

# Limited-Memory BFGS for Large-Scale Unconstrained Optimization

---

Sam Dawley

May 7, 2024

Princeton University

# Introduction

Exact and iterative solutions to nonlinear systems of equations are of great theoretical and numerical interest due to their prolific appearance across many of the economic, financial, and scientific fields which drive modern academic and industrial research [1, 2].

Under this umbrella exist optimization problems, generally written

$$\min f(x) \tag{1}$$

such that  $x \in \mathbf{D}$

where we seek the value of some variable  $x$  within a predefined domain  $\mathbf{D}$  which minimizes the objective function  $f(x)$ .

In case the domain  $\mathbf{D}$  defines some set of equalities or inequalities on  $x$  the problem is said to be **constrained**; in this instance there are a plethora of solution techniques spanning fields in calculus and linear programming [3, 4]. In contrast, if  $\mathbf{D}$  does not define any constraints on the variables  $x$  our problem is **unconstrained** and the set of techniques for finding solutions is dramatically reduced.

## Introduction: Quasi-Newton Methods

When the objective function is unwieldy and complicated a class of **quasi-Newton methods** have been developed which seek to approximate the Jacobian matrix at each iteration rather than compute it outright. From the point of view of Broyden [5] the iterative step is written

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbb{B}_k^{-1} f(\mathbf{x}_k) \quad (2)$$

with  $\mathbb{B}_k$  a transformation which approximates the Jacobian at  $f(\mathbf{x}_k)$ . Efficient methods of developing and updating  $\mathbb{B}_k$  for the purposes of optimization are an active area of research which spawned from major accomplishments during the latter half of the twentieth century [3, 4, 6].

## Introduction: BFGS Update

Among the most important quasi-Newton developments in the 20th and 21st century is the *Broyden–Fletcher–Goldfarb–Shanno* (BFGS) update which uses Eq. (2) with

$$\bar{\mathbb{B}}^{-1} = \bar{\mathbb{H}} = \mathbb{H} + \frac{\mathbf{s}\mathbf{s}^T}{\mathbf{y}^T\mathbf{s}} \left[ \frac{\mathbf{y}^T\mathbb{H}\mathbf{y}}{\mathbf{y}^T\mathbf{s}} + 1 \right] - \frac{\mathbf{s}\mathbf{y}^T\mathbb{H} + \mathbb{H}\mathbf{s}^T}{\mathbf{y}^T\mathbf{s}} \quad (3)$$

where  $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$  and  $\mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$  are the differences in position and gradient of the objective function.

Where memory is constrained, Nocedal [7] introduces the alternative representation of (3)

$$\bar{\mathbb{H}} = \mathbb{V}^T \mathbb{H} \mathbb{V} + \rho \mathbf{s} \mathbf{s}^T \quad (4)$$

with  $\mathbb{V} = \mathbb{I} - \rho \mathbf{s} \mathbf{y}^T$  and  $\rho = 1/\mathbf{y}^T \mathbf{s}$ . Equation (4) illustrates that termination—in the sense  $\bar{\mathbb{H}} = \mathbb{H}$  is unchanged—occurs if  $\mathbb{V} = \mathbb{I}$  the identity matrix and  $\rho = 0$ .

## Introduction: LBFGS Update

The extended representation of Eq. (4) at each iteration looks like

$$\begin{aligned}\mathbb{H}_1 &= \mathbb{V}_0^T \mathbb{H}_0 \mathbb{V}_0 + \rho_0 \mathbf{s}_0 \mathbf{s}_0^T \\ \mathbb{H}_2 &= \mathbb{V}_1^T \mathbb{V}_0^T \mathbb{H}_0 \mathbb{V}_0 \mathbb{V}_1 + \mathbb{V}_1^T \rho_0 \mathbf{s} \mathbf{s}_0^T \mathbb{V}_1 + \rho_1 \mathbf{s}_1 \mathbf{s}_1^T \\ &= \mathbb{V}_1^T \mathbb{H}_0 \mathbb{V}_1 + \rho_1 \mathbf{s}_1 \mathbf{s}_1^T, \quad \text{etc.}\end{aligned}$$

with the final line a result of dropping “old information” where  $\mathbb{V}_0 = \mathbb{I}$  and  $\rho_0 \mathbf{s}_0 \mathbf{s}_0^T = \mathbf{0}$ . In general we may choose to discard old information beyond the previous  $m$  iterations, giving us a scheme for updating  $\bar{\mathbb{H}}$  which relies only the previous  $m$  realizations of  $\mathbb{H}$ , in contrast to the update formula (3) which *requires* knowledge of all previous iterations. For this reason Nocedal’s method is referred to as the *limited-memory BFGS (LBFGS)* update.

## Motivation: Collective Density Variables

Expressing the distribution of points in  $N$ -body systems is often done most naturally by using **collective density variables** [8–15], conventionally defined by the transformation

$$\tilde{\rho}(\mathbf{k}) = \sum_{j=1}^N \exp(i\mathbf{k} \cdot \mathbf{r}_j) \quad (5)$$

whereby  $\mathbf{r}_j$  are the positions of all points  $j = 1, \dots, N$  and  $\mathbf{k}$  is a wave vector in the appropriately chosen reciprocal space of a given  $d$ -dimensional domain  $\Omega$  containing all  $N$  points. At present we will be interested in studying  $\tilde{\rho}(\mathbf{k})$  as it relates to the *static structure factor*,  $S(\mathbf{k})$ , directly proportional to the total scattering intensity of a material:

$$S(\mathbf{k}) = \frac{|\tilde{\rho}(\mathbf{k})|^2}{N} = 1 + \frac{2}{N} C(\mathbf{k}), \quad (6)$$

$$C(\mathbf{k}) = \sum_{j < k} \cos[\mathbf{k} \cdot (\mathbf{r}_k - \mathbf{r}_j)] \quad \text{with} \quad -\frac{N}{2} \leq C(\mathbf{k}) \leq \frac{N(N-1)}{2}$$

## Motivation: Constraints on Collective Variables

In the simplest setting we consider a square-integrable pairwise additive potential  $v(\mathbf{r})$  acting on the domain  $\Omega$  described above with volume  $V_\Omega$ . The total potential energy becomes

$$\Phi(\mathbf{r}^N) = \sum_{\mathbf{k} \in Q} V(\mathbf{k}) [C(\mathbf{k}) - C_0(\mathbf{k})]^2$$
$$V(\mathbf{k}) = \begin{cases} V_0 > 0 & 0 \leq |\mathbf{k}| \leq K \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Crucially, for pairwise potentials  $V(\mathbf{k})$  supported on the compact interval  $0 \leq |\mathbf{k}| \leq K$ ,  $K > 0$  whereby  $S(\mathbf{k})$  is constrained to its absolute minimum value of zero (wherever  $V(\mathbf{k})$  is supported) then

- (I) the system must be at a global energy minimum and
- (II) for all  $|\mathbf{k}| \leq K$  density variations are completely suppressed.

Such point patterns are referred to as *stealthy*.

## Motivation: Constraints on Collective Variables (cont.)

The generation of stealthy hyperuniform materials is highly nontrivial and amounts to the *collective coordinate optimization* procedure which seeks a target  $S_0(\mathbf{k})$  beginning from an arbitrary initial structure factor.

Minimizing  $\Phi$  of Eq. (7) for a  $d$ -dimensional system of  $N$  particles is tantamount to the unconstrained optimization of  $dN$  variables, the dimensionality of which is gargantuan for even the smallest of physically realizable systems. It is for this reason we employ the LBFGS update step in our optimization algorithm in generating stealthy hyperuniform materials beginning from arbitrary initial conditions.

## Methodology

Unconstrained optimization of the objective function (7) will be done using the LBFGS update step (4) as outlined by Nocedal [7]. We will be most interested in systems containing a number of points useful for modelling simple to complex physical systems in the thermodynamic limit for general two- and three-dimensional spaces.

The most important aspect of our implementation will be ensuring that the **optimization performs well for a host of arbitrary initial conditions** and as such we will test the algorithm using various unique point patterns as initial conditions.

## Methodology: A Special BFGS Update Formula

Implementation of the LBFGS update step follows that of Nocedal [7] closely. Beginning from the same problem statement as in (1), consider the objective function  $f : \mathbb{R}^N \rightarrow \mathbb{R}$  with a gradient  $g$  and hessian matrix  $h$ . Define

$$\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k \quad \text{and} \quad \mathbf{y}_k = g_{k+1} - g_k \quad (8)$$

The quasi-Newton update step becomes

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbb{B}_k^{-1} f(\mathbf{x}_k) = \mathbf{x}_{k+1} - \mathbb{H}_k f(\mathbf{x}_k) \quad (9)$$

where  $\mathbb{B}_k$  is a suitable approximation to the Jacobian matrix of  $f(\mathbf{x}_k)$ . We introduce the matrix  $\mathbb{H}_k$ , an approximation to the *inverse* of the Jacobian matrix. In particular, the *special BFGS matrix*  $\mathbb{H}_{\text{new}}$  may be written

$$\mathbb{H}_{\text{new}} = [\mathbb{I} - \rho \mathbf{s} \mathbf{y}^T] \mathbb{H}_{\text{old}} [\mathbb{I} - \rho \mathbf{y} \mathbf{s}^T] + \rho \mathbf{s} \mathbf{s}^T = \mathbb{V}^T \mathbb{H}_{\text{old}} \mathbb{V} + \rho \mathbf{s} \mathbf{s}^T \quad (10)$$

with  $\rho = 1/\mathbf{y}^T \mathbf{s}$ .

## Methodology: A Special BFGS Update Formula (cont.)

Choose  $m$  to be the number of previous corrections included to the modified BFGS update  $\mathbb{H}_{k+1}$ . In general, for  $k + 1 \leq m$  the usual BFGS update is

$$\begin{aligned}\mathbb{H}_{k+1} = & (\mathbb{V}_k^T \mathbb{V}_{k-1}^T \cdots \mathbb{V}_0^T \mathbb{H}_0 \mathbb{V}_0 \cdots \mathbb{V}_{k-1} \mathbb{V}_k) \\ & + (\mathbb{V}_k^T \cdots \mathbb{V}_1^T \rho_0 \mathbf{s}_0 \mathbf{s}_0^T \mathbb{V}_1 \cdots \mathbb{V}_k + \mathbb{V}_k^T \mathbb{V}_{k-1}^T \rho_{k-2} \mathbf{s}_{k-2} \mathbf{s}_{k-2}^T \mathbb{V}_{k-1} \mathbb{V}_k) \\ & \cdots + (\mathbb{V}_k^T \rho_{k-1} \mathbf{s}_{k-1} \mathbf{s}_{k-1}^T \mathbb{V}_k + \rho_k \mathbf{s}_k \mathbf{s}_k^T)\end{aligned}\quad (11)$$

In the special case that  $k + 1 > m$ , the BFGS update becomes

$$\begin{aligned}\mathbb{H}_{k+1} = & (\mathbb{V}_k^T \mathbb{V}_{k-1}^T \cdots \mathbb{V}_{k-m+1}^T \mathbb{H}_0 \mathbb{V}_{k-m+1} \cdots \mathbb{V}_{k-1} \mathbb{V}_k) \\ & + (\mathbb{V}_k^T \cdots \mathbb{V}_{k-m+2}^T \rho_{k-m+1} \mathbf{s}_{k-m+1} \mathbf{s}_{k-m+1}^T \mathbb{V}_{k-m+2} \cdots \mathbb{V}_k) \\ & \cdots + (\mathbb{V}_k^T \rho_{k-1} \mathbf{s}_{k-1} \mathbf{s}_{k-1}^T \mathbb{V}_k + \rho_k \mathbf{s}_k \mathbf{s}_k^T)\end{aligned}\quad (12)$$

## Methodology: A Special BFGS Update Formula (cont.)

Eqs. (11) and (12) are our special BFGS update matrices. Importantly, if  $\mathbb{H}_0$  is a positive definite matrix—most easily realized by allowing our initial guess to be the identity matrix—it is straightforward to verify that all subsequent matrices derived from (11) and (12) are positive definite also, provided  $\mathbf{y}_i^T \mathbf{s}_i > 0$  for all iterations.

The formulas (11) and (12) may be used for quasi-Newton updates of the form

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k, \quad \mathbf{d}_k = -\mathbb{H}_k \mathbf{g}_k \quad (13)$$

where each iteration uses a limited memory update to determine the search direction  $\mathbf{d}_k$  and a line search method for determining the scalar  $\alpha_k$ . In any case, the special BFGS update yields an algorithm with quadratic termination. For details regarding this proof the reader is referred to Ref. [7].

## Results: Initial Conditions

We consider the following set of initial conditions as benchmark cases for our algorithm:

| Initial Conditions         |              |        |        |      |     |
|----------------------------|--------------|--------|--------|------|-----|
| Packing                    | Final $\chi$ | $\rho$ | $\phi$ | $N$  | $M$ |
| URL                        |              | 1.0    | 0.08   | 1024 | 50  |
| Poisson                    | 0.02         | 1.0    | 0.54   | 1024 | 50  |
| Saturated RSA <sup>1</sup> |              | 1.0    | 0.54   | 1005 | 44  |

**Table 1:** Initial conditions considered to for the unconstrained optimization of Eq. (7) using the LBFGS update formula. Here  $\rho$  is the number density of particles,  $\phi$  is the dimensionless density, also known as the volume fraction,  $N$  is the number of points considered,  $M$  is the number of members within each ensemble, and  $\chi$  is a parameter which controls the exclusion region  $K$  chosen to enforce  $K = 1$  in the ground state.

---

<sup>1</sup>This is an ensemble average number of particles  $N$ .

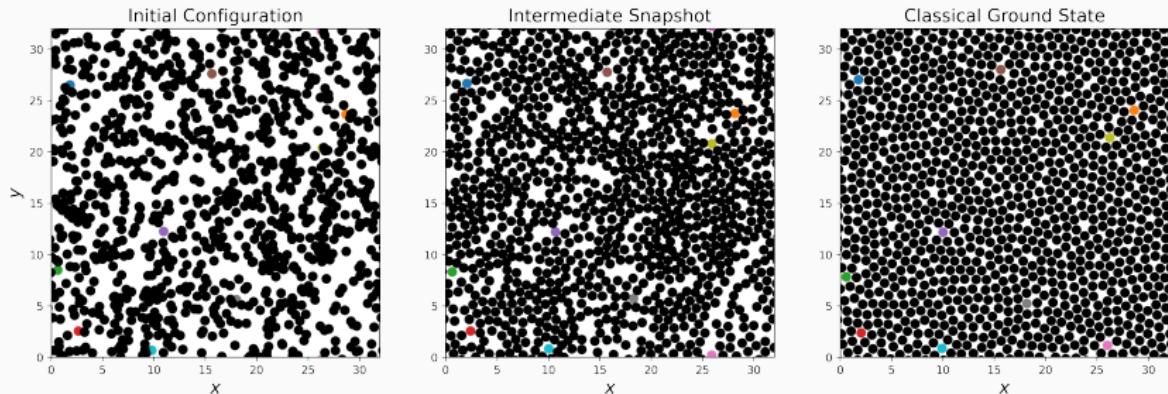
## Results (cont.)

The initial and final configurations provided by our algorithm are illustrated by Figs. 1 through 5 for each of the initial conditions listed in Table 1. Some arbitrary points are colored so we get an idea of how much each particle was displaced from its original position.

The structure factors for the same initial conditions and final ground states are shown in Figs. 2 through 6. A dotted red line running vertical along these graphs depicts the extent of the exclusion region in terms of the dimensionless wave number  $kD$ .

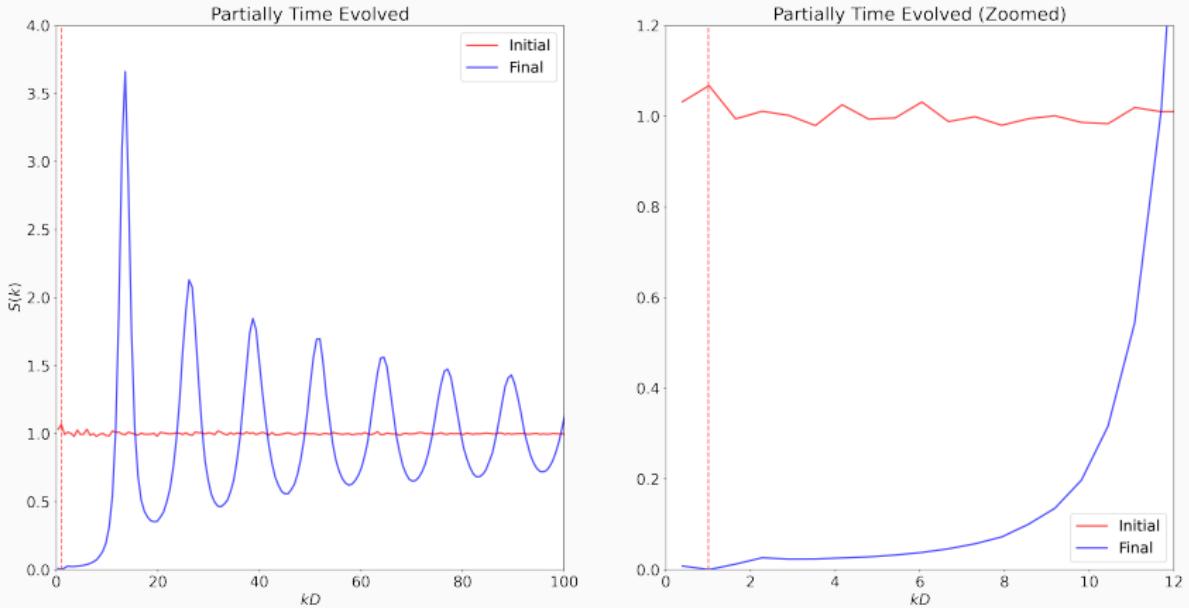
# Results: Poisson Point Patterns

Configurational Snapshots of Poisson Point Pattern Optimized to  $\chi = 0.02$  Stealthy Ground State



**Figure 1:** Temporal evolution of Poisson point pattern from initial condition to stealthy ground state with  $\chi = 0.02$  following collective coordinate optimization.

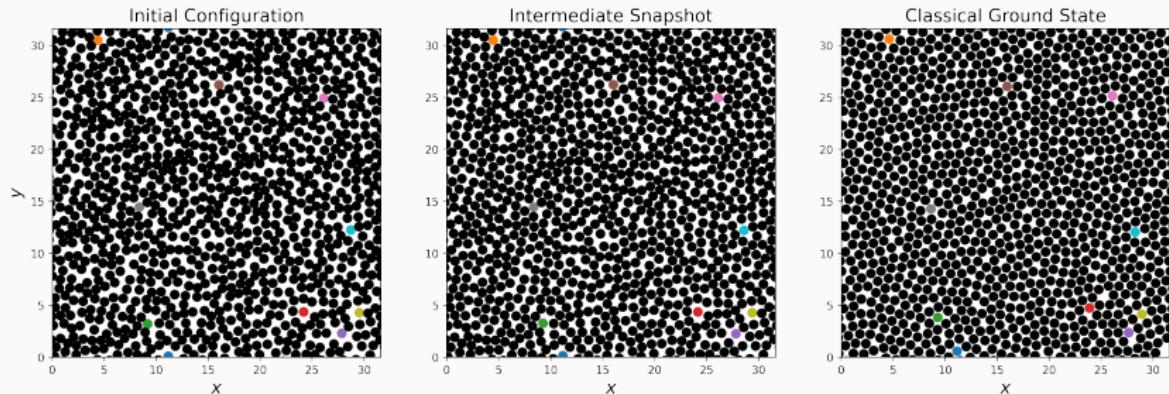
## Results: Poisson Point Patterns (cont.)



**Figure 2:** Time evolution of  $S(k)$  from initial Poisson point pattern to stealthy ground state with  $\chi = 0.02$ .

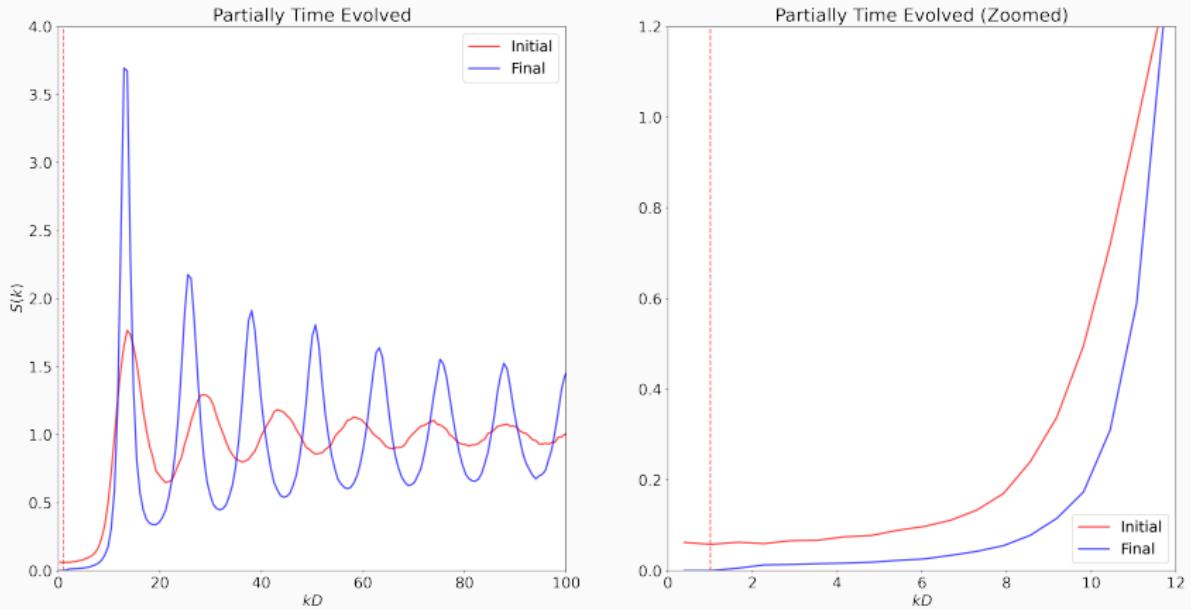
# Results: Saturated RSA Sphere Packings

Configurational Snapshots of RSA Point Pattern Optimized to  $\chi = 0.02$  Stealthy Ground State



**Figure 3:** Temporal evolution of saturated RSA point pattern from initial condition to stealthy ground state with  $\chi = 0.02$  following collective coordinate optimization.

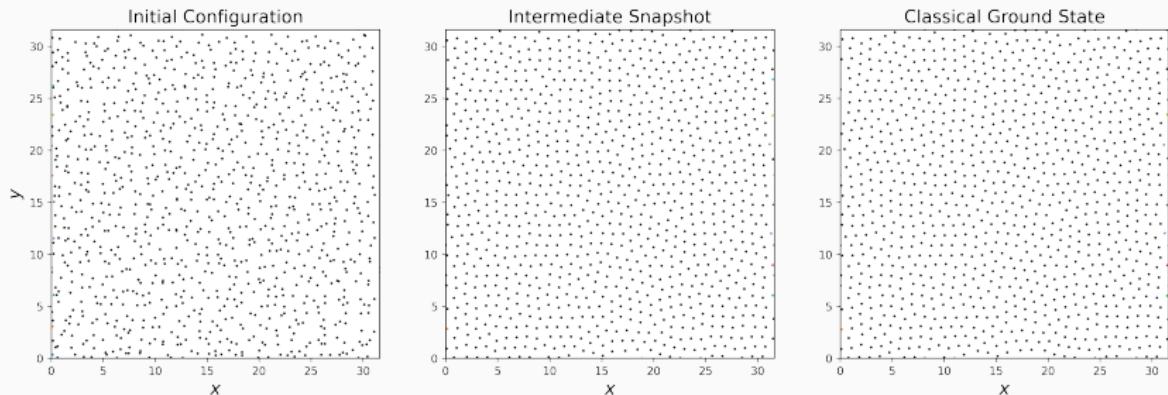
## Results: Saturated RSA Sphere Packings (cont.)



**Figure 4:** Time evolution of  $S(k)$  from initially saturated RSA sphere packing to stealthy ground state with  $\chi = 0.02$ .

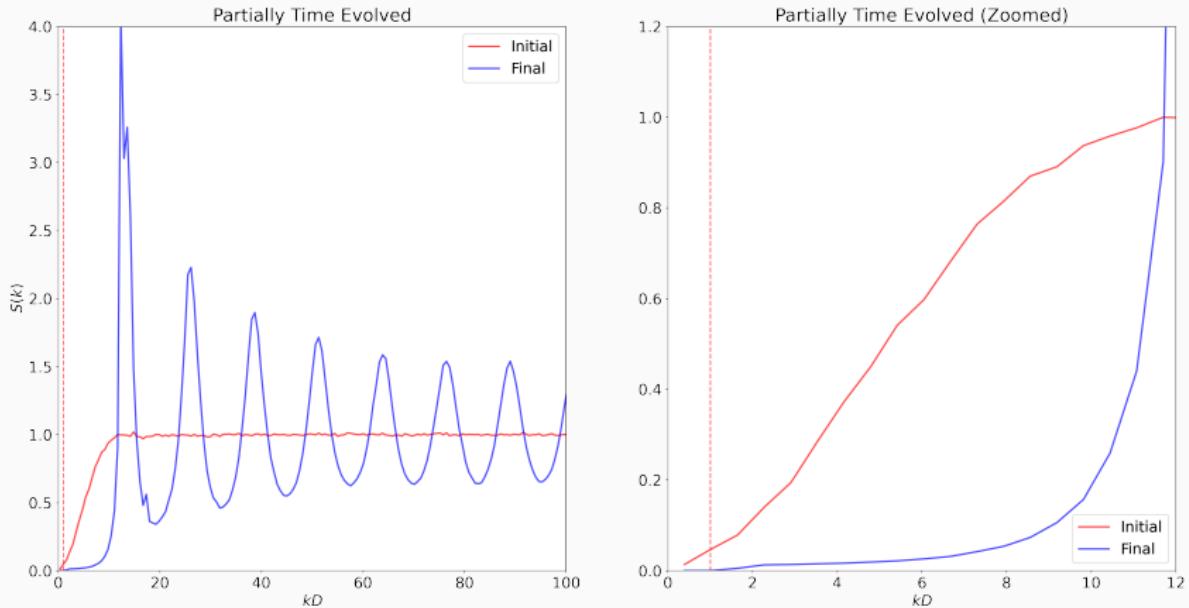
# Results: URL Point Patterns

Configurational Snapshots of URL Point Pattern Optimized to  $\chi = 0.02$  Stealthy Ground State



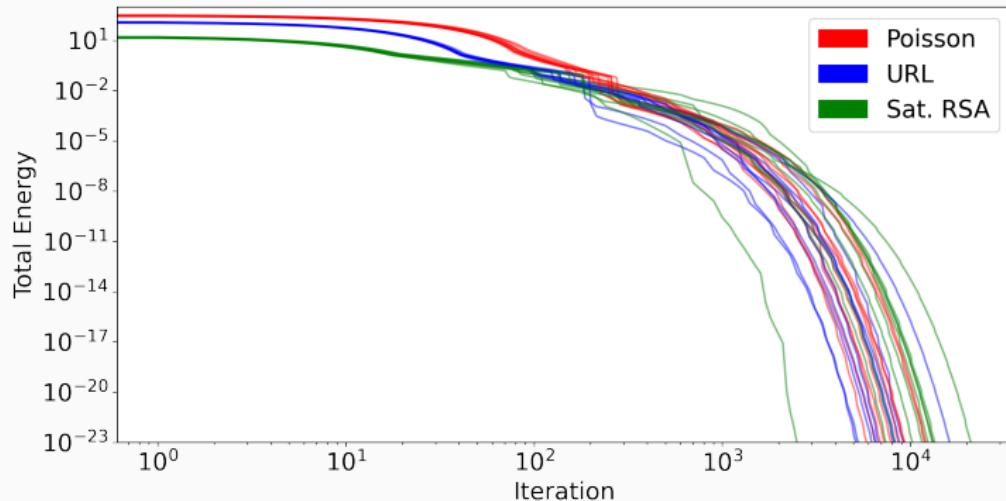
**Figure 5:** Temporal evolution of URL point pattern from initial condition to stealthy ground state with  $\chi = 0.02$  following collective coordinate optimization.

## Results: URL Point Patterns (cont.)



**Figure 6:** Time evolution of  $S(k)$  from initial URL point pattern to stealthy ground state with  $\chi = 0.02$ .

## Results: Energy Minimization



**Figure 7:** Energy as a function of number of iterations required by collective coordinate optimization using the LBFGS update step.

## Outlook

---

Although the implemented optimization algorithm taking advantage of a limited-memory BFGS update step provides excellent accuracy and efficiency for our purposes the final algorithm is far from optimized.

An analysis of the optimal number  $m$  of previous iterations to store could not be performed and likely contributes to suboptimal performance of the optimizer.

## References i

---

- [1] Z. Chen, Y. Liu, H. Sun, *Nature Communications* **2021**, *12*, 6136.
- [2] B. Oksendal, **Stochastic differential equations: an introduction with applications**, Springer Science & Business Media, 2013.
- [3] J. E. Dennis, Jr, J. J. Moré, *SIAM review* **1977**, *19*, 46–89.
- [4] R. Fletcher, **Practical Methods of Optimization**, Wiley, 2000.
- [5] C. G. Broyden, *Mathematics of computation* **1965**, *19*, 577–593.
- [6] R. H. Byrd, J. Nocedal, R. B. Schnabel, *Mathematical Programming* **1994**, *63*, 129–156.
- [7] J. Nocedal, *Mathematics of computation* **1980**, *35*, 773–782.
- [8] S. Ghosh, J. L. Lebowitz, *Communications in Mathematical Physics* **2018**, *363*, 97–110.
- [9] E. Leutheusser, *Phys. Rev. A* **1984**, *29*, 2765–2773.
- [10] R. D. Mountain, D. Thirumalai, *Physica A: Statistical Mechanics and its Applications* **1993**, *192*, 543–549.
- [11] J. K. Percus, G. J. Yevick, *Physical Review* **1958**, *110*, 1.

## References ii

---

- [12] D. Pines in *Solid state physics volume 1*, Elsevier, **1955**, pages 367–450.
- [13] A. Sütő, *Physical review letters* **2005**, *95*, 265501.
- [14] G. J. Yevick, J. K. Percus, *Physical Review* **1956**, *101*, 1186.
- [15] G. Zhang, F. H. Stillinger, S. Torquato, *The Journal of Chemical Physics* **2016**, *145*, DOI 10.1063/1.4972862.