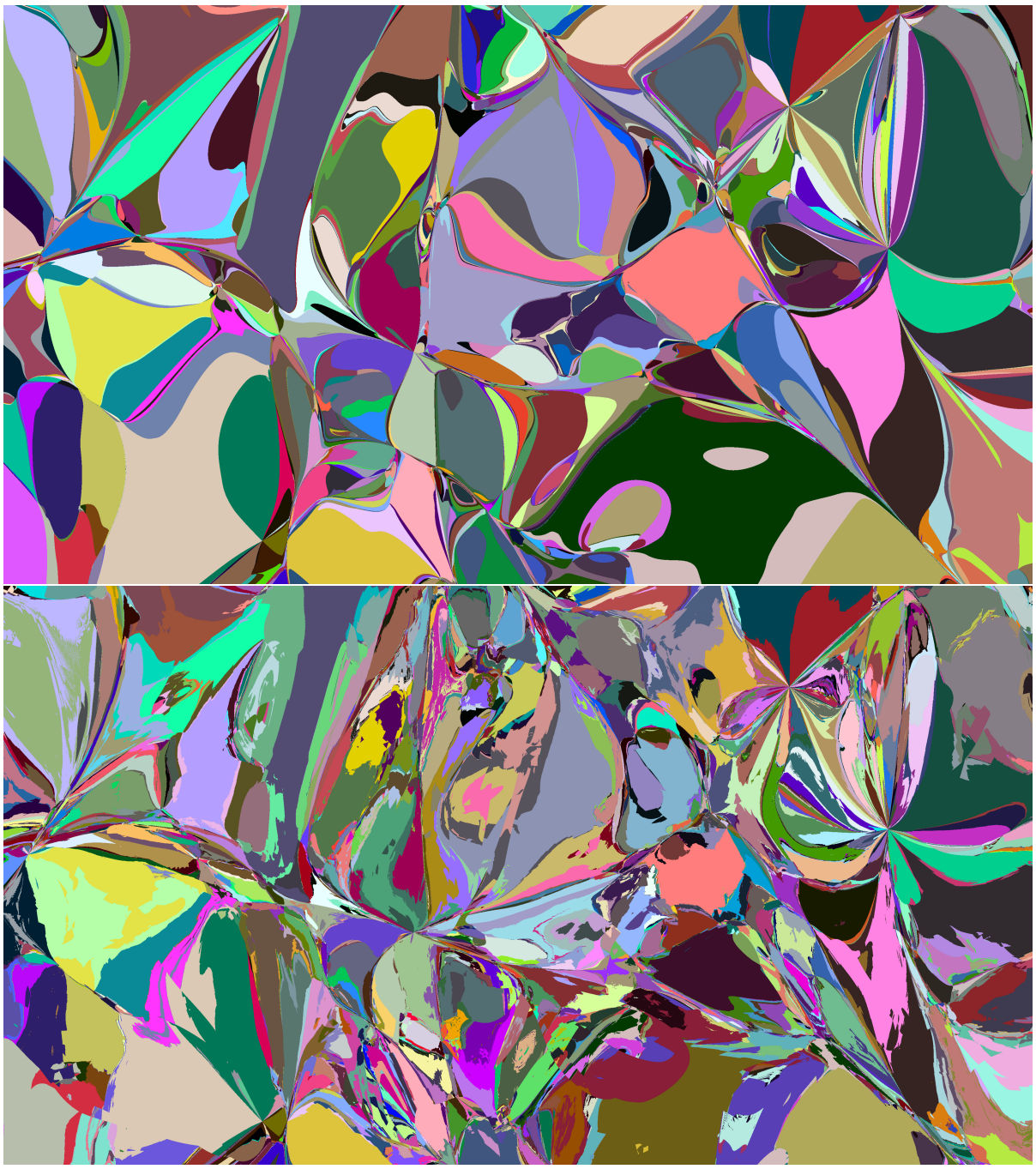

Figure 3.5: Slicing the energy landscape. 1350×2400 pixels in a random $2d$ plane in the configuration space of $N = 16$ disks. At each pixel, we use CVODE (top) and FIRE (bottom), to identify which basin of attraction it belongs to. Each basin is uniquely encoded by one color across both panels.

where optimizer accuracies are higher (about 70% for FIRE and 40% for L-BFGS). Even at this small N, there are noticeable basin deformations, and optimizers make basins appear less regular,

more "fractal-like." Fig. 3.7 presents additional slices at $N = 128$ with different random seeds, confirming that the basin scrambling is generic rather than an artifact of a particular choice of slice.

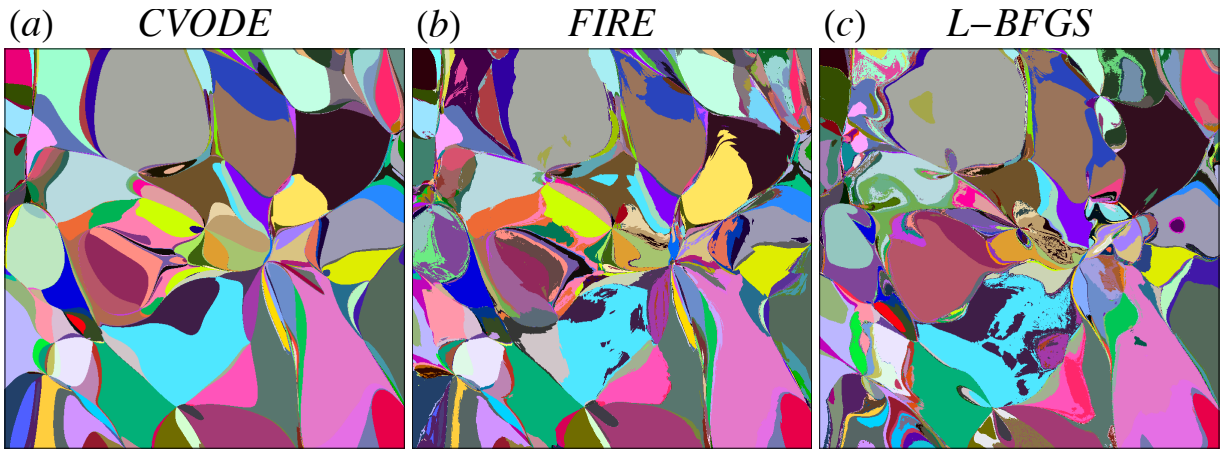

Figure 3.6: Small-system slices. Random $2d$ slice of the energy landscape of $N = 8$ particles at $\phi = 0.9$ using (a) CVODE, (b) FIRE, and (c) L-BFGS. Minima are matched across slices then color-coded.

3.6.3 Dependence on Packing Fraction

The results presented above were obtained at $\phi = 0.9$, well above the jamming transition. To demonstrate that these findings are generic to the overcompressed regime, we examine how accuracy depends on packing fraction. In Fig. 3.8, we present results at $\phi = 0.86$, closer to the jamming point. Panel (a) shows that accuracies follow the same trends as at $\phi = 0.9$: exponentially decaying accuracies for optimizers. Panel (b) confirms that time scalings remain $O(N^{3/2})$ for optimizers and $O(N^{5/2})$ for CVODE. The landscape slices in panels (c)–(e) are qualitatively similar to those at $\phi = 0.9$. We do not consider GD at this packing fraction due to its prohibitive cost at large N.

Fig. 3.9 examines accuracy as a continuous function of ϕ for fixed system sizes $N = 16$ and $N = 32$. The data is obtained by comparing configurations obtained by optimization to CVODE solutions for the same points, ignoring fluid states. Both L-BFGS and FIRE show degradation of

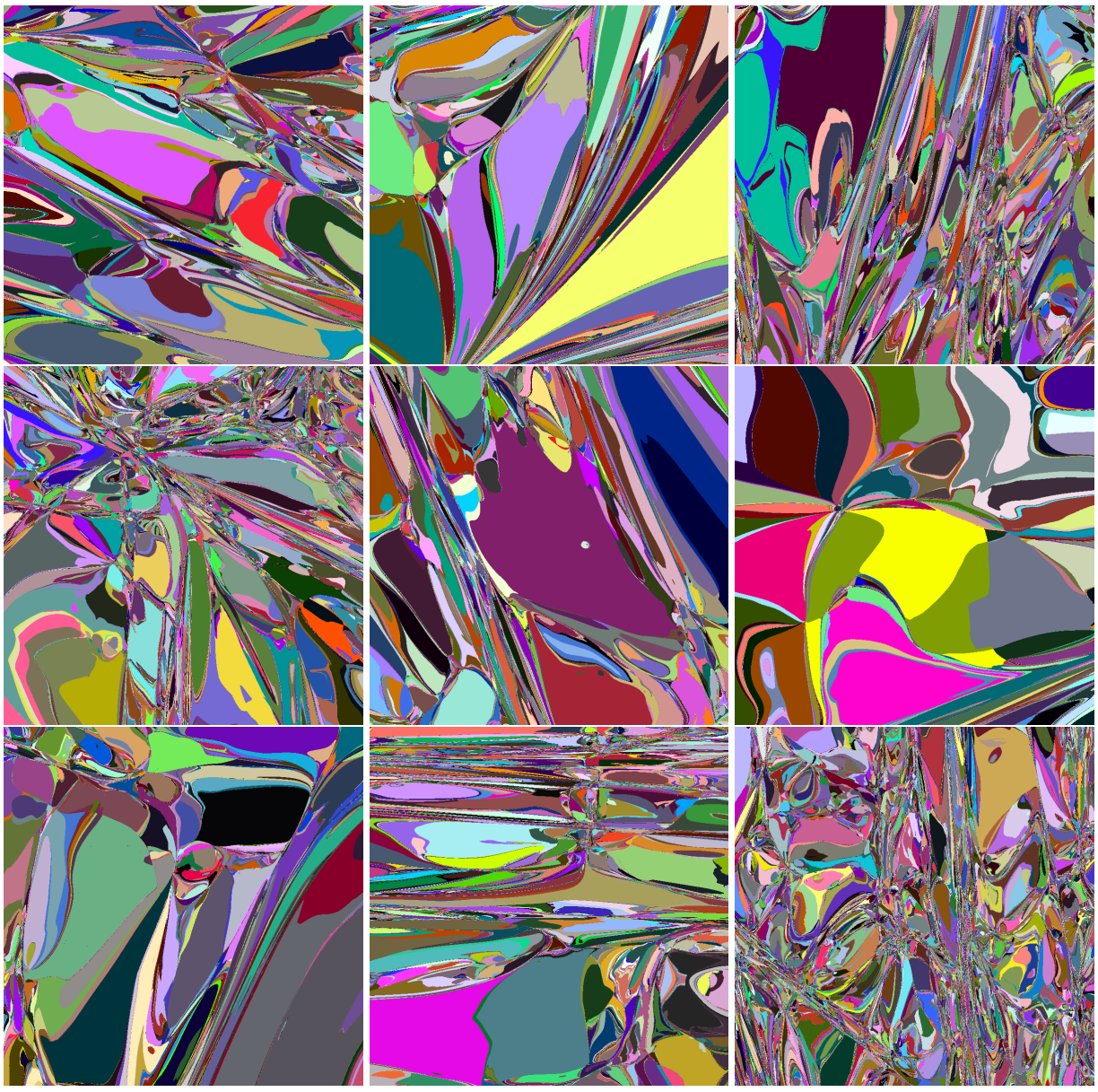

Figure 3.7: Gallery of energy landscape slices. Slices of the landscape of $N = 128$ particles at $\phi = 0.9$, obtained with the same parameters as in Fig. 3.4 but with different random seeds.

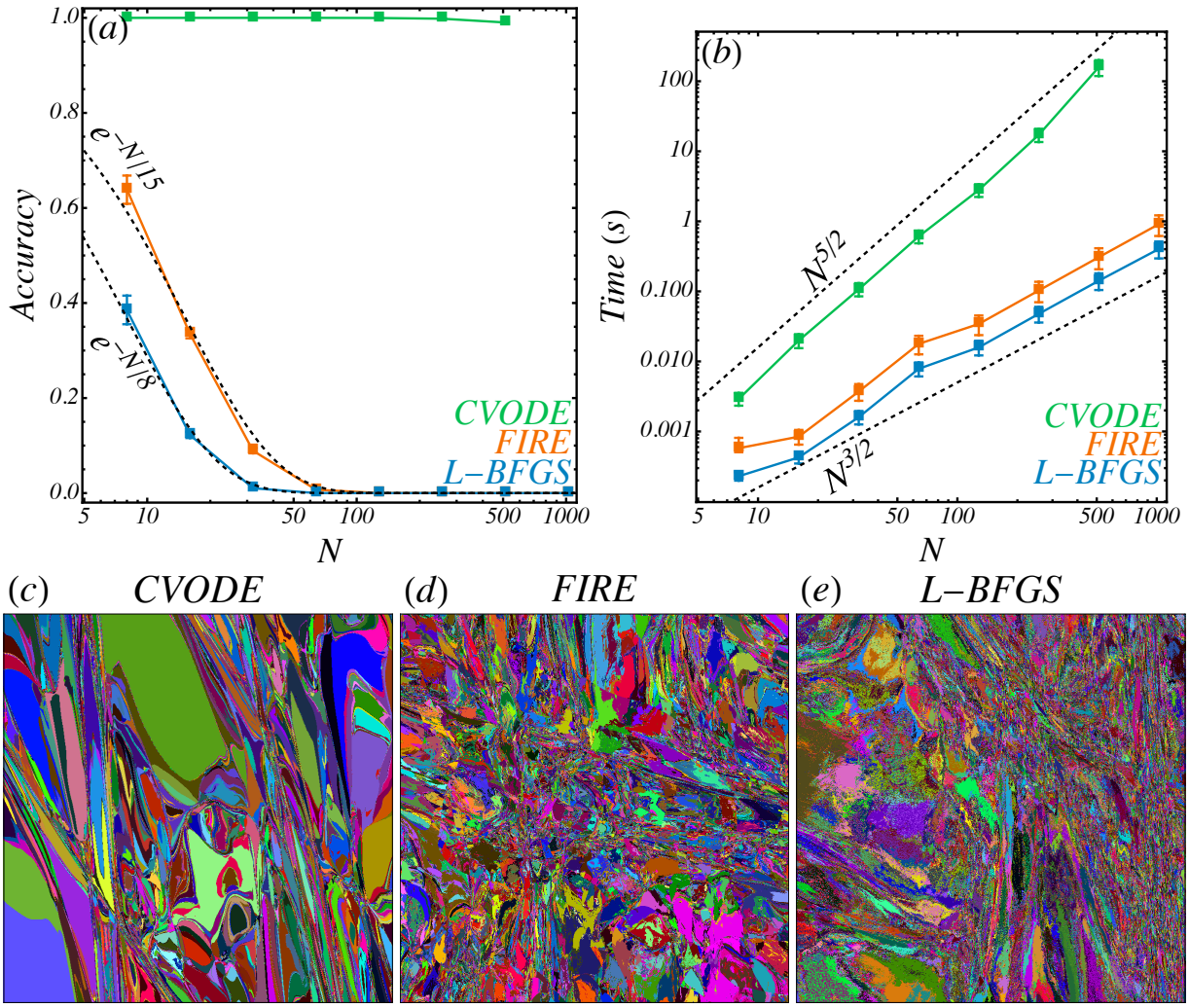

Figure 3.8: Optimizers: fast but inaccurate at every ϕ. (a) Average accuracy of algorithms compared to low-tolerance CVODE, computed over 10^4 random uniform initial conditions. Error bars are 95% confidence intervals, obtained using a Clopper-Pearson estimator. Dashed lines are exponential fits, and the long-dashed line is a stretched exponential fit. (b) Corresponding scalings of the average computation times, with error bars obtained from a Student-T 95% confidence interval. (c)–(e) 800 × 800-pixel slices for $N = 128$ particles for (c) CVODE, (d) FIRE, and (e) L-BFGS.

accuracy as the jamming point is approached from above.

To illustrate the effect of density on basins by reproducing the cuts from Fig. 3.4 at several ϕ, but using the same relative coordinates $\{\boldsymbol{r}_i/L\}_{i=1..N}$ and the same size polydispersity up to a global factor on all diameters, so that the slices correspond to the same initial point positions but different global dilations of the system. The results are shown in Fig. 3.10. In this figure, colors

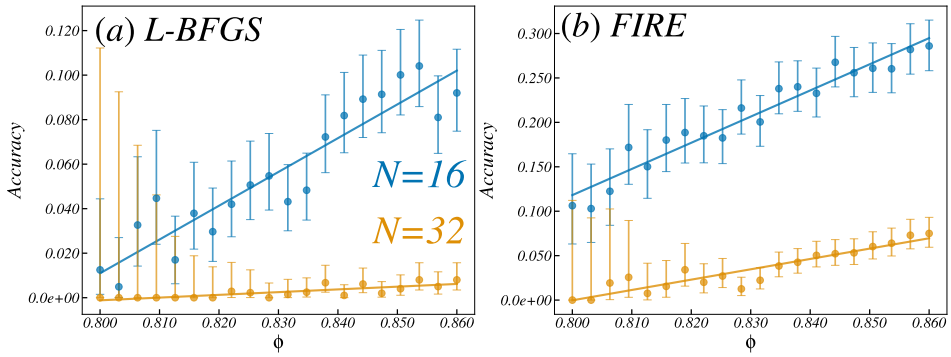

Figure 3.9: Optimizer accuracy dependence on volume fraction. Accuracy of (a) L-BFGS and (b) FIRE optimizers compared to high-precision CVODE solutions as a function of volume fraction ϕ. Data points show the fraction of random initial conditions that yield identical minima to the CVODE reference, ignoring fluid states, for system sizes $N = 16$ (blue circles) and $N = 32$ (orange circles). Solid lines represent linear regression fit guidelines. Error bars are Clopper-Pearson 95% confidence intervals.

are *not* matched across slices as minima move across densities, and black indicate non-jammed (fluid) states. Going from $\phi = 0.828$ to $\phi = 0.86$, we show that the configuration space goes from being mostly made of fluid states and sparsely populated by basins to being tiled entirely by basins, at a packing fraction $\phi \approx 0.84$ comparable to the final plateau from Fig. 3.3 for $N = 128$.

3.7 Discussion

This chapter establishes that commonly used optimizers in the jamming community—particularly FIRE and L-BFGS—fail catastrophically at the fundamental task of basin identification as system size increases. We demonstrated that their accuracy decays exponentially with particle number, dropping to near zero for systems with more than approximately 64 particles. At $N = 128$, not a single point in representative two-dimensional slices of configuration space was mapped to the correct basin by these methods.

A key insight enabling these findings is moving away from the false frame of basin identification from an optimization problem to an ODE integration problem. While the jamming community has historically conflated energy minimization with solving the steepest descent equa-

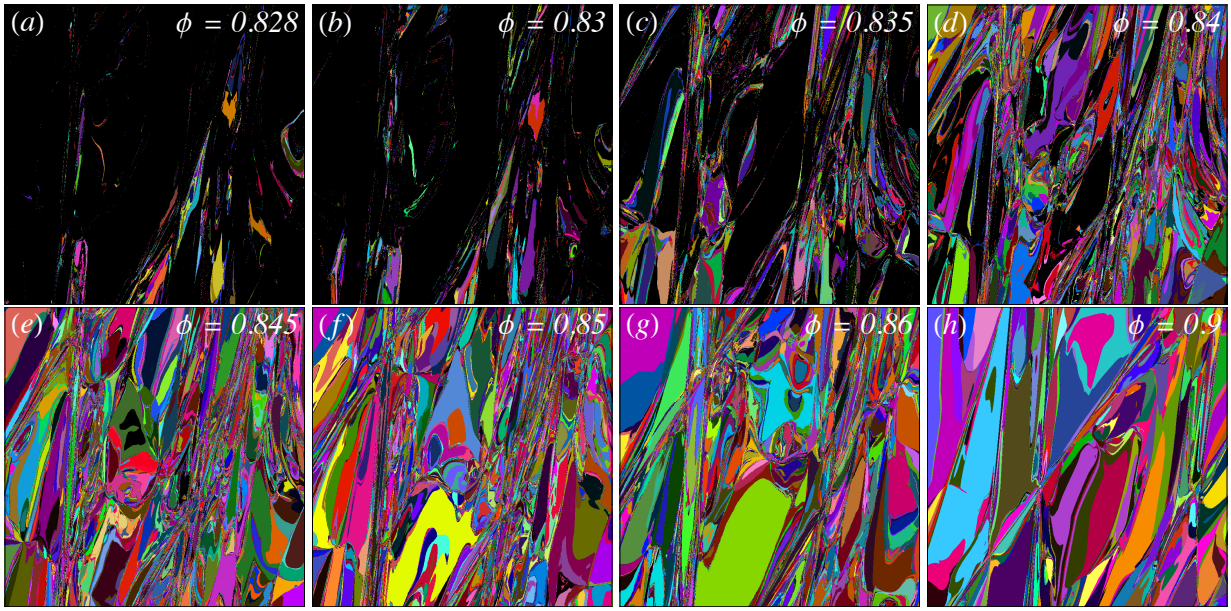

Figure 3.10: Density and landscape slices. Slices of the energy landscape for $N = 128$ and at ϕ equal to (a) 0.828, (b) 0.830, (c) 0.835, (d) 0.84, (e) 0.845, (f) 0.85, (g) 0.86, (h) 0.9. Individual basins are encoded by colors, while liquid states are shown as black pixels.

tion, systematically benchmarking ODE solvers revealed that CVODE, an adaptive implicit BDF method, offers the best time-for-error trade-off. Crucially, CVODE (and other production grade ODE solvers) provides explicit control over numerical accuracy through its tolerance parameters, and maintains reliable performance even at looser tolerances—a property absent from optimizers, whose errors cannot be systematically reduced without fundamentally changing the algorithm.

While CVODE incurs greater computational cost per minimization—scaling as $O(N^{5/2})$ compared to $O(N^{3/2})$ for optimizers—this cost is necessary for reliable basin identification. Adaptive gradient descent can achieve comparable accuracy but only with prohibitively small step sizes that render it impractical for large systems.

We further showed that optimizer errors introduce systematic biases, including shifts in the estimated jamming transition density and alterations to the distribution of minimum energies. The shift in the jamming point is relevant to get a precise estimate of critical exponents.

The present analysis focused on a single metric: the accuracy of assigning configuration-

space points to their correct basins of attraction. The two-dimensional slices presented in this chapter reveal that the apparent fractal roughness of basin boundaries, previously argued to be an intrinsic geometric feature, is in fact an artifact of optimizer-induced misattribution. With accurate ODE integration, basins appear considerably smoother. This observation raises a natural question: what is the true geometry of these basins of attraction? We address this question in the following chapter.

4 Geometry of basins

> Distress not yourself if you cannot at first understand the deeper mysteries of Spaceland. By degrees they will dawn upon you.
>
> — Edwin A. Abbott, *Flatland*

4.1 Introduction

The previous chapter established that commonly used optimizers fail catastrophically at basin identification as system size increases, and that accurate mapping of basins requires proper integration of the steepest descent ordinary differential equation. The two-dimensional slices presented there revealed a striking visual contrast: basins that appear rough and fragmented when mapped with optimizers become smooth and well-defined when mapped with CVODE. This observation motivates the central question of the present chapter: what is the true geometry of basins of attraction in the energy landscape of soft sphere packings?

The question of basin geometry connects to a broader theme in the study of complex systems: the role of fractality. Fractal basin boundaries are well-established in nonlinear dynamical systems [58–61], where they imply that arbitrarily small perturbations can redirect a system to a qualitatively different attractor.

In the study of glasses and jammed packings, various forms of fractality have been discussed. Mean-field theories predict a hierarchical organization of metastable states in the Gardner

phase [82], and the paths traced by relaxation dynamics have been argued to exhibit fractal structure [29, 35, 85]. More directly relevant to the present work, several studies have suggested that the basins of attraction themselves may be fractal objects [23, 28, 36, 94, 95]. Evidence cited for this claim includes the apparent roughness of basin boundaries in low-dimensional slices [23, 28], power-law distributions of basin intersection lengths [95], and scale-free distributions of basin volumes [36, 94]. Similar claims have appeared in the context of neural networks [89, 90] and constraint satisfaction problems [91, 92].

However, a number of prior studies of potential energy landscapes share a common methodological feature: they relied on optimizers rather than solving the steepest descent ODE. As we demonstrated in Chapter 2, such optimizers introduce systematic errors in basin identification that grow with system size. This raises a natural concern: could there be a case in where a claim about the geometry or reported fractality be an artifact of the optimizer mapping rather than a genuine geometric property of the landscape?

In this chapter, we demonstrate that for soft sphere packings, the answer is yes: apparent fractality in basin geometry can be an artifact of optimizer errors rather than a genuine landscape property.

The chapter is organized as follows. Section 4.2 analyzes the distribution of chord lengths obtained from random line intersections with basins, while Section 4.3 computes box-counting dimensions from two-dimensional slices. Section 4.4.1 turns to full-dimensional measurements through survival probabilities that quantify how far one can perturb a configuration before leaving its basin. Finally, Section 4.5 examines the arrangement of neighboring basins in configuration space. Throughout, we compare results obtained with CVODE against those from FIRE and L-BFGS, demonstrating that the choice of basin-mapping method fundamentally alters the measured geometry and produces apparent fractal behavior that is not intrinsic to the landscape.

4.2 Line Cuts

4.2.1 Method

To characterize basin geometry through low-dimensional intersections, we measure how random one-dimensional line segments intersect with basins of attraction in configuration space. The procedure proceeds as follows:

1. **Random Configuration and Line Selection.** A configuration is selected uniformly at random, followed by sampling a random direction uniformly to make a line. On the selected line, a segment is defined symmetrically around the starting configuration, with a total length $L_S \approx 10\mu_s$ for $N = 16$, comparable to the sidelengths of the two-dimensional slices shown in Fig. 3.5.

2. **Discretization and Basin Tagging.** The segment is divided into 10^6 regularly spaced points, which serve as initial conditions. Each point is tagged according to the basin it falls into upon minimization.

3. **Boundary Identification.** From the discretized basin map, all pairs of neighboring pixels falling into distinct basins are identified.

4. **Boundary Refinement.** At each such boundary pair, a new collection of points is defined on the line with a resolution 10^2 times finer, linking the two neighboring pixels. Optimizations are performed at these new points to better resolve the junction between basins and potentially uncover new basins.

5. **Basin Length Reconstruction.** The intersection lengths ℓ_{ij} between basins and the line are reconstructed using both resolution levels.

This adaptive zooming approach bypasses the limitation that an enormous number of regularly spaced points would otherwise be needed to adequately sample the broad distribution of intersection lengths. The base resolution before zooming is approximately 1000 times finer than that of the two-dimensional slices, and the zoomed resolution is thus 10^5 times finer. Although selective zooming at junctions could in principle introduce biases, empirical checks confirm that no new basins appear within the bulk of existing basins when switching to finer scales, indicating that basins are sufficiently regular to avoid measurable bias from this procedure.

4.2.2 Results and Discussion

The distribution $p(\ell)$ of intersection lengths across a collection of random segments is shown in Fig. 4.1(a) for both CVODE and L-BFGS minimization methods. While both methods produce a power-law-like decay, the distributions differ between the two approaches. L-BFGS overestimates

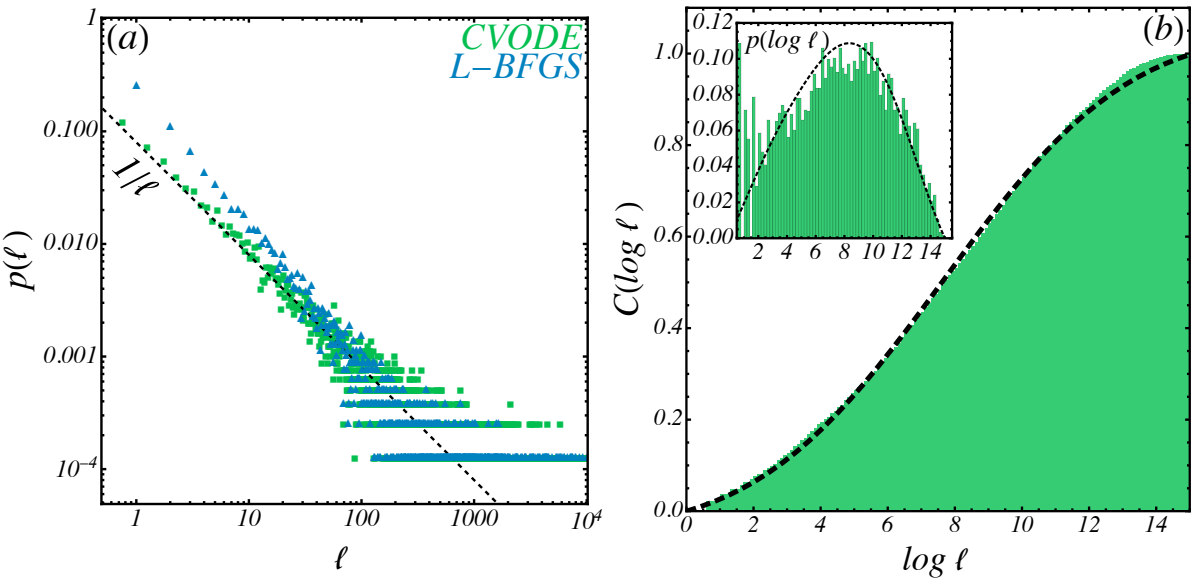

Figure 4.1: Linear intersects of basins. (a) Intersection lengths distributions obtained with L-BFGS (blue triangles) and CVODE (green squares) over 10 lines of 10^6 pixels for $N = 16$, in log scales. A dashed line indicates $1/\ell$ behavior. (b) CDF of the distribution of $\log \ell$ obtained with CVODE by zooming 100× on each basin boundary found from panel (a). The dashed black line is a truncated Gaussian fit. Inset: Corresponding histogram of the pdf, the dashed line shows a kernel regression.

the prevalence of small basins and consequently the decay exponent of the distribution, consistent with the "confetti" fragmentation observed in Fig. 3.4(c)–(f).

Taking advantage of CVODE's accuracy, we investigate the true distribution of line intersections. Given the broad nature of this distribution, we focus on the distribution of the logarithm of lengths, $p(\log \ell)$, presented in Fig. 4.1(b). The results demonstrate that this distribution is normal, indicating that the intersection lengths themselves are log-normally distributed.

This finding connects with prior results on basin volumes, which have been argued to follow log-normal distributions in soft sphere systems based on both numerical measurements [17, 30, 134] and theoretical arguments [135]. The volume of the intersection of a basin with an n-dimensional affine space may be approximated by a product of n independent log-normal lengths, yielding a log-normal distribution of volumes.

The log-normal distribution invalidates claims that basin intersection lengths are scale-free [95]. The power-law tail with exponent -1 observed in some studies arises from the asymptotic behavior of log-normal distributions with large variances [136]. Similarly, claims of scale-free distributions of basin volumes [36, 94] which contradict direct numerical measurements [17, 30, 134] likely stem from inadequate sampling of a log-normal distribution where only the tail was observed, a common issue with small sample sizes [137].

Notably, the peak in the distribution of log-lengths obtained with CVODE is already visible before the adaptive refinement step, confirming that the zooming procedure helps resolve single-pixel intersections but does not artificially introduce the observed peak.

4.3 Box-Counting Dimension of Basin Boundaries

To complete the data on distributions of $1d$ intersection lengths between random lines and basins, we discuss a $2d$ measure of the fractal dimension of intersections of basins with random $2d$ planes, such as the slices of Fig. 3.5. To quantify the fractal dimension of basin boundaries, we rely on

the Minkowski-Bouligand, or "box-counting" dimension [138], d_B. The box-counting dimension is defined as

$$d_B = \frac{\log(N(\ell_B))}{\log(\ell_B)}, \tag{4.1}$$

where $N(\ell_B)$ is the number of cubic boxes with sidelength ℓ_B that are needed to cover a shape. For non-fractal shapes $d_B = 1$, while fractal shapes have $d_B > 1$.

4.3.1 Measurement Procedure

Starting from a $2d$ slice of configuration space such as Fig. 3.5 of the main text, we crop the image to the smallest bounding box that encloses the basin boundary, with a padding of $L/16$ to mitigate boundary effects, then for the top n_B basins we create n_B binary map pictures with values 1 in the basin and 0 elsewhere. For each picture we perform the following procedure

1. We then feed each resulting picture into the `porespy` package [139] to count $N(\ell_B)$, defined here as the number of boxes that contain pixels both inside the basin and outside the basin

2. To estimate d_B, we perform a linear fit on the $\log(N(\ell_B))$ versus $\log(\ell_B)$ data, excluding the largest scales so as to mitigate finite-size effects due to the finite slice size. We thus measure a fractal dimension for the selected basin

4.3.2 Results

In Fig. 4.2(a), we show Gaussian kernel density estimates (KDE) obtained from the histograms of d_B for each method. We show that the mode of the distribution is close to unity ($d_B^{mode} \in [1.05; 1.15]$) when using CVODE or GD, but is shifted to significantly higher values ($d_B^{mode} \in [1.3; 1.4]$) when using FIRE or L-BFGS.

To estimate the effects of finite resolution, in Fig. 4.2(b), we plot the average box counting dimension $\langle d_B \rangle$ calculated for the n_B largest basins, against n_B (error bars are 95% confidence

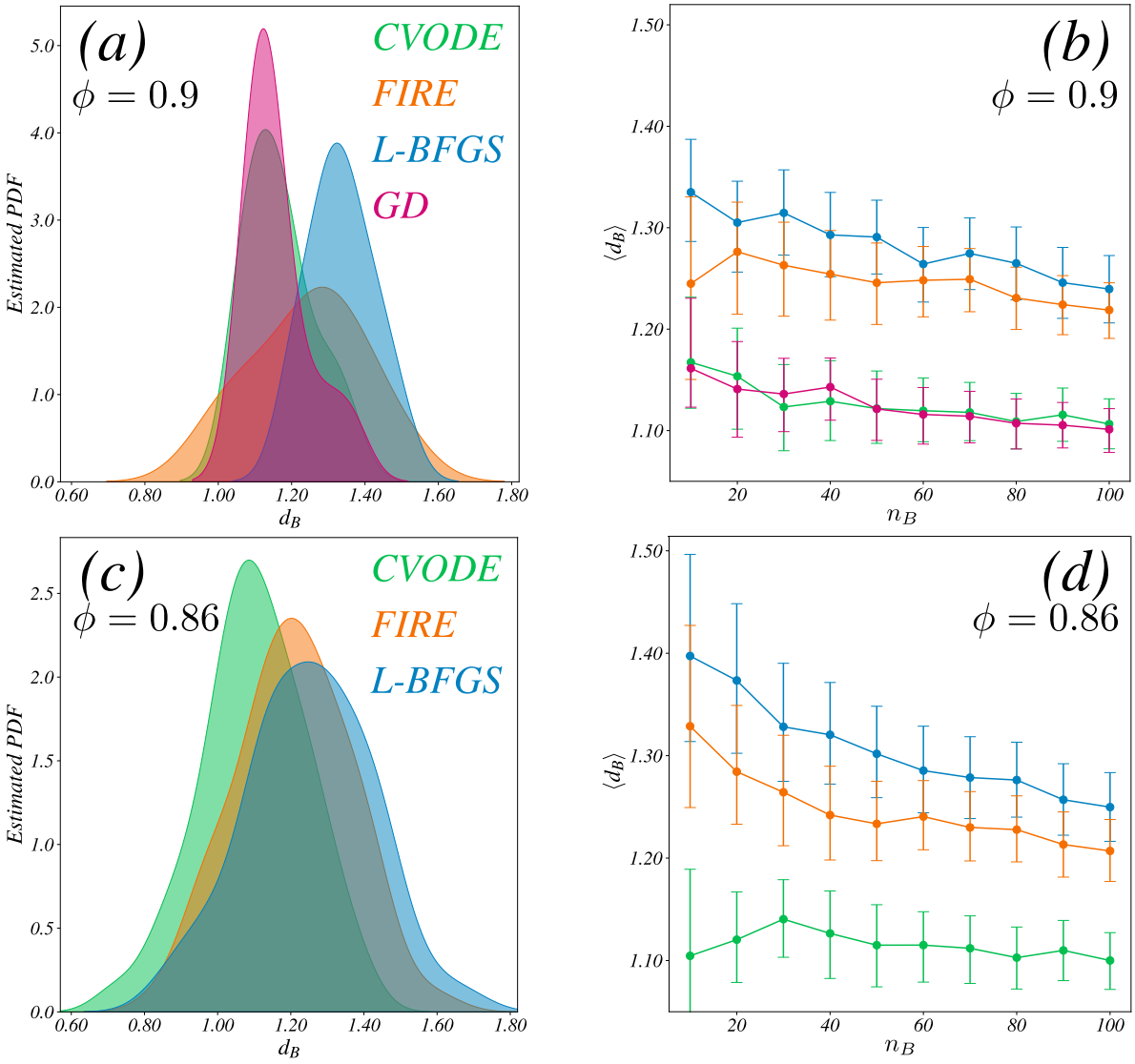

Figure 4.2: Box-counting dimension. (a) Gaussian kernel density estimate of the PDF of the box counting dimension d_B from top 10 largest basins for $N = 128$ and $\phi = 0.9$, and (b) corresponding average box counting dimension estimated from the top n_B largest basins. In (c), (d), we show the same plots for $N = 128$ and $\phi = 0.86$.

ϕ	d_B (L-BFGS)	d_B (FIRE)	d_B (CVODE)	d_B (GD)
0.9	1.34 ± 0.05	1.24 ± 0.09	1.17 ± 0.05	1.16 ± 0.06
0.86	1.40 ± 0.09	1.33 ± 0.09	1.10 ± 0.08	N/A

Table 4.1: Box-counting dimensions. Table of d_B values (with bootstrapped 95% confidence intervals) estimated from the 10 largest basins in a slice, for each method.

intervals calculated by bootstrap). We show that as smaller basins are included in the average, the estimated fractal dimension decreases significantly for FIRE and L-BFGS, while it essentially plateaus for ODE solvers. This is due to the finite resolution of slices: as we include smaller basins, the scale of the pixel size begins to dominate. As a result, for optimizers, d_B is underestimated when n_B is large.

We repeat the same measurement at $\phi = 0.86$, save for GD which becomes prohibitively expensive near jamming. The corresponding plots are shown in Fig. 4.2(c) and (d) (the slices used for this measurement are shown in 3.8). The average d_B for $n_B = 10$ are reported in Table 4.1.

Basins look more fractal ($d_B \gtrsim 1.25$) when identified with optimizers rather than ODE solvers ($d_B \approx 1.1$). As the jamming point is approached, the fractal dimension estimated with CVODE slices does not change significantly, while it becomes larger for L-BFGS and FIRE. Like line measurements, these results suggest that individual basins do not get intrinsically more fractal near jamming, implying this is just another mirage appearing in optimizer measurements. In particular, the analogy proposed in past work between basins of attraction and Apollonian gaskets [94] quoted a fractal dimension close to 1.3, which is close to what optimizers report but significantly larger than the CVODE values.

4.4 Survival Probability

4.4.1 Method

The survival probability measures the fraction of configurations that, after being displaced by a distance R, remain within their original basin of attraction. The measurement proceeds as follows:

1. **Reference Point Selection.** A point $\boldsymbol{X}$ is sampled uniformly at random from the full configuration space. The associated minimum is identified by solving steepest descent using either CVODE or an optimizer.

2. **Perturbation Sampling.** A set of n random perturbations is generated, each displacing $\boldsymbol{X}$ by a fixed distance R. We use Muller's method [140]: draw an isotropic Gaussian vector $\mathbf{g} \sim \mathcal{N}(0, I_{dN})$, then project onto the hypersphere surface:

$$\boldsymbol{X}_i = \boldsymbol{X} + R \cdot \frac{\mathbf{g}_i}{\|\mathbf{g}_i\|}, \tag{4.2}$$

   for $i = 1, \ldots, n$. This generates perturbation directions uniformly on the $(dN-1)$-dimensional unit hypersphere.

3. **Basin Identification.** Each perturbed configuration is minimized using the same method employed for the reference point. For periodic systems, structure comparison accounts for translational drift by aligning coordinates, then computing particle-wise displacements under periodic boundary conditions:

$$\Delta\mathbf{r}_i = \left(\mathbf{r}_i^{(a)} - \mathbf{r}_i^{(b)}\right) \mod L, \tag{4.3}$$

   where displacements exceeding $L/2$ are wrapped: $\Delta\mathbf{r}_i \to L - \Delta\mathbf{r}_i$. Rattlers (mechanically unstable particles with fewer than $d+1$ contacts) are excluded from the comparison. Two

structures match if $\max_i \|\Delta \mathbf{r}_i\| < \epsilon_{\text{tol}}$ with $\epsilon_{\text{tol}} = 10^{-2}$.

4. **Survival Probability.** The survival probability $p_{\text{in}}(R)$ for this reference point is the fraction of perturbed configurations returning to the same minimum:

$$p_{\text{in}}(R) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}[\text{sample } i \text{ returns to same basin}]. \tag{4.4}$$

5. **Ensemble Averaging.** The procedure is repeated for K independent reference points sampled from configuration space. The ensemble-averaged survival probability is then

$$P_{\text{in}}(R) = \langle p_{\text{in}}(R) \rangle_X, \tag{4.5}$$

where the average is taken over all reference points.

Since the number of perturbations per reference point satisfies $n \gg K$, the statistical uncertainty is dominated by variation across reference points rather than sampling noise within each basin. Accordingly, error bars are computed using BCa (bias-corrected and accelerated) bootstrap [141] with 10,000 resamples over the K reference points, yielding 95% confidence intervals for the mean survival probability.

4.4.2 Fractality

To demonstrate that the survival probability can detect fractality when present, consider a fractal object S with dimension d_f embedded in D-dimensional space. When a $(D-1)$-dimensional hypersphere of radius R intersects S, the intersection generically has dimension $d_f - 1$.

The survival probability $P_{\text{in}}(R)$ measures the fraction of the hypersphere surface occupied by S. The $(D-1)$-dimensional hypersphere has surface measure scaling as R^{D-1}, while the intersection with the fractal has $(d_f - 1)$-dimensional measure scaling as $R^{d_f - 1}$. The survival probability

therefore scales as

$$P_{\text{in}}(R) \sim \frac{R^{d_f-1}}{R^{D-1}} = R^{d_f-D}. \tag{4.6}$$

For a fractal object with $d_f < D$, this yields a power-law decay with exponent $d_f - D < 0$. For example, the Sierpinski carpet in $D = 2$ has $d_f = \log 8/\log 3 \approx 1.893$, predicting $P_{\text{in}}(R) \sim R^{-0.107}$. We verify this prediction numerically in Fig. 4.3, where the measured survival probability follows the expected power law over several decades.

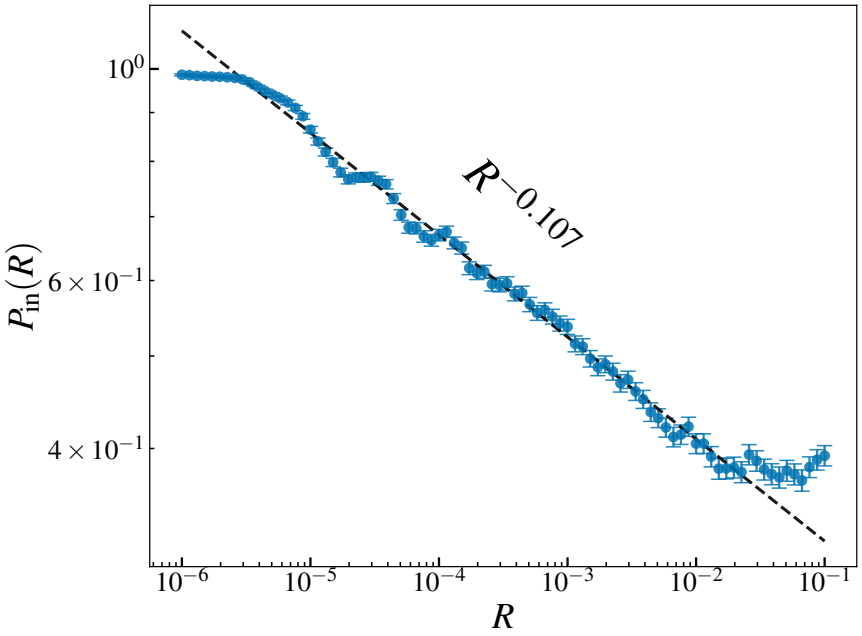

Figure 4.3: Survival probability for the Sierpinski carpet. The survival probability $P_{\text{in}}(R)$ as a function of perturbation radius R, averaged over 100 reference points sampled uniformly from inside the carpet. The dashed line shows the predicted scaling $P_{\text{in}}(R) \sim R^{d_f-D}$ with $d_f - D = \log 8/\log 3 - 2 \approx -0.107$. Error bars are Clopper-Pearson 95% confidence intervals.

To generate Fig. 4.3, we construct a Sierpinski carpet with 50 iterations and sample 100 reference points uniformly inside the carpet via rejection sampling. For each reference point and each of 86 radii (logarithmically spaced from 10^{-6} to 10^{-1}), we draw 10,000 random perturbations uniformly on a circle of that radius and check whether they fall inside the carpet. The survival probability is averaged over all reference points, with Clopper-Pearson intervals computed from the binomial counts.

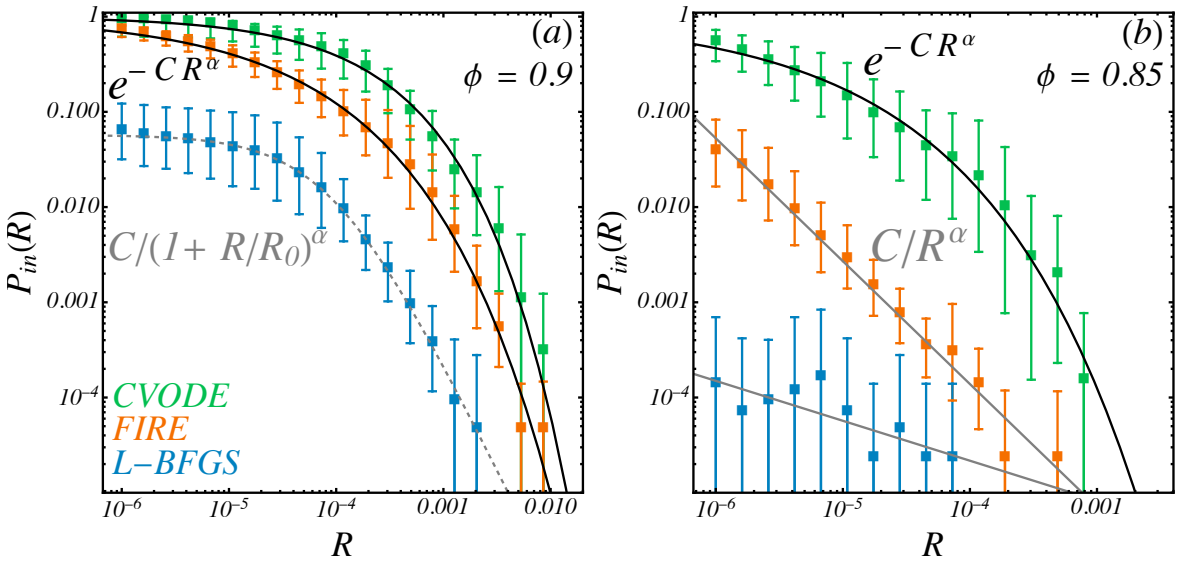

Figure 4.4: Survival (*a*) Survival from a random point in the landscape of $N = 1024$ disks, in log-log scale. Solid black lines are stretched-exponential fits for CVODE and FIRE. Dashed gray line: saturating power-law fit to L-BFGS. Error bars are BCa bootstrap 95% confidence intervals [141]. (*b*) Same curve at $\phi = 0.85$. Gray lines are power-law fits for FIRE and L-BFGS. The black line is a stretched-exponential fit to CVODE.

We note that the survival probability could serve as a measure of basin stability, analogous to the uncertainty exponent approach developed by Grebogi et al. [58–61] for characterizing sensitivity to initial conditions near fractal basin boundaries (see Section 1.1.1). However, an important distinction is that survival probability probes stability in the bulk of basins rather than near their boundaries. This difference may be relevant for certain systems. It is also important to note that the boundary-detection method of Grebogi et al. scales as $O(N^D)$ where D is the dimension of phase space, making it prohibitively expensive in high dimensions. In contrast, the survival probability measurement relies on Monte Carlo sampling, for which the number of samples required to achieve a given statistical precision is independent of dimension.

4.4.3 Method Dependence of Survival Curves

Figure 4.4(a) compares survival probability curves for L-BFGS, FIRE, and CVODE at $\phi = 0.9$. For a system of $N = 1024$ particles, we sample reference points uniformly from configuration space—13 for CVODE and 43 for the optimizers and around each reference point we sample 1000 perturbations on each of 30 nested hyperspheres of increasing radius.

The results reveal a method dependence. Optimizers consistently escape from the basin at much smaller displacements than CVODE, even at this high packing fraction. L-BFGS is particularly sensitive, showing significant basin escape even for minute perturbations.

Panel (b) shows the same measurement closer to jamming at $\phi = 0.85$. Here, both L-BFGS and FIRE display power-law decays: straight lines in log-log scale which would suggest fractal basin boundaries. However, CVODE tells a different story: it reveals basins that extend over much larger distances, with survival probability following a stretched exponential form $P_{\text{in}}(R) = \exp(-CR^{\alpha})$. This functional form has a characteristic length scale, indicating that the basins are *not* scale-free. The apparent fractality observed with optimizers is therefore another artifact of the minimization method, not an intrinsic property of the energy landscape.

4.4.4 Survival Probability Across Packing Fractions

Having established that the choice of minimization method significantly affects survival probability measurements, we now examine how this method dependence evolves as the system approaches jamming. Figure 4.5 shows survival curves across a range of packing fractions for each method.

For these measurements, we use 13 random reference configurations for CVODE and 43 for FIRE and L-BFGS. The larger number of reference points for the optimizers is necessary because their lower survival probabilities lead to higher variance at the same confidence level. Around each reference point, we sample 1000 perturbations at each of 30 different radii R, measuring the

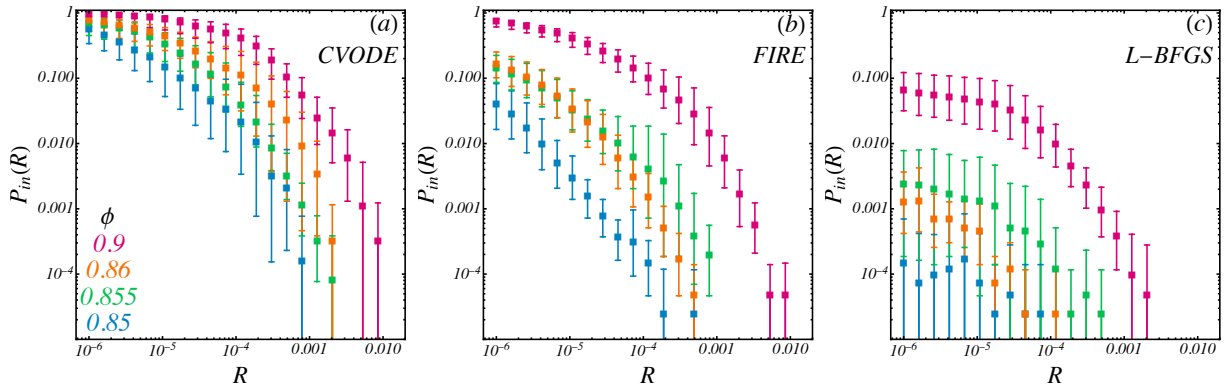

Figure 4.5: Survival probability approaching jamming. Survival probability as a function of perturbation radius R for several packing fractions approaching the jamming transition, shown in log-log scale. Panels show results for (a) CVODE, (b) FIRE, and (c) L-BFGS. Error bars are BCa bootstrap 95% confidence intervals [141].

survival probability $P_{\text{in}}(R)$ as the fraction of perturbed configurations that return to the same minimum.

The results reveal that the discrepancy between methods grows more pronounced as jamming is approached ($\phi_J \approx 0.842$). At packing fractions as high as $\phi = 0.85$, the optimizers already exhibit power-law-like decays, appearing as straight lines in log-log scale while CVODE maintains stretched-exponential behavior with a clear characteristic length scale.

4.4.4.1 Fitting Methodology

To quantify whether the survival curves exhibit scale-free (fractal) or characteristic-scale behavior, we fit each curve to two functional forms. The first is a stretched exponential,

$$P_{\text{in}}(R) = \exp\left(-\left(\frac{R}{R_{\text{se}}}\right)^{\alpha}\right), \tag{4.7}$$

where R_{se} is a characteristic length scale and α is the stretching exponent. This form describes basins with a well-defined size scale.

The second is a saturating power law,

$$P_{\text{in}}(R) = \frac{A}{\left(1 + \frac{R}{R_{\text{pl}}}\right)^{\beta}}, \tag{4.8}$$

where A is a saturation factor, R_{pl} sets the crossover scale, and β is the power-law exponent in the tail. This form captures scale-free behavior at large R, which would indicate fractal basin structure.

We perform weighted nonlinear least-squares fits in log space, with weights $1/\delta_i^2$ where $\delta_i \equiv \Delta_i / P_{\text{in}}(r_i)$ and Δ_i is the width of the 95% confidence interval at each point. When the Central Limit Theorem applies and the error can be assumed Gaussian, $\Delta_i \propto \sigma(r_i)$ where $\sigma(r_i)$ is the variance at each point. This weighting ensures that the fits accurately capture the tail behavior, which is most diagnostic for distinguishing between the two functional forms.

4.4.4.2 Fit Quality Comparison

To systematically compare fit quality across methods and packing fractions, we compute the difference in weighted residual sum of squares,

$$\Delta\text{WRSS}_{\text{log}} \equiv \text{WRSS}_{\text{power law}} - \text{WRSS}_{\text{stretched exp}}. \tag{4.9}$$

Negative values indicate that the power law provides a better fit; positive values favor the stretched exponential.

We assess the robustness of this comparison using leave-one-out (jackknife) resampling [141]. For each resampled ensemble, we compute $\Delta\text{WRSS}_{\text{log}}$ and determine the proportion P_{pl} of ensembles for which the power law fits better, with 95% Clopper-Pearson confidence intervals. The results are shown in Figure 4.6(a).

The results are clear: L-BFGS yields $P_{\text{pl}} \approx 1$ at all packing fractions, indicating that power-law

behavior appears regardless of density. FIRE shows $P_{\mathrm{pl}} \approx 0$ at high ϕ but transitions to $P_{\mathrm{pl}} \approx 1$ near jamming, suggesting emergent power-law behavior as $\phi \to \phi_J$. In contrast, CVODE maintains $P_{\mathrm{pl}} < 0.5$ at all packing fractions, with no systematic trend as jamming is approached.

These results demonstrate that the apparent power-law behavior in survival curves, and by extension, the apparent fractality of basins, can appear as an artifact of the minimization method rather than an intrinsic property of the energy landscape.

Figure 4.6(b)–(f) reports the fitted parameters for both models across all methods and packing fractions. Notably, the characteristic scale R_{se} from stretched-exponential fits with CVODE (green symbols in panel b) remains approximately constant as jamming is approached, while the corresponding L-BFGS values (blue symbols) become vanishingly small. This provides additional evidence that the basins are well-behaved shapes with finite, ϕ-independent length scales when measured correctly.

4.5 Basin Arrangement

We now turn to how the choice of minimization method affects the perceived arrangement of basins in configuration space, following the approach of Ref. [33]. We start with 13 random initial configurations uniformly sampled from phase space, For each configuration, we use CVODE with `rtol` $= 10^{-11}$ and `atol` $= 10^{-12}$ (one order of magnitude below the values in Table C.1) to locate the corresponding minimum within its basin of attraction.

For each of these 13 minima, we then apply 1000 random kicks of length R, choosing the direction uniformly in space, we do this for 60 R values. Of these 60 values, 10 are sampled uniformly in log space from 0.1 to 1 and 50 uniformly from 1 to $R = L$. In 4.7 we only present the data up to $R = L/2$, confining ourselves to observe differences in intermediate kick size regimes. After each kick, we minimize the system using CVODE, FIRE, and L-BFGS, employing the parameters listed in Tables C.1, C.2, and C.3, respectively. Finally, excluding any case in which

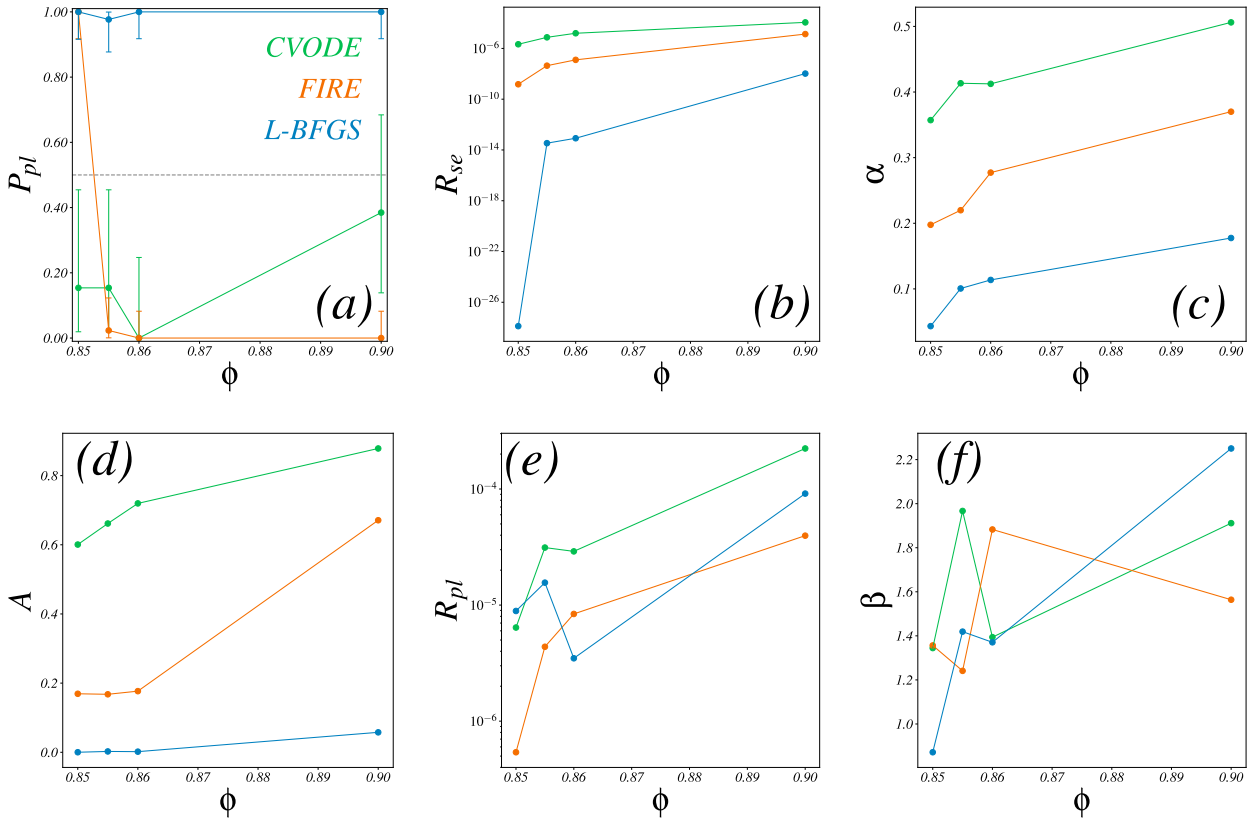

Figure 4.6: Fit quality comparison. (a) Proportion P_{pl} of leave-one-out ensembles for which the power-law fit outperforms the stretched exponential, as a function of packing fraction. Panels (b)–(f) show the fitted parameters for each model and method: (b) R_{se}, (c) α, (d) A, (e) R_{pl}, and (f) β.

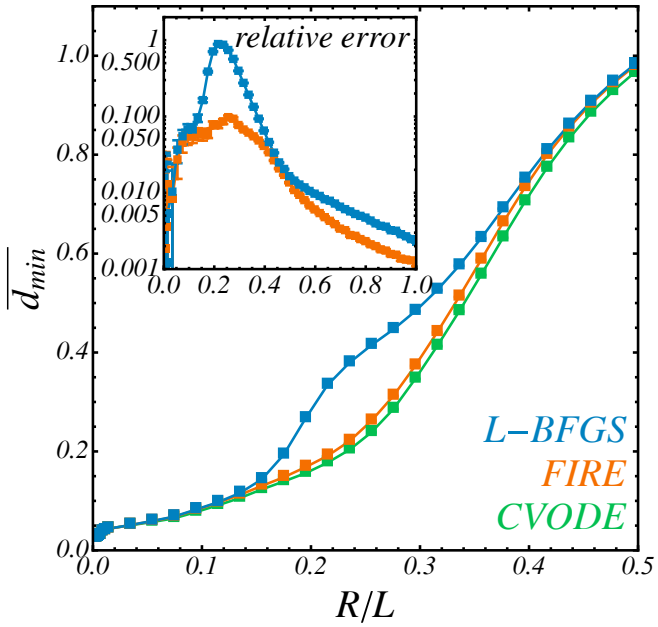

Figure 4.7: Distance between neighboring minima. Average normalized distance $\overline{d_{\min}}$ between an initial minimum and minima reached after perturbations of size R, for $N = 1024$ particles. Inset: relative error in $\overline{d_{\min}}$ for optimizers compared to CVODE, shown on a logarithmic scale.

the new minimum coincides with the original one, we compute the normalized metric distance between the new minimum M' and the original one M,

$$d_{min}(M, M') \equiv \frac{1}{2\langle R_i \rangle} \sqrt{\sum_{i<j} \left(C_{ij} - C'_{ij} \right)^2} \tag{4.10}$$

with $\langle R_i \rangle$ the average particle radius and C_{ij}, C'_{ij} are the "stable contact vectors" between particles i and j in minima M and M', respectively. These contact vectors are defined as

$$C_{ij} = \boldsymbol{r}_{ij} \mathbb{1}(r_{ij} \leq R_i + R_j), \tag{4.11}$$

that is, the distance vector if the particles are in contact, and 0 otherwise. This definition ensures that the measured distance is unambiguous in spite of the presence of rattlers and of translational symmetry.

Figure 4.7 shows that for $N = 1024$, the average normalized metric distance $\overline{d_{min}}$ between

original and neighboring minima is systematically shifted when using optimizers compared to CVODE, with relative deviations of 10 to 80% at intermediate perturbation radii R, with optimizers reaching further than than CVODE, including a illusory "hump" with L-BFGS. This suggest that measured basin organization is also affected by minimization method

4.6 Discussion

While Chapter 2 established that optimizers fail at basin identification with increasing system size, this chapter demonstrates that these failures produce systematic biases in measured basin geometry that are both quantitative and qualitative. The results presented here show that apparent fractal properties of basins of attraction in soft sphere packings can be attributed to optimizer errors rather than genuine features of the energy landscape.

The evidence is consistent across measurement approaches. Box-counting dimensions of basin boundaries drop from $d_B \approx 1.3$–1.4 with optimizers to $d_B \approx 1.1$ with CVODE, quantifying the roughness apparent in the slices in Chapter 2. Survival probabilities exhibit stretched-exponential decay with CVODE, indicating basins with finite length scales, whereas optimizers produce power-law decays that would incorrectly suggest scale-free, fractal structure. Finally, measurements of basin arrangement show that optimizers systematically overestimate the distance between neighboring minima, distorting the perceived organization of configuration space.

While the majority of our results are in the overjammed regime, the discrepancy between methods grows more pronounced in survival probability measurements as the jamming transition is approached, including qualitative changes in the functional form. Near jamming, optimizers produce increasingly fragmented basin maps while CVODE reveals basins that remain geometrically well-behaved. The characteristic length scale extracted from CVODE persists across packing fractions, indicating that basin geometry does not undergo a fundamental change near jamming—contrary to what optimizer-based measurements would suggest.

One note regarding methodology: while we used the survival probability primarily as a tool to distinguish fractal from non-fractal basins, it can also measure the fractal dimension of a basin directly. Unlike the boundary-detection methods of Grebogi et al. [58, 59], which scale as $O(N^D)$ in the dimension D of configuration space, survival probability relies on Monte Carlo sampling whose computational cost is dimension independent. This may make it a valuable tool for extracting fractal dimensions to quantify global stability of basins in high-dimensional systems, although this direction is not explored deeply in this thesis.

The results of this chapter and the preceding one establish a clear methodological prescription: reliable characterization of basin geometry requires solving the steepest descent ODE with appropriate numerical methods rather than relying on optimizers. For soft sphere packings above the jamming transition, this reveals basins that are smooth, log-normally distributed in size, and possess well-defined characteristic length scales. This builds a case to re-verify previous studies which used optimizers[29, 35, 36] to extract geometric information.

5 | Sampling uniformly in a high-dimensional basin

> The total unsuitability of the harmonic mean estimator should have been apparent within an hour of its discovery
>
> — Radford Neal, *Blog*

5.1 Introduction

As discussed in Section 1.2.3, the volume of a basin can be expressed as an integral over configuration space weighted by an indicator function. Evaluating this indicator requires solving the steepest descent equation from each sampled configuration—a computationally expensive operation that makes direct volume estimation intractable. Moreover, because jammed packings possess an astronomical number of minima, individual basins occupy vanishingly small fractions of configuration space, rendering naive sampling over the full phase space hopeless.

This chapter describes a method that circumvents these difficulties by generating uniform samples directly within a basin of interest using Markov Chain Monte Carlo (MCMC) that was developed by Frenkel and collaborators [16–18, 30]. Rather than estimating volumes by counting hits from global sampling, we constrain random walkers to remain inside a single basin and use

free energy estimation techniques to reconstruct the absolute volume. The central challenge in high dimensions is that basins concentrate their volume in narrow, elongated regions far from the minimum, causing single random walks to become trapped. We overcome this using parallel tempering with harmonic biasing potentials: an ensemble of walkers, each confined by springs of different stiffness, collectively explores all radial regions of the basin while exchanging configurations to maintain ergodicity.

We begin in Section 5.2 by formulating the sampling problem and explaining why single-chain approaches fail in high dimensions. We then introduce parallel tempering and the Multistate Bennett Acceptance Ratio (MBAR) estimator for free energy differences. Section 5.3 details the practical implementation: determining the spring constant range, running the parallel tempering simulation, sampling an inner reference sphere, and combining these elements to estimate the basin volume.

5.2 Drawing uniform samples from a high-dimensional basin

As described in Sec. 1.2, the volume of a basin of attraction $\mathcal{B}_\alpha$ associated with minimum α can be written as an integral over configuration space:

$$v_\alpha = \int d\boldsymbol{X}\, \mathbb{1}_{\mathcal{B}_\alpha}(\boldsymbol{X}), \tag{5.1}$$

where $\mathbb{1}_{\mathcal{B}_\alpha}(\boldsymbol{X})$ is the indicator function that returns 1 if $\boldsymbol{X}$ belongs to basin α and 0 otherwise. Evaluating this indicator function requires solving the steepest descent equation from configuration $\boldsymbol{X}$ and checking whether the resulting minimum matches $\boldsymbol{X}^\alpha_{IS}$.

The most naive approach to estimating this integral would be to sample uniformly over the entire configuration space and count the fraction of points that land in basin α, $p_\alpha = n_\alpha/N_s$,

where n_α is the number of samples landing in basin α and N_s is the total number of samples. The volume can then be estimated as $\hat{v}_\alpha = p_\alpha \cdot V_{\text{tot}}$. However, this strategy is hopelessly inefficient for jammed sphere packings. As discussed in Sec. 1.1.2, the number of minima in these systems is astronomically large, and correspondingly, individual basin volumes are vanishingly small fractions of the total configuration space. The probability of hitting any specific basin even twice becomes negligible for systems of even modest size, meaning that it is impossible to estimate v_α this way.

The approach we take, as briefly introduced in Sec. 1.2.3, is to instead generate samples within the basin itself using a Markov Chain Monte Carlo (MCMC) walk constrained to remain inside $\mathcal{B}_\alpha$. The samples generated can then be used to reconstruct the volume through a process we will discuss in detail below. As an additional benefit, these uniform samples within the basin can also be used to extract information about the basin structure.

5.2.1 The problem with a single chain

The standard approach to sample uniformly within a basin is to use a Metropolis-Hastings random walk. We initialize a walker at some configuration $\boldsymbol{X}_0 \in \mathcal{B}_\alpha$, typically at the minimum $\boldsymbol{X}_{IS}^\alpha$ itself. At each step t, the walker proposes a move to a new configuration:

$$\boldsymbol{X}' = \boldsymbol{X}_t + \delta\boldsymbol{\eta}, \tag{5.2}$$

where $\boldsymbol{\eta}$ is a vector of independent standard normal random variables and δ is a step size parameter that controls the typical displacement. Since we aim to sample uniformly within the basin, the target distribution is flat and the Metropolis-Hastings acceptance criterion simplifies to:

$$P_{\text{accept}} = \begin{cases} 1 & \text{if } \boldsymbol{X}' \in \mathcal{B}_\alpha, \\ 0 & \text{otherwise.} \end{cases} \tag{5.3}$$

In other words, the walker accepts any proposed move that remains inside the basin and rejects moves that would exit it. Checking basin membership requires minimizing from $\boldsymbol{X}'$ and verifying that the resulting minimum matches $\boldsymbol{X}_{IS}^{\alpha}$. After sufficient equilibration, random samples X_n from the uniform distribution over $\mathcal{B}_\alpha$ can be obtained by decorrelating the sequence of configurations $\{\boldsymbol{X}_t\}$.

However, a naive approach using a single such random walk faces fundamental challenges from high-dimensional geometry. In high dimensions, even convex bodies concentrate their volume far from the center—a hypercube, for example, has most of its mass in its 2^d corners (see Figure 5.1). Basins of attraction exhibit similar structure: narrow, elongated regions extending from a small convex core [21, 27, 30, 142]. A random walker becomes trapped in these regions, and transitioning between them requires traversing the core—an event whose probability decreases exponentially with dimension. Adequate sampling demands either $O(2^d)$ independent walkers to sample every corner or equivalently long equilibration times for one walker to sample every corner, rendering direct estimation intractable.

5.2.2 Parallel Tempering

The trapping problem can be circumvented by running multiple random walks simultaneously, each biased to explore a different radial region of the basin. Instead of a single free walker, we introduce a collection of walkers tethered to the minimum $\boldsymbol{X}_{IS}^{\alpha}$ by harmonic springs of varying stiffness k. Each walker samples configurations according to a biasing potential

$$U_k(\boldsymbol{X}) = \frac{k}{2}\left|\boldsymbol{X} - \boldsymbol{X}_{IS}^{\alpha}\right|^2, \tag{5.4}$$

which confines the walker to a region whose typical distance from the minimum depends on k. A walker with large positive k remains tightly localized near the minimum, sampling primarily the small convex core of the basin. As k decreases toward zero, the confining force weakens and

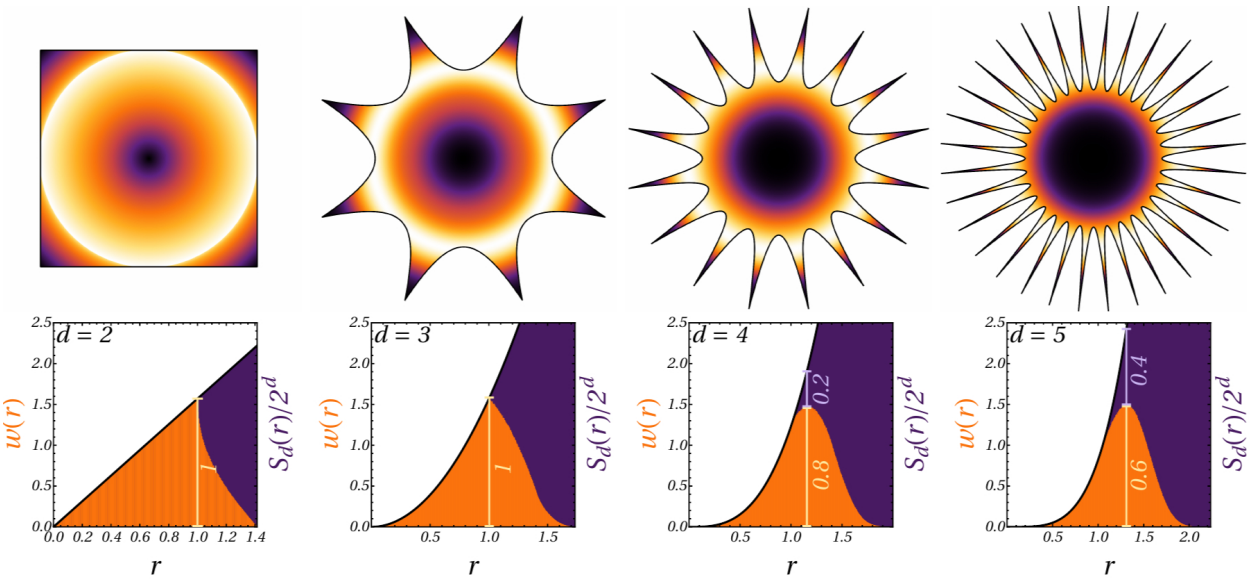

Figure 5.1: Volume concentration in high-dimensional hypercubes. *Top*: Visualizations of hypercubes in $d = 2, 3, 4$, and 5 dimensions, with color indicating the radial distribution of volume. As dimensionality increases, volume shifts from the center (dark) toward the corners (light). *Bottom*: The relative weight $w(r)$ of each radial shell—the fraction of total volume contained in a thin spherical shell at distance r from the center—for the hypercube (orange) compared to a hyperball (purple). In low dimensions, the hypercube's volume peaks near its inscribed sphere, but as d grows, the distribution shifts to larger radii where the corners dominate, illustrating why a random walker in high dimensions becomes trapped in corners far from the central core. Figure adapted from Ref. [142].

the walker explores progressively larger distances from the minimum. At $k = 0$, the walker is completely unbiased and free to explore the entire basin volume including the "tentacles" where most of the volume resides.

Crucially, negative values of k can also be employed. A negative spring constant creates a repulsive potential that pushes the walker away from the minimum, encouraging exploration of the outermost regions of the basin where the tentacles extend furthest. By spanning a range of k values from large positive (tight confinement) through zero (free exploration) to moderately negative (outward bias), the ensemble of walkers collectively samples all radial regions of the basin.

The key ingredient that makes this approach tractable is parallel tempering [143, 144]: periodic exchanges of configurations between walkers with neighboring values of k. Consider two walkers with spring constants k_i and k_{i+1}, currently at configurations $\boldsymbol{X}_i$ and $\boldsymbol{X}_{i+1}$. A swap of their configurations is proposed and accepted with probability

$$P(\boldsymbol{X}_i, k_i \mid \boldsymbol{X}_{i+1}, k_{i+1}) = \min\left(1, \exp\left[-\Delta U\right]\right), \tag{5.5}$$

where $\Delta U = (k_{i+1} - k_i)\left(\left|\boldsymbol{X}_i - \boldsymbol{X}_{IS}^{\alpha}\right|^2 - \left|\boldsymbol{X}_{i+1} - \boldsymbol{X}_{IS}^{\alpha}\right|^2\right)/2$ is the change in total biasing energy upon exchange. This Metropolis-type acceptance criterion is derived by imposing detailed balance on the transition matrix, ensuring that each walker continues to sample from its target distribution.

In practice, exchanges are attempted simultaneously for all even-indexed pairs $(0{-}1, 2{-}3, \ldots)$ or all odd-indexed pairs $(1{-}2, 3{-}4, \ldots)$, alternating between the two sets [145]. This even-odd scheme does not satisfy detailed balance for individual pairs, and hence is non-reversible. However, it does satisfy *global balance*: the total probability flux into any configuration equals the flux out at steady state. Global balance suffices to guarantee convergence to the correct equilibrium distribution while being less restrictive than detailed balance. Saifuddin et al. [146] have also shown that such non-reversible schemes significantly outperform their reversible counterparts.

The exchange mechanism solves the trapping problem as follows. When a walker at low k becomes stuck in a tentacle far from the center, it can swap configurations with a neighboring walker at higher k that happens to be closer to the core. The low-k walker then finds itself in a new region of the basin, having effectively tunneled through the narrow core without actually traversing it. Meanwhile, the higher-k walker that received the distant configuration will tend to drift back toward the center due to its stronger confining potential. Through repeated exchanges cascading up and down the ladder of k values, configurations diffuse efficiently between the core and the periphery, enabling exploration of the entire basin.

A key advantage of this approach is its favorable scaling with dimension. While direct sampling would require exponentially many walkers to cover the $O(2^d)$ corners of a high-dimensional basin, the number of replicas needed for parallel tempering grows only roughly linearly with dimension d [142]. This improvement arises because each walker need not visit every corner individually; instead, the exchange mechanism allows information about distant regions to propagate through the ensemble. The required number of replicas is determined by ensuring sufficient overlap between the radial distributions sampled by neighboring walker.

5.2.3 Estimating a normalizing constant: MBAR

While parallel tempering generates samples from each biased distribution, these samples only reflect relative probabilities. The volume, i.e. the absolute normalization constant at $k = 0$ (the partition function $Z_k = \int e^{-u_k(X)} d\boldsymbol{X}$, evaluated at $k = 0$), remains unknown. To obtain it, we estimate the free energy difference $\Delta f = -\ln(Z_0/Z_{\text{ref}})$ between the unbiased state and a reference state with a known normalization constant.

Given samples from multiple replicas with different biasing potentials, we need a method to estimate the free energy difference between them—and ultimately, the volume of the unbiased basin. The Multistate Bennett Acceptance Ratio (MBAR) [98] provides a principled solution by finding the maximum likelihood estimate for the free energies $f_i = -\ln Z_i$ of each state. The

resulting estimator is:

$$\hat{f}_i = -\ln \sum_{j=1}^{K} \sum_{n=1}^{N_j} \frac{e^{-u_i(\boldsymbol{X}_{jn})}}{\sum_{\ell=1}^{K} N_\ell \, e^{\hat{f}_\ell - u_\ell(\boldsymbol{X}_{jn})}} \tag{5.6}$$

where $u_i(\boldsymbol{X})$ is the reduced potential for state i, and the sums run over all K states and all samples $\boldsymbol{X}_{jn}$ collected from each state. This set of coupled nonlinear equations must be solved self-consistently; every sample from every state contributes to each free energy estimate, optimally combining all available information. MBAR is a *maximum likelihood estimator*, therefore it has special properties amongst the class of estimators that estimate free-energy differences with the same assumptions [147].

First, it is *consistent*: as the number of samples grows, the estimates converge to the true free energy differences. Second, it is *asymptotically normal*: the distribution of estimates becomes Gaussian in the large-sample limit, which allows straightforward uncertainty quantification. Third, it is *asymptotically efficient*: among all unbiased estimators, MLEs achieve the smallest possible variance (the Cramér-Rao lower bound) as sample size increases.

In practice, we use the `pymbar` library [98] to solve these equations iteratively.

5.3 Procedure

We start by randomly sampling a configuration in phase space and map it to it's corresponding inherent structure via solving steepest descent/using a proxy optimizer. we then perform the following procedure

5.3.1 Finding $k_{\max}$

Before running parallel tempering, we must determine the maximum spring constant $k_{\max}$ that keeps the most tightly confined walker within the basin. The goal is to find the largest k such that a configuration sampled from the harmonic bias still belongs to the original basin with high

probability—typically 90%. A walker confined by a stiffer spring explores a smaller region around the minimum, making it easier to remain in the basin; one with a weaker spring samples larger displacements and is more likely to escape.

The algorithm proceeds as follows:

1. Begin with an initial guess for the spring constant k.
2. Sample a configuration by displacing all particles from the minimum according to a Gaussian distribution whose width is set by the spring constant (tighter springs yield smaller displacements).
3. Quench the displaced configuration to its nearest energy minimum and check whether it matches the original packing.
4. Repeat for many samples and compute the acceptance fraction—the proportion that remain in the basin.
5. Adjust k based on the observed acceptance:
    - If the acceptance exceeds the target, decrease k to allow larger displacements.
    - If the acceptance falls below the target, increase k to further constrain the walker.
6. Iterate until the acceptance converges to the target value.

The resulting $k_{\max}$ defines the tightest confinement used in the parallel tempering simulation and sets the scale for the innermost replica's sampling region.

5.3.2 Implementation details of parallel tempering

The parallel tempering simulation is organized around a coordinator-worker architecture, developed by Klicpera [148]. A single coordinator process manages the simulation, while multiple

worker processes run Monte Carlo simulations in parallel. This allows replicas to be simulated concurrently, with the coordinator handling exchanges and convergence testing.

The procedure operates as follows:

1. **Initialize replicas.** Create a set of replicas spanning a range of spring constants: positive values from $k_{\max}$ down to $k = 0$ (the free walker), plus a subset of negative values that push the walker outward to sample the basin periphery.

2. **Distribute work.** The coordinator assigns each replica to an available worker. Workers that finish early receive new replicas, balancing the computational load dynamically.

3. **Run Monte Carlo.** Each worker evolves its replica for a fixed number of steps under the assigned harmonic bias. At each step, the squared displacement from the minimum,

$$r^2(t) = \left|\boldsymbol{X}(t) - \boldsymbol{X}_{IS}^{\alpha}\right|^2, \tag{5.7}$$

   is recorded, building up a time series $\{r^2(t)\}$ for each replica.

4. **Collect results.** Workers return their updated configurations and squared-displacement time series to the coordinator.

5. **Attempt exchanges.** The coordinator proposes swaps between neighboring replicas according to the acceptance criterion in Eq. (5.5). Exchanges alternate between even-indexed pairs (replicas 0–1, 2–3, etc.) and odd-indexed pairs (replicas 1–2, 3–4, etc.) on successive iterations as described in 5.2.2.

6. **Repeat until convergence.** Steps 2–5 continue until the r^2 time series has equilibrated and accumulated enough uncorrelated samples. For replica i with time series $\{r_i^2(t)\}$ having post-equilibration mean $\mu_i = \langle r_i^2 \rangle$, variance σ_i^2, and integrated autocorrelation time τ_i

(computed via FFT), the correlated standard error after n samples is [149]

$$\mathrm{SE}_i = \sqrt{\frac{\sigma_i^2(1 + 2\tau_i)}{n}}. \tag{5.8}$$

Convergence is achieved when the relative standard error SE_i/μ_i falls below a threshold ϵ for all replicas. If not yet converged, the simulation is extended adaptively: the required sample size for replica i is estimated as [148]

$$M_i = \frac{\sigma_i^2(1 + 2\tau_i)}{(\mu_i \epsilon)^2}, \tag{5.9}$$

and additional PT iterations are scheduled to collect $\max_i(M_i) - n$ more samples. In practice we also have a maximum number of iterations

The dynamic loading system accounts for the disparity in basin mapping times—fast for high k, slow for low k—leading to significantly improved run times compared to running each replica on it's own *CPU*.

5.3.3 Inner sphere sampling

The MBAR estimator described in Section 5.2.3 provides free energy *differences* between states, but to compute the absolute basin volume we need a reference state with a known normalization. The parallel tempering simulation samples configurations at various spring constants k, but does not directly probe the innermost region very close to the minimum where the geometry is simplest. The inner sphere sampling step provides this missing reference.

The idea is to sample uniformly within a small ball of radius $r_{\min}$ centered at the minimum $\boldsymbol{X}_{IS}^{\alpha}$:

$$\mathcal{S}_{\min} = \{\boldsymbol{X} : \left|\boldsymbol{X} - \boldsymbol{X}_{IS}^{\alpha}\right| < r_{\min}\}. \tag{5.10}$$

If this ball were entirely contained within the basin, its volume would simply be the volume of an N-dimensional ball,

$$V_N(r) = \frac{\pi^{N/2}}{\Gamma(N/2+1)} r^N. \tag{5.11}$$

In practice, even for small $r_{\min}$, some configurations within the ball may minimize to a different inherent structure. We therefore measure the *acceptance fraction* p_{acc}—the probability that a uniformly sampled configuration within the ball belongs to basin α—by drawing many samples and checking basin membership for each. The effective reference volume is then

$$V_{\text{ref}} = V_N(r_{\min}) \cdot p_{\text{acc}}. \tag{5.12}$$

The radius $r_{\min}$ is chosen to be small enough that p_{acc} remains high (typically above 50%), ensuring a reliable reference, while still providing overlap with the innermost parallel tempering replica. In practice, $r_{\min}$ is determined from the mean squared displacement observed at the highest spring constant in the parallel tempering simulation: given $\langle r^2 \rangle_{k_{\max}}$, we set $r_{\min} \propto \sqrt{\langle r^2 \rangle_{k_{\max}}}$.

In addition to measuring the acceptance fraction, we also record a time series of displacements $r(t) = \left| \boldsymbol{X}(t) - \boldsymbol{X}_{IS}^{\alpha} \right|$ from configurations sampled with a Gaussian distribution of width $\sigma \sim r_{\min}$. This time series is included in the MBAR analysis as an additional state, providing crucial information for connecting the reference volume to the unbiased distribution.

5.3.4 Estimating volume

With the parallel tempering time series and inner sphere data in hand, we now describe how to compute the basin volume. The key insight is that the negative logarithm of the basin volume,

$$F_\alpha \equiv -\ln v_\alpha, \tag{5.13}$$

can be interpreted as a free energy: it is the negative log of a partition function (the integral over the basin). This interpration allows us to put it in context of free-energy techniques developed in computational chemistry.

The inner sphere provides a reference free energy based on the measured acceptance fraction:

$$F_{\min} = -\ln V_N(r_{\min}) - \ln p_{\text{acc}}, \tag{5.14}$$

where $V_N(r_{\min})$ is the volume of the reference ball (Eq. 5.11) and p_{acc} is the acceptance fraction. This anchors the free energy scale at a known absolute value corresponding to the effective volume $V_{\text{ref}} = V_N(r_{\min}) \cdot p_{\text{acc}}$ of the innermost region.

To connect this reference to the full basin, we must estimate the free energy difference $\Delta f_{\text{inner}\to k=0}$ between the inner sphere state and the unbiased ($k = 0$) state. Recall that each state i in our umbrella sampling scheme has a partition function $Z_i = \int_{\mathcal{B}_\alpha} e^{-u_i(X)} d\boldsymbol{X}$ and corresponding free energy $f_i = -\ln Z_i$. The free energy difference

$$\Delta f_{\text{inner}\to k=0} = f_{k=0} - f_{\text{inner}} \tag{5.15}$$

quantifies how the accessible phase space volume changes as we remove the biasing potential. The MBAR estimator (Section 5.2.3) provides the maximum likelihood estimate of this quantity by optimally combining all samples from the parallel tempering replicas and the inner sphere.

When constructing the reduced potentials for this analysis, the inner sphere state requires special treatment:

$$u_{\text{inner}}(r) = (N-1)\ln r + \frac{k_{\text{inner}}}{2} r^2, \tag{5.16}$$

where $r = \left|\boldsymbol{X} - \boldsymbol{X}_{IS}^{\alpha}\right|$ and k_{inner} is the effective spring constant corresponding to the Gaussian sampling width. The first term is the log density of states for an N-ball: for uniform density in configuration space, the radial density of states scales as $g(r) \propto r^{N-1}$ (the surface area of an

$(N-1)$-sphere), so $\ln g(r) = (N-1)\ln r$. This accounts for the greater entropy associated with regions further from the origin, and must be included to correctly reweight the inner sphere samples alongside the parallel tempering data. The parallel tempering replicas use the standard harmonic form $u_i(r) = \frac{k_i}{2} r^2$.

The negative log basin volume is then estimated as

$$\widehat{F}_\alpha = F_{\min} - \widehat{\Delta f}_{\text{inner}\to k=0}, \tag{5.17}$$

where the hat denotes that $\widehat{\Delta f}$ is the MBAR estimate of the true free energy difference. Hence, the estimate the volume by $\hat{v}_\alpha = e^{-\widehat{F}_\alpha}$.

5.3.5 Computing the radial density of states

Beyond the total volume, the parallel tempering data also yield the *radial density of states* $\xi(r)$—the density of configurations at distance $r = \left|X - X^\alpha_{IS}\right|$ from the minimum. This quantity provides detailed information about basin geometry and satisfies

$$v_\alpha = \int_0^\infty \xi(r)\, dr. \tag{5.18}$$

For comparison, an N-dimensional ball of radius R has $\xi(r) = S_{N-1} r^{N-1}$ for $r < R$ and zero otherwise, where S_{N-1} is the surface area of the unit $(N-1)$-sphere.

Each replica k samples configurations from the biased distribution

$$p_k(r) = \frac{\xi(r)\, e^{-u_k(r)}}{Z_k}, \tag{5.19}$$

where $u_k(r)$ is the reduced biasing potential and $Z_k = \int \xi(r)\, e^{-u_k(r)}\, dr$ is the partition function. For the parallel tempering replicas with harmonic bias, $u_k(r) = \frac{k}{2} r^2$. For the inner sphere state, which samples uniformly from a Gaussian distribution in configuration space, the effective re-

duced potential is

$$u_{\text{inner}}(r) = (N-1)\ln r + \frac{k_{\text{inner}}}{2}r^2, \tag{5.20}$$

where the logarithmic term accounts for the radial density of states of an N-ball: uniform sampling in configuration space yields a radial distribution proportional to r^{N-1}.

Taking the logarithm of Eq. (5.19) and rearranging gives

$$\ln \xi(r) = \ln p_k(r) + u_k(r) + f_k, \tag{5.21}$$

where $f_k = -\ln Z_k$ is the dimensionless free energy of state k. This expression shows that the unbiased density of states can be recovered from the biased distribution by adding back the biasing potential and an overall normalization constant.

To estimate $\xi(r)$ from the simulation data, we bin the time series $\{r(t)\}$ from each replica into histograms. Let $n_k(r)$ denote the (normalized) histogram count for replica k at radial bin r, which provides an estimate $\hat{p}_k(r) \approx p_k(r)$. The free energy offsets $\hat{f}_k$ are obtained from MBAR (Section 5.2.3). Each replica thus provides an estimate of $\ln \xi(r)$ via Eq. (5.21), but only at radii where it has samples.

To combine the estimates from all replicas, we take a weighted average using the histogram counts as weights:

$$\ln \hat{\xi}(r) = \frac{\sum_{k=1}^{K} n_k(r)\left[\ln n_k(r) + u_k(r) + \hat{f}_k\right]}{\sum_{k=1}^{K} n_k(r)}. \tag{5.22}$$

This estimator gives more weight to replicas that sample a given radial bin more frequently, and naturally handles regions where only a subset of replicas contribute. The denominator $\sum_k n_k(r)$ represents the total number of visits to bin r across all replicas.

To characterize basin shape, we compare $\xi(r)$ to the density of states of a hyperball in the same dimension, which scales as r^{N-1}. Plotting $\log\left(\xi(r)/r^{N-1}\right)$ yields a constant for a hyperball; deviations from constancy reveal how the basin geometry differs from spherical. As shown in

Chapter 6, this quantity decreases with r in real basins, reflecting the narrowing of "tentacles" far from the minimum.

5.4 Validation on hypercubes

To validate the volume estimation procedure, we apply it to a potential with known basin geometry:

$$U(\boldsymbol{x}) = \left(1 + d + \sum_{i=1}^{d} \cos(2\pi x_i)\right)^{1/2}, \tag{5.23}$$

The cosine terms have period 1 in each coordinate, so the potential has a periodic structure that tiles configuration space into unit hypercubes. Each minimum lies at an integer lattice point $\boldsymbol{x}^* \in \mathbb{Z}^d$, and its basin of attraction is the surrounding hypercube $[x_i^* - \frac{1}{2}, x_i^* + \frac{1}{2}]^d$. Consequently, every basin has volume $v = 1$, and the negative log volume should vanish: $F_0 = -\ln(1) = 0$. Deviation from zero indicates systematic bias in the estimation method. We note that in this thesis we're interested in basin volumes, hence we create this test potential to test basin assigment. For results on how the free-energy volume calculation performs purely on hypercubes, we refer readers to Casiulis et al [142] showing correctness upto $d = 1000$.

We test CVODE, FIRE, and L-BFGS across dimensions ranging from $d = 8$ to $d = 512$. For each dimension and minimizer, we run the procedure on hypercubes with initial conditions sampled uniformly at random. Figure 5.2 shows the results.

CVODE and FIRE produce volume estimates consistent with the true value ($F_0 = 0$) across all dimensions tested. L-BFGS, however, begins to deviate from zero at high dimensions, suggesting that it may introduce systematic bias in this regime. We examine how the methods compare on real basins in the next section.

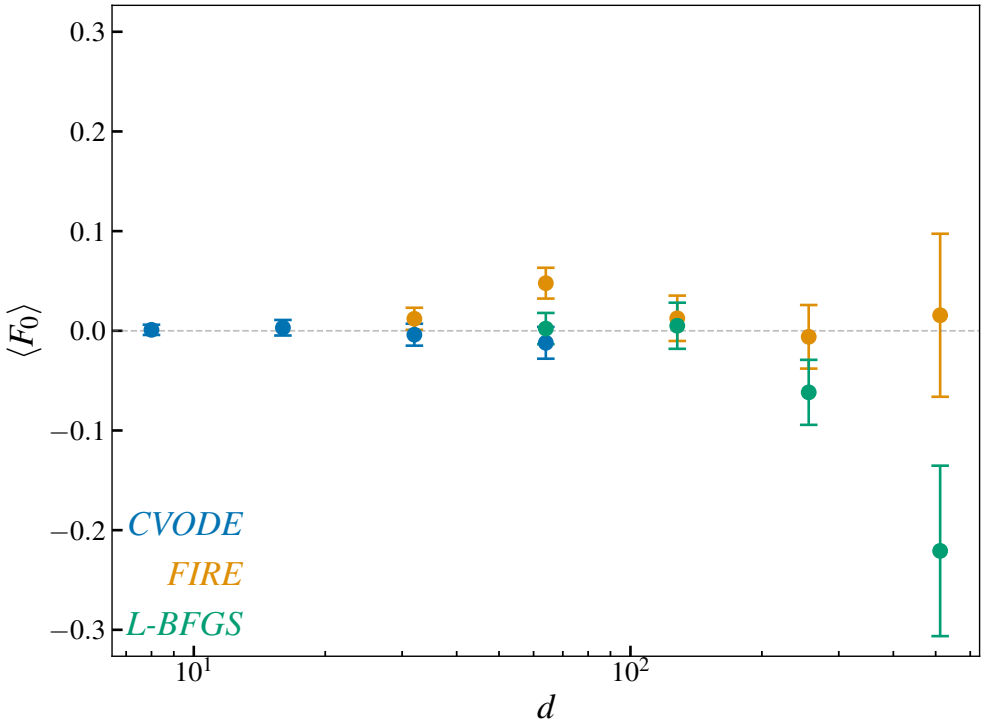

Figure 5.2: Validation of the volume estimation method on unit hypercubes. The estimated free energy $\langle F_0 \rangle$, which should equal zero for a unit hypercube, is plotted against dimension d for three minimizers: CVODE (blue), FIRE (orange), and L-BFGS (green). Error bars represent the standard error ($n = 10$, except $n = 3$ for $d = 512$). CVODE and FIRE remain consistent with zero across all dimensions, while L-BFGS begins to deviate at high d.

6 | Volume and Radial Structure of Basins

Somehow everything just fits perfectly together, and it's sort of a miracle.

— Henry Cohn, on E_8 and the Leech lattice

6.1 Introduction

As discussed in Section 1.1.2, basin volumes provide a natural measure of global stability in dynamical systems: basins occupying larger fractions of phase space are more likely to be reached from random initial conditions. In the dynamical systems community, naive Monte Carlo sampling suffices when basins occupy significant volume fractions but this approach is limited to systems with either low dimensional state spaces or where the basins of interest occupy a significant fraction of phase space. For jammed sphere packings, the astronomical number of minima renders individual basins vanishingly small, making naive sampling hopeless. Frenkel and collaborators [16–18, 30] therefore developed the MCMC-based approach described in the previous chapter, which constrains random walkers within a single basin and reconstructs the volume through free energy estimation.

This method has a critical dependence on accurate basin identification: checking whether a

configuration belongs to a given basin requires minimization at every MCMC step—approximately 10^6 minimizations per basin. In Chapter 3, we demonstrated that commonly used optimizers exhibit exponentially decaying accuracy with system size, failing catastrophically for systems larger than a few dozen particles. In Chapter 4, we showed that these errors produce qualitative artifacts in measured basin geometry, with apparent fractality arising from optimizer misattribution rather than intrinsic landscape properties. This raises an immediate concern: if optimizers cannot reliably identify which basin a point belongs to, how trustworthy are the volumes calculated using them?

This chapter addresses that question directly. We apply the parallel tempering method to measure basin volumes using both CVODE and optimizer proxies, revealing systematic biases that grow with system size. By examining the radial density of states within basins, we trace these discrepancies to their origin: optimizer accuracy depends strongly on distance from the minimum, with errors accumulating precisely in the "tentacle" regions where most of the volume resides in high dimensions. These results demonstrate that accurate volume measurements require solving the steepest descent ODE, and provide new insight into the structure of basins far from their minima.

6.2 Basin volumes

We now apply the parallel tempering method described in the previous chapter to measure basin volumes using different minimizers. Recall that the procedure requires checking basin membership at each MCMC step—approximately 10^6 minimizations per basin [31, 142]. Given the accuracy differences observed in Figure 3.4, we expect this method to be significantly affected by optimizer inaccuracy. We test this hypothesis at $\phi = 0.9$ across system sizes that remain amenable to CVODE.

The results for the intensive free energy $F_0/N = -\ln(v_\alpha)/N$ are shown in Figure 6.1. While

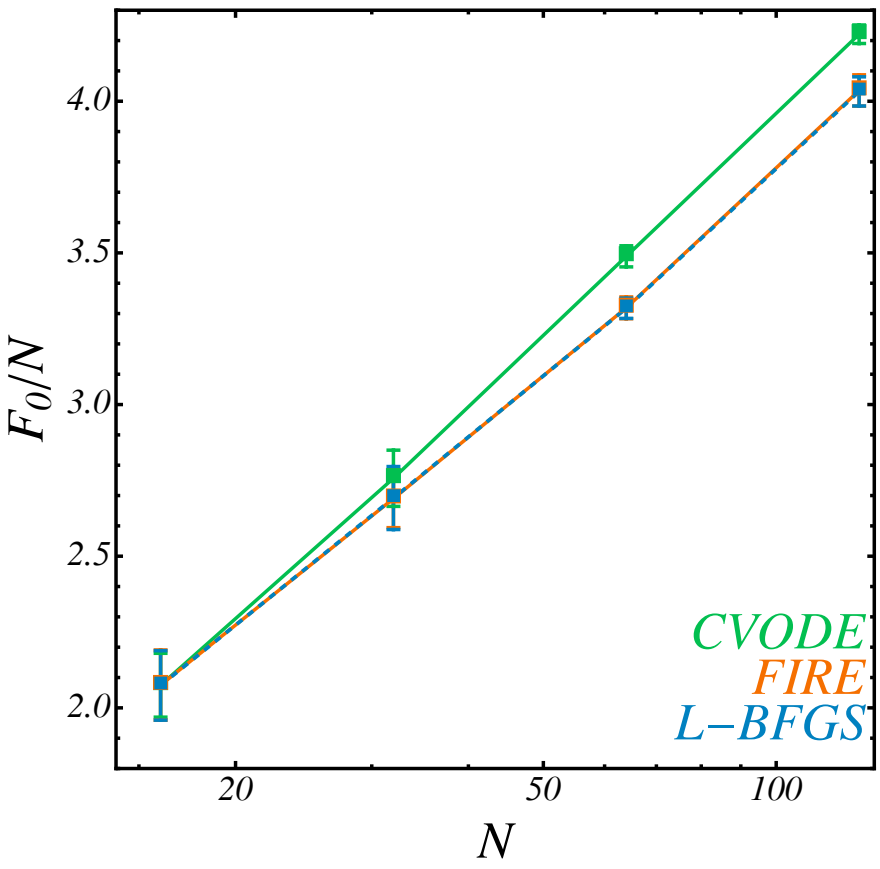

Figure 6.1: Free energies. Intensive free energies F_0/N across methods, against N, each averaged over the same 5 basins for each N.

FIRE and L-BFGS yield near-indistinguishable values, there is a systematic bias between CVODE and the approximate optimizers. Worse, this bias grows with system size as FIRE and L-BFGS become increasingly inaccurate. Notably, the volumes measured by optimizers are *larger* than the true volumes—a counter-intuitive result we explain below.

6.3 Density of States

The volume overestimate by optimizers can be understood through the radial density of states (DOS), computed from the parallel tempering samples as described in Section 5.3.5. Figure 6.2 compares the DOS for example basins using CVODE and FIRE. We plot the DOS against the

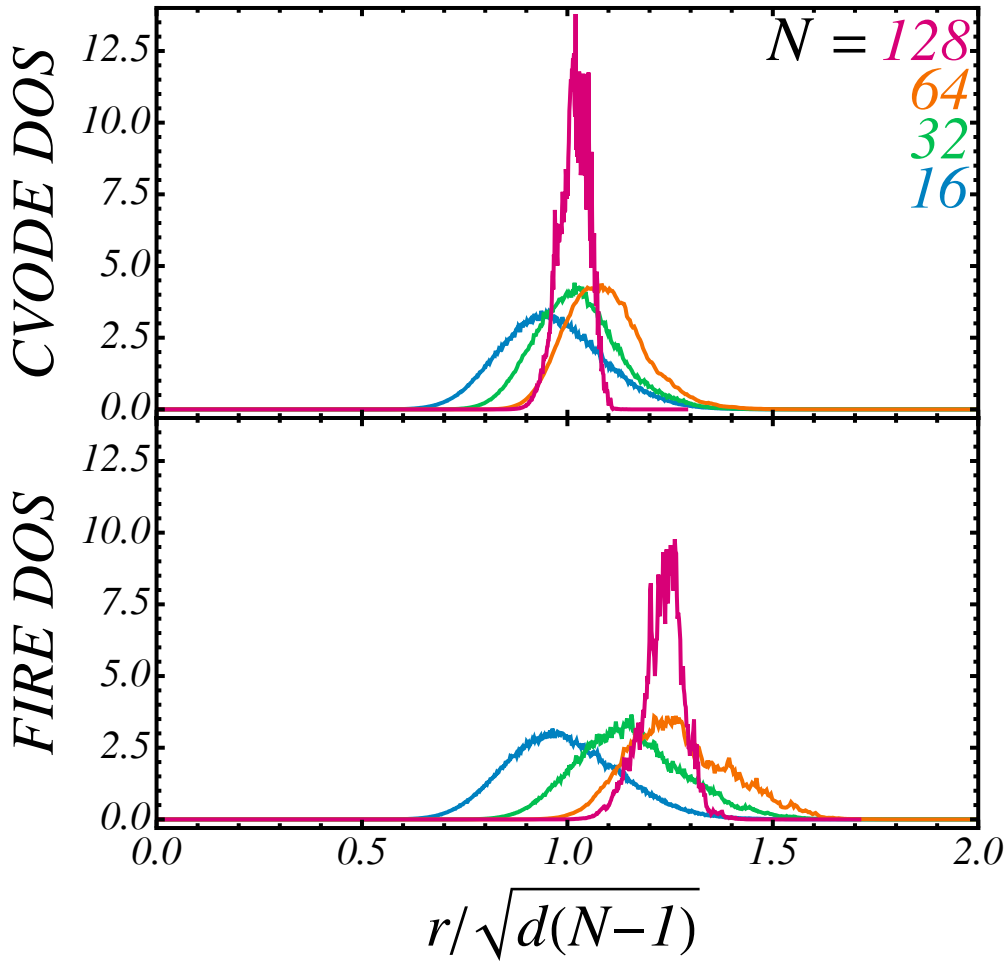

Figure 6.2: Density of states. Example densities of states (DOS) for one basin per N, comparing CVODE (left) and FIRE (right).

rescaled radial distance $r/\sqrt{(N-1)d}$, which keeps the length of a long diagonal of a unit cube constant across dimensions. The CVODE DOS are all maximal around 1, while the FIRE distributions systematically shift to higher values—FIRE finds samples lying further away from the minimum.

6.4 Accuracy vs Distance

The explanation for the shifted DOS lies in how optimizer accuracy depends on distance from the minimum. Figure 6.3 shows the accuracy of FIRE on samples from the same basins as Figure 6.2,

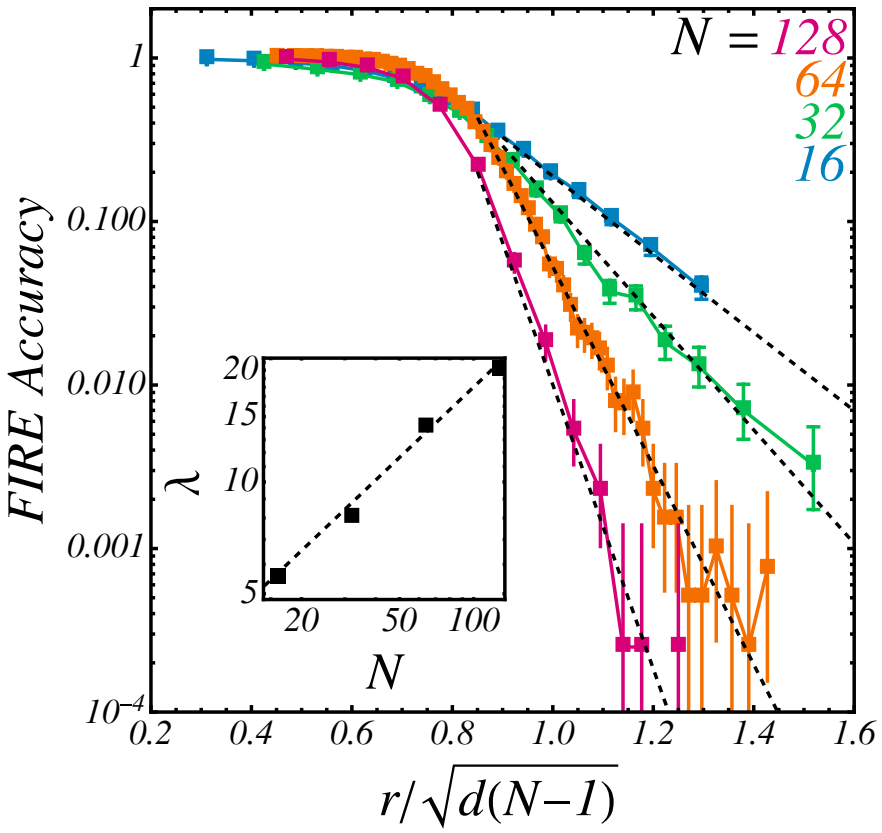

Figure 6.3: Accuracy vs distance. Averaged accuracy over $O(10^5)$ FIRE samples used, against their distance to the minimum, in semi-log scale. Dashed lines are exponential fits $y = C\exp(-\lambda x)$ of each curve. Inset: best decay rate λ against N in log-log, with a dashed power-law $\lambda \sim N^{0.6}$.

plotted against rescaled radial distance. The accuracy plummets after $r/\sqrt{(N-1)d} \approx 0.7$, with an exponential decay that becomes steeper with increasing dimensionality (inset).

This reveals the mechanism: FIRE wrongly tags points outside the basin as belonging to it, likely because inertia facilitates ridge crossing. In high dimensions, a slightly larger spherical shell contributes an enormous volume, so these erroneous points lead to systematic volume overestimates. This provides quantitative evidence that optimizers bias the sampling of minima, consistent with the biased estimates of energies and ϕ_J discussed in Chapter 3.

6.5 Basin Shape Comparison

To connect accuracy decay to basin geometry, Figure 6.4 shows the CVODE DOS divided by that of a hyperball in $N(d-1)$ dimensions [30, 142]. This ratio measures how much sparser than a ball the basin becomes as r grows—where it drops, the basin narrows into "tentacles."

The accuracy of FIRE (Figure 6.3) begins falling at distances slightly larger than where this ratio starts dropping, meaning inaccuracy sets in precisely when entering the tentacles [1]. This also explains why the disagreement between CVODE and optimizers grows with N: in higher dimensions, the volume of the largest inscribed ball becomes vanishingly small [142], so most of the basin lies in the tentacles where optimizers fail.

These discrepancies call into question the scope of results on basin geometry obtained with FIRE [30, 31, 123].

6.6 Half-Survival Radius

The density of states and accuracy measurements presented above characterize how basins deviate from hyperballs and where optimizers fail. However, they do not directly measure the local *width* of the basin at different radial distances. The half-survival radius $R_{1/2}$ provides precisely this information: it quantifies how far one can perturb a configuration before it escapes to a different minimum with 50% probability.

6.6.1 Methods

We measure $R_{1/2}$ using a procedure that extends the survival probability method described in Section 4.4.1. The key difference is that we sample perturbations from points distributed throughout

[1] The decay exponents in Figure 6.4 (see inset) are much larger than those in Figure 6.3. Thus, interpreting Figure 6.4 as the success rate of naïve Monte Carlo on a sphere [142], the FIRE basin, while inaccurate, remains much more correlated with the true basin than uniform sampling.

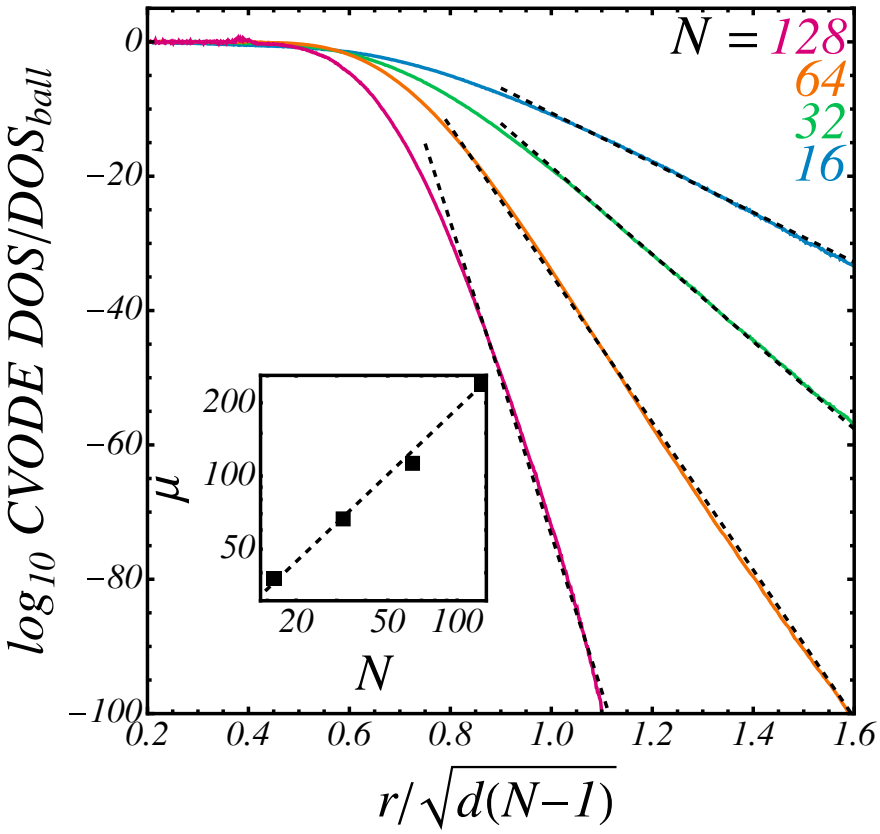

Figure 6.4: Basin shape. $\log_{10}$ of the ratio between the CVODE DOS and that of a hyperball in $N(d-1)$ dimensions. Dashed lines are exponential fits. Inset: decay rate μ against N in log-log, with a dashed power-law $\mu \sim N^{0.9}$.

the basin interior rather than from a single reference point, allowing us to track how basin width varies with distance from the minimum.

Binning parallel tempering samples by distance. Starting from the parallel tempering trajectory samples, we compute the radial distance of each sample $\mathbf{x}$ from the basin minimum $\mathbf{x}_0$:

$$r = \|\mathbf{x} - \mathbf{x}_0\|. \tag{6.1}$$

Samples are binned by distance into intervals of width $\Delta r = 0.1$ in configuration space units. From each bin, we select 10 representative center points for the subsequent sphere sampling.

Survival probability at multiple shell radii. For each center point $\mathbf{c}$ at distance r from the minimum, we compute the survival probability $S(R)$ at 15 shell radii spanning multiple orders of magnitude, from $R \approx 0.01$ to $R \approx 2$ in configuration space units. For each distance bin, we average the survival curves across all 10 center points:

$$\bar{S}(R) = \frac{1}{10} \sum_{k=1}^{10} S_k(R). \tag{6.2}$$

Interpolation to find $R_{1/2}$. Given the averaged measurements $(R_j, \bar{S}_j)$, we find $R_{1/2}$ where $\bar{S}(R_{1/2}) = 0.5$ by linear interpolation in log-probability space. Survival probability typically decays exponentially or stretched-exponentially with radius:

$$S(R) \sim \exp\left(-\lambda R^{\beta}\right), \tag{6.3}$$

so in log-probability space the relationship is nearly linear, improving interpolation accuracy. We build an interpolating function mapping $\log S \to R$ and evaluate at $\log(0.5) \approx -0.693$.

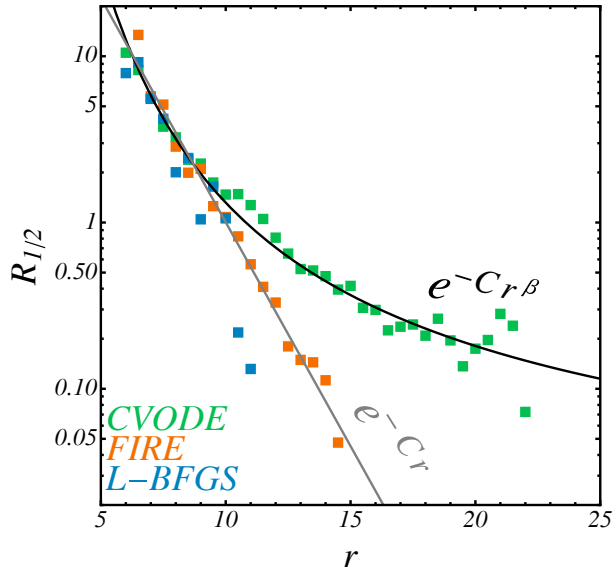

Figure 6.5: Half-survival radius. Half-survival radius $R_{1/2}$ inside a single basin against distance r to its minimum for $N = 128$. Lines indicate proposed fits: exponential (gray) for FIRE and L-BFGS, stretched-exponential (black) for CVODE.

6.6.2 Results

Results for $N = 128$ particles at packing fraction $\phi = 0.9$ are shown in Figure 6.5. The measurement involves 10 center points per distance bin, 1000 samples per hypersphere, and 15 shell radii—approximately 10^6 total samples per minimization method.

The half-survival radius provides a direct measure of sensitivity to perturbations: smaller $R_{1/2}$ means a configuration is more sensitive, requiring only a small displacement to escape the basin with appreciable probability. In this sense, $1/R_{1/2}$ quantifies the local sensitivity at each distance from the minimum.

The functional form of $R_{1/2}(r)$ differs qualitatively between methods. Optimizers (FIRE and L-BFGS) yield exponential decays:

$$R_{1/2}(r) \sim \exp(-\lambda r), \tag{6.4}$$

while CVODE reveals stretched exponential behavior:

$$R_{1/2}(r) \sim \exp\left(-\lambda r^{\beta}\right), \quad \beta < 1. \tag{6.5}$$

This distinction has geometric significance. For hypercubes, cross-sections perpendicular to a main diagonal decay exponentially with distance from the center [150]. The optimizer results are therefore consistent with a hypercube-like basin geometry. However, the stretched exponential observed with CVODE indicates that real basins have "thicker" tentacles than hypercubes—they maintain finite width further into their periphery than hypercubic analogies would predict. In terms of sensitivity, this means configurations in the tentacle regions are less sensitive to perturbations than hypercubic models suggest: they can tolerate larger displacements before escaping to a different basin.

This finding is consistent with observations that tentacles contribute significantly to basin volumes at large distances from the minimum in sphere packings [27, 30], Kuramoto models [21, 123], and neural networks [151]. Crucially, this qualitative difference in basin geometry is invisible to optimizer-based measurements, which systematically report the wrong functional form.

6.7 Discussion

The central finding of this chapter is that optimizer choice within the parallel tempering volume calculation introduces systematic bias. Comparing free energies obtained with CVODE against those from FIRE and L-BFGS reveals a discrepancy that widens as N increases. Counterintuitively, optimizers yield volumes that are too large—the opposite of what one might naively expect from a method that "misses" the correct minimum.

The resolution lies in understanding where optimizers fail. The radial density of states and accuracy-versus-distance measurements together paint a clear picture: FIRE maintains reason-

able accuracy near the minimum but deteriorates rapidly at larger distances. Because high-dimensional basins concentrate their volume in narrow peripheral regions rather than near the center, these distant points dominate the integral. When FIRE misassigns an outside point as inside, it effectively expands the basin's apparent reach, and the geometric amplification of volume with radius does the rest.

Beyond quantifying the bias, the half-survival analysis uncovers a qualitative difference in inferred basin shape. The functional form of how cross-sectional width shrinks with distance differs between methods: optimizers produce a simple exponential falloff, while CVODE reveals a slower stretched-exponential decline. The latter implies that real basins maintain finite width further into their periphery than hypercubic analogies would predict—a distinction invisible to optimizer-based measurements.

These findings have direct implications for the literature on basin volumes in jammed packings. Previous work [16–18, 30, 31, 123] employed optimizers for the basin membership checks required for volume estimation. In Chapter 4, we showed that qualitative changes in survival probability emerge near jamming that are seen with optimizers but not when solving the ODE. Notably, the verification of the Edwards hypothesis in two dimensions [31] relied on measurements extrapolated to the jamming point. It remains to be seen how those conclusions are affected when solving the steepest descent ODE.

The present calculations also present a new unchartered frontier Xu's original study included a steepest-descent verification, but only for $N = 8$ $d = 14$; Zhang et al. [21] computed volumes in Kuramoto systems in $84d$, where the potential is easier to evaluate where some basins span appreciable phase-space fractions with naive methods. The present measurements represent the first ODE-based volume calculations at the scale of 254 dimensions with basins that occupy negligible fractions of configuration space.

7 | Conclusions and Future directions

This thesis has analyzed the geometry and volume of basins of attraction in soft sphere packings. In doing so, it has demonstrated that commonly used numerical techniques for studying potential energy landscapes in physics and chemistry produce biased results that affect the conclusions drawn from them. While most of our measurements were conducted in the overjammed regime at high packing fraction, in cases where we systematically approached the jamming transition, we observed that optimizer accuracy degrades (Section 3.6.3), and survival probability results change both qualitatively and quantitatively (Section 4.4.3). It is therefore critical to re-verify results in the field that depend on fidelity to steepest descent pathways, particularly as we get closer to jamming.

7.1 Primary Contributions

The contributions of this thesis fall into three interconnected themes: numerical methodology for basin identification, the geometric structure of basins, and basin volume measurements.

Numerical methodology. We showed that commonly used optimizers used for energy minimization, FIRE and L-BFGS, produce erroneous basin assignments at an increasing rate as system size grows, reaching near-complete failure for systems with $N \gtrsim 64$ particles (Section 3.5). By moving aways from the false frame of basin identification as a minimization problem, back to an ODE integration problem, we identified CVODE as a practical solver capable of accurate steepest

descent integration up to $dN \sim 10^3$. These optimizer errors propagate into physical observables, biasing both the distribution of minimum energies and finite-size estimates of the jamming density (Section 3.4).

Basin geometry. Optimizer failures produce systematic biases in measured basin geometry that are both quantitative and qualitative; apparent fractal properties of basins can be attributed to optimizer errors rather than genuine features of the landscape. Box-counting dimensions of basin boundaries drop with CVODE (Section 4.3), quantifying the roughness visible in configuration-space slices. Survival probabilities exhibit stretched-exponential decay with CVODE, indicating basins with finite length scales, whereas optimizers produce power-law decays that would incorrectly suggest scale-free structure (Section 4.4.3). Intersection lengths follow log-normal distributions (Section 4.2.2), further confirming that basins possess well-defined characteristic scales. We also note that the survival probability method, while employed here primarily to distinguish between fractal and non-fractal basins, can serve as a direct probe of fractal dimension (Section 4.4.2). Because it relies on Monte Carlo sampling rather than the systematic boundary detection of Grebogi et al. [58, 59], its computational cost does not grow with configuration-space dimension, making it potentially useful for characterizing basin stability in high-dimensional systems. Geometric conclusions drawn from optimizer-based measurements in previous studies [29, 35, 36] may therefore warrant re-examination.

Basin volumes. We extended volume measurements to 254 dimensions using ODE-based basin membership checks (Section 6.2), a significant increase over previous ODE-verified calculations in packings, which reached 14 dimensions [16]. Volume calculations in dynamical systems have reached 86 dimensions [21], but these relied on naive sampling in regimes where some individual basins occupy substantial fractions of phase space and the potential is less expensive. A comparison with optimizer-based measurements revealed that optimizers systematically inflate volume estimates, with the discrepancy widening at larger system sizes. This inflation originates from

distance-dependent accuracy: optimizers perform adequately near the minimum but fail in the peripheral "tentacle" regions where high-dimensional volume concentrates (Section 6.4). Measuring how basin cross-sections narrow with radial distance, we found a stretched-exponential decay, indicating that these tentacles are thicker than those of hypercubes (Section 6.6).

7.2 Where can we go from here

7.2.1 Jamming and glasses

We have shown CVODE used correctly can reliably give inherent structures properly at system sizes researchers care about. We note the problem of not being able to converge to the correct inherent structure has been noted[79, 81] although attributed to chaoticity instead of numerical methodology. We give a tool that is able to do so without issue, so we can be assured of the correctness of results obtained. Some possible use cases are precise estimates of the jamming transition, proper calculations of configurational entropy.

7.2.2 Basin volume methodology

The computational cost of accurate basin volume calculations remains substantial: measuring the volume of a single basin correctly for $N = 128$ particles in 2D required over two weeks using 16 cores. Several avenues for improvement exist.

First, significant advances in parallel tempering methodology have emerged in recent years [152–154] that have yet to be applied to basin volume calculations.

Second, our implementation uses standard Metropolis sampling with random-walk proposals, which suffers from diffusive slowdown: in d dimensions, a random walker requires $O(d^2)$ steps to traverse the configuration space [155]. We note that our current approach discards the information contained in steepest descent trajectories. Each basin membership query integrates an

ODE from the proposed configuration to its minimum, tracing a path through the basin interior. This information is discarded because we only retain the binary outcome of whether the configuration belongs to the basin. Schemes that exploit this geometric information, perhaps by using gradient information to guide proposals or by caching trajectory data to inform future queries, remain unexplored.

7.2.2.1 Caveats

While the free-energy method to calculate volumes presented in this thesis is powerful, it involved several moving parts, from optimizing potential evaluations in C++ via template metaprogramming to managing simulations where computing the volume of a single basin can take weeks. Other approaches should therefore be considered first when applicable. For systems with basins that dominate phase space, naive Monte Carlo sampling may suffice, as demonstrated for the Kuramoto model [20, 21]. In other cases, analytical solutions are available, for instance, the basins of the random Lorentz gas [156], the smaller basins of the Kuramoto model [66] and those of certain high-dimensional axion potentials in string theory [54]. However we note that research is limited by the tools we have on hand. The specific code we use is extremely specialized to jammed packings. Making an extensible modular implementation that is not fragile that everybody can use is critical not only for broader use, but to make it easy for other researchers to contribute ideas to making volume calculations faster. These could be useful to measure high-dimensional volumes that can be mapped to a basin of a flow.

7.2.3 Other fields

We note that with basin volumes we have only touched the surface of what is possible to do. We talk about a few problems in other fields we can potentially tackle with the same approach. We also note that there is interest in calculating basin volumes in the dynamical systems community using the methods that we use in this thesis [157]. There is work on calculating volumes in the

machine learning context to understand generalization, although currently published work does a naive linear estimate based on Hessian eigenvalues [5]. Although there are many use cases [20, 142] we discuss two directions in detail, particularly in the context of stability.

Ecosystem dynamics In ecology, Lotka–Volterra equations model species interactions through coupled ordinary differential equations, where the number of species s determines the system's dimensionality. Analytical results are largely confined to two regimes: small communities ($s \leq 3$) [158] where explicit solutions remain tractable, and the mean-field limit ($S \to \infty$) [74]. Between these extremes lies the a more realistic regime of 10^2–10^3 [49, 159] species, where numerical approaches may be useful. Basin volume methods could be useful to characterize ecosystem stability in this intermediate regime and understand how stability correlates with equilibrum properties such as species diversity. Notably, Lotka–Volterra dynamics with symmetric competitive interactions and no noise maps directly onto gradient descent on a glassy energy landscape [160], making the methods developed in this thesis directly applicable.

The human heart The normal human heart exhibits two stable attractors [161]: the normal sinus rhythm and ventricular fibrillation, a lethal arrhythmia that represents the most frequent cause of sudden death[162]. As Strogatz [20] has noted, characterizing the stability of both basins and understanding what controls their relative stability would be valuable beyond purely phenomenological descriptions. Recent advances in modeling the human heart to clinical accuracy [163] may make characterizing the stability of these basins tractable.

7.3 Concluding remarks

In summary, I have analyzed the geometry and volumes of basins in soft sphere packings and, in the process, shown that numerical methods commonly used since the 1980s to study potential energy landscapes can lead to misleading results for soft sphere packings. There are two broad

goals this thesis aims to achieve beyond presenting results. First, I believe Chapter 2 is critical to understand before commenting on landscape properties from optimization dynamics; the intended audience is researchers who study potential energy landscapes in physics and chemistry. Second, while I tried to explicitly motivate the tools I use in this thesis through the idea of global stability in dynamical systems, the most I have done in this particular context is introduce two measures and illustrate where the assumed chaoticity of the energy landscape of glasses comes from in Chapter 4 and show how correct basin volumes can be calculated at large d in Chapter 6. There are many different measures introduced in the context of the global stability/resilience of an attractor, and a general broadly applicable theory of stability is still lacking [164]. Whether the stability of a basin per se can be connected to statistical mechanical quantities of interest is intriguing. I will throw that speculation up in the air. There is much to be explored, and fun to be had.

A | Convergence Criterion

Throughout this study, a convergence criterion is needed to decide whether an optimizer has run for sufficiently long that it has approached its infinite-time attractor. Far from jamming (here, empirically, for packing fractions $\phi \gtrsim 0.86$), the Hessian eigenvalues are relatively large throughout the landscape, so that we use a simple gradient-based convergence criterion: we end ODE solving and optimizations when $\|\mathbf{g}\| < 10^{-10}$. However, as the system gets closer to jamming, the Hessian eigenvalues may become small, as the energy landscape flattens. As a result, it is possible to have very small forces on particles far from the minimum, so that a purely gradient-based criterion can trigger premature termination of the minimization. To circumvent this issue, we employ a Newton-step-based convergence criterion that uses a second-order estimate of the distance to the minimum,

$$\delta \mathbf{X}_{\text{min}} \approx \left(\overline{\overline{\boldsymbol{H}}}(\boldsymbol{X}) + \lambda(\boldsymbol{X})\,\overline{\overline{\boldsymbol{I}}}\right)^{-1} \mathbf{g}(\boldsymbol{X}), \tag{A.1}$$

where $\mathbf{g}$ is the gradient, $\overline{\overline{\boldsymbol{H}}}$ is the Hessian, $\overline{\overline{\boldsymbol{I}}}$ is the identity matrix, and λ is chosen to ensure that $H + \lambda I$ remains well-conditioned. In practice, $\lambda = \max\left(2|\lambda_{\text{min}}|,\ \gamma\,\lambda_{\text{avg}}\right)$, where λ_{min} and λ_{avg} are the minimum and average Hessian eigenvalues at $\boldsymbol{X}$, respectively, and $\gamma = 0.1$. We consider the dynamics converged when

$$\|\delta \mathbf{X}_{\text{min}}\| < 10^{-5}. \tag{A.2}$$

This threshold is three orders of magnitude smaller than the maximal distance between non rattler particles used to declare two minima as different, 10^{-2}. The reason for using a smaller value is

that Newton steps are only a heuristic and may lower accuracy if one does not make its range smaller than every other scale of the landscape near minima. We check that this value does not introduce errors compared to a very small gradient criterion in the landscape of Hertzian disks, but comes at a lower computational cost.

B | Minima Matching Procedure

It is crucial to be able to accurately determine whether two points obtained as large-time limits of numerical dynamics map to one and the same minimum of the energy landscape. In the context of jammed packings, this procedure is complicated by two factors: (i) the presence of rattlers—particles that can move over a finite-measure set of positions without changing the overall energy of the system—and (ii) the translational invariance of the system under periodic boundary conditions. We address these complications by first identifying all rattlers, then comparing only the positions of non-rattler (backbone) particles after accounting for translational symmetry.

B.1 Rattler Identification

A rattler is a particle that is not mechanically constrained by its neighbors and can therefore move continuously without changing the energy of the packing. Because rattler positions are not uniquely determined by the minimum, their coordinates must be excluded when comparing two configurations.

B.1.1 Geometric Criterion

A particle i is mechanically constrained (not a rattler) if and only if it satisfies two conditions:

1. It has at least $z_{\min} = 3$ contacts with other non-rattler particles.

2. Its center is contained within the convex hull of its contact vectors.

The first condition ensures that particle i has enough constraints to potentially be fixed in place. The second condition ensures that the constraints actually prevent motion: a particle with three or more contacts can still be a rattler if all contacts lie on one side, allowing it to roll away from them.

To formalize the second condition, let $\mathcal{N}_i$ denote the set of particles in contact with particle i. For each $j \in \mathcal{N}_i$, define the contact vector

$$\mathbf{r}_{ij} = \mathbf{x}_j - \mathbf{x}_i, \tag{B.1}$$

pointing from particle i toward its neighbor j. Particle i is mechanically constrained if and only if the origin lies within the convex hull of these contact vectors:

$$\mathbf{0} \in \text{ConvexHull}\left(\{\mathbf{r}_{ij}\}_{j\in\mathcal{N}_i}\right). \tag{B.2}$$

Geometrically, this means that the particle cannot move in any direction without approaching at least one of its neighbors, and is therefore trapped.

B.1.2 Convex Hull Test in Two Dimensions

To test whether the origin lies inside the convex hull of a set of 2D vectors, we use a winding algorithm:

1. Sort the contact vectors $\{\mathbf{r}_{ij}\}$ by polar angle $\theta_{ij} = \arctan\left(r_{ij,y}/r_{ij,x}\right)$.

2. Traverse the sorted vertices counter-clockwise.

3. For each pair of consecutive vertices $\mathbf{r}_{ij}$ and $\mathbf{r}_{ik}$ (where k follows j in the sorted order), check whether the origin lies to the left of the directed edge from $\mathbf{r}_{ij}$ to $\mathbf{r}_{ik}$. This is determined by

the sign of the cross product:

$$\mathbf{r}_{ij} \times \mathbf{r}_{ik} = r_{ij,x}\, r_{ik,y} - r_{ik,x}\, r_{ij,y}. \tag{B.3}$$

4. The origin is inside the convex hull if and only if it lies to the left of (or on) every edge, *i.e.*, if the cross product is non-negative for all consecutive pairs.

B.1.3 Iterative Identification

The identification of rattlers must be performed iteratively because removing one rattler can cause previously stable particles to become rattlers themselves. The algorithm proceeds as follows:

1. Compute the neighbor list for each particle, identifying all pairs (i, j) that are in contact.
2. Initialize all particles as potentially stable.
3. Iterate until no changes occur:
   (a) For each particle i still marked as stable:
      - If particle i has fewer than 3 contacts with other stable particles, mark it as a rattler.
      - Otherwise, if the origin does not lie within the convex hull of its contact vectors, mark it as a rattler.
   (b) If particle i is newly marked as a rattler, remove it from the neighbor lists of all its former contacts and add those contacts to a queue to be rechecked.
4. Return the final classification of each particle as either rattler or backbone.

B.1.4 Jamming Stability Check

After identifying all rattlers, we verify that the remaining backbone forms a mechanically stable structure. For a 2D packing to be jammed, the number of contacts N_c among backbone particles must satisfy the isostatic condition [109]:

$$N_c \geq 2(N_{\text{backbone}} - 1) + 1 = 2N_{\text{backbone}} - 1, \tag{B.4}$$

where N_{backbone} is the number of non-rattler particles. This condition ensures that there are enough constraints to fix all degrees of freedom (minus overall translations). If this condition is not satisfied, the configuration is classified as a fluid state rather than a jammed packing.

B.2 Structure Comparison

Given two configurations that have both been identified as jammed (not fluid), we must determine whether they represent the same minimum. This comparison must account for both the presence of rattlers (whose positions are arbitrary) and translational invariance under periodic boundary conditions.

B.2.1 Pre-checks

Before performing a detailed coordinate comparison, we apply two quick checks:

1. If the two configurations have different numbers of rattlers, they correspond to different minima.

2. If either configuration fails the jamming stability criterion, it is classified as fluid and not compared further.

B.2.2 Translational Alignment

Due to translational invariance, two configurations representing the same minimum may be shifted relative to each other. To account for this, we align the configurations by superimposing a reference particle. Specifically:

1. Select the first non-rattler particle (by index) as the reference particle. Since our system uses a fixed set of particle radii, particle identity is well-defined.

2. Compute the drift between the reference particle positions:

$$\mathbf{d} = \mathbf{x}_{\text{ref}}^{(B)} - \mathbf{x}_{\text{ref}}^{(A)}. \tag{B.5}$$

3. Shift all coordinates of configuration B:

$$\tilde{\mathbf{x}}_i^{(B)} = \mathbf{x}_i^{(B)} - \mathbf{d}. \tag{B.6}$$

B.2.3 Distance Computation with Periodic Boundaries

After alignment, we compute the distance between corresponding non-rattler particles, accounting for periodic boundary conditions. For each non-rattler particle i, the displacement vector is:

$$\boldsymbol{\Delta}_i = \left(\mathbf{x}_i^{(A)} - \tilde{\mathbf{x}}_i^{(B)}\right) \mod L, \tag{B.7}$$

where L is the box length. To obtain the minimum-image displacement, we apply:

$$\Delta_{i,\alpha} \leftarrow \begin{cases} \Delta_{i,\alpha} & \text{if } |\Delta_{i,\alpha}| \le L/2, \\ \Delta_{i,\alpha} - L\,\text{sign}(\Delta_{i,\alpha}) & \text{otherwise,} \end{cases} \tag{B.8}$$

for each Cartesian component α. The distance for particle i is then $d_i = \|\Delta_i\|$.

B.2.4 Matching Criterion

Two configurations are declared to represent the same minimum if the maximum displacement across all non-rattler particles is below a threshold:

$$\max_{i \in \text{backbone}} d_i < d_{\text{tol}}. \tag{B.9}$$

We use $d_{\text{tol}} = 10^{-2}$ throughout this work. This threshold is three orders of magnitude larger than the convergence criterion (see Appendix A), ensuring that numerical noise in the minimization does not cause false negatives. In practice, for packing fractions $\phi > 0.85$, configurations that match have maximum displacements smaller than 10^{-4} when using a gradient tolerance of 10^{-10} in the convergence criterion. Our choice of d_{tol} is the largest value that does not lead to appreciable changes in the identified landscape structure at any value of (N, ϕ).

C | Solver Parameters

C.0.1 Choices of solver parameters

In this section, we provide complete lists of parameter for all solvers used in the main text—namely, CVODE, FIRE, L-BFGS, and Gradient Descent. Whenever relevant, we also provide a rationale for these choices.

C.0.1.1 CVODE Parameters

Many adaptive ODE solvers, including CVODE, adjust their step size by imposing an upper bound on the estimated local error for each step. As noted in Sec. **??**, this bound is specified by two tolerance settings: relative error (`rtol`) and absolute error (`atol`). However, constraining this local error estimate does not guarantee that the global error will adhere to the same bounds, particularly in systems where small errors can lead to diverging trajectories. To address this limitation, we set values for tolerances based on an analysis of the accuracy of the mapping $\boldsymbol{X}_0 \mapsto \boldsymbol{X}_\infty$ between random initial points and their corresponding final minima.

To assess this accuracy, one needs a reference (a ground truth). Relying on the fact that CVODE is an ODE solver, and thus asymptotically converges to the true steepest-descent trajectories at vanishing `rtol`, we expect minimal error when `rtol` is small enough that the mapping remains unchanged over large random sets of points as we decrease `rtol`. Across all our simulation parameters, this is achieved when `rtol` is set to 10^{-14}.

Using this reference, we assess performance at larger tolerances by running CVODE from approximately 10000 random initial points at each `rtol` value. We define an accuracy for each `rtol` as the fraction of initial points that converge to the same minimum as when `rtol` $= 10^{-14}$. Results are shown in Fig. C.1, where we plot accuracies and computation times against `rtol` at all system sizes for which CVODE is considered in the paper. We also report performance based on whether we switch on an iterative Newton-Krylov scheme [165] provided within CVODE [120, 166] to solve an implicit time-stepping equation instead of using the dense Hessian. For each N, we highlight the `rtol` and choice of scheme (iterative or dense) corresponding to the fastest time with $> 98\%$ accuracy at identifying basins accurately. This `rtol` is the value we use in practice.

We summarize our choices of `rtol` and scheme for each N in Table C.1, where we also give values for accuracies larger than 95%. Note that the optimal time to attain these accuracies involves switching on the iterative scheme at sizes $N > 64$, which leads to a slight change of trend of time scalings at the junction $N = 64$.

System Size	`rtol` (acc $\gtrapprox$ 95)	`rtol` (acc > 98)	Iterative
8	10^{-4}	10^{-7}	False
16	10^{-5}	10^{-7}	False
32	10^{-5}	10^{-7}	False
64	10^{-6}	10^{-7}	False
128	10^{-7}	10^{-8}	True
256	10^{-8}	10^{-9}	True
512	10^{-9}	10^{-10}	True
1024	10^{-10}	10^{-11}	True

Table C.1: Tolerances for CVODE. Table of CVODE tolerances we choose. For small systems, while a higher `rtol` is optimal for performance, the resulting numerical noise can prevent the gradient from satisfying the 10^{-10} convergence threshold close to a minimum. Here iterative means that an iterative Newton-Krylov scheme is used to solve the Newton's equation instead of using the dense Hessian.

Also note that, while the matrix structure for the Hessian is sparse, we are not aware of already available methods that may take advantage of this property, because the sparsity structure constantly changes as particles start or stop interacting through the minimization process. Finally, from the values in Tab. C.1, it is interesting to notice that `rtol` seemingly scales with N

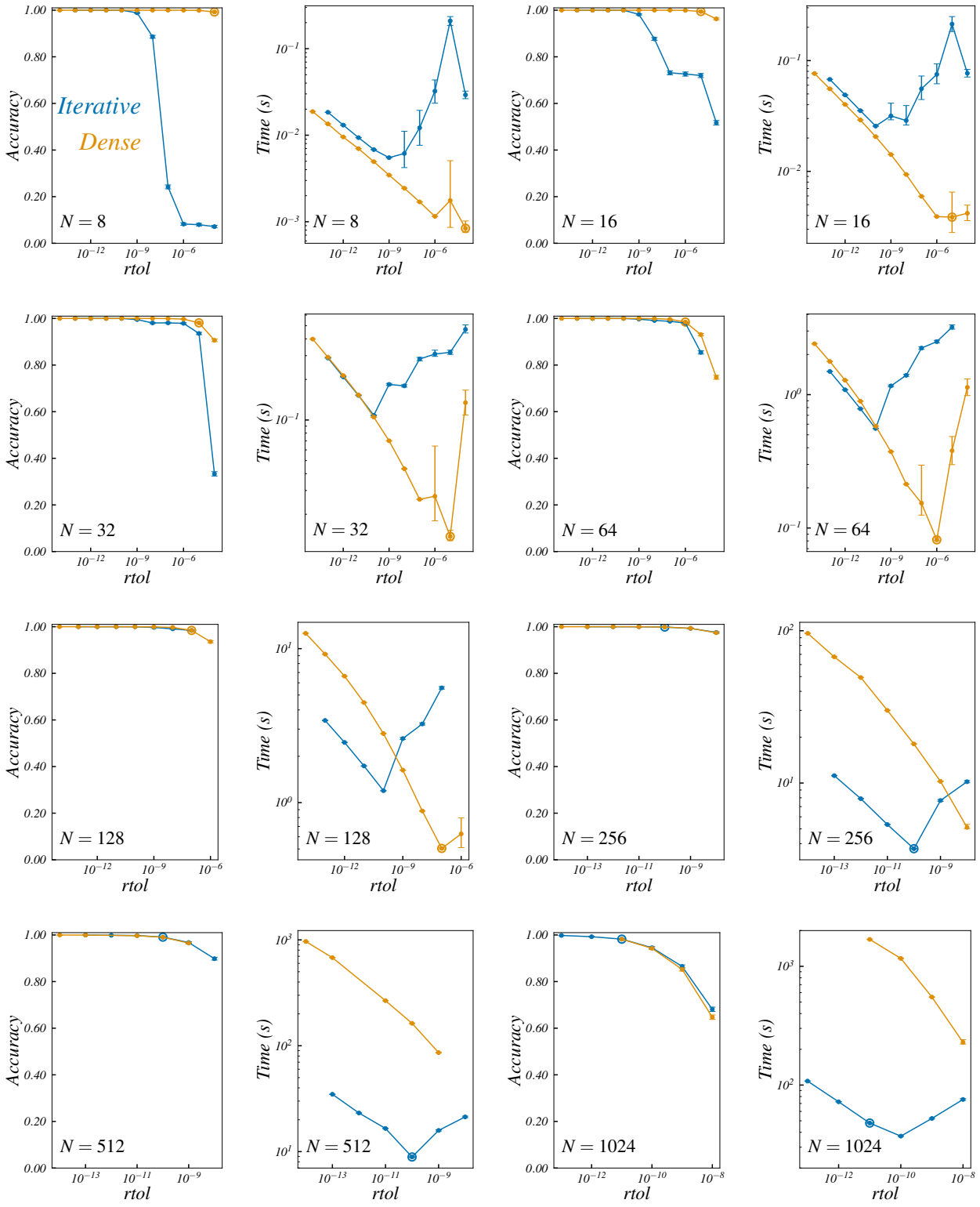

Figure C.1: CVODE Accuracy and Time vs Tolerance. For each N, we vary `rtol` and report accuracy and time, averaged across 10^4 random initial conditions. Error bars represent Clopper-Pearson confidence intervals for accuracy and standard error for time. Highlighted points correspond to points used in the main text, corresponding to > 98% accuracy. The point at the lowest rtol is taken to have accuracy 1.0

with a power close to N^{-3}, that is reminiscent of the timestep scaling with N reported in Ref. [80] – although we emphasize that `rtol` is not simply a timestep.

C.0.1.2 FIRE, LBFGS, and Gradient Descent parameters

We employ the FIRE and L-BFGS algorithms as implemented in the pele package [167], which adhere to the recommendations by Ref. [28] to generate accurate basins. To prevent excessively large step sizes, we cap the maximum step at

$$\Delta_{\max} = \frac{\mu_s}{10} \tag{C.1}$$

with μ_s the mean radius of the smaller particles in our bidisperse distribution of radii. In the basin volume calculations, however, we use a different bound to be consistent with parameter choices in Ref. [31],

$$\Delta^{\mathrm{BV}}_{\max} = \frac{R_{min}}{4} \tag{C.2}$$

with R_{min} the smallest radius. In practice, for our choice of distributions, we typically have $\Delta^{\mathrm{BV}}_{\max} \approx 2\Delta_{\max}$.

In our FIRE implementation, we incorporate an additional safeguard: the minimizer is prevented from taking uphill steps by halting and resetting momentum, as in Ref. [31]. The list of FIRE parameters for the pele implementation used in the paper are given in Tab. C.2. `finc` determines the factor to be applied to dt after `Nmin` steps, dt' = `finc` dt, if the velocity is downhill (i.e pointing in the direction of the gradient). If we encounter an uphill step, then we reduce dt by multiplying it by `fdec`, dt' = `fdec` dt. The maximum timestep,`dtmax`, is set to 1.

Our L-BFGS parameters are listed in Tab. C.3. Here, `M` controls how well the algorithm approximates the Hessian, `maxErise` determines how much the energy can rise due to the method, (in this case 0), `H0` determines the initial approximate inverse Hessian. We use the default parameter

Parameter	Value
dtstart	0.1
dtmax	1
maxstep	0.5
Nmin	5
finc	1.1
fdec	0.5
fa	0.99
astart	0.1
stepback	True
tol	10^{-10}

Table C.2: FIRE algorithm parameters

M = 1 as recommended by Ref. [28].

Parameter	Value
tol	10^{-10}
M	1
maxstep	0.1
maxErise	10^{-10}
H0	0.1

Table C.3: L-BFGS algorithm parameters

Finally, in the variant of Gradient Descent adapted from Ref. [79] and described in Sec. **??**, we use the parameters given in Tab. C.4. Here ϵ is a bound on the cosine similarity between successive gradients. $\mathrm{d}t_{\text{initial}}$ is the starting value of the timestep $\mathrm{d}t$. If $n_{\text{backtrack}}$ successive steps satisfy the cosine similarity condition then we modify the timestep $\mathrm{d}t = n_{\text{backtrack}}\mathrm{d}t$, and if the condition is violated, we reduce the timestep by doing $\mathrm{d}t = \mathrm{d}t/n_{\text{backtrack}}$.

Parameter	Value
ϵ	10^{-2}
dt_{initial}	10^{-5}
$n_{\text{backtrack}}$	5

Table C.4: Gradient Descent algorithm parameters

D | Video Description

We show a video slice of basins of attraction in <link>. attached to the article is obtained as a succession of $2d$ slices of the energy landscape similar to the ones shown in Figs. 1 and 2 of the main text, for a system of $N = 16$ Hertzian disks at $\phi = 0.9$. To obtain motion along a closed trajectory, and thus a looping video, we first define a random $2d$ slice in the usual way, see Sec. 4.2. That slice is defined by a random point $\boldsymbol{X}$ in configuration space, and two orthogonal unit vectors $\hat{\boldsymbol{n}}_1$ and $\hat{\boldsymbol{n}}_2$. We then sample a third and fourth random direction $\hat{\boldsymbol{e}}_3$ and $\hat{\boldsymbol{e}}_4$, and use them to construct the unit vectors $\hat{\boldsymbol{n}}_3$ and $\hat{\boldsymbol{n}}_4$ such that $\hat{\boldsymbol{n}}_i \cdot \hat{\boldsymbol{n}}_j = \delta_{ij}$ for $(i, j) \in \{1, 2, 3, 4\}^2$. Using the two vectors perpendicular to the starting slice, we define a circular trajectory in configuration space and generate regularly spaced slices along it, such that the successive slices are always perpendicular to the circle and are spaced by a distance equal to the size of a pixel. We then match minima across all slices so that any one basin is encoded by the same color across slices. The resulting collection of slices forms a continuous-looking video that hints at a smooth $3d$ structure of basins.

Bibliography

[1] José Nelson Onuchic, Zaida Luthey-Schulten, and Peter G. Wolynes. "Theory of Protein Folding: The Energy Landscape Perspective". In: *Annual Review of Physical Chemistry* 48.1 (1997), pp. 545–600.

[2] Frank H. Stillinger. *Energy Landscapes, Inherent Structures, and Condensed-Matter Phenomena*. Princeton University Press, 2015. ISBN: 9780691166803.

[3] Matteo Aldeghi et al. "Roughness of Molecular Property Landscapes and Its Impact on Modellability". In: *Journal of Chemical Information and Modeling* 62 (2022), pp. 4660–4671.

[4] Cyrus Levinthal. "How to Fold Graciously". In: *Mössbauer Spectroscopy in Biological Systems: Proceedings of a Meeting held at Allerton House, Monticello, Illinois*. University of Illinois Press, 1969, pp. 22–24.

[5] Lei Wu, Zhanxing Zhu, and Weinan E. "Towards Understanding Generalization of Deep Learning: Perspective of Loss Landscapes". In: *ICML 2017 Workshop on Principled Approaches to Deep Learning*. Sydney, 2017.

[6] Hao Li et al. "Visualizing the loss landscape of neural nets". In: *Advances in Neural Information Processing Systems* NeurIPS 2018 (2018), pp. 6389–6399.

[7] Carlo Baldassi, Fabrizio Pittorino, and Riccardo Zecchina. "Shaping the learning landscape in neural networks around wide flat minima". In: *Proceedings of the National Academy of Sciences of the United States of America* 117.1 (2020), pp. 161–170.

[8] Fabrizio Pittorino et al. “Entropic gradient descent algorithms and wide flat minima”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2021.12 (2021), p. 124015.

[9] Yu Feng and Yuhai Tu. “The inverse variance-flatness relation in stochastic gradient descent is critical for finding flat minima”. In: *Proceedings of the National Academy of Sciences of the United States of America* 118.9 (2021), e2015617118.

[10] Carlo Baldassi et al. “Typical and atypical solutions in nonconvex neural networks with discrete and continuous weights”. In: *Physical Review E* 108.2 (2023), p. 24310.

[11] Alison Pouplin et al. “On the curvature of the loss landscape”. In: *Arxiv Preprint* (2023), p. 2307.04719.

[12] Nicholas Lourie, He He, and Kyunghyun Cho. “Hyperparameter Loss Surfaces Are Simple Near their Optima”. In: *Second Conference on Language Modelling*. Montréal, 2025.

[13] Chiara Cammarota. *Schematic rugged energy landscape*. Simons Collaboration on Cracking the Glass Problem. Figure credit: Chiara Cammarota.

[14] Werner Krauth and Marc Mézard. “Storage capacity of memory networks with binary couplings”. In: *Journal de Physique* 50.20 (1989), pp. 3057–3066. ISSN: 0302-0738. DOI: 10.1051/jphys:0198900500200305700.

[15] Werner Krauth, Marc Mézard, and Jean-Pierre Nadal. “Basins of attraction in a perceptron-like neural network”. In: *Complex Systems* 2.4 (1988), pp. 387–408.

[16] Ning Xu, Daan Frenkel, and Andrea J. Liu. “Direct determination of the size of basins of attraction of jammed solids”. In: *Physical Review Letters* 106.24 (2011), p. 245502.

[17] Daniel Asenjo, Fabien Paillusson, and Daan Frenkel. “Numerical calculation of granular entropy”. In: *Physical Review Letters* 112.9 (2014), p. 098002.

[18] Stefano Martiniani et al. “Turning intractable counting into sampling: Computing the configurational entropy of three-dimensional jammed packings”. In: *Physical Review E* 93.1 (2016), p. 012906.

[19] Peter J. Menck et al. “How basin stability complements the linear-stability paradigm”. In: *Nature Physics* 9.2 (2013), pp. 89–92.

[20] Daniel A. Wiley, Steven H. Strogatz, and Michelle Girvan. “The size of the sync basin”. In: *Chaos* 16.1 (2006), p. 015103. ISSN: 10541500.

[21] Yuanzhao Zhang and Steven H. Strogatz. “Basins with tentacles”. In: *Physical Review Letters* 127.19 (2021), p. 194101.

[22] Yuanzhao Zhang et al. “Deeper but smaller: Higher-order interactions increase linear stability but shrink basins”. In: *Science Advances* 10.October (2024), eado8049.

[23] David J Wales. “Basins of Attraction for Stationary Points on a Potential-Energy Surface”. In: *Journal of the Chemical Society: Faraday Transactions* 88.5 (1992), pp. 653–657.

[24] Andreas Heuer. “Properties of a glass-forming system as derived from its potential energy landscape”. In: *Physical Review Letters* 78.21 (1997), pp. 4051–4054.

[25] A. Saksaengwijit, B. Doliwa, and A. Heuer. “Description of the dynamics in complex energy landscapes via metabasins: A simple model study”. In: *Journal of Physics Condensed Matter* 15.11 (2003), S1237.

[26] Tetyana V. Bogdan, David J. Wales, and Florent Calvo. “Equilibrium thermodynamics from basin-sampling”. In: *Journal of Chemical Physics* 124.4 (2006), p. 044102.

[27] S. S. Ashwin et al. “Calculations of the structure of basin volumes for mechanically stable packings”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 85.6 (2012), p. 061307.

[28] Daniel Asenjo et al. “Visualizing basins of attraction for different minimization algorithms”. In: *Journal of Physical Chemistry B* 117.42 (2013), pp. 12717–12723.

[29] Hyun Joo Hwang, Robert A. Riggleman, and John C. Crocker. “Understanding soft glassy materials using an energy landscape approach”. In: *Nature Materials* 15.9 (2016), pp. 1031–1036.

[30] Stefano Martiniani et al. “Structural analysis of high-dimensional basins of attraction”. In: *Physical Review E* 94.3 (2016), 031301(R). ISSN: 24700053.

[31] Stefano Martiniani et al. “Numerical test of the Edwards conjecture shows that all packings are equally probable at jamming”. In: *Nature Physics* 13.9 (2017), pp. 848–851.

[32] David J. Wales. “Exploring Energy Landscapes”. In: *Annual Review of Physical Chemistry* 69 (2018), pp. 401–425.

[33] R. C. Dennis and E. I. Corwin. “Jamming Energy Landscape is Hierarchical and Ultrametric”. In: *Physical Review Letters* 124.7 (2020), p. 78002.

[34] Horst Holger Boltz, Jorge Kurchan, and Andrea J. Liu. “Fluctuation distributions of energy minima in complex landscapes”. In: *Physical Review Research* 3.1 (2021), p. 013061.

[35] Amruthesh Thirumalaiswamy, Robert A Riggleman, and John C Crocker. “Exploring canyons in glassy energy landscapes using metadynamics”. In: *Proceedings of the National Academy of Sciences* 119.43 (2022), e2210535119.

[36] Varda F. Hagh and Sidney R. Nagel. “Permutation Symmetry Restoration in Disordered Materials”. In: *Arxiv Preprint* (2024), p. 2403.03926.

[37] Florent Krzakala and Jorge Kurchan. “Landscape analysis of constraint satisfaction problems”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 76.2 (2007), p. 021122.

[38] Silvio Franz et al. “Universality of the SAT-UNSAT (jamming) threshold in non-convex continuous constraint satisfaction problems”. In: *SciPost Physics* 2.3 (2017), p. 019.

[39] Antonio Sclocchi and Pierfrancesco Urbani. “High-dimensional optimization under non-convex excluded volume constraints”. In: *Physical Review E* 105.2 (2022), p. 024134.

[40] Anna Choromanska et al. “The loss surfaces of multilayer networks”. In: *Artificial intelligence and statistics*. PMLR. 2015, pp. 192–204.

[41] Marco Baity-Jesi et al. “Comparing dynamics: Deep neural networks versus glassy systems”. In: *International Conference on Machine Learning*. PMLR. 2018, pp. 314–323.

[42] Silvio Franz, Antonio Sclocchi, and Pierfrancesco Urbani. “Critical Jammed phase of the linear perceptron”. In: *Physical Review Letters* 123.11 (2019), p. 115702.

[43] Stefano Sarao Mannelli et al. “Who is afraid of big bad minima? Analysis of gradient-flow in a spiked matrix-tensor model”. In: *Advances in Neural Information Processing Systems* 32.NeurIPS (2019). ISSN: 10495258.

[44] Riccardo Fabbricatore and Vladimir V. Palyulin. “Gradient dynamics in reinforcement learning”. In: *Physical Review E* 106 (2022), p. 025315.

[45] Persia Jana Kamali and Pierfrancesco Urbani. “Stochastic Gradient Descent outperforms Gradient Descent in recovering a high-dimensional signal in a glassy energy landscape”. In: *Arxiv Preprint* (2023), p. 2309.04788.

[46] Max Kerr Winter and Liesbeth M.C. Janssen. “Glassy dynamics in deep neural networks: A structural comparison”. In: *Physical Review Research* 7.2 (2025), p. 23010.

[47] Satyam Anand, Guanming Zhang, and Stefano Martiniani. “Emergent universal long-range structure in random-organizing systems”. In: *Arxiv Preprint* (2025), p. 2505.22933.

[48] Felix Roy et al. “Numerical implementation of dynamical mean field theory for disordered systems: application to the Lotka-Volterra model of ecosystems”. In: *Journal of Physics A: Mathematical and Theoretical* 52 (2019), p. 484001.

[49] Guy Bunin. “Ecological communities with Lotka-Volterra dynamics”. In: *Physical Review E* 95.4 (2017), p. 042414.

[50] Alfred J. Lotka. “Contribution to the Theory of Periodic Reaction”. In: *Journal of Physical Chemistry* 14.3 (1910), pp. 271–274.

[51] Tobias Galla. “Generating-functional analysis of random Lotka-Volterra systems: A step-by-step guide”. In: *Scipost Physics Lecture Notes* (2024), pp. 1–81.

[52] Ada Altieri et al. “Properties of Equilibria and Glassy Phases of the Random Lotka-Volterra Model with Demographic Noise”. In: *Physical Review Letters* 126.25 (2021), p. 258301.

[53] Mark A Miller and David J Wales. “Energy landscape of a model protein”. In: *The Journal of chemical physics* 111.14 (1999), pp. 6610–6616.

[54] Naomi Gendler et al. “Axion minima in string theory”. In: *Journal of High Energy Physics* 2025.2 (2025), pp. 1–24.

[55] Asimina Arvanitaki et al. “String axiverse”. In: *Physical Review D - Particles, Fields, Gravitation and Cosmology* 81.12 (2010), pp. 1–23. ISSN: 15507998. DOI: 10.1103/PhysRevD.81.123530.

[56] Steven H. Strogatz. *Nonlinear Dynamics and Chaos, with applications to physics, biology, chemistry and engineering*. Addison-Wesley, 1994.

[57] Stephen Wiggins. *Introduction to applied nonlinear dynamical systems and chaos*. Springer, 2003.

[58] Steven W. McDonald et al. “Fractal basin boundaries”. In: *Physica D: Nonlinear Phenomena* 17.2 (1985), pp. 125–153. ISSN: 0167-2789. DOI: https://doi.org/10.1016/0167-2789(85)90001-6.

[59] Celso Grebogi, Edward Ott, and James A. Yorke. “Chaos, Strange Attractors, and Fractal Basin Boundaries in Nonlinear Dynamics”. In: *Science* 238 (1987), pp. 632–638.

[60] Celso Grebogi, Edward Ott, and James A. Yorke. “Metamorphoses of Basin Boundaries in Nonlinear Dynamical Systems”. In: *Phys. Rev. Lett.* 56 (10 Mar. 1986), pp. 1011–1014. DOI: 10.1103/PhysRevLett.56.1011.

[61] Celso Grebogi et al. “Final state sensitivity: An obstruction to predictability”. In: *Physics Letters A* 99.9 (1983), pp. 415–418. ISSN: 0375-9601. DOI: https://doi.org/10.1016/0375-9601(83)90945-3.

[62] Jacobo Aguirre, Ricardo L Viana, and Miguel AF Sanjuán. “Fractal structures in nonlinear dynamics”. In: *Reviews of Modern Physics* 81.1 (2009), pp. 333–386.

[63] John Vandermeer, Lewi Stone, and Bernd Blasius. “Categories of chaos and fractal basin boundaries in forced predator–prey models”. In: *Chaos, Solitons & Fractals* 12.2 (2001), pp. 265–276.

[64] Jef Huisman and Franz J Weissing. “Fundamental unpredictability in multispecies competition”. In: *The American Naturalist* 157.5 (2001), pp. 488–494.

[65] Robin Delabays, Melvyn Tyloo, and Philippe Jacquod. “The size of the sync basin revisited”. In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 27.10 (Oct. 2017), p. 103109. ISSN: 1054-1500. DOI: 10.1063/1.4986156.

[66] Pablo Groisman et al. “The size of the sync basin resolved”. In: *arXiv preprint arXiv:2506.03419* (2025).

[67] Peter J Menck et al. “How dead ends undermine power grid stability”. In: *Nature communications* 5.1 (2014), p. 3969.

[68] Paul Schultz, Jobst Heitzig, and Jürgen Kurths. “Detours around basin stability in power networks”. In: *New Journal of Physics* 16.12 (2014), p. 125001.

[69] Nico Wunderling et al. “Basin stability and limit cycles in a conceptual model for climate tipping cascades”. In: *New Journal of Physics* 22.12 (2020), p. 123031.

[70] Oleg V. Maslennikov, Vladimir I. Nekorkin, and Jürgen Kurths. “Basin stability for burst synchronization in small-world networks of chaotic slow-fast oscillators”. In: *Phys. Rev. E* 92 (4 Oct. 2015), p. 042803. DOI: 10.1103/PhysRevE.92.042803.

[71] Frank H Stillinger. “Exponential multiplicity of inherent structures”. In: *Physical Review E* 59.1 (1999), p. 48.

[72] Frank H. Stillinger and Thomas A. Weber. “Hidden structure in liquids”. In: *Physical Review A* 25.2 (1982), pp. 978–989.

[73] Frank H. Stillinger. “A topographic view of supercooled liquids and glass formation”. In: *Science* 267.5206 (1995), pp. 1935–1939. ISSN: 00368075. DOI: 10.1126/science.267.5206.1935.

[74] Valentina Ros and Yan Fyodorov. “The High-dimensional Landscape Paradigm: Spin-Glasses, and Beyond”. In: *Spin Glass Theory and Far Beyond: Replica Symmetry Breaking after 40 Years* (2023), pp. 95–114.

[75] Marc Mézard and Andrea Montanari. *Information, Physics, and Computation*. Oxford University Press, 2009.

[76] S. F. Edwards and R. B.S. Oakeshott. “Theory of powders”. In: *Physica A: Statistical Mechanics and its Applications* 157.3 (1989), pp. 1080–1090. ISSN: 03784371.

[77] Frank H Stillinger and Thomas A Weber. “Inherent pair correlation in simple liquids”. In: *The Journal of chemical physics* 80.9 (1984), pp. 4434–4437.

[78] Frank H Stillinger, Thomas A Weber, et al. “Computer simulation of local order in condensed phases of silicon”. In: *Physical review B* 31.8 (1985), pp. 5262–5271.

[79] Patrick Charbonneau and Peter K. Morse. “Jamming, relaxation, and memory in a minimally structured glass former”. In: *Physical Review E* 108.5 (2023), p. 054102. ISSN: 24700053.

[80] Eddie Bautista and Eric I. Corwin. “Numerically Discovered Inherent States are Always Protocol Dependent in Jammed Packings”. In: *Arxiv Preprint* (2025), p. 2508.09284.

[81] Yoshihiko Nishikawa et al. “Relaxation Dynamics in the Energy Landscape of Glass-Forming Liquids”. In: *Physical Review X* 12.2 (2022), p. 21001.

[82] Patrick Charbonneau et al. “Fractal free energy landscapes in structural glasses”. In: *Nature Communications* 5 (2014), p. 4725.

[83] D. A. Lidar et al. “Fractal analysis of protein potential energy landscapes”. In: *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics* 59.2 (1999), pp. 2231–2243. ISSN: 1063651X. DOI: 10.1103/PhysRevE.59.2231.

[84] Yan Wei Li et al. “Softness, anomalous dynamics, and fractal-like energy landscape in model cell tissues”. In: *Physical Review E* 103.2 (2021), p. 22607. ISSN: 24700053. DOI: 10.1103/PhysRevE.103.022607.

[85] Chaoyi Liu and Yue Fan. “Emergent Fractal Energy Landscape as the Origin of Stress-Accelerated Dynamics in Amorphous Solids”. In: *Physical Review Letters* 127.21 (2021), p. 215502.

[86] Asaf Levi et al. “From local uncertainty to global predictions: Making predictions on fractal basins”. In: *PLoS ONE* 13.4 (2018), p. 0194926. ISSN: 19326203.

[87] Alvar Daza, Alexandre Wagemakers, and Miguel A.F. Sanjuán. “Classifying basins of attraction using the basin entropy”. In: *Chaos, Solitons and Fractals* 159 (2022), p. 112112. ISSN: 09600779.

[88] Erik Bollt et al. “Fractal basins as a mechanism for the nimble brain”. In: *Scientific Reports* 13.1 (2023), p. 20860.

[89] Jascha Sohl-Dickstein. “The boundary of neural network trainability is fractal”. In: *Arxiv Preprint* (2024), p. 2402.06184.

[90] Andrew Ly and Pulin Gong. “Optimization on multifractal loss landscapes explains a diverse range of geometrical and dynamical properties of deep learning”. In: *Nature Communications* 16.1 (2025), p. 3252.

[91] Mária Ercsey-Ravasz and Zoltán Toroczkai. “Optimization hardness as transient chaos in an analog approach to constraint satisfaction”. In: *Nature Physics* 7.12 (2011), pp. 966–970.

[92] Melinda Varga et al. “Order-to-chaos transition in the hardness of random Boolean satisfiability problems”. In: *Physical Review E* 93.5 (2016), p. 052211.

[93] Charusita Chakravarty, Pablo G. Debenedetti, and Frank H. Stillinger. “Generating inherent structures of liquids: Comparison of local minimization algorithms”. In: *Journal of Chemical Physics* 123.20 (2005), pp. 2005–2006. ISSN: 00219606. DOI: 10.1063/1.2129327.

[94] Claire P. Massen and Jonathan P.K. Doye. “Power-law distributions for the areas of the basins of attraction on a potential energy landscape”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 75.3 (2007), p. 037101.

[95] Eduardo Bautista and Eric I. Corwin. “Basins of Attraction in the Jamming Energy Landscape Have Power-Law Length Distributions”. In: *Bulletin of the American Physical Society*. Minneapolis, 2023.

[96] Kevin A. Interiano-Alberto, Peter K. Morse, and Robert S. Hoy. “Critical-like slowdown in thermal soft-sphere glasses via energy minimization”. In: *Physical Review E* 109.6 (2024), p. L062603.

[97] Daan Frenkel and Berend Smit. *Understanding molecular simulation: from algorithms to applications*. Academic Press, 2001.

[98] Michael R. Shirts and John D. Chodera. “Statistically optimal analysis of samples from multiple equilibrium states”. In: *Journal of Chemical Physics* 129.12 (2008).

[99] Wangang Xie et al. “Improving marginal likelihood estimation for Bayesian phylogenetic model selection”. In: *Systematic biology* 60.2 (2011), pp. 150–160.

[100] Erik Bitzek et al. “Structural relaxation made simple”. In: *Physical Review Letters* 97.17 (2006), p. 170201.

[101] Dong C. Liu and Jorge Nocedal. “On the limited memory BFGS method for large-scale optimization”. In: *Mathematical Programming* 45 (1989), pp. 503–528.

[102] William W. Hager and Hongchao Zhang. “The Limited Memory Conjugate Gradient Method”. In: *SIAM Journal on Optimization* 23.4 (2013), pp. 2150–2168. DOI: 10.1137/120898097.

[103] William W. Hager and Hongchao Zhang. “A Survey of Nonlinear Conjugate Gradient Methods”. In: *Pacific Journal of Optimization* 2.1 (2006), pp. 35–58.

[104] William W. Hager and Hongchao Zhang. “Algorithm 851: CG_DESCENT, A Conjugate Gradient Method with Guaranteed Descent”. In: *ACM Transactions on Mathematical Software* 32.1 (2006), pp. 113–137. DOI: 10.1145/1132973.1132979.

[105] William W. Hager and Hongchao Zhang. “A New Conjugate Gradient Method with Guaranteed Descent and an Efficient Line Search”. In: *SIAM Journal on Optimization* 16.1 (2005), pp. 170–192. DOI: 10.1137/030601880.

[106] JA Northby. “Structure and binding of Lennard-Jones clusters: 13≤ N≤ 147”. In: *The Journal of chemical physics* 87.10 (1987), pp. 6166–6177.

[107] Lev Davidovich Landau et al. *Theory of elasticity: volume 7*. Vol. 7. Elsevier, 2012.

[108] Guo Jie Gao, Jerzy Bławzdziewicz, and Corey S. O’Hern. “Frequency distribution of mechanically stable disk packings”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 74.6 (2006), p. 061304.

[109] Corey S. O’Hern et al. “Jamming at zero temperature and zero applied stress: The epitome of disorder”. In: *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics* 68.1 (2003), p. 011306. ISSN: 1063651X.

[110] Ethan Stanifer and M Lisa Manning. “Avalanche dynamics in sheared athermal particle packings occurs via localized bursts predicted by unstable linear response”. In: *Soft Matter* 18.12 (2022), pp. 2394–2406.

[111] Yurii Nesterov. “A method for solving the convex programming problem with convergence rate $O(1/k^2)$”. In: *Doklady Akademii Nauk SSSR* 269 (1983), pp. 543–547.

[112] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer, 2004.

[113] Charles George Broyden. “The convergence of a class of double-rank minimization algorithms 1. General considerations”. In: *IMA Journal of Applied Mathematics* 6.1 (1970), pp. 76–90. DOI: 10.1093/imamat/6.1.76.

[114] Roger Fletcher and Colin M Reeves. “Function minimization by conjugate gradients”. In: *The Computer Journal* 7.2 (1964), pp. 149–154. DOI: 10.1093/comjnl/7.2.149.

[115] Donald Goldfarb. “A family of variable-metric methods derived by variational means”. In: *Mathematics of Computation* 24.109 (1970), pp. 23–26. DOI: 10.1090/S0025-5718-1970-0258249-6.

[116] David F Shanno. “Conditioning of quasi-Newton methods for function minimization”. In: *Mathematics of Computation* 24.111 (1970), pp. 647–656. DOI: 10.1090/S0025-5718-1970-0572855-7.

[117] Jorge Nocedal. “Updating quasi-Newton matrices with limited storage”. In: *Mathematics of Computation* 35.151 (1980), pp. 773–782. DOI: 10.1090/S0025-5718-1980-0572855-7.

[118] Jorge Nocedal and Stephen Wright. *Numerical Optimization*. Springer, 1999.

[119] Ernst Hairer and Gerhard Wanner. “Solving ordinary differential equations. II, Vol. 14 of”. In: *Springer Series in Computational Mathematics (Springer Berlin Heidelberg, Berlin, Heidelberg, 1996)* 10 (1996), pp. 978–3.

[120] Alan C. Hindmarsh et al. “SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers”. In: *ACM Transactions on Mathematical Software* 31.3 (2005), pp. 363–396.

[121] Christopher Rackauckas and Qing Nie. “DifferentialEquations.jl – A Performant and Feature-Rich Ecosystem for Solving Differential Equations in Julia”. In: *Journal of Open Research Software* 5.1 (2017), p. 15.

[122] J M Kosterlitz. “The critical properties of the two-dimensional \text{XY} model”. In: *Journal of Physics C: Solid State Physics* 7.6 (1974), p. 1046.

[123] Stefano Martiniani. “On the complexity of energy landscapes: algorithms and a direct test of the Edwards conjecture”. PhD thesis. 2017.

[124] Jacques Villain. “Spin glass with non-random interactions”. In: *Journal of Physics C: Solid State Physics* 10 (1977), p. 1717.

[125] Jacques Villain. “Two-level systems in a spin-glass model. I. General formalism and two-dimensional model”. In: *Journal of Physics C: Solid State Physics* 10 (1977), p. 4793.

[126] Jooyoung Lee, J Michael Kosterlitz, and Enzo Granato. “Monte Carlo Study of Frustrated XY Models on a Triangular and Square Lattice”. In: *Physical Review B* 43.13 (1991), pp. 11531–11534.

[127] Ch Tsitouras. “Runge–Kutta pairs of order 5 (4) satisfying only the first column simplifying assumption”. In: *Computers & Mathematics with Applications* 62.2 (2011), pp. 770–775.

[128] Frank H. Stillinger and Thomas A. Weber. “Computer simulation of local order in condensed phases of silicon”. In: *Physical Review B* 31.8 (1985), pp. 5262–5271.

[129] Randall A LaViolette and Frank H Stillinger. “Thermal disruption of the inherent structure of simple liquids”. In: *The Journal of chemical physics* 85.10 (1986), pp. 6027–6033.

[130] Daniel Vågberg et al. “Finite-size scaling at the jamming transition: Corrections to scaling and the correlation-length critical exponent”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 83.3 (2011), 030303(R).

[131] C. J. Clopper and E. S. Pearson. “The use of confidence or fiducial limits illustrated in the case of the binomial”. In: *Biometrika* 26.4 (1934), pp. 404–413.

[132] Chris Glasbey et al. “Colour displays for categorical images”. In: *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur* 32.4 (2007), pp. 304–309.

[133] Bogdan I. Epureanu and Henry S. Greenside. “Fractal basins of attraction associated with a damped Newton’s method”. In: *SIAM Review* 40.1 (1998), pp. 102–109.

[134] Daan Frenkel, Daniel Asenjo, and Fabien Paillusson. “The other entropy”. In: *Molecular Physics* 111.22-23 (2013), pp. 3641–3650.

[135] Fabien Paillusson. “Devising a protocol-related statistical mechanics framework for granular materials”. In: *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 91.1 (2015), p. 012204. ISSN: 15502376.

[136] Aaron Clauset and Cosma Rohilla Shalizi. “Power-Law Distributions in Empirical Data”. In: *SIAM Review* 51.4 (2009), pp. 661–703.

[137] Anna D Broido and Aaron Clauset. “Scale-free networks are rare”. In: *Nature Communications* 10 (2019), p. 1017.

[138] Kenneth Falconer. *Fractal geometry: mathematical foundations and applications*. John Wiley & Sons, 2013.

[139] Jeff Gostick et al. “PoreSpy: A Python Toolkit for Quantitative Analysis of Porous Media Images”. In: *Journal of Open Source Software* 4.37 (2019), p. 1296.

[140] Mervin E. Muller. “A Note on a Method for Generating Points Uniformly on N-dimensional Spheres”. In: *Communications of the ACM* 2.4 (1959), pp. 19–20. DOI: 10.1145/377939.377946.

[141] B Efron. “Bootstrap methods: another look at the jackknife”. In: *Annals of Statistics* 7.1 (1979), pp. 1–26.

[142] Mathias Casiulis and Stefano Martiniani. “When you can’t count, sample! Computable entropies beyond equilibrium from basin volumes”. In: *Papers in Physics* 15 (2023), p. 150001.

[143] Alex Bunker and Burkhard Dünweg. “Parallel excluded volume tempering for polymer melts”. In: *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics* 63.1 (2001), p. 016701. ISSN: 1063651X.

[144] Hiroaki Fukunishi, Osamu Watanabe, and Shoji Takada. “On the Hamiltonian replica exchange method for efficient sampling of biomolecular systems: Application to protein structure prediction”. In: *Journal of Chemical Physics* 116.20 (2002), pp. 9058–9067.

[145] Tsuneyasu Okabe et al. “Replica-exchange Monte Carlo method for the isobaric–isothermal ensemble”. In: *Chemical Physics Letters* 335 (2001), pp. 435–439. DOI: 10.1016/S0009-2614(01)00055-0.

[146] Saifuddin Syed et al. “Non-Reversible Parallel Tempering: A Scalable Highly Parallel MCMC Scheme”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 84.2 (Dec. 2021), pp. 321–350. ISSN: 1369-7412. DOI: 10.1111/rssb.12464.

[147] George Casella and Roger L. Berger. *Statistical Inference*. 2nd. Duxbury Press, 2002.

[148] Johannes Klicpera. “Measuring the Energy Landscapes of Large Granular Systems: A Numerical Test of the Edwards Conjecture in 3D and a Basin Volume Analysis of the Yielding Transition”. Master’s Thesis. Technische Universität München, Nov. 2017.

[149] Matthias Troyer and Uwe-Jens Wiese. “Computational Complexity and Fundamental Limitations to Fermionic Quantum Monte Carlo Simulations”. In: *American Journal of Physics* 78.2 (2010), pp. 150–157. DOI: 10.1119/1.3266847.

[150] Shiri Artstein-Avidan, Apostolos Giannopoulos, and Vitali D. Milman. *Asymptotic Geometric Analysis , Part I*. Vol. 202. Providence, Rhode Island: American Mathematical Society, 2015.

[151] Brandon Livio Annesi et al. “The star-shaped space of solutions of the spherical negative perceptron”. In: *Physical Review Letters* 131.22 (2023), p. 227301.

[152] Saifuddin Syed et al. “Parallel Tempering on Optimized Paths”. In: *Proceedings of the 38th International Conference on Machine Learning (ICML)*. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 10033–10042.

[153] Nikola Surjanovic et al. “Parallel Tempering With a Variational Reference”. In: *Advances in Neural Information Processing Systems*. Vol. 35. 2022, pp. 565–577.

[154] Nikola Surjanovic et al. *Uniform Ergodicity of Parallel Tempering With Efficient Local Exploration*. 2024.

[155] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[156] Giampaolo Folena et al. "Jamming the Random Lorentz Gas: Configurational Entropy, Crunching Geometry, and Critical Universality". In: *Arxiv Preprint* (2024), p. 2410.05784.

[157] George Datseris et al. *Feature request: Compute basin volumes via nested sampling*. GitHub Issue. Attractors.jl Issue #60. 2022.

[158] Robert M May. *Stability and complexity in model ecosystems*. Vol. 6. Princeton university press, 2001.

[159] Matthieu Barbier et al. "Generic assembly patterns in complex ecological communities". In: *Proceedings of the National Academy of Sciences of the United States of America* 115.9 (2018), pp. 2156–2161.

[160] Giulio Biroli, Guy Bunin, and Chiara Cammarota. "Marginally stable equilibria in critical ecosystems". In: *New Journal of Physics* 20.8 (2018), p. 083051.

[161] Leon Glass. "Multistable spatiotemporal patterns of cardiac activity". In: *Proceedings of the National Academy of Sciences* 102.30 (2005), pp. 10409–10410.

[162] Mayo Clinic. *Ventricular fibrillation - Symptoms & causes*. Mayo Clinic. Accessed: 2025-12-05. 2022.

[163] Shanlin Qin et al. "A highly parallel fully implicit domain decomposition method for the simulation of the hemodynamics of a patient-specific artery at the full-body scale". In: *Journal of Computational Physics* 472 (2023), p. 111730.

[164] Hana Krakovská, Christian Kuehn, and Iacopo P Longo. "Resilience of dynamical systems". In: *European Journal of Applied Mathematics* 35.1 (2024), pp. 155–200.

[165] Youcef Saad and Martin H Schultz. “GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear”. In: *SIAM Journal on Scientific Statistical Computing* 7.3 (1986), pp. 856–869.

[166] David J. Gardner et al. “Enabling New Flexibility in the SUNDIALS Suite of Nonlinear and Differential/Algebraic Equation Solvers”. In: *ACM Transactions on Mathematical Software* 48.3 (2022), p. 31.

[167] Jacob D. Stevenson, Stefano Martiniani, and Julian Schrenk. *pele*. 2024.