Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Hybrid Particle Filtering and Local Optimization Framework for Solving Challenging Lennard-Jones Cluster Problems

François Bertrand Akoa

francoisakoa@gmail.com

MAATAI-LAB https://orcid.org/0009-0007-2652-3386

Research Article

Keywords: Global optimization, Lennard-Jones clusters, particle filtering, invertible particle flow, sequential Monte Carlo, local optimization

Posted Date: November 24th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11833624/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

A Hybrid Particle Filtering and Local Optimization Framework for Solving Challenging Lennard-Jones Cluster Problems

François Bertrand Akoa[1,*]

[1] *R&D Director, MAATAI-LAB.* francois.akoa@maatai-lab.com

*Corresponding author

Abstract Lennard-Jones (LJ) cluster optimization is a demanding benchmark for global search because competing energy funnels can trap otherwise effective methods. We introduce a hybrid particle filtering (HPF) framework coupling population exploration with local refinement without problem-specific seeding. An Invertible Particle-Flow (IPF) operator transports particles toward promising configurations, while diversity-aware weighting, adaptive annealing, resampling and rejuvenation balance selection and exploration. Separate accessibility and target-concentration arguments motivate exploration and annealed selection in an idealized uncapped setting; they do not establish convergence of the implemented algorithm. Experiments on LJ38, LJ75–77, LJ98 and LJ102–104 use 30 distinct box-uniform initial populations per size, with independently recorded search seeds. Comparators include basin-hopping with occasional jumping (BH-OJ), genetic algorithms, particle swarm optimization, comprehensive learning particle swarm optimization and sequential Monte Carlo. Baseline evaluation caps were fixed from a separate HPF calibration and do not equal realized computational costs; all BH-OJ/LJ98 runs are time-limited. HPF achieves 100% target attainment on LJ38 and LJ102–104, and 96.7%, 86.7%, 56.7% and 20.0% on LJ75, LJ76, LJ77 and LJ98. BH-OJ succeeds only on LJ38 (10%); other baseline cells have no successes. Ablation on LJ75 and LJ102 identifies local refinement as the largest contributor to target attainment under the tested limits. Removing IPF preserves observed success but increases mean evaluations by factors of 3.32 and 9.11, respectively. These results concern the tested implementations and parameterizations, not their algorithmic families; validation is restricted to LJ clusters.

Keywords Global optimization, Lennard-Jones clusters, particle filtering, invertible particle flow, sequential Monte Carlo, local optimization.

1 Introduction

GLOBAL optimization is a cornerstone of scientific and engineering research, where the goal is to find the absolute minimum of a function over its entire domain. For non-convex functions, characterized by multiple local minima, this task becomes particularly challenging, as standard local optimization methods often converge to sub-optimal solutions. Developing efficient global optimization algorithms is thus essential for addressing complex real-world problems.

One of the most well-studied benchmarks for global optimization is the Lennard-Jones (LJ) cluster problem. In this problem, we seek to determine the configuration of N atoms that minimizes the total potential energy, where the interaction between each pair of atoms is described by the LJ potential:

$$V(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right]. \tag{1}$$

The total energy of the cluster is then

$$E = \sum_{1 \le i < j \le N} V(r_{ij}), \tag{2}$$

where r_{ij} is the distance between atoms i and j, ϵ is the interaction strength, and σ is the effective atomic diameter. The LJ cluster problem is challenging because its non-convex energy landscape contains numerous local minima. Although most cluster sizes have icosahedral global minima, several exceptions are notorious:

- $N = 38$: an FCC truncated octahedron [1];
- $N = 75$–77 and 102–104: Marks decahedral minima [1];

- $N = 98$: a tetrahedral reference structure [3].

These non-icosahedral structures lie in narrow energy funnels, making them stringent tests for global optimization algorithms.

Traditional methods for LJ optimization include basin-hopping [1] and genetic algorithms [2], but even they can struggle with the hardest cases. In this paper, we propose a hybrid particle-filter/local-solver framework that overcomes these difficulties through a combination of global exploration mechanisms (particle filtering with IPF, adaptive kicks, island exchange, diversity metrics) and local refinement.

1.1 Contributions

Our main contributions include:

1. An Invertible Particle-Flow (IPF) operator that biases particles toward elite configurations through invertible, contractive transport while practical diversity is supported by the combined action of jitter, mutation, resampling, and rejuvenation; its contribution in the LJ75/LJ102 ablation concerns evaluation cost, whereas local optimization has the largest effect on target attainment
2. Comprehensive experimental validation with 30 search runs per instance, comparing against five representative stochastic baselines
3. Ablation studies demonstrating the contribution of individual algorithmic components
4. Sensitivity analysis of key hyperparameters on representative problem instances
5. Detailed algorithmic specifications enabling full reproducibility, including state representation, symmetry handling, transition operators, resampling methods, and local optimization configuration
6. Separate theoretical motivations for exploration and annealed selection in an idealized uncapped setting, rather than a convergence guarantee for the implemented algorithm, together with a discussion of their practical interpretation and relation to existing theory

To clarify the relationship between novel and adapted elements: the *novel* contributions of this work are (a) the Invertible Particle-Flow (IPF) operator as an optimization-oriented transport mechanism within particle filtering, (b) the specific integration of diversity-weighted importance sampling (sharing, novelty, crowding) with local gradient-based refinement, and (c) the ablation-driven analysis indicating the value of this tight integration on LJ75 and LJ102. Elements *adapted* from existing literature include particle filtering with adaptive resampling [26, 29], simulated annealing schedules [12], basin-hopping local refinement [1], island-model parallelism [20], and diversity metrics from evolutionary computation [22]. The key contribution lies not in any single component but in their principled integration into a coherent framework. On LJ75 and LJ102, the ablation study (Section 4.2) compares removal of individual mechanisms and two reduced configurations; it does not evaluate each component in isolation.

The remainder of this paper is organized as follows: Section 2 provides an expanded literature review contextualizing our work. Section 3 details the algorithmic framework. Section 4 presents experimental results. Section 5 concludes with future directions.

2 Literature Review

The optimization of Lennard-Jones (LJ) clusters represents a fundamental challenge in computational chemistry and physics, serving as a rigorous benchmark for global optimization algorithms due to its complex energy landscape characterized by exponentially numerous local minima [1, 4]. The core objective is to identify atomic configurations that minimize the potential energy, a task that becomes particularly challenging for specific cluster sizes where the global minimum resides in narrow energy funnels [8]. These problematic cases—notably clusters with 38, 75–77, 98, and 102–104 atoms—feature non-icosahedral structures including face-centered cubic (FCC) truncated octahedra, Marks decahedra, and Leary tetrahedra [3, 5], presenting significant convergence difficulties for traditional optimization methods.

2.1 Traditional Optimization Approaches

Early approaches relied heavily on stochastic and heuristic methods. The basin-hopping algorithm introduced by Wales and Doye [1] transformed the potential energy surface into a series of interpenetrating staircases, successfully identifying global minima for clusters up to 110 atoms. Genetic algorithms, particularly Deaven and Ho's molecular geometry optimizer [2], demonstrated remarkable efficacy by evolving candidate solutions through simulated natural selection. However, these methods exhibit three key limitations for challenging LJ clusters: (1) exponential scaling of computational cost with cluster size, (2) susceptibility to entrapment in local minima, and (3) poor performance on non-icosahedral structures [6, 7].

To address these limitations, researchers developed specialized enhancements. Leary's stochastic tunneling method [3] enabled escape from deep local minima by dynamically modifying the potential landscape, successfully identifying problematic non-icosahedral structures. Schoen's two-phase local search [4] significantly enlarged basins of attraction through strategic perturbation techniques, improving performance on 75-, 98-, and 102-atom clusters. Iwamatsu further refined basin-hopping with occasional jumping heuristics [5], accelerating convergence through controlled exploration-exploitation balancing.

Specialized LJ algorithms provide a stronger problem-specific context than generic stochastic baselines alone. Pullan's population-based search combines cut-and-paste moves, structural niching and directed optimization, recovering the reported putative minima for $2 \leq N \leq 372$ [32]. Dynamic lattice searching constructs candidate atomic sites adaptively and was evaluated up to 309 atoms [33]. Fuzzy global optimization combines discrete-space Monte Carlo moves with real-space local minimization and reports putative minima for a test set extending to 1,000 atoms [34]. Multicanonical basin hopping was tested on LJ_{150}–LJ_{185} [35], while revised basin hopping integrates additional move strategies for clusters and nanoparticles [36]. These studies include larger systems and, for PBS and FGO, the difficult sizes considered here. Our five baseline implementations are not an exhaustive representation of these specialized methods. Their published results use different protocols and are not pooled with our runs or interpreted as matched-budget comparisons.

2.2 Particle Swarm and Evolutionary Methods

Particle Swarm Optimization (PSO) has shown promise for LJ clusters through its population-based search strategy. Modern PSO variants incorporating adaptive inertia weights, constriction factors, and neighborhood topologies have achieved competitive performance [17, 18]. Recent advances include:

- Comprehensive Learning PSO (CLPSO) [19] using exemplars from different particles
- Dynamic Multi-Swarm PSO (DMS-PSO) [20] with regrouping mechanisms
- Unified PSO (UPSO) [21] combining multiple PSO variants

These methods motivate comparison with representative swarm optimizers, but their generic algorithmic specifications do not establish a common percentage improvement on LJ clusters. The experiments below assess the stated implementations under their recorded protocol.

2.3 Deterministic Global Optimization

Recent work has explored deterministic approaches that distinguish rigorous certification from finding low-energy configurations. Kuznetsov and Sahinidis [15] derive sharper interatomic-distance bounds, symmetry-reducing formulations and convex underestimators. Their computational study proves global optimality for the five- and six-atom LJ problems using a general-purpose global solver. These results illustrate the difficulty of certification; they do not certify the larger clusters used in the present stochastic benchmark.

2.4 Particle Filtering and Sequential Monte Carlo

A paradigm shift emerged with particle filtering techniques reformulating optimization as sequential state estimation. The foundational framework by Stinis [9] treats stochastic optimization as a filtering problem, concentrating computational effort in promising regions. Subsequent innovations include:

- **Gerber and Douc (2022)** [13]: Developed a particle-filter algorithm for objectives defined through expectations, using a concentrating target distribution and averaging to balance exploration and

convergence. Their assumptions and limit regime are distinct from those of the finite-budget HPF implementation.

- **Zhang, Taghvaei and Mehta** [10]: Introduced a controlled particle filter based on mean-field optimal control and density transport, avoiding the resampling-based construction. The result supplies a distinct variational viewpoint, not a measured LJ speedup for HPF.
- **Zhou et al.** [11, 12]: Developed randomized optimization frameworks exploiting annealing principles and adaptive importance sampling, demonstrating effectiveness on benchmark problems.
- **Eslami and Babazadeh (2024)** [14]: Formulated particle-filter optimization for noisy objective evaluations, with a customized transition model and an unscented-transform treatment of measurement noise.

These particle filtering methods have demonstrated promising results, but their application to challenging LJ clusters remains limited, particularly for cases requiring tight integration of global exploration and local refinement.

Annealed flow transport [37] combines learned transport, importance reweighting, resampling and Markov moves, with particle-limit results under explicit regularity assumptions. This differs from HPF's standardized diversity weights, finite-tolerance local solves and changing population. Such results, like Feynman–Kac particle theory [26], are relevant to a possible tracking analysis but do not by themselves supply the missing implication in Section 3.12.

2.5 Nested Sampling and Advanced Sampling Methods

Nested sampling, originally developed for Bayesian integration, is also used to explore molecular energy landscapes. Maillard et al. (2025) [16] study the partition functions, phase transitions and stable configurations of LJ7 and LJ36 with slice-sampling-based nested sampling. This complements optimization-oriented searches by addressing thermodynamic exploration; it is not a matched-budget comparison on the eight difficult instances considered here.

2.6 Diversity-Driven Strategies

Maintaining particle diversity represents a critical challenge in high-dimensional optimization. Goldberg and Richardson's sharing mechanisms [22] inspired diversity-preserving adaptations in evolutionary algorithms, later extended to particle swarm optimization [17]. Normalizing flows provided breakthrough density estimation capabilities through invertible transformations [23, 24], enabling effective diversity maintenance in complex landscapes. Neural ODE frameworks [25] further enhanced landscape topology prediction, creating synergies between deep learning and optimization.

2.7 Research Gaps and Contributions

Despite these advancements, several critical gaps remain in the literature:

1. **Limited integration of diversity mechanisms**: Existing particle filter methods for optimization often lack sophisticated diversity-preserving mechanisms, leading to premature convergence in multimodal landscapes.
2. **Insufficient local refinement**: Most SMC-based optimizers focus primarily on global exploration, with limited integration of gradient-based local optimization that could accelerate convergence once promising regions are identified.
3. **Lack of component-wise analysis**: Few studies provide ablation studies demonstrating which algorithmic components contribute most significantly to performance, making it difficult to identify essential mechanisms.
4. **Limited benchmark comparisons**: Comprehensive comparisons against multiple state-of-the-art methods (PSO variants, modern SMC approaches, deterministic methods) are rare in the literature.
5. **Insufficient experimental rigor**: Many studies report results from 10 or fewer independent runs, limiting statistical confidence, particularly for highly stochastic methods.

Our work addresses these gaps through a hybrid framework that:

1. Introduces an Invertible Particle-Flow (IPF) operator that supports practical diversity through its interaction with jitter, mutation, resampling, and rejuvenation, via invertible, contractive transport transformations toward elite configurations
2. Tightly integrates gradient-based local optimization with particle filtering
3. Incorporates multiple complementary diversity mechanisms (sharing, novelty, crowding distance)
4. Provides comprehensive experimental validation with 30 search runs per instance
5. Includes ablation studies measuring the effect of removing individual components on LJ75 and LJ102
6. Compares against multiple representative stochastic baselines
7. Provides detailed algorithmic specifications enabling reproducibility

3 Methods

3.1 Overview of the Hybrid Optimization Framework

The global optimization of the rugged Lennard-Jones (LJ) potential surface requires both exploration, to navigate the myriad local minima, and exploitation, to descend efficiently once a promising energy funnel is identified [1, 6]. Our hybrid framework couples particle filtering (PF), which maintains a probabilistic ensemble of candidate configurations, with deterministic local refinement. To overcome particle collapse in high-dimensional spaces, we introduce an Invertible Particle-Flow (IPF) operator (Section 3.4), while an adaptive kick mechanism (Section 3.8) reignites diversity if progress stalls. Diversity metrics, including sharing and novelty, enhance weight updates to promote exploration (Section 3.5). Island exchange further improves global search efficiency by sharing high-quality solutions across subpopulations (Section 3.9). The framework dynamically adjusts parameters to balance exploration and exploitation, ensuring robust performance across challenging LJ cluster sizes.

Figure 1 illustrates the data flow: particles are initialized within a box-shaped region, periodically exchanged, transformed via IPF and jitter, refined locally, re-weighted with diversity metrics, resampled based on effective sample size (ESS), and kicked if stagnant.

3.2 State Representation and Symmetry Handling

3.2.1 Coordinate System

Each LJ cluster configuration is represented as a vector $x \in \mathbb{R}^{3N}$ containing Cartesian coordinates:

$$x = [x_1, y_1, z_1, x_2, y_2, z_2, \ldots, x_N, y_N, z_N]^T \tag{3}$$

where (x_i, y_i, z_i) denotes the position of atom i. This representation is chosen for its simplicity and compatibility with gradient-based local optimization. The domain is bounded: $x \in \Omega = [-R, R]^{3N}$, where R is the container radius, initialized to $R_0 = 5.0$ and adaptively adjusted during optimization (Section 3.8).

Let $P_0 = 3N + 56$ denote the nominal initial population size and m_t the size actually present. At $t = 0, \ldots, T-1$, scheduling requests $\max(5, \lfloor P_0(1 - t/T) \rfloor)$ particles, with $T = 1000$. This step only selects existing particles; subsequent exchanges and injections can increase the size. Diagnostics use m_t, while the resampling thresholds use P_0.

3.2.2 Symmetry Considerations

The LJ potential exhibits several symmetries:

- **Translational invariance**: Energy is unchanged under rigid translations $x \to x + \mathbf{1}_N \otimes c$ where $c \in \mathbb{R}^3$ and $\mathbf{1}_N$ is the N-dimensional vector of ones.
- **Rotational invariance**: Energy is unchanged under rotations $x \to (I_N \otimes A)x$ where $A \in SO(3)$.

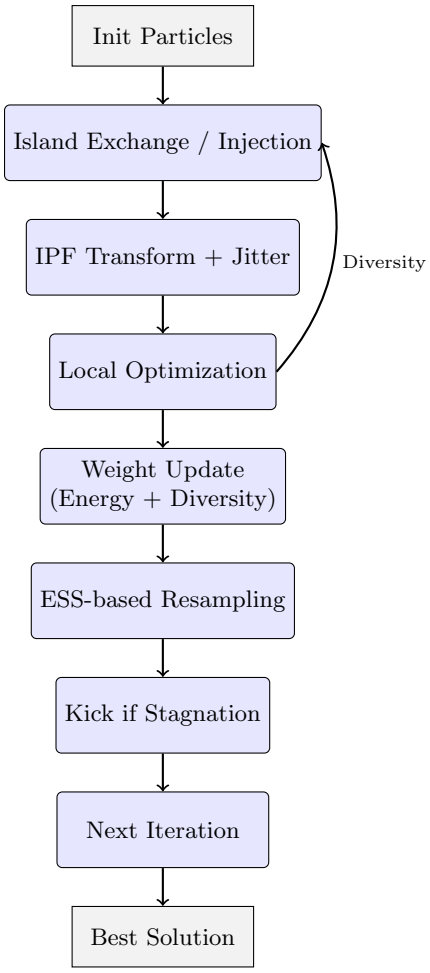

Figure 1: Data flow diagram of the hybrid particle filtering and local optimization framework. After initialization, periodic exchanges and IPF with jitter precede local refinement, diversity-aware weighting and ESS-based resampling. Kicks are applied upon stagnation; the best evaluated configuration is retained independently.

- **Permutational symmetry**: Swapping identical atoms does not change energy.

Clustering, sharing and novelty use Cartesian distances with the supplied atom ordering. The evaluated iteration does not apply principal-axis alignment or canonical atom relabeling after local optimization. The initial populations are frozen with their coordinate conventions. Thus the distance-based mechanisms can distinguish symmetry-equivalent coordinate representations.

3.3 Particle Perturbations and Advanced Mutations

The perturbation mechanisms balance local refinement with global exploration. In the evaluated configuration, the active proposal is cluster-wise IPF followed by Gaussian jitter; the alternative mutation branch is inactive.

3.3.1 Gaussian Perturbation

The active Gaussian perturbation is the box-relative jitter of Section 3.4, applied after transport, with an initial per-coordinate standard deviation of 0.2.

3.3.2 Differential Evolution Mutation

The alternative mutation branch includes $x_i + 0.8(x_a - x_b)$ when its initial uniform draw is below 0.4. This operator is not executed on the IPF branch used for the reported main experiment.

3.3.3 Crossover

The alternative branch also mixes a candidate with a fresh uniform configuration or an archive member. These mixtures are inactive on the evaluated IPF path; removing IPF in an ablation is distinguished from switching the complete proposal branch.

3.3.4 Boundary Handling

Perturbed particles are clipped to the feasible region:

$$x_k^{(i)\prime} \leftarrow \mathrm{clip}(x_k^{(i)\prime}, \mathrm{lb}, \mathrm{ub}) \tag{4}$$

where $\mathrm{lb} = -R \cdot \mathbf{1}_{3N}$ and $\mathrm{ub} = R \cdot \mathbf{1}_{3N}$.

3.4 Invertible Particle-Flow (IPF) Operator

The IPF operator addresses a fundamental challenge in particle filtering: weight collapse due to resampling can eliminate diversity. We introduce invertible transport transformations that gently move particles toward elite configurations while supporting exploration capacity through stochastic perturbations, diversity-aware weighting, and rejuvenation steps.

3.4.1 Motivation and Theoretical Foundation

Traditional particle filtering can suffer from particle degeneracy, where a few particles dominate the weight distribution [26]. Resampling addresses this but can reduce diversity. The IPF operator supports diversity by applying invertible transformations with nonsingular Jacobian, inspired by normalizing flows [23, 24]. In the linear variant used here, the mapping is contractive rather than volume-preserving; its role is to bias particles toward promising regions while practical diversity is supported through subsequent jitter, mutation, resampling, and rejuvenation steps.

3.4.2 Flow Options

We consider four invertible mapping families $\Phi_\alpha : \mathbb{R}^{3N} \to \mathbb{R}^{3N}$:

$$\begin{cases} \text{Linear:} & \Phi_\alpha(x) = (1-\alpha)x + \alpha x^\star \\ \text{Affine coupling [24]:} & y_a = x_a,\ y_b = x_b \odot e^{s(x_a)} + t(x_a) \\ \text{Residual:} & \Phi_\alpha(x) = x + \alpha \tanh(x^\star - x) \\ \text{ODE-flow [25]:} & \text{Solve } \frac{dy}{dt} = \alpha \tanh(x^\star - y),\ y(0) = x \end{cases} \tag{5}$$

where $x^\star$ is the lowest-energy member of the current particle's cluster. For affine coupling, $s(\cdot)$ and $t(\cdot)$ are scaling and translation networks. The evaluated linear mapping uses k-means with $K = \min(m_t, \max(2, \lfloor\sqrt{m_t}\rfloor))$ and the coefficient

$$\alpha_t = 0.98(1 - t/T) + 0.05, \qquad t = 0, \ldots, T-1, \quad T = 1000. \tag{6}$$

It starts at 1.03 and ends at 0.05098. Thus early steps slightly overshoot the elite. No discrete iteration in this schedule has $\alpha_t = 1$.

3.4.3 Choice of Flow and Justification

In the current implementation, we adopt the **linear flow** for the following reasons:

1. **Computational efficiency**: Linear flow requires only vector operations, making it extremely fast to compute even for large N.

2. **Guaranteed invertibility**: For a fixed elite, the Jacobian is $(1-\alpha)I$, so $|\det \nabla\Phi_\alpha| = |1-\alpha|^{3N} > 0$ for $\alpha \neq 1$. The map is bijective on $\mathbb{R}^{3N}$ and on a restricted domain onto its image, not generally onto the original domain. For the implemented coefficients, $|1 - \alpha_t| < 1$, so the transformation is contractive, not volume-preserving. This calculation does not establish invertibility of the full composition with clustering, clipping and resampling.

3. **Empirical assessment**: The contribution of the linear IPF is examined within the integrated framework in Section 4.2; no comparative success-rate estimate for the other mappings is established by the main experiment.

4. **Interpretability**: Linear transport relative to the cluster elite has clear geometric interpretation and tunable exploration-exploitation trade-off via α.

The more sophisticated flow options (affine coupling, residual, ODE-flow) may provide benefits for higher-dimensional problems or landscapes with more complex topology. We reserve their investigation for future work and acknowledge this as a limitation of the current study.

3.4.4 Application and Diversity Preservation

After applying the IPF transformation:

$$x_k^{(i)} \leftarrow \Phi_{\alpha_k}(x_k^{(i)}), \tag{7}$$

we add small isotropic Gaussian jitter to prevent exact collapse:

$$x_k^{(i)} \leftarrow x_k^{(i)} + \zeta\,(\mathrm{ub}_j - \mathrm{lb}_j)\cdot\nu, \quad \nu \sim \mathcal{N}(0, I_{3N}) \tag{8}$$

where $\zeta = 0.02$ is the jitter magnitude. Initially, $\mathrm{ub}_j - \mathrm{lb}_j = 10$, yielding a per-coordinate jitter standard deviation of 0.2; it decreases when the box contracts. Projection onto the current box is applied after transport and after jitter.

Remark 1. *The key property used here is invertibility with nonsingular Jacobian, not exact volume preservation. In the linear IPF used in this work, the map is contractive and therefore tends to concentrate particles toward elite regions. Diversity is preserved in practice not by Jacobian preservation, but by the interaction of IPF with Gaussian jitter, mutation, ESS-triggered resampling, particle injection, and kick-based rejuvenation. Accordingly, IPF should be interpreted as an optimization-oriented transport operator rather than as an exact measure-preserving flow.*

3.5 Weight Update and ESS-Based Resampling

3.5.1 Effective Objective Definition

We combine energy and diversity information into an effective objective, using the post-local population energies. For each particle i, the effective objective is:

$$\text{eff-objs}^{(i)} = E_{\mathrm{LJ}}(x^{(i)}) + \lambda_{\text{share}}\cdot\text{sharing}^{(i)} - \lambda_{\text{novelty}}\cdot\text{novelty}^{(i)}, \tag{9}$$

where the diversity metrics are:

Sharing metric: Penalizes particles in crowded regions

$$\text{sharing}^{(i)} = \sum_{j\neq i}\max\left(0, 1 - \frac{\|x^{(i)} - x^{(j)}\|}{\sigma_{\text{share}}}\right), \tag{10}$$

$$\sigma_{\text{share}} = 0.1\|\mathrm{ub}_0 - \mathrm{lb}_0\|_2 = \sqrt{3N}. \tag{11}$$

This metric increases for particles with many neighbors within radius σ_{share}, encouraging exploration of sparse regions. The coefficient is fixed at $\lambda_{\text{share}} = 0.01$. The implementation includes the self-contribution 1 in the sharing sum; because it is common to all particles, it cancels in the subsequent minimum shift.

Novelty metric: Rewards particles far from previously discovered solutions

$$\text{novelty}^{(i)} = \frac{1}{|\text{archive}|}\sum_{a\in\text{archive}}\|x^{(i)} - a\|, \quad \lambda_{\text{novelty}} = 0.01. \tag{12}$$

The archive stores the ten most recent strictly improving population-best configurations. The novelty term is zero while this archive is empty.

Distinction between sharing and novelty: While both metrics promote diversity, they serve different purposes:

- **Sharing** operates on the current population, preventing clustering within a single iteration
- **Novelty** compares against historical best solutions, preventing the population from repeatedly rediscovering the same local minima across iterations

The removal experiments in Section 4.2 assess these metrics within the integrated framework on LJ75 and LJ102; they do not establish their necessity on all instances.

3.5.2 Particle Weights and Normalization

For finite scores, the implemented weights are

$$z_i = \frac{F_i - \min_j F_j}{\max\{\mathrm{std}(F), 10^{-8}\}}, \quad a_i = \max\{10^{-8}, \exp(-\beta_t z_i)\}, \quad w_i = \frac{a_i}{\sum_j a_j}. \tag{13}$$

Here std uses the population divisor m_t, unlike the sample SD used in the result tables. The positive floor on a_i is part of the implementation. These are not exact Gibbs weights, even before considering score standardization and diversity terms. Nonfinite objectives are handled by a separate validity policy in the source.

3.5.3 Effective Sample Size and Resampling

The pre-resampling effective sample size is

$$e_t = \left(\sum_{i=1}^{m_t} w_i^2 \right)^{-1}. \tag{14}$$

For a nonterminal iteration, the implementation resamples when $e_t < 0.1P_0$. It uses *stratified* points $u_i = (i + v_i)/m_t$, $i = 0, \ldots, m_t - 1$, with independent $v_i \sim U(0, 1)$, followed by inversion of the cumulative weights. [29].

The resampled coordinates receive Gaussian jitter at 0.01 times the current box width. When the same pre-resampling e_t is below $0.05P_0$, $\lfloor 0.1P_0 \rfloor$ uniform points are appended and the population is sampled back to P_0 if needed. The stored weights and ESS refer to the population *before* these operations. They are not recomputed from uniform weights after a reset to decide severe degeneracy.

3.6 Adaptive Annealing

The inverse temperature β_k controls the selection pressure, with higher values focusing on exploitation. We use an adaptive schedule balancing ESS and iteration progress.

3.6.1 Initialization

The initial and maximum inverse temperatures are set based on problem characteristics:

$$\beta_{\min} = 10^{-4}, \tag{15}$$

$$\beta_{\max} = 5.0, \tag{16}$$

These values set weak initial selection pressure and a finite upper bound. The adaptation factor starts at $b_0 = 1$. The sensitivity of results to $\beta_{\min}$ is analyzed in Section 4.3.

3.6.2 Update Rule

For $p_t = t/(T-1)$ and the pre-resampling e_t, the update is

$$a_t^{\mathrm{ESS}} = \mathrm{clip}(e_t/(0.5P_0), 0.1, 1), \tag{17}$$

$$\beta_{t+1} = \mathrm{clip}\left(\beta_t + b_t(\beta_{\max} - \beta_t)p_t a_t^{\mathrm{ESS}},\ \beta_0(1 - 0.5p_t),\ \beta_{\max}\right), \tag{18}$$

$$b_{t+1} = \begin{cases} \min(1.1b_t, 5), & e_t < 0.05P_0, \\ \max(0.95b_t, 1), & \text{otherwise.} \end{cases} \tag{19}$$

The factor is updated after computing β_{t+1}. At a target-reaching iteration, the nonterminal update is skipped, so the logged next temperature can equal the used temperature.

For fixed scores, increasing β increases selection pressure. It is not by itself a proof of protection against weight degeneracy. The finite cap also excludes the divergent-temperature assumption used for the idealized target family in Section 3.12.

3.7 Local Optimization

Local optimization refines promising particles using gradient-based methods, accelerating convergence once good basins are identified.

3.7.1 Selection of Particles for Refinement

At each iteration, all current particles are selected for local optimization:

$$k_{\text{ref}} = m_t. \tag{20}$$

After a non-target exit, up to five current candidates receive a final refinement when permitted by the stopping limits. These evaluations count toward the total cost.

3.7.2 IPOPT Configuration

We use IPOPT (Interior Point OPTimizer) [27] with the following configuration:

- **Algorithm**: Interior point with exact Hessians computed from the LJ potential
- **Linear solver**: MUMPS for sparse symmetric systems
- **Convergence tolerance**: IPOPT option `tol=1e-8`, with zero bound relaxation
- **Maximum iterations**: 5000
- **Derivative test**: Disabled (known analytic gradients/Hessians)

The following formulas describe the unregularized LJ energy gradient and Hessian. The implementation differentiates the regularized potential in Eq. (33) analytically. Solver status and the best admissible evaluated point are recorded separately.

The LJ energy gradient and Hessian are:

$$\frac{\partial E}{\partial x_i} = \sum_{j \neq i} \frac{\partial V(r_{ij})}{\partial r_{ij}} \frac{\partial r_{ij}}{\partial x_i}, \tag{21}$$

$$\frac{\partial^2 E}{\partial x_i \partial x_j} = \frac{\partial^2 V(r_{ij})}{\partial r_{ij}^2} \frac{\partial r_{ij}}{\partial x_i} \frac{\partial r_{ij}}{\partial x_j} + \frac{\partial V(r_{ij})}{\partial r_{ij}} \frac{\partial^2 r_{ij}}{\partial x_i \partial x_j}, \tag{22}$$

where the LJ potential derivatives are:

$$\frac{\partial V}{\partial r} = 4\epsilon \left[-12 \frac{\sigma^{12}}{r^{13}} + 6 \frac{\sigma^6}{r^7} \right], \tag{23}$$

$$\frac{\partial^2 V}{\partial r^2} = 4\epsilon \left[156 \frac{\sigma^{12}}{r^{14}} - 42 \frac{\sigma^6}{r^8} \right]. \tag{24}$$

3.7.3 Constraints

We apply hard per-atom spherical constraints in the local subproblem. Denoting by $\mathbf{r}_i = (x_i, y_i, z_i)$ the position of atom i, we require

$$\|\mathbf{r}_i\|_2^2 \leq R_t^2, \qquad i = 1, \ldots, N, \tag{25}$$

in addition to the current box bounds. IPOPT receives these constraints, their gradients $2\mathbf{r}_i$, and their Hessian contributions $2\lambda_i I_3$. Outer proposals need not initially satisfy the balls. Target attainment is evaluated using unpenalized E_τ, not a penalty objective.

3.7.4 Local Solver Return Policy

IPOPT, native Newton and projected L-BFGS are available as alternative local backends. The experiments use IPOPT without automatic fallback to another backend. The local return policy can retain a lower-energy admissible evaluated trial even when the solver status is unsuccessful or limited; status and incumbent usefulness are separate records.

For the BH-OJ and GA implementations, the local method is SciPy L-BFGS-B with bounds and a scaled, guarded stopping policy [28]. Objective and gradient are scaled consistently within each solve. Budgeted recovery attempts follow exits with $\|x - \Pi_\Omega(x - \nabla E_\tau(x))\|_\infty > 10^{-3}$; the strict stationarity check remains at 10^{-10}. The recorded nominal tolerances are $f_{\text{tol}} = 10^{-12}$ and $g_{\text{tol}} = 10^{-10}$; an exit on functional tolerance is not labeled as verified stationarity. Local trial energies contribute to the same run-level objective counter and best-point record. The numerical diagnostics are reported in the supplementary material.

3.8 Kick Mechanism for Stagnation

Stagnation occurs when the best energy remains unchanged for multiple iterations, indicating the population may be trapped in a local minimum basin.

3.8.1 Stagnation Detection

We declare stagnation if:

$$|E_{\text{best}}^{(k)} - E_{\text{best}}^{(k-1)}| < \epsilon_{\text{tol}} \tag{26}$$

for $\epsilon_{\text{tol}} = 10^{-6}$. A running counter Δ_{stagn} is incremented by one each iteration without improvement; when $\Delta_{\text{stagn}} \geq 10$ (i.e., 10 consecutive iterations without improvement), a kick is triggered and the counter is reset to zero. The counter is also reset upon any energy improvement (Section 3.8.4).

3.8.2 Kick Perturbation

The kick factor is initialized to 10. After a non-improving comparison it increases by 0.5; an improving comparison resets it to 10.5. At $c \geq 10$, it is updated once more as

$$K \leftarrow \min\{100, K + 0.5c + 0.01d\}. \tag{27}$$

Each coordinate then receives

$$x_j \leftarrow [\Pi_t(x + 0.38K\eta)]_j, \qquad \eta_j \sim U(-2, 2). \tag{28}$$

The counter resets after the kick, but the factor is not reset until an improving comparison. The counter and amplitude therefore have distinct reset rules.

3.8.3 Container Adaptation

Before the kick displacement is projected, the half-width is contracted:

$$R \leftarrow \max\{0.98R, 3\}. \tag{29}$$

The radius remains between 3 and its initial value 5. The compact set used for the theoretical analysis is specified independently in Section 3.12.

3.8.4 Reset

A kick alters the population, not the external incumbent. The next local solves use the updated radius. The active local radius reported in a result and the last population radius need not refer to the same moment if a terminal kick preceded finalization; the timestamped history and options disambiguate them.

3.9 Island Exchange

Island exchange promotes diversity and information sharing across subpopulations.

3.9.1 Partitioning

For $t > 0$ divisible by ten, the population is partitioned into three groups by index modulo three. The best evaluated member of each nonempty group is appended to the population. A separate random stream then samples exactly P_0 members, using replacement only when necessary.

3.9.2 Cluster-Based Injection

For $t > 0$ divisible by twenty, k-means with three groups and fixed seed 42 is applied to the Cartesian population. The best existing member of each nonempty cluster is appended. The injected representatives are existing cluster members rather than geometric centroids. These representatives subsequently enter the same IPF, objective-accounting and local-refinement pipeline as other candidates.

3.10 Diversity Metrics and Parameter Tuning

Beyond sharing and novelty (Section 3.5), we track additional diversity indicators:

3.10.1 Population Spread

The active population-wide spread statistic is the mean of all unordered pairwise Euclidean distances,

$$D(P) = \frac{2}{m(m-1)} \sum_{i<j} \|x_i - x_j\|_2. \tag{30}$$

The threshold is $0.05\|u_0 - \ell_0\|_2 = 0.5\sqrt{3N}$. If spread is smaller, $\lfloor 0.2P_0 \rfloor$ fresh uniform configurations are appended using the injection random stream. The population can therefore temporarily exceed its nominal scheduled size.

3.10.2 Dynamic Parameter Adaptation

The coupled changes are the population-size schedule, the temperature updates, diversity injection and the kick/radius rules above. On the linear-IPF path, the active proposal noise is the box-relative jitter. The mutation-strength parameter of the alternative branch is inactive.

3.11 Integrated Algorithm

Algorithm 1 outlines the hybrid iteration, incorporating IPF, local refinement, diversity-driven weights, resampling, island exchange and adaptive kicks. Counter and best-point updates take place at evaluations, including evaluations inside local-solver callbacks. Stored objectives and weights are aligned with the post-local population before resampling; subsequent injections are reflected in the next iteration, not retrospectively inserted into that history row.

3.12 Surrogate Asymptotic Convergence Analysis

We state two independent idealized results. Their role is to motivate exploration and annealed selection, not to prove convergence of the implemented particle process. In particular, the actual weight floor, standardized diversity scores, memory, changing population, local constraints and capped temperatures are not covered by an energy-only target theorem.

Let $D = \{x \in [-R_{\max}, R_{\max}]^{3N} : r_{ij} \geq \delta \text{ for all } i < j\}$ be nonempty, with $R_{\max} > 0$ and $\delta > 0$. This is an idealized compact domain, not an assertion that the executed candidates satisfy a hard separation. Let ν be a Borel probability measure with full topological support on D. Ordinary box clipping does not enforce the separation condition.

Assumption 1. *The energy $E : D \to \mathbb{R}$ is continuous. Define $E_D^* = \min_{x \in D} E(x)$ and $X_D^* = \{x \in D : E(x) = E_D^*\}$.*

Both the unregularized pair energy away from collisions and E_τ restricted to D satisfy this continuity hypothesis. The latter is also continuous at collisions because $\tau > 0$, but its numerical regularization is not a hard geometric separation. The minimum in the proposition is a minimum over D, not a certificate for an unrestricted physical cluster problem.

Algorithm 1 Hybrid Particle-Filter/Local-Solver Algorithm with IPF, Adaptive Kick, and Diversity Metrics

Load the frozen P_0-particle population; initialize PF and auxiliary streams.
Evaluate initial candidates; initialize external incumbent and counters.
for $t = 0, \ldots, T-1$ **do**
  Select the scheduled population size without replacement when shrinking.
  Apply modulo-island exchange if $t > 0$ and $t \bmod 10 = 0$.
  Append best three-cluster members if $t > 0$ and $t \bmod 20 = 0$.
  Cluster-wise linear IPF, project, jitter and project.
  Evaluate candidates; refine all current particles with IPOPT/MUMPS.
  Update population-best archive and compute weights from post-local energies.
  Store objectives, weights, pre-resampling ESS and incumbent.
  if target not reached **then**
    Stratified resampling and severe-degeneracy injection when triggered.
    Update β using the stored pre-resampling ESS.
    Update stagnation counter and adaptation factor; inject if spread is low.
    If stagnating, update kick factor, contract radius, kick and reset counter.
  end if
  Store post-operation metadata and cumulative counters.
  if target reached or a stopping limit is encountered **then**
    Terminate with the recorded reason.
  end if
end for
Apply final top-five refinement after a non-target exit when permitted.
Return retained evaluated incumbent; audit energy and record final coordinates.

Assumption 2. *For a fixed population $z \in D^m$, an idealized exploration proposal draws the m configurations conditionally independently with complete kernels $q_i(z, \cdot)$ on D. For every nonempty relatively open $U \subseteq D$, $q_i(z, U) > 0$. Define $Q(z, \cdot) = \bigotimes_{i=1}^{m} q_i(z, \cdot)$.*

The q_i are the entire kernels in this definition, not Gaussian perturbations followed by unspecified transformations. A concrete example is a positive density proportional to $\exp[-\|y - z_i\|^2/(2\sigma^2)]$ relative to $\nu(dy)$, with $\sigma > 0$. This is a theoretical construction supported on D, not a claim that clipping implements it. Full support concerns relatively open sets; for a density with respect to Lebesgue measure, a prescribed singleton has probability zero.

Proposition 1 (Neighborhood accessibility)**.** *Under Assumption 2, $Q(z, U) > 0$ for every $z \in D^m$ and every nonempty relatively open $U \subseteq D^m$.*

This statement establishes one-step accessibility for Q alone. It is not an irreducibility, aperiodicity, recurrence or invariant-distribution statement for the composed HPF transition.

Assumption 3. *For the idealized target sequence, $\beta_k \geq 0$ and $\beta_k \to \infty$. This is a separate hypothesis, not a consequence of the executed capped update* (18).

Define

$$\pi_\beta(A) = \frac{\int_{A \cap D} e^{-\beta E(x)} \nu(dx)}{\int_D e^{-\beta E(y)} \nu(dy)}. \tag{31}$$

Normalized Lebesgue measure is a possible choice only when it has positive mass and support equal to D. These properties are not inferred without justification for an arbitrary constrained or symmetry-reduced set.

Proposition 2 (Concentration of energy-only target measures)**.** *Under Assumption 1, for every relatively open neighborhood U of X_D^*, $\pi_\beta(U) \to 1$ as $\beta \to \infty$.*

This is concentration on the minimizer *set*, not necessarily convergence to a Dirac mass at a unique coordinate vector. Appendix A supplies independent proofs of the two propositions.

Remark 2 (Interpretation and absence of a convergence implication)**.** *Propositions 1 and 2 concern different mathematical objects. They do not imply that a particle-state law ρ_k tracks π_{β_k}. With $\|\rho - \pi\|_{\mathrm{TV}} = \sup_A |\rho(A) - \pi(A)|$,*

$$\rho_k(U^c) \leq \|\rho_k - \pi_{\beta_k}\|_{\mathrm{TV}} + \pi_{\beta_k}(U^c). \tag{32}$$

Only the second term is controlled by Proposition 2. No result here controls the first, or transfers concentration to the implemented adaptive algorithm.

The practical process depends on archives, counters, schedules and random-generator states; a Markov description on particle coordinates alone is not presumed. Invertibility of one affine step does not repair this missing tracking argument. Neither do repeated local solves. The practical algorithm uses bounded temperatures and does not explicitly enforce δ-separation. Its performance is assessed by the finite experiments in Section 4.

3.13 Computational Complexity

Each iteration incurs:

- $O(m_t N^2)$ for energy evaluations of all particles (pairwise interactions);
- $O(m_t \cdot 3N)$ for linear IPF transformations once elites are selected;
- $O(m_t^2 \cdot 3N)$ for sharing and population spread, and $O(m_t|\mathcal{A}_t| \cdot 3N)$ for novelty;
- $O(m_t)$ for cumulative weights and ordered resampling positions;
- $k_{\mathrm{ref}} C_{\mathrm{IP}}$ for local refinement, where $k_{\mathrm{ref}} = m_t$ in the evaluated configuration;
- $O(I_K m_t \cdot 3N \cdot K)$ for k-means with K clusters and I_K iterations.

The IPF operator and kick episodes add minor overhead, and island exchange occurs periodically. Local refinement cost includes objective and derivative evaluations, factorizations and linear solves. Its elapsed time also depends on parallel execution; counting energy calls alone does not account for all these operations.

3.14 Implementation Details

The framework is implemented in C++ with IEEE754 binary64 arithmetic, Eigen 3.4.0, IPOPT 3.14.20 and OpenMP. Local optimization uses MUMPS and exact analytic derivatives, as described in Section 3.7. The baselines use Python 3.12.13, NumPy 2.3.5 and SciPy 1.17.0, with L-BFGS-B for the local refinement in BH-OJ and GA. BH-OJ and GA use baseline implementation 3.1.0, whose local policy records the projected-gradient residual of the returned point rather than the solver's convergence flag; PSO, CLPSO and SMC do not call that routine and use 3.0.0. All five optimize the objective of Eq. (33), so HPF and the baselines minimize the same function.

In reduced units where the pair-equilibrium distance is one, $y = r/(2^{1/6}\sigma)$ and $V/\epsilon = y^{-12} - 2y^{-6}$. All numerical experiments use the same squared-distance regularization:

$$E_\tau(x) = \sum_{i<j} (q_{ij}^6 - 2q_{ij}^3), \qquad q_{ij} = \frac{1}{\|r_i - r_j\|_2^2 + \tau}, \qquad \tau = 10^{-8}. \tag{33}$$

At finite τ, this differs from the singular pair potential introduced in Section 1. Reference-energy agreement is assessed at the stated tolerance, with this numerical convention explicit.

All experiments were conducted on a workstation equipped with an AMD Ryzen Threadripper 7960X CPU (24 physical cores, 48 logical processors) and 256 GB RAM. HPF uses eight local process workers and ten threads for population evaluations, with one BLAS/OpenMP thread within each local worker. The five baselines run on CPU with one BLAS/OpenMP thread per run. No GPU is used in these experiments.

Wall-clock times reflect the tested CPU implementations and their differing resource allocations, rather than equal computational resources or hardware-neutral speed rankings. Objective-evaluation counts offer a complementary cost measure, although derivative and factorization costs differ. Complete algorithmic parameters are given in Table 1.

Table 1: Complete Algorithmic Parameters. m denotes the current population size; $t = 0, \ldots, T-1$.

Parameter	Value
Initial population P_0	$3N + 56$ independent box-uniform configurations per run
Population schedule	$\max(5, \lfloor P_0(1 - t/T) \rfloor)$ before exchange
Outer iteration limit T	1,000; target and elapsed-time stops also active
Elapsed-time limit	36,000 s per run; no energy-evaluation cap
Flow / clustering	Linear, cluster-wise IPF / k-means
Flow groups	$\min(m, \max(2, \lfloor\sqrt{m}\rfloor))$
IPF coefficient schedule	$\alpha_t = 0.98(1 - t/1000) + 0.05$
IPF jitter	0.02 times the current coordinate-wise box width
Initial / minimum box half-width	5 / 3; multiply by 0.98 at each kick
Local solver	IPOPT 3.14.20; MUMPS; exact Hessian
Candidates refined	All current particles (`refine_all=true`)
Final refinement	Up to five current candidates after a non-target exit
Local iteration limit / tolerance	5,000 / `tol=1e-8`
Local geometric constraints	Hard per-atom balls $\|r_i\|_2^2 \le R_t^2$
Local execution	8 persistent process workers
Population evaluation	10 OpenMP threads; binary64 CPU arithmetic
Initial / maximum β	10^{-4} / 5
Sharing / novelty coefficients	0.01 / 0.01
Sharing radius	$0.1\|u_0 - \ell_0\|_2 = \sqrt{3N}$
Improvement archive	10 most recent improving configurations
Resampling	Stratified; trigger ESS $< 0.1P_0$
Severe-degeneracy injection	$\lfloor 0.1P_0 \rfloor$ at ESS $< 0.05P_0$
Diversity injection	$\lfloor 0.2P_0 \rfloor$ appended when mean distance $< 0.5\sqrt{3N}$
Stagnation tolerance / trigger	10^{-6} / ten non-improving iterations
Kick cap / displacement multiplier	100 / 0.38
Islands / cluster representatives	Every ten / twenty iterations after $t = 0$
Base seed / run indices	2024 / 1–30; distinct population and PF seeds
Target criterion	$E_\tau \le E_{\mathrm{ref}} + 10^{-4}$
Pair regularization	$\tau = 10^{-8}$ added to squared distance

Parameters for the evaluated HPF configuration. The alternative mutation branch and local backends are inactive.

3.15 Experimental Setup

We evaluate the framework on LJ cluster sizes $N \in \{38, 75, 76, 77, 98, 102, 103, 104\}$, selected for their competing energy funnels and non-icosahedral reference structures. In the main benchmark, we perform 30 runs per cluster size, indexed by $r = 1, \ldots, 30$. Each HPF run starts from a distinct population of $3N + 56$ configurations, with every coordinate drawn uniformly from $[-5, 5]^{3N}$ and the initialization and search seeds recorded. A separate HPF calibration using one fixed initial population per size is documented as a secondary comparison in Supplement S3; its observations are not pooled with the main benchmark.

For a base seed $s_0 = 2024$, cluster size N, run index r and offset o, the seed is the unsigned 32-bit integer obtained from the first eight hexadecimal digits of

$$\mathrm{MD5}(\texttt{s0_N_r_o}). \tag{34}$$

The PF stream uses $o = 1000$ and the initialization stream uses $o = 0$. Initial coordinates use the PCG64/SeedSequence convention. Both seeds and initial-population hashes are recorded in the supplementary data.

HPF stops when the reference target is reached, after 1,000 outer iterations, or after 36,000 elapsed seconds. No objective-call cap is imposed on HPF. The operational test is $E_\tau \leq E_{\mathrm{ref}} + 10^{-4}$; all reported successful runs also satisfy $|E_\tau - E_{\mathrm{ref}}| \leq 10^{-4}$, where E_{ref} is the rounded LJ reference energy [1].

Baselines include:

1. **Basin-Hopping with Occasional Jumping (BH-OJ)** [5]: Metropolis-based basin-hopping with temperature 1, Gaussian step size 0.5, stagnation trigger after 60 iterations, forty unrelaxed jumps of scale 3.5, radial contraction 0.85, and L-BFGS-B refinement. Local iteration limits are 4,000, or 1,200 after jumps.

2. **Genetic Algorithm (GA)**: real-coded GA with SBX crossover ($\eta_c = 15$, probability 0.7), polynomial mutation ($\eta_m = 20$, per-coordinate probability 0.1), tournament selection of size 3, 5% elitism, population size 300, and L-BFGS-B refinement of elites every 25 generations, with at most 5,000 local iterations.

3. **Particle Swarm Optimization (PSO)**: global-best PSO with $w = 0.7298$, $c_1 = c_2 = 1.49618$, and $m = \min(300, 3N + 56)$ particles.

4. **Comprehensive Learning PSO (CLPSO)** [19]: per-dimension exemplar learning with $c = 1.49445$, refresh gap 7, learning probabilities 0.05–0.50 and inertia decreasing from 0.9 to 0.4 with the fraction of evaluation budget consumed; $m = \min(300, 3N + 56)$.

5. **SMC-based optimizer**: tempered resample–move SMC beginning at $\beta = 0$, with ESS target $0.7m$, systematic resampling and two reflected Gaussian Metropolis sweeps per level. The maximum inverse temperature is $5 \log(1 + N)$, the initial proposal scale is 0.6, and $m = \min(300, 3N + 56)$.

The baselines also draw initial coordinates uniformly from $[-5, 5]^{3N}$ without centering; the individual coordinates are not asserted to be the same as HPF's. Baseline L-BFGS-B uses box constraints, whereas HPF also imposes per-atom ball constraints. The comparison therefore evaluates the stated implementations, including their initialization and local solvers, rather than isolating their global-search moves.

For each cluster size, the executed baseline evaluation cap was fixed from a separate common-population HPF calibration and is unchanged (Table 2). Local objective evaluations count toward each cap, with each evaluated configuration counted separately. Baselines also have a 36,000-second limit per run. These limits do not imply identical realized evaluations, minimization effort or CPU resources.

Performance metrics are:

- Success rate: fraction of runs satisfying $|E_\tau - E_{\mathrm{ref}}| \leq 10^{-4}$;
- Best energy reached across 30 runs;
- Mean energy and sample standard deviation;
- Actual energy-evaluation count and wall-clock time in seconds.

Energy and timing summaries include all runs, including time-limited outcomes, with the reason for stopping reported separately. These comparisons concern the specific implementations and parameterizations tested under this protocol, not the corresponding algorithmic families in general.

Table 2: HPF objective-evaluation counts from 30 distinct starting populations per cluster size and the fixed baseline budgets.

N	HPF max evals	HPF mean evals	Baseline budget
38	230 981	178 925	973 472
75	19 852 440	6 573 502	27 857 692
76	19 845 989	9 089 272	38 146 196
77	28 816 794	15 310 801	57 259 400
98	40 292 501	25 716 020	82 633 222
102	4 542 882	1 143 638	9 085 764
103	6 016 238	1 232 786	9 583 880
104	9 331 737	4 108 697	19 340 034

HPF means and maxima include all 30 outcomes and both population and local objective calls. Baseline limits derive from a separate HPF calibration and have not been changed; they are not generally twice the HPF maximum shown here. Each baseline also has a 36,000-second limit.

4 Results

4.1 Main Results

We evaluated the hybrid particle filtering algorithm on the challenging Lennard-Jones (LJ) cluster sizes $N \in \{38, 75, 76, 77, 98, 102, 103, 104\}$, known for their multi-funnel energy landscapes and non-icosahedral global minima [3]. To account for the stochastic nature of the search, we performed 30 runs for each cluster size. For HPF, the 30 runs per size use 30 independently generated box-uniform initial populations and separate search seeds, as specified in Section 3.15. The initial container radius is $R_0 = 5.0$, with contraction toward 3 at kicks (Section 3.8).

To ensure reproducibility, the base seed is $s_0 = 2024$, the run index is $1, \ldots, 30$, and the first eight hexadecimal digits of the MD5 hash of `s0_N_r_offset` are used as an unsigned 32-bit seed, without reduction modulo 2^{31}. The PF offset is 1000. The initialization metadata, streams and population fingerprints accompany the results. The baseline and HPF populations follow the same box-uniform family but are not assumed to share coordinates.

Results are reported as means ± sample standard deviations over the 30 runs, compared to reference energies from the Cambridge Cluster Database [1] and baseline methods. Table 3 compares HPF's best energies against the reference targets using the numerical convention of Eq. (33). In every successful

Table 3: Comparison with reference energies from the Cambridge Cluster Database. Columns show the reference energy, HPF's best energy across 30 distinct initial populations, its absolute gap, and the success rate at tolerance 10^{-4}.

N	Cambridge DB	HPF Best	$\lvert\Delta E\rvert$	Success
38	−173.928427	−173.928427167	1.67×10^{-7}	100%
75	−397.492331	−397.492332572	1.57×10^{-6}	96.7%
76	−402.894866	−402.894867631	1.63×10^{-6}	86.7%
77	−409.083517	−409.083518768	1.77×10^{-6}	56.7%
98	−543.665361	−543.665363134	2.13×10^{-6}	20.0%
102	−569.363653	−569.363654967	1.97×10^{-6}	100%
103	−575.766131	−575.766133373	2.37×10^{-6}	100%
104	−582.086642	−582.086644595	2.60×10^{-6}	100%

HPF energies use Eq. (33). All successful final configurations agree with their target within 10^{-4}; reference agreement is not a global-optimality certificate.

run, the final energy matches the reference value to within 10^{-4}. This is a numerical target-agreement criterion, not a new global-optimality certificate.

Table 4 summarizes the performance comparison. HPF is the only tested method to achieve nonzero success rates on all eight instances under the stated protocol, including LJ77 and LJ98. These statements concern the specific baseline implementations and parameterizations tested, not the corresponding algorithmic families in general.

Key findings include: (1) On LJ38, HPF achieves 100% target attainment and BH-OJ 10%, while the other baselines achieve none. (2) On LJ75 and LJ76, HPF achieves 96.7% and 86.7%, respectively, while

Table 4: Performance Comparison on Challenging LJ Clusters (30 runs per method). Success rates (%) and best energies from the distinct-population HPF experiment and the fixed baseline protocols.

N	Known	Success Rate (%) / Best Energy Found					
	Minimum	HPF	BH-OJ	GA	PSO	CLPSO	SMC
38	−173.928427	100 / −173.93	10 / −173.93	0 / −170.65	0 / −104.48	0 / −33.64	0 / −22.64
75	−397.492331	96.7 / −397.49	0 / −396.19	0 / −386.83	0 / −200.36	0 / −56.99	0 / −39.48
76	−402.894866	86.7 / −402.89	0 / −402.38	0 / −392.18	0 / −208.96	0 / −59.46	0 / −38.07
77	−409.083517	56.7 / −409.08	0 / −408.49	0 / −398.99	0 / −205.58	0 / −62.84	0 / −41.65
98	−543.665361	20.0 / −543.67	0† / −543.03	0 / −534.50	0 / −272.04	0 / −71.20	0 / −51.22
102	−569.363653	100 / −569.36	0 / −567.26	0 / −551.76	0 / −261.14	0 / −56.84	0 / −51.09
103	−575.766131	100 / −575.77	0 / −573.67	0 / −560.88	0 / −272.99	0 / −54.56	0 / −55.00
104	−582.086642	100 / −582.09	0 / −580.97	0 / −568.71	0 / −284.04	0 / −62.81	0 / −53.06

†: all BH-OJ/LJ98 runs ended at 36,000 seconds, consuming 62.50% of their evaluation cap on average. Other unsuccessful baseline runs exhausted their cap. Percentages are rounded; full-precision energies and counts are supplied in the data.

Table 5: Mean final energy and sample standard deviation from 30 distinct-population HPF runs and 30 runs per baseline under their respective stopping limits.

N	Mean Energy ± SD						Fn Eval
	HPF	BH-OJ	GA	PSO	CLPSO	SMC	Budget
38	$-173.928427085 \pm 4.30 \times 10^{-7}$	-172.78 ± 0.66	-167.61 ± 1.25	-70.78 ± 10.76	-27.52 ± 1.94	-18.85 ± 1.15	973 472
75	$-397.492327965 \pm 2.52 \times 10^{-5}$	-395.35 ± 0.69	-380.79 ± 3.00	-177.43 ± 11.79	-53.31 ± 1.20	-36.11 ± 1.31	27 857 692
76	-402.826829358 ± 0.176	-401.53 ± 0.45	-386.91 ± 2.31	-184.84 ± 13.12	-55.79 ± 1.63	-36.83 ± 0.89	38 146 196
77	-408.833557034 ± 0.291	-407.53 ± 0.52	-392.41 ± 2.53	-186.33 ± 11.63	-58.83 ± 1.95	-37.90 ± 1.26	57 259 400
98	$-543.647440354 \pm 9.11 \times 10^{-3}$	-539.51 ± 1.37	-519.78 ± 3.94	-237.09 ± 17.34	-63.62 ± 2.07	-48.26 ± 1.38	82 633 222
102	$-569.363653118 \pm 1.01 \times 10^{-5}$	-563.37 ± 1.65	-545.61 ± 3.44	-215.15 ± 16.49	-51.78 ± 1.88	-47.93 ± 1.12	9 085 764
103	$-575.766133240 \pm 7.17 \times 10^{-7}$	-569.84 ± 1.56	-550.93 ± 4.30	-223.60 ± 18.21	-52.33 ± 1.15	-48.71 ± 1.84	9 583 880
104	$-582.086644574 \pm 7.16 \times 10^{-8}$	-576.78 ± 1.42	-557.76 ± 4.28	-237.70 ± 18.82	-56.97 ± 2.35	-50.25 ± 1.36	19 340 034

All runs, including unsuccessful and time-limited outcomes, contribute to the means and sample standard deviations. HPF values are the all-run moments of Table 11; the budget column applies to baselines only. Small HPF standard deviations are shown in scientific notation.

all five baselines remain at 0%. (3) On LJ77 and LJ98, HPF reaches the targets in 56.7% and 20.0% of runs, respectively; the five baseline implementations reach neither target. (4) On LJ102–104, HPF achieves 100% and no baseline reaches the targets. The calibrated evaluation caps, different local solvers and resource allocations prevent treating these outcomes as a controlled ranking of algorithmic families.

The auxiliary fixed-population experiment provides complementary evidence about initialization sensitivity (Supplement S3). With one initial population reused for 30 different search seeds per size, HPF attains 208/240 targets, compared with 198/240 when a new population is drawn for each run. On LJ98, success falls from 13/30 to 6/30; the unsuccessful runs in both series repeatedly reach an energy approximately 0.022401 above the reference. On LJ77, both protocols achieve 17/30 successes, but only nine seed-index-matched pairs succeed under both protocols, with eight successes exclusive to each. Thus, equal aggregate rates can conceal substantial changes in the individual outcomes. The fixed-population sample measures search variability conditional on one chosen population, while the main experiment varies initialization and search together; these observations do not isolate a causal initialization effect and are not pooled.

Table 6 reports mean wall-clock times for the five baselines under the evaluation-budget protocol. All unsuccessful baseline runs exhaust their evaluation caps except the thirty BH-OJ/LJ98 runs, which

Table 6: Computational Cost Comparison: Mean Wall-Clock Time ± SD (seconds) under the Evaluation-Budget Protocol (30 runs).

N	Budget	BH-OJ	GA	PSO	CLPSO	SMC
38	973 472	120.2 ± 28.2	61.5 ± 0.5	31.6 ± 0.2	44.0 ± 0.3	46.2 ± 0.3
75	27 857 692	12062.9 ± 259.4	5741.9 ± 56.4	4832.2 ± 101.7	5346.4 ± 84.2	5159.7 ± 60.7
76	38 146 196	16636.6 ± 223.9	8021.3 ± 87.0	6751.2 ± 66.9	7463.3 ± 78.0	7224.9 ± 80.6
77	57 259 400	25560.6 ± 267.0	12427.7 ± 156.3	10593.4 ± 251.2	11574.7 ± 161.4	11186.3 ± 119.5
98	82 633 222	36000.0 ± 0.0	29560.1 ± 371.9	26269.4 ± 385.6	27945.1 ± 418.6	27013.2 ± 377.8
102	9 085 764	6326.6 ± 208.6	3240.7 ± 133.6	2554.3 ± 27.7	2777.8 ± 29.0	2964.8 ± 103.1
103	9 583 880	7152.7 ± 119.9	3693.9 ± 44.5	3302.2 ± 30.0	3484.3 ± 35.9	3386.9 ± 39.1
104	19 340 034	15085.4 ± 822.6	7698.8 ± 172.4	6805.8 ± 197.3	7306.2 ± 161.8	7074.4 ± 120.4

All 30 BH-OJ/LJ98 runs reached the 36,000-second limit, not their evaluation caps. Timings include non-successful runs and were measured with different solver resources and CPU load; they are not hardware-neutral or matched-resource time-to-target comparisons.

stop at 36,000 seconds. Those runs consume 51,643,503 energy evaluations on average, or 62.50% of the nominal cap of 82,633,222. Their outcomes are included in all summaries and remain identified as time-limited, not as full-budget failures. Elapsed times reflect the recorded CPU implementations rather than matched-resource timing.

Table 5 reports HPF and baseline means and sample standard deviations side by side; HPF values use all 30 runs, as in Table 11. The baseline energies in Tables 4 and 5 do not all lie within a few percent of the reference energy. On LJ98, the best values are approximately -543.034 for BH-OJ, -534.499 for GA, -272.045 for PSO, -71.197 for CLPSO and -51.222 for SMC. Energy–coordinate consistency was checked independently, but this does not establish local stationarity or explain the performance differences. The tested methods differ in local refinement and initialization as well as in global-search moves. In BH-OJ, 1.636% of local calls record a large unresolved projected-gradient residual; a functional-tolerance exit is not automatically a stationarity certificate. The diagnostics accompany the reproducibility materials.

These results are consistent with the exploration-exploitation rationale of the method. The two idealized results of Section 3.12 motivate exploration and annealed selection separately, but provide no convergence guarantee for the capped, adaptive, archive-dependent implementation, which does not explicitly impose the theoretical separation condition. Evidence for the implemented algorithm is empirical.

4.2 Ablation Studies

To measure the effect of removing individual algorithmic mechanisms from the integrated framework, we consider LJ75 and LJ102 (30 paired runs per configuration), together with two reduced configurations. Results appear in Table 7. For each instance and run index, variants share the same initial population and PF seed; different run indices use different populations and seeds. All variants have limits of 1,000 PF iterations and 36,000 seconds, with no objective-evaluation cap. These common stopping limits do not imply equal evaluation counts. All 600 ablation runs terminate at the target or iteration limit.

Table 7: Ablation Study: Success Rates (%) When Removing Components (30 runs per instance; 1,000-iteration and 36,000-second limits)

Configuration	**LJ75**	**LJ102**
Full HPF method	100	100
- IPF operator	100	100
- Sharing metric	96.7	100
- Novelty metric	100	100
- Kick mechanism	36.7	100
- Island exchange	96.7	100
- Local optimization	0	0
- Adaptive annealing	100	100
Only PF + local opt (reduced controls)	10	100
Only PF (no local, reduced controls)	0	0

The reference configuration refines all current particles with IPOPT/MUMPS during the PF loop and retains terminal top-five refinement when needed. Both local stages are disabled in the no-local variants. The study uses $\alpha_t = 0.98 + (0.05 - 0.98)t/(T-1)$, $t = 0, \ldots, T-1$, with $T = 1000$, rather than the shifted coefficient of Eq. (6); its control and the main-experiment samples are not pooled. Removing IPF disables the transport map without replacing its clustering and jitter branch. Removing adaptive annealing fixes $\beta = 10^{-4}$. The reduced PF-plus-local and PF-only configurations disable IPF, sharing, novelty, kicks, island mechanisms and diversity/ESS rejuvenation; ordinary jitter and ESS-triggered resampling remain enabled.

Key findings:

- **Local optimization has the largest target-attainment effect**: Removing it reduces success from 30/30 to 0/30 on both instances under these stopping limits. The no-local variants reach the iteration cap after far fewer evaluations; this is not an equal-evaluation comparison.
- **IPF reduces evaluation cost**: The no-IPF variant also succeeds in 30/30 runs on each instance, but its mean evaluation counts are 12.710 versus 3.833 million for the full control on LJ75, and 8.632 versus 0.947 million on LJ102 (ratios 3.32 and 9.11). Mean times are 5,096 versus 1,349 seconds and 4,843 versus 313 seconds, respectively; these timings reflect the recorded execution conditions.
- **Kicks have an instance-dependent effect**: Removing kicks reduces LJ75 success to 11/30 (36.7%), while LJ102 remains at 30/30.
- **Other individual removals do not establish universal necessity**: Removing sharing or island exchange gives 29/30 on LJ75 and 30/30 on LJ102. Removing novelty or adaptive annealing gives 30/30 on both. One-run differences are descriptive and do not by themselves establish a reliable effect.
- **Reduced configurations**: PF plus local refinement succeeds in 3/30 LJ75 runs and 30/30 LJ102 runs; PF without local refinement succeeds in neither. These configurations remove several controls together and do not test individual mechanisms in isolation.

These experiments support the integrated framework on LJ75 and LJ102, with distinct effects on success and cost. In particular, they establish no observed success-rate gain from IPF on these samples, despite its lower evaluation cost. They do not establish that every mechanism is necessary on all sizes or on other problem classes.

For a post hoc paired check, exact two-sided McNemar tests compare each of the nine variants with the full control on each instance; Holm adjustment covers all 18 comparisons (Supplement S1). The one-run sharing and island-exchange differences have $p = 1$, whereas removing kicks on LJ75 gives adjusted $p = 4.96 \times 10^{-5}$. An absence of discordant outcomes does not establish equivalence or universal necessity. Paired-bootstrap 95% intervals for the no-IPF/full ratio of mean evaluation counts are $[2.40, 4.61]$ on

LJ75 and [7.25, 11.27] on LJ102. These assessments retain the unequal-evaluation stopping protocol and are conditional on the sampled initializations.

4.3 Sensitivity Analysis

We analyze sensitivity to key hyperparameters on LJ75 and LJ102 (10 completed runs per configuration and instance), using the study configuration of Section 4.2. Times and iteration counts are averaged over all ten runs, including unsuccessful runs. The default rows $\beta_{\min} = 10^{-4}$ and $\alpha_0 = 0.98$ reuse the first ten runs of the full ablation control for each instance; they are not additional independent experiments. The 600 ablation runs and 220 additional sensitivity runs thus provide 820 unique executions for Tables 7–10.

4.3.1 Number of Particles

Table 8: Effect of particle count on success rate (%) and mean time (s), 10 runs per configuration and instance

	Success (%)		Mean time (s)	
N_p	LJ75	LJ102	LJ75	LJ102
100	70	100	1417.1	165.5
200	80	100	1898.4	420.1
300	100	100	2582.3	285.5
400	100	100	1201.1	354.2
500	100	100	2534.9	411.1

On LJ75, the observed success rate increases from 70% at $N_p = 100$ to 80% at 200 and 100% at 300–500. LJ102 reaches 100% at every tested particle count. Mean time is not monotone in N_p: on LJ75, 400 particles have the lowest mean time among the settings with 10/10 successes, whereas on LJ102 the lowest mean time is obtained with 100 particles. These observations do not establish a universally optimal population size.

4.3.2 Annealing Schedule

Table 9: Effect of $\beta_{\min}$ on success rate (%) and mean iteration count, 10 runs per configuration and instance

	Success (%)		Mean iterations	
$\beta_{\min}$	LJ75	LJ102	LJ75	LJ102
10^{-5}	100	100	127.2	10.5
10^{-4} (default)	100	100	127.2	10.5
10^{-3}	100	100	70.7	10.5
10^{-2}	100	100	92.5	10.5

All four settings yield 10/10 successes on both instances. On LJ75, $\beta_{\min} = 10^{-3}$ has the lowest observed mean iteration count (70.7) and mean evaluation count (3.068 million), but the difference from the default is driven by only one of the ten paired runs. On LJ102, all four settings have the same mean counts: 10.5 iterations and 861,146.9 evaluations. Their paired objective histories are identical while weight histories differ, and no resampling is triggered in these runs; equal outcomes therefore do not imply an inactive parameter. The two lowest beta settings likewise give identical paired objective histories on LJ75, with different weights. Success rates do not distinguish the tested settings in these samples.

4.3.3 IPF Step Size

All four initial α_0 settings, including 1.00, achieve 10/10 successes on each instance. They are the actual first-step coefficients of the endpoint schedule in Section 4.2, with final endpoint 0.05. Mean evaluation counts range from 4.198 to 5.300 million on LJ75 and from 0.742 to 0.861 million on LJ102. At $\alpha_0 = 1.00$,

Table 10: Effect of the IPF parameter setting α_0 on success rate (%), 10 runs per configuration and instance

α_0	LJ75	LJ102
0.95	100	100
0.98 (default)	100	100
0.99	100	100
1.00	100	100

the first deterministic map collapses each cluster to its elite and is singular; subsequent jitter and local refinement remain enabled. This boundary test shows no success-rate deterioration, not preservation of invertibility or diversity by that first map.

Overall, success rates are stable over the tested annealing and IPF settings, while particle-count sensitivity is visible on LJ75. These conclusions are restricted to LJ75 and LJ102 and ten runs per configuration; they do not establish general parameter insensitivity.

4.4 Search Dynamics and Topology Diagnostics

To understand how HPF navigates Lennard-Jones energy landscapes, we analyze search dynamics from two complementary viewpoints. Figures 2 and 3 visualize the evolution of particle objectives and normalized weights over PF iterations. These preserved illustrations are qualitative and do not depict the 240 distinct-population runs summarized in Table 11.

Figures 4 and 5 report four complementary diagnostics: weight entropy $H(\mathbf{w}_t) = -\sum_i w_t^{(i)} \log w_t^{(i)}$, normalized effective sample size, the best particle energy before local refinement, and an estimated energy-mode count obtained by kernel density estimation (KDE). Together, these provide a compact picture of when the algorithm maintains broad exploration and when selection pressure induces concentration. The panels are complementary illustrations and are not assumed to describe the same runs.

Weight concentration depends on both the landscape and the search dynamics. For LJ38 (Figure 2), weights remain nearly uniform. The illustrated LJ102–104 trajectories show progressive changes in weight magnitude over several iterations. These observations do not establish that the underlying potentials have simple or smooth landscapes.

The intermediate cases LJ75 and LJ76 exhibit a different pattern. The particle population maintains a diffuse weight distribution before a late concentration event in Figure 3. The corresponding entropy and ESS diagnostics in Figure 5 illustrate a gradual transition toward stronger selection.

The most informative behavior is observed on LJ77 and LJ98. Figure 3 shows persistent weight dispersion with isolated higher-weight particles. The complementary entropy and ESS diagnostics in Figure 5 illustrate stronger concentration than for LJ38. Such trajectory-dependent patterns are indirect diagnostics, not a reconstruction or proof of landscape topology.

An important practical point is the division of labor between the PF phase and the local optimizer. The figures illustrate search dynamics rather than isolate a pre-local PF contribution. The removal experiments in Table 7 address the role of local refinement separately on LJ75 and LJ102, where removing it causes the largest drop in target attainment under the tested limits.

These observations retain the exploration–exploitation interpretation of the figures without turning trajectory-specific features into universal landscape classifications. They also suggest a direction for future improvement: adapting particle count, annealing aggressiveness, and rejuvenation intensity online as a function of entropy, ESS, and mode-count trajectories.

Table 11 presents the detailed HPF results extracted from the optimization logs, including per-instance success rates, best and mean energies, function evaluation counts, and wall-clock times.

Table 12 summarizes the empirical relationship between energy landscape complexity, convergence behavior, and weight concentration patterns observed across the tested Lennard-Jones instances.

This synthesis highlights the link between structural complexity and the timing or abruptness of exploitation during PF iterations, providing valuable insights for future algorithmic improvements.

5 Conclusion

This paper introduces a hybrid optimization framework that combines Particle Filtering with local refinement techniques to tackle the challenging Lennard-Jones (LJ) cluster optimization problem across the

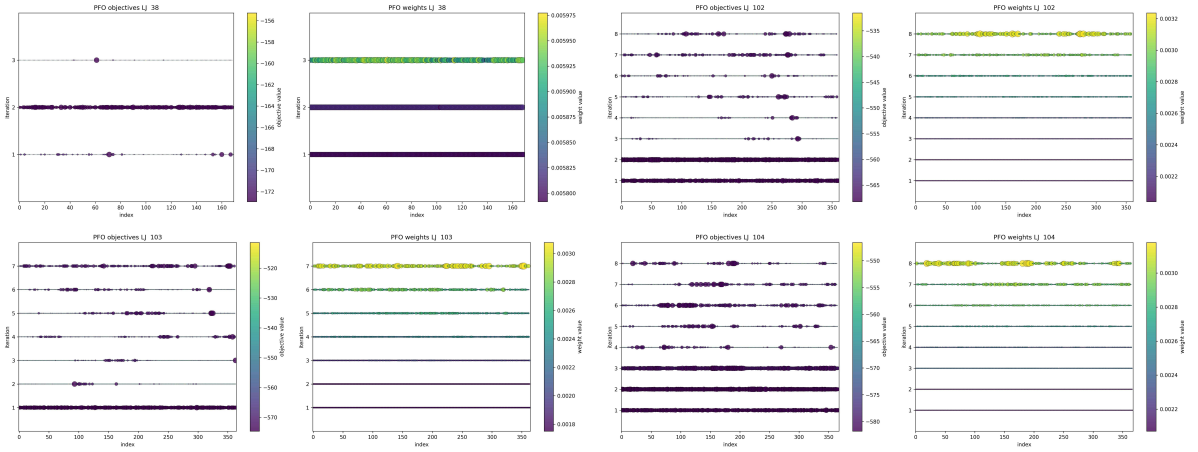

Figure 2: Evolution of particle objectives (left) and normalized weights (right) over PF iterations for LJ38 and LJ102–104. Each row shows one PF iteration. Color bars give the objective and weight scales; larger bubbles in the weight panels highlight higher weights.

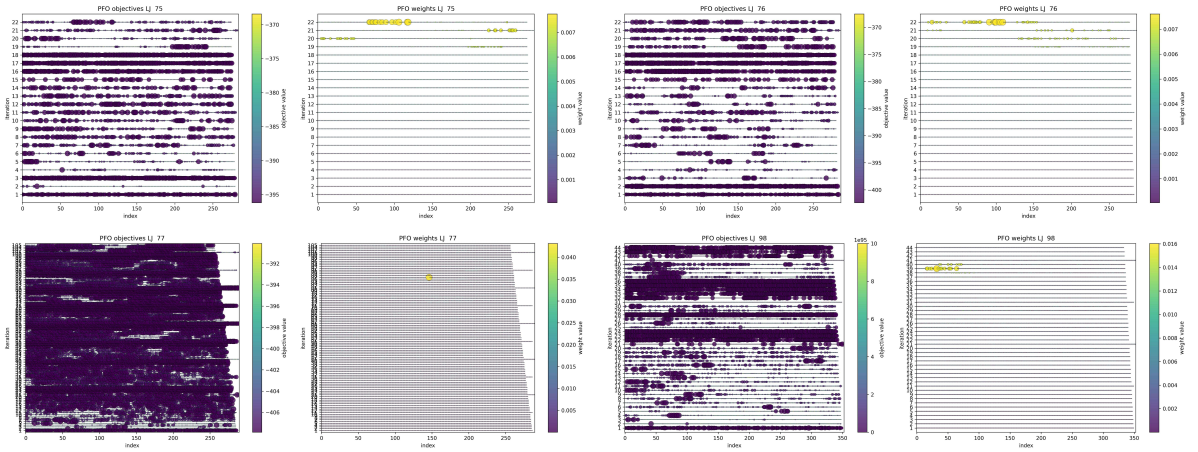

Figure 3: Evolution of particle objectives (left) and normalized weights (right) over PF iterations for LJ75, LJ76, LJ77 and LJ98. Weights remain diffuse for longer, with late or isolated concentration events. Color bars give the objective and weight scales; marker sizes provide a qualitative indication of weight magnitude.

tested range of cluster sizes, namely $N \in \{38, 75, 76, 77, 98, 102, 103, 104\}$. The method operates without problem-specific seeding and is evaluated with 30 distinct box-uniform initial populations per instance.

5.1 Main Findings

From 30 distinct populations per size, HPF reaches the reference targets in 30/30 runs on LJ38 and LJ102–104, 29/30 on LJ75, 26/30 on LJ76, 17/30 on LJ77 and 6/30 on LJ98. The corresponding target-attainment rates range from 20.0% to 100%, indicating markedly instance-dependent reliability. Under the executed evaluation and time limits, BH-OJ attains the target in 10% of LJ38 runs, and the other baseline cells have no successes. These outcomes support the value of the integration relative to the tested implementations and parameterizations, but differing local solvers, calibrated limits and resources preclude a general ranking of the corresponding algorithmic families. The supplementary fixed-population series sharpens this limitation: it yields 13/30 successes on LJ98 against 6/30 with distinct populations, while identical LJ77 totals conceal sixteen discordant, search-seed-matched outcomes. This comparison illustrates initialization sensitivity without isolating a causal effect of the starting population (Supplement S3).

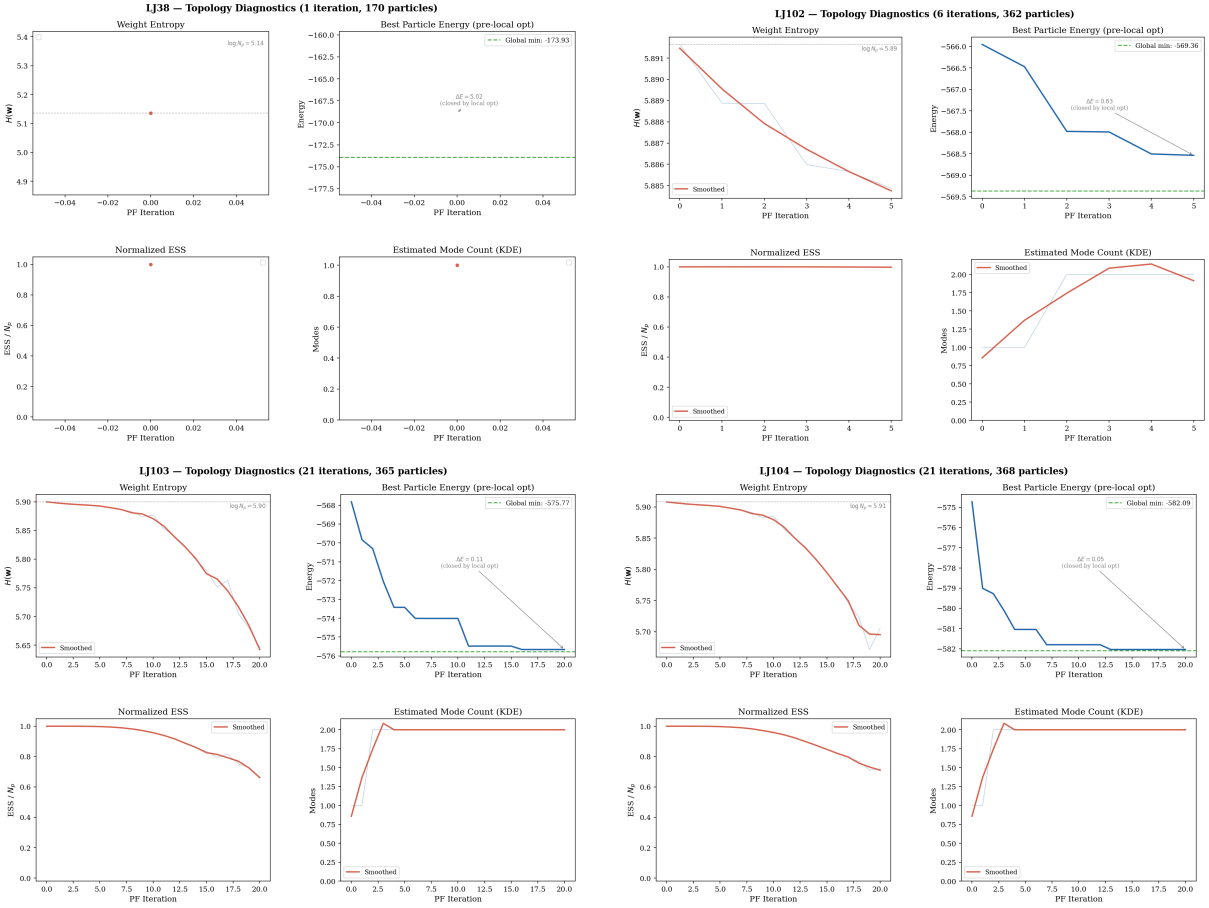

Figure 4: Topology diagnostics for LJ38 and LJ102–104. Each instance panel shows weight entropy $H(\mathbf{w})$ with $\log N_p$ reference (top-left), best particle energy before local optimization with the reference minimum (top-right), normalized ESS (bottom-left), and estimated mode count via KDE (bottom-right). Raw traces in light blue; Savitzky–Golay smoothed curves in red.

The ablation study (Section 4.2), restricted to LJ75 and LJ102 and its own reference configuration, identifies local optimization as the largest contributor to target attainment under the stopping limits used. Removing IPF leaves success unchanged at 30/30 on both instances but increases mean evaluations by factors of 3.32 and 9.11, respectively. Kicks affect LJ75 success strongly, whereas the individual removals of novelty and adaptive annealing show no success-rate loss. These findings support the integration without implying that every component is necessary. Removing mechanisms is not the same as testing each one in isolation, and conclusions are not extended to all sizes or other problem classes.

5.2 Key Design Elements

The principal design element is the integration of particle filtering with local refinement, adaptive kicks and diversity-aware weighting. Within this framework, the *Invertible Particle-Flow* (IPF) operator biases particles toward elite configurations through smooth, invertible transport steps while practical diversity is supported by the combined action of jitter, mutation, resampling, and rejuvenation. Its contribution should be interpreted within that integration, rather than as the sole source of the observed gains. Weight evolution analysis (Section 4.4) illustrates nearly uniform weights for LJ38, late concentration on LJ75–LJ76 and isolated weight spikes on LJ77–LJ98.

Theoretical motivation is provided by the separate accessibility and target-concentration statements of Propositions 1–2. They do not combine into a convergence guarantee for the capped, adaptive, archive-dependent algorithm used in the experiments, which does not explicitly impose the theoretical separation condition. Section 3.12 and Remark 2 specify the missing tracking argument.

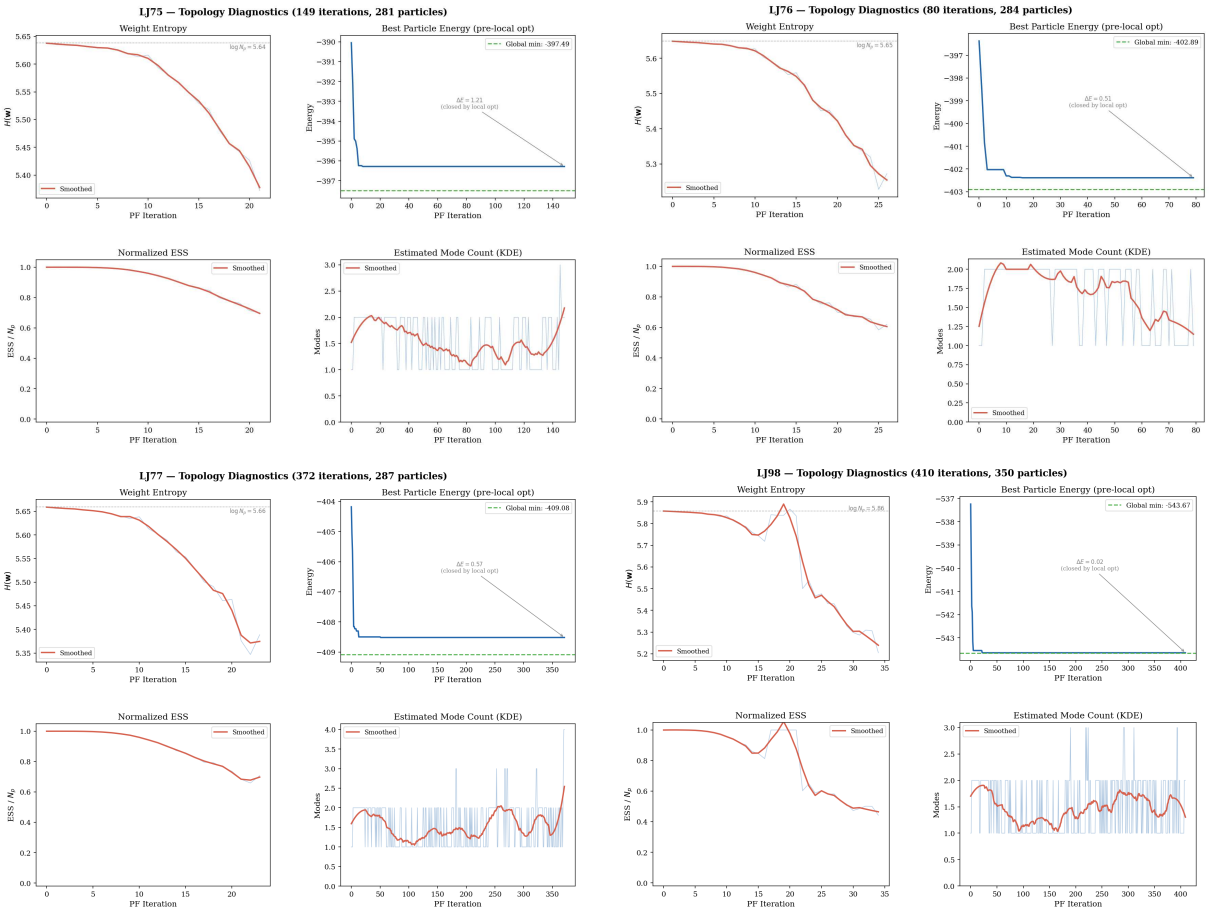

Figure 5: Topology diagnostics for LJ75–LJ98. Entropy and ESS are reported for iterations with numerically valid normalized weights. Best-particle energies are measured before local refinement. LJ75 and LJ76 show gradual entropy decline; LJ77 and LJ98 exhibit sharper drops indicating stronger concentration on fewer particles.

5.3 Limitations

Despite these achievements, several limitations remain:

- **Reliability on hardest cases**: With renewed initial populations, target attainment is 56.7% on LJ77 and 20.0% on LJ98. Reliability is therefore instance-dependent and is not established for arbitrary initialization distributions.
- **Computational cost**: LJ98 includes individual runs exceeding 20,000 seconds on the stated CPU configuration; mean times and their dispersion are reported in Table 11
- **Limited flow exploration**: Only linear IPF flow was thoroughly tested; more complex flows may provide further gains
- **Single problem domain**: Validation focused on LJ clusters; generalization requires further study

5.4 Future Directions

While the present validation is restricted to Lennard-Jones clusters, the framework's modular design suggests potential applicability to other molecular and non-convex optimization problems. This remains an untested hypothesis; systematic cross-domain assessment is necessary before a broader claim can be made. Several natural extensions can be identified:

1. **Advanced normalizing flows**: Investigate trainable affine coupling and neural ODE flows for IPF

Table 11: Detailed HPF results from 30 distinct initial populations per instance. Energy and timing moments use all runs; the evaluation mean is conditional on success.

N	Success	Best Energy	Mean Energy ± SD	Mean Evals (success)	Max Evals	Mean Time (s) ± SD
38	30/30 (100%)	−173.928427167	$-173.928427085 \pm 4.30 \times 10^{-7}$	178 925	230 981	16.2 ± 2.8
75	29/30 (96.7%)	−397.492332572	$-397.492327965 \pm 2.52 \times 10^{-5}$	6 115 607	19 852 440	2583.4 ± 1667.5
76	26/30 (86.7%)	−402.894867631	-402.826829358 ± 0.176	7 478 919	19 845 989	4312.4 ± 2918.6
77	17/30 (56.7%)	−409.083518768	-408.833557034 ± 0.291	9 873 759	28 816 794	8293.6 ± 4505.9
98	6/30 (20.0%)	−543.665363134	$-543.647440354 \pm 9.11 \times 10^{-3}$	10 078 857	40 292 501	16424.5 ± 6183.1
102	30/30 (100%)	−569.363654967	$-569.363653118 \pm 1.01 \times 10^{-5}$	1 143 638	4 542 882	447.5 ± 481.8
103	30/30 (100%)	−575.766133373	$-575.766133240 \pm 7.17 \times 10^{-7}$	1 232 786	6 016 238	503.2 ± 589.2
104	30/30 (100%)	−582.086644595	$-582.086644574 \pm 7.16 \times 10^{-8}$	4 108 697	9 331 737	2358.0 ± 2042.2

Means and sample standard deviations of energies and elapsed times include all 30 runs (sample SD divisor 29). Evaluation means are restricted to successful runs, as labeled; evaluation maxima include all runs. Full-precision statistics are available in the data.

Table 12: Landscape complexity and observed search behavior (HPF target attainment from 30 distinct initial populations per size; weight patterns illustrate the retained, separate Figure 2–3 trajectories).

Instance	**Landscape / observed regime**	**Success Rate**	**Weight Pattern**
LJ38	Rapid observed descent	100%	Nearly uniform
LJ75	Moderately rugged	96.7%	Late concentration
LJ76	Moderately rugged	86.7%	Late concentration
LJ77	Highly rugged	56.7%	Strong isolated spikes
LJ98	Highly rugged	20.0%	Isolated spikes
LJ102	Broad funnel	100%	Progressive weight changes
LJ103	Broad funnel	100%	Progressive weight changes
LJ104	Broad funnel	100%	Progressive weight changes

2. **Adaptive hyperparameter selection**: Develop meta-learning approaches to automatically tune N_p, annealing schedule, diversity weights
3. **Parallel and GPU acceleration**: Exploit parallelizability for larger clusters ($N > 150$)
4. **Domain extension**: Apply to protein folding, molecular docking, materials design, and standard global optimization benchmarks (e.g., Morse clusters, Thomson problem, binary Lennard-Jones mixtures) to systematically assess the framework's generalizability beyond homogeneous LJ systems
5. **Tighter convergence rates**: Derive finite-sample complexity bounds
6. **Machine learning potentials**: Integrate learned surrogates to reduce evaluation costs

Overall, this work presents a methodology for global optimization of Lennard-Jones clusters evaluated against the specified baseline implementations under the adopted protocol. Potential applicability to other multi-funnel molecular optimization problems remains untested.

Acknowledgements

We thank the anonymous reviewers for their insightful comments, which substantially improved this manuscript.

Funding

No funding was received for conducting this research.

Conflict of Interest

The author declares that he has no conflict of interest.

Data and Code Availability

The implementation, experimental scripts and numerical data are available to the Editor and reviewers at:

`https://github.com/francoisakoa/hpfo/releases/tag/pf131-jogo-r2`

The release includes the complete records of the 240-run HPF benchmark with independently generated initial populations, together with the native populations, seeds, final coordinates, optimization histories and associated scripts. The baseline and ablation/sensitivity datasets are also provided.

The repository README gives a file-by-file guide to the data, source code, analysis scripts and integrity checks:

`https://github.com/francoisakoa/hpfo`

Ethics

This work consists of numerical simulations of model particle systems and involves no human participants or animal experiments.

References

[1] D. J. Wales and J. P. K. Doye. Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms. *J. Phys. Chem. A*, 101(28):5111–5116, 1997.

[2] D. M. Deaven and K. M. Ho. Molecular geometry optimization with a genetic algorithm. *Phys. Rev. Lett.*, 75(2):288–291, 1995.

[3] R. H. Leary. Global optimization on funneling landscapes. *J. Global Optim.*, 18(4):367–383, 2000.

[4] F. Schoen. Two-phase methods for global optimization. In P. M. Pardalos and H. E. Romeijn, editors, *Handbook of Global Optimization*, volume 2, pages 151–177. Kluwer, Dordrecht, 2003.

[5] M. Iwamatsu and Y. Okabe. Basin hopping with occasional jumping. *Chem. Phys. Lett.*, 399(4–6):396–400, 2004.

[6] C. A. Floudas et al. *Handbook of Test Problems in Local and Global Optimization*. Kluwer, Dordrecht, 1999.

[7] C. Barron, S. Gomez, and L. Romero. Archimedean polyhedron structure yields a lower energy atomic cluster. *Appl. Math. Lett.*, 12(7):97–100, 1999.

[8] J. P. K. Doye and D. J. Wales. Thermodynamics and the global optimization of Lennard-Jones clusters. *J. Chem. Phys.*, 109(19):8143–8153, 1998.

[9] P. Stinis. Stochastic global optimization as a filtering problem. In *AIP Conf. Proc.*, volume 1168, pages 740–743, 2009.

[10] C. Zhang, A. Taghvaei, and P. G. Mehta. A controlled particle filter for global optimization, 2017. arXiv:1701.02413.

[11] E. Zhou, M. C. Fu, and S. I. Marcus. A particle filtering framework for randomized optimization algorithms. In *Proc. Winter Simulation Conf.*, pages 647–654, 2008.

[12] E. Zhou, M. C. Fu, and S. I. Marcus. Solving continuous-state POMDPs via density projection. *IEEE Trans. Autom. Control*, 55(5):1101–1116, 2010.

[13] M. Gerber and R. Douc. A global stochastic optimization particle filter algorithm. *Biometrika*, 109(4):937–955, 2022. doi: 10.1093/biomet/asab067.

[14] M. Eslami and M. Babazadeh. Particle filter optimization: A Bayesian approach for global stochastic optimization, 2024. arXiv:2406.03089.

[15] A. Kuznetsov and N. V. Sahinidis. New bounds and formulations for the deterministic global optimization of Lennard–Jones clusters. *J. Global Optim.*, 2025. doi: 10.1007/s10898-025-01476-7.

[16] L. Maillard, F. Finocchi, C. Godinho, and M. Trassinelli. Nested sampling for exploring Lennard-Jones clusters. *Phys. Sci. Forum*, 12(1):8, 2025. doi: 10.3390/psf2025012008.

[17] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proc. IEEE Int. Conf. Neural Networks*, volume 4, pages 1942–1948, 1995.

[18] M. Clerc and J. Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput.*, 6(1):58–73, 2002.

[19] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.*, 10(3):281–295, 2006.

[20] J. J. Liang and P. N. Suganthan. Dynamic multi-swarm particle swarm optimizer. In *Proc. IEEE Swarm Intell. Symp.*, pages 124–129, 2005.

[21] K. E. Parsopoulos and M. N. Vrahatis. UPSO: A unified particle swarm optimization scheme. In Lecture Series on Computer and Computational Sciences, vol. 1, pp. 868–873, 2004.

[22] D. E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization. In *Proc. Int. Conf. Genetic Algorithms*, pages 41–49, 1987.

[23] D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. In *Proc. Int. Conf. Mach. Learn.*, pages 1530–1538, 2015.

[24] L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using real NVP. In *Proc. Int. Conf. Learn. Represent.*, 2017.

[25] T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations. In *Proc. Adv. Neural Inf. Process. Syst.*, pages 6571–6583, 2018.

[26] P. Del Moral. *Feynman-Kac Formulae: Genealogical and Interacting Particle Systems with Applications*. Springer, New York, 2004.

[27] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.*, 106(1):25–57, 2006.

[28] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu. A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.*, 16(5):1190–1208, 1995.

[29] R. Douc and O. Cappé. Comparison of resampling schemes for particle filtering. In *Proc. Int. Symp. Image Signal Process. Anal.*, pages 64–69, 2005.

[30] C. Andrieu and A. Doucet. Particle filtering for partially observed Gaussian state space models. *J. R. Stat. Soc. B*, 67(3):827–836, 2005.

[31] S. P. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Springer, London, 1993.

[32] W. Pullan. An unbiased population-based search for the geometry optimization of Lennard–Jones clusters: $2 \leq N \leq 372$. *J. Comput. Chem.*, 26(9):899–906, 2005. doi: 10.1002/jcc.20226.

[33] X. Shao, L. Cheng, and W. Cai. A dynamic lattice searching method for fast optimization of Lennard–Jones clusters. *J. Comput. Chem.*, 25(14):1693–1698, 2004. doi: 10.1002/jcc.20096.

[34] K. Yu, X. Wang, L. Chen, and L. Wang. Unbiased fuzzy global optimization of Lennard-Jones clusters for $N \leq 1000$. *J. Chem. Phys.*, 151:214105, 2019. article 214105; Publisher's Note, vol. 152, p. 019901 (2020), doi: 10.1063/1.5142504; primary doi: 10.1063/1.5127913.

[35] L. Zhan, B. Piwowar, W.-K. Liu, P. J. Hsu, S. K. Lai, and J. Z. Y. Chen. Multicanonical basin hopping: a new global optimization method for complex systems. *J. Chem. Phys.*, 120(12):5536–5542, 2004. doi: 10.1063/1.1649728.

[36] G. G. Rondina and J. L. F. Da Silva. Revised basin-hopping Monte Carlo algorithm for structure optimization of clusters and nanoparticles. *J. Chem. Inf. Model.*, 53(9):2282–2298, 2013. doi: 10.1021/ci400224z.

[37] M. Arbel, A. Matthews, and A. Doucet. Annealed flow transport Monte Carlo. In *Proc. 38th Int. Conf. Mach. Learn.*, volume 139, pages 318–330. PMLR, 2021.

A Surrogate Convergence Argument for the Idealized Uncapped Variant

A.1 Idealized Exploration Kernel

The state and kernels are exactly those of Assumption 2. The reference measure ν has full support on the nonempty compact metric space D. The positive-density example there has a positive finite normalization constant. It is not a composition of arbitrary mutation, clipping and selection operations. Fixing an archive is not equivalent to omitting a nonconstant novelty term. Neither operation is needed for the two propositions because no complete adaptive Markov chain is analyzed.

A.2 Accessibility of the Exploration Kernel

Proof of Proposition 1. Let $U \subseteq D^m$ be nonempty and relatively open, and choose $y \in U$. The product topology provides nonempty relatively open $U_i \subseteq D$ such that $y \in \prod_i U_i \subseteq U$. The definition of Q and conditional independence give

$$Q(z, U) \geq Q\left(z, \prod_i U_i\right) = \prod_{i=1}^{m} q_i(z, U_i) > 0.$$

Every factor is positive by Assumption 2. □

The conclusion is one-step neighborhood accessibility. It does not assert positive probability of drawing a prescribed point, recurrence of a practical trajectory, or existence of a stationary distribution for the composed algorithm. Subsequent state-dependent transformations require their own analysis.

A.3 Concentration on the Global Minimizer Set

Proof of Proposition 2. By compactness and continuity, X_D^* is nonempty and compact. Let U be a relatively open neighborhood of X_D^*. If $D \setminus U = \varnothing$, the conclusion is immediate. Otherwise the compact complement is disjoint from X_D^*, so

$$\eta = \min_{x \in D \setminus U} E(x) - E_D^* > 0.$$

The relatively open set $V = \{x \in D : E(x) < E_D^* + \eta/2\}$ is nonempty, contained in U, and satisfies $\nu(V) > 0$ by full support. Hence

$$\pi_\beta(D \setminus U) \leq \frac{e^{-\beta(E_D^* + \eta)} \nu(D \setminus U)}{e^{-\beta(E_D^* + \eta/2)} \nu(V)} \leq \frac{e^{-\beta\eta/2}}{\nu(V)} \longrightarrow 0.$$

Therefore $\pi_\beta(U) \to 1$. □

The bound depends on U, E and ν. It is a bound for target measures, not for the runtime, errors or probability of success of the implemented solver.

A.4 Scope of the Appendix

The proofs do not show that standardized diversity-modified empirical weights converge to energy-only targets. A positive denominator floor does not force β_k times an energy gap to diverge when that gap shrinks: for example, gaps $1/k^2$ and $\beta_k = k$ yield a product tending to zero once a fixed positive floor dominates. The practical weight floor introduces a further difference from exact Gibbs weights.

No local-solver convergence follows from Lipschitz continuity of the energy alone. No exact projection onto a critical-point set is assumed for finite-tolerance optimization. Ordinary box clipping does not establish invariance of a collision-free theoretical domain. Capped annealing does not meet Assumption 3; removing a cap from an ESS-based rule would not, by itself, prove divergence either.

The propositions thus provide separate mathematical motivations for exploration and annealed selection. As Remark 2 specifies, the missing step for a convergence result on HPF is control of the discrepancy between its particle-state law and the target family. That step is not claimed here.