GLOBAL OPTIMIZATION VIA SOFTMIN ENERGY MINIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Non-convex optimization in high dimensions requires balancing gradient information with global exploration, a tension that Langevin dynamics and other gradient-informed methods address only partially. We introduce a gradient-based swarm method driven by a Softmin energy interaction $J_\beta(\mathbf{x})$, a smooth approximation of the minimum over the ensemble that couples particles through their relative energies, and add Brownian noise to its gradient flow. We show that at every stable equilibrium the interaction splits the ensemble into particles at local minima and particles at local maxima; that without noise the ensemble's best value converges at least as fast as gradient flow under a Polyak–Łojasiewicz condition; and that the interaction caps the Freidlin–Wentzell quasipotential barrier between adjacent basins at order $1/\beta$, independently of the true barrier height and strictly below the barrier an overdamped Langevin particle faces. Consistently with this cap, on double-well and quadruple-well benchmarks Softmin reaches the target in every run at every barrier height tested, while Simulated Annealing, Langevin dynamics, and variants of Particle Swarm Optimization and Consensus-Based Optimization all eventually fail; counting failed runs, it is more than an order of magnitude cheaper than Simulated Annealing overall, though not on low barriers. Nor is this a mere effect of ensemble size: Softmin succeeds in every run with five particles, whereas Simulated Annealing gains from more particles only what independent restarts predict, enough on moderate barriers but not on high ones. On Lennard-Jones clusters, Softmin's energy gap to the best known minimum stays nearly flat as the cluster grows while every baseline's rises, and the method trains transformers with millions of parameters. Its limits are equally clear: Particle Swarm Optimization is faster on a high-dimensional Ackley function, and Adam on transformer training, where no method struggled.

1 INTRODUCTION

Non-convex objectives trap gradient-based methods in suboptimal local minima, a recognized bottleneck in settings from deep network training to reinforcement learning. The classical remedy is noise, as in Langevin dynamics (Coffey et al., 2004) and noise-perturbed stochastic gradient descent (Fehrman et al., 2020); but the mean escape time of a noisy particle grows exponentially in the height of the surrounding barrier (Berglund, 2011). Simulated Annealing and its variants (Kirkpatrick et al., 1983; Gelfand & Mitter, 1990; Tang et al., 2024) shift from noisy exploration to local refinement through a cooling schedule. Running many particles helps, but independent particles gain only through statistical averaging. Letting them cooperate, as in Interacting Particle Langevin Dynamics (Akyildiz et al., 2025; Cordero Encinar et al., 2025) or in metaheuristics such as Consensus-Based Optimization (CBO) and Particle Swarm Optimization (PSO) (Bandaru & Deb, 2016; Yang, 2020), can do better, but metaheuristics largely lack the guarantees available for gradient methods (Hussain et al., 2019; Rajwar et al., 2023).

We combine stochastic gradients with an interaction that has a precise, analyzable effect on barriers. Given $f : \mathbb{R}^d \to \mathbb{R}$ and an ensemble $\mathbf{x} = (x_1, \dots, x_n)$, $x_i \in \mathbb{R}^d$, the Softmin energy

$$J_\beta(\mathbf{x}) = \sum_{i=1}^{n} \frac{\exp(-\beta f(x_i))}{\sum_{j=1}^{n} \exp(-\beta f(x_j))} f(x_i) \tag{1}$$

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

is the Gibbs-weighted average of the particles' values (Bertsekas, 1999), interpolating between their mean ($\beta \to 0$) and their minimum ($\beta \to \infty$). We minimize it with the stochastic gradient flow on $\mathbb{R}^{nd}$

$$\mathrm{d}\mathbf{x}_t = -n\nabla J_{\beta_t}(\mathbf{x}_t)\mathrm{d}t + \sqrt{2\sigma}\mathrm{d}B_t, \tag{2}$$

where B_t is an nd-dimensional Brownian motion and $\sigma > 0$ the noise intensity. The inverse temperature β_t may vary in time for the descent guarantee of Theorem 2.4; the barrier analysis of Section 3 fixes it, and empirically a fixed, well-tuned β outperformed annealing it (Appendix D).

Contributions.

- **Ensemble partitioning and no slowdown.** The gradient of nJ_β rescales each particle's gradient by a state-dependent, sign-changing effective learning rate. At every stable equilibrium, particles with a positive rate sit at local minima and those with a negative rate at local maxima, and at least one particle is always at a local minimum (Theorem 2.1). The ensemble's best value never increases and, under a Polyak–Łojasiewicz condition, converges at least as fast as gradient flow, for any temperature schedule (Theorem 2.4).
- **Barrier flattening.** The interaction caps the Freidlin–Wentzell quasipotential barrier between adjacent basins at order $1/\beta$, uniformly in the true barrier height and strictly below the exit exponent of an overdamped Langevin particle (Theorem 3.4).
- **Reliability beyond restarts, and its limits.** Softmin reaches the target in every run on double-well and quadruple-well landscapes at every barrier height tested, where every baseline eventually fails. With as few as five particles it keeps this reliability, while Simulated Annealing gains from more particles only what independent restarts predict. On Lennard-Jones clusters its gap to the best known minimum stays flat as the cluster grows, and it trains transformers with millions of parameters. PSO is faster on a high-dimensional Ackley function and Adam on transformer training. We present agreement with the theory as consistency, not confirmation (Section 4).

Proofs are in Appendix C and experimental settings in Appendix E.

1.1 Related work

We summarize the closest lines of work here; Appendix A gives an extended discussion.

Metaheuristics. Population-based metaheuristics include Genetic Algorithms (Goldberg, 1989), Ant Colony Optimization (Dorigo & Stützle, 2004), CBO (Carrillo et al., 2018; Pinnau et al., 2017; Fornasier et al., 2024) and PSO (Kennedy & Eberhart, 1995). Their classical forms coordinate particles through position-based updates, without gradients: consensus averaging toward a weighted mean in CBO, attraction toward the best positions found in PSO. Gradient-informed variants (Carrillo et al., 2021; Noel, 2012; Riedl et al., 2023; Bolte et al., 2024) add the gradient to these updates, but keep consensus or attraction as the source of interaction. Softmin replaces them with the Softmin of the particles' values as the sole interaction. This assigns each particle a minimizing or maximizing role without basin-switching heuristics (Theorem 2.1), and it yields a provable barrier cap (Theorem 3.4).

Langevin and interacting-particle methods. SGLD (Welling & Teh, 2011) and its non-asymptotic analysis (Raginsky et al., 2017) concern a single diffusion driven by ∇f, with guarantees governed by the spectral gap of the associated Gibbs measure. With several particles, the interaction usually reduces to averaging over independent runs. IPLD (Akyildiz et al., 2025; Cordero Encinar et al., 2025) genuinely couples particles, through their empirical measure, to estimate marginal-likelihood gradients in latent-variable models. Our coupling is competitive rather than mean-field: particles interact through their relative energies, which gives them a sign-changing effective learning rate, and the aim is global optimization of f.

Gibbs ensembles and softmin. Population Annealing (Wang et al., 2015) also uses a Gibbs-weighted ensemble, but it duplicates low-energy particles and removes high-energy ones, whereas Softmin keeps a fixed population and explores through continuous repulsion. The log-sum-exp free energy $F_\beta(\mathbf{x}) = -\beta^{-1}\log\sum_i e^{-\beta f(x_i)}$ has gradient $w_k\nabla f(x_k)$, so trailing particles with small weight simply stagnate. The gradient of J_β can instead reverse their motion, which is what destabilizes suboptimal minima. The Gibbs softmin has been studied as a smooth minimum (Bertsekas, 1999) and

for policy gradients and exploration in reinforcement learning (Mei et al., 2020; Song et al., 2019). Swarm-gradient methods anneal a particle density toward the minimizer (Bolte et al., 2024). To our knowledge, J_β has not been used as the objective of a global search.

2 Gradient Flow dynamics for Softmin energy

We first study the deterministic flow

$$\mathrm{d}\mathbf{x}_t = -n\nabla J_\beta(\mathbf{x}_t)\mathrm{d}t. \tag{3}$$

Let $w_k = e^{-\beta f(x_k)}(\sum_j e^{-\beta f(x_j)})^{-1}$ be the Gibbs weight of particle x_k and define the *pre-factor* $A_k(\mathbf{x}) = nw_k\,[1-\beta(f(x_k)-J_\beta(\mathbf{x}))]$. Then (Lemma C.1)

$$\nabla_{x_k}(nJ_\beta(\mathbf{x})) = A_k(\mathbf{x})\nabla f(x_k), \tag{4}$$

so (3) reads $\dot{x}_k = -A_k(\mathbf{x})\nabla f(x_k)$: each particle follows its own gradient with a state-dependent *effective learning rate*. As $\beta \to 0$, $w_k \to 1/n$ and $A_k(\mathbf{x}) \to 1$, recovering n independent gradient flows; this fixes the scaling factor n. The rate is positive, and the particle *minimizing*, when $f(x_k) < J_\beta(\mathbf{x}) + 1/\beta$. It is negative when $f(x_k) > J_\beta(\mathbf{x}) + 1/\beta$, and the particle then performs gradient *ascent*. A particle trapped in a poor minimum is therefore pushed uphill once the rest of the ensemble has found sufficiently lower values. Two facts drive the equilibrium analysis. First, since $\sum_k w_k = 1$ and $\sum_k w_k f(x_k) = J_\beta(\mathbf{x})$,

$$\sum_{k=1}^{n} A_k(\mathbf{x}) = n \qquad \text{for every } \mathbf{x} \in \mathbb{R}^{nd}, \tag{5}$$

so some pre-factor is always positive. Second, when every particle sits at a critical point of f, the mixed second derivatives of J_β vanish, and the Hessian of nJ_β is block diagonal with blocks $A_k(\mathbf{x})\nabla^2 f(x_k)$ (Lemma C.1).

2.1 Stationary points of the dynamics

Theorem 2.1. *Let $f \in C^2(\mathbb{R}^d)$, let $\beta > 0$, and let $\bar{\mathbf{x}} = (\bar{x}_1, \dots, \bar{x}_n)$ be a locally asymptotically stable equilibrium of* (3)*. Then, for every $k \in \{1, \dots, n\}$, evaluated at $\bar{\mathbf{x}}$:*

1. $\nabla f(\bar{x}_k) = 0$;
2. $A_k(\bar{\mathbf{x}}) \neq 0$;
3. *the sign of $A_k(\bar{\mathbf{x}})$ fixes the type of $\bar{x}_k$:*

$$A_k(\bar{\mathbf{x}}) > 0 \implies \bar{x}_k \text{ is a local minimum of } f,$$
$$A_k(\bar{\mathbf{x}}) < 0 \implies \bar{x}_k \text{ is a local maximum of } f,$$

with $A_k(\bar{\mathbf{x}}) > 0 \iff f(\bar{x}_k) < J_\beta(\bar{\mathbf{x}}) + 1/\beta$.

Moreover, by (5)*, at least one particle sits at a local minimum of f.*

No convexity is assumed. In particular, no particle rests at a strict local minimum above the threshold $J_\beta(\bar{\mathbf{x}}) + 1/\beta$. Conversely:

Corollary 2.2 (Sufficient condition for stability)**.** *Let $\bar{\mathbf{x}} = (\bar{x}_1, \dots, \bar{x}_n)$ satisfy $\nabla f(\bar{x}_k) = 0$ for every k, and assume that for each k either*

$$A_k(\bar{\mathbf{x}}) > 0 \text{ and } \nabla^2 f(\bar{x}_k) \succ 0, \qquad \text{or} \qquad A_k(\bar{\mathbf{x}}) < 0 \text{ and } \nabla^2 f(\bar{x}_k) \prec 0,$$

evaluated at $\bar{\mathbf{x}}$. Then $\nabla^2(nJ_\beta)(\bar{\mathbf{x}}) \succ 0$; in particular $\bar{\mathbf{x}}$ is a strict local minimum of nJ_β and a locally asymptotically stable equilibrium of (3)*.*

Proof. By Lemma C.1 the Hessian at a fully critical configuration is block diagonal with blocks $A_k(\bar{\mathbf{x}})\nabla^2 f(\bar{x}_k)$, and each hypothesis makes the corresponding block positive definite. □

The deterministic flow thus has stable equilibria that mix minima and maxima of f, not only consensus on one minimum, and which one is reached depends on the initialization. Escaping a suboptimal equilibrium requires noise, which Section 3 analyzes.

2.2 DESCENT OF THE ENSEMBLE MINIMUM

The sign of $A_k(\mathbf{x}(t))$ is not conserved along a trajectory, so we track the running best value $m(t) := \min_k f(x_k(t))$. We do this for an arbitrary, possibly time-dependent schedule $\beta_t > 0$.

Definition 2.3 (Polyak–Łojasiewicz condition)**.** A differentiable $f : \mathbb{R}^d \to \mathbb{R}$ satisfies the *Polyak–Łojasiewicz (PL) condition* on $S \subseteq \mathbb{R}^d$ with constant $\mu > 0$ if $\frac{1}{2}\|\nabla f(x)\|^2 \geq \mu\big(f(x) - f_\star\big)$ for every $x \in S$, where $f_\star := \inf f$.

Theorem 2.4 (Descent of the ensemble minimum)**.** *Let $f \in C^1(\mathbb{R}^d)$ have locally Lipschitz gradient, let $\beta_t > 0$ be measurable, and let $\mathbf{x}(t) = (x_1(t), \ldots, x_n(t))$ be a global solution of*

$$\dot{x}_k(t) = -A_k(\mathbf{x}(t))\,\nabla f(x_k(t)), \qquad k = 1, \ldots, n.$$

Suppose f satisfies the Polyak–Łojasiewicz condition (Definition 2.3) with constant $\mu > 0$ on the sublevel set $S_0 := \{x : f(x) \leq m(0)\}$. Then m is nonincreasing and

$$m(t) - f_\star \leq e^{-2\mu t}\big(m(0) - f_\star\big), \qquad t \geq 0.$$

This is a no-slowdown guarantee: the Softmin coupling never makes the ensemble's leading particle slower than ordinary gradient flow. It complements, rather than replaces, the basin-escape mechanism of Section 3. Without the PL hypothesis m is still nonincreasing; only the exponential rate requires it. Global existence of solutions is discussed in Remark C.2 (Appendix C).

3 BARRIER FLATTENING AND EXIT ASYMPTOTICS

We now fix $\beta_t \equiv \beta$ in (2), so that componentwise

$$\mathrm{d}x_k(t) = -A_k(\mathbf{x}_t)\nabla f(x_k(t))\mathrm{d}t + \sqrt{2\sigma}\mathrm{d}B_t^k, \tag{6}$$

and carry out the exit analysis in physical dimension $d = 1$ (see the end of this section for $d > 1$). Fix $n \geq 2$ and three consecutive nondegenerate critical points $z_- < s < z_+$ of f: two local minima $z_\pm$ and the local maximum s separating them, with

$$f'(x) > 0 \text{ on } (z_-, s), \qquad f'(x) < 0 \text{ on } (s, z_+). \tag{7}$$

Set the relative energy and barrier height

$$r(x) := f(x) - f(z_-), \qquad h := r(s) > 0, \qquad \rho_+ := r(z_+) < h. \tag{8}$$

The case of interest is $\rho_+ < 0$, in which z_+ is the deeper minimum.

Assumption 3.1. $f \in C^2(\mathbb{R})$; $z_- < s < z_+$ are nondegenerate critical points of f (i.e. $f''(z_-) > 0$, $f''(s) < 0$, $f''(z_+) > 0$) satisfying the monotonicity condition (7); and $n \geq 2$, $\beta > 0$, $d = 1$.

The consensus $(z_-, \ldots, z_-)$ is a strict local minimum of nJ_β, hence genuinely metastable (Remark C.3, Appendix C). We use the Freidlin–Wentzell theory of exit times (Appendix B): under its standard hypotheses, the exit time from a metastable domain D satisfies $\lim_{\sigma\downarrow 0} \sigma \log \mathbb{E}_{\mathbf{x}} \tau_D^\sigma = V_\beta(\mathbf{x}, \partial D)$, where V_β is the quasipotential of (2) at fixed β. To build an escape path, freeze $n - 1$ particles at z_- and let one move, $\Gamma(x) := (x, z_-, \ldots, z_-)$. Factoring $e^{-\beta f(z_-)}$ out of the Gibbs weights gives the exact identity

$$nJ_\beta(\Gamma(x)) = nf(z_-) + u(r(x)), \qquad u(r) := \frac{n\, r\, e^{-\beta r}}{n - 1 + e^{-\beta r}}, \quad r \in \mathbb{R}, \tag{9}$$

where u is the *reduced potential*.

Lemma 3.2 (Channel invariance and reduced gradient flow)**.** *The set $\{\Gamma(x) : x \in \mathbb{R}\}$ is invariant under* (3)*, since $f'(z_-) = 0$ forces the frozen coordinates' drift to vanish identically. On this invariant manifold, the dynamics reduces to the one-dimensional gradient flow $\dot{x} = -W'(x)$ of the reduced potential $W(x) := u(r(x))$ of* (9)*.*

Lemma 3.3 (Channel drift and shape of the reduced potential)**.** *Let $A_1(\Gamma(x))$ denote the pre-factor of the active particle along the channel. Then, for every $x \in \mathbb{R}$,*

$$A_1(\Gamma(x)) = u'(r(x)) = \frac{n\, e^{-\beta r(x)}\, q(r(x))}{\big(n - 1 + e^{-\beta r(x)}\big)^2}, \qquad q(r) := (n-1)(1 - \beta r) + e^{-\beta r}. \tag{10}$$

The function q is strictly decreasing on $\mathbb{R}$ and has a unique zero r_c, which satisfies

$$\frac{1}{\beta} < r_c < \frac{n}{(n-1)\beta}. \tag{11}$$

Consequently u is strictly increasing on $(-\infty, r_c]$ and strictly decreasing on $[r_c, \infty)$, and r_c is exactly the relative-energy level at which the active particle switches from minimizing to maximizing, i.e. the level at which $f(x) - J_\beta(\Gamma(x)) = 1/\beta$.

We abbreviate the two relevant channel heights by

$$u_c := u(r_c) = \max_{r\in\mathbb{R}} u(r), \qquad u_h := \max_{0\le r\le h} u(r) = u\big(\min\{h, r_c\}\big). \tag{12}$$

Theorem 3.4 (One-dimensional basin exit and post-saddle passage)**.** *Under Assumption 3.1, the following hold.*

1. ***(Uniform barrier cap.)***
$$u_c = \max_{r\in\mathbb{R}} u(r) < \frac{n}{e(n-1)\beta}. \tag{13}$$
Both u_c and, by (11)*, r_c are of order β^{-1}, uniformly in h.*

2. ***(Initial basin exit: quasipotential bound.)*** *The infimum of the action among channel paths joining $\Gamma(z_-)$ to $\Gamma(s)$ equals u_h. Consequently*
$$V_\beta\big(\Gamma(z_-), \Gamma(s)\big) \le u_h < \min\{h, r_c\} \le h. \tag{14}$$

3. ***(Post-saddle well and passage.)*** *Assume*
$$r_c < h. \tag{15}$$
Then $\Gamma(s)$ is a strict local minimum of nJ_β, with channel barrier $u_c - u(h)$ on the z_- side. On the z_+ side:

   - *If $\rho_+ < r_c$, there is a unique $\xi_+ \in (s, z_+)$ with $r(\xi_+) = r_c$, the channel well at $\Gamma(s)$ has matching barrier height $u_c - u(h)$ on this side, and*
   $$V_\beta\big(\Gamma(s), \Gamma(\xi_+)\big) \le u_c - u(h) < u_c; \tag{16}$$
   - *If instead $\rho_+ \ge r_c$, no point ξ_+ exists and the channel potential rises monotonically from $\Gamma(s)$ to $\Gamma(z_+)$, so no intermediate barrier separates them, with total rise $u(\rho_+) - u(h) \le u_c - u(h)$.*

   When $\rho_+ < r_c$, concatenating the two crossings gives
   $$V_\beta\big(\Gamma(z_-), \Gamma(\xi_+)\big) \;\le\; u_h + \big(u_c - u(h)\big) \;\le\; 2u_c - u(h) \;<\; \frac{2n}{e(n-1)\beta}, \tag{17}$$
   by (14)*,* (16)*, and $u_h \le u_c$. Under the standing hypothesis $r_c < h$ of this part one has $u_h = u_c$, so the second inequality in* (17) *is in fact an equality. If instead $h \le r_c$, $\Gamma(s)$ is not a local minimum of nJ_β and no post-saddle well forms.*

Corollary 3.5 (Exit-time consequence, conditional)**.** *Suppose in addition that z_- admits a metastable domain $D_- \subset \mathbb{R}^n$ for the flow* (3)*, in the sense that $\Gamma(z_-)$ is the unique attractor of D_- for the noiseless flow and the hypotheses of the Freidlin–Wentzell exit theorem (Appendix B) hold on D_-, with $\Gamma(x_m) \in \partial D_-$ realizing the minimal quasipotential to ∂D_-, where $x_m \in (z_-, s]$ is the unique point with $r(x_m) = \min\{h, r_c\}$. Then*

$$\lim_{\sigma\downarrow 0} \sigma \log \mathbb{E}_{\Gamma(z_-)} \tau^\sigma_{D_-} = V_\beta\big(\Gamma(z_-), \partial D_-\big) \le u_h < h.$$

We do not construct D_- or verify these hypotheses; Theorem 3.4 supplies only the quasipotential bound (14), which holds unconditionally.

Remark 3.6 (Comparison with independent Langevin dynamics). For n *independent* overdamped Langevin particles started at z_-, the small-noise exit exponent is the physical barrier h: a minimum over n i.i.d. exit times changes only the pre-factor, not the exponent. By (14), the joint Softmin diffusion admits a path of cost $u_h < h$ in every regime of h, and one capped at order $1/\beta$ however large h is. This compares an *upper bound* on the interacting system's quasipotential with the *exact* exponent of the noninteracting one. A statement about exit *times* holds only under the hypotheses of Corollary 3.5.

The channel identity extends algebraically to $d > 1$: at a critical point s the active Hessian block is $u'(h)\nabla^2 f(s)$, so at an index-one saddle with $u'(h) < 0$ the induced well is a one-dimensional phenomenon. Transverse directions can only add escape routes, and since V_β is an infimum over paths, the channel cost remains an upper bound on it. Part 3 does not carry over as stated: with $d-1$ stable transverse directions of f at s, $\Gamma(s)$ is a minimum of nJ_β along the channel only. The cap decreases like β^{-1}, but a large β also suppresses the drift of non-leading particles through their Gibbs weights. Neither a decaying nor an increasing schedule β_t improved on a fixed, well-chosen β in our experiments. The scope of the theorem, the full comparisons and these negative results are discussed in Appendix D.

4 Numerical Experiments

We run the Euler–Maruyama discretization of (2) at a fixed β (Algorithm 1, Appendix E) against seven baselines: Simulated Annealing (Gelfand & Mitter, 1990), overdamped Langevin dynamics (Welling & Teh, 2011), CBO with gradient information with and without annealing (Carrillo et al., 2021), PSO with gradient information (Noel, 2012), and the canonical gradient-free CBO (Carrillo et al., 2018; Pinnau et al., 2017) and PSO (Kennedy & Eberhart, 1995). Each benchmark makes one thing hard at a time: the height of a barrier (Section 4.1), the number of local minima as the dimension grows (Sections 4.2 and 4.4), and the size of the ensemble (Section 4.3). We always report *reliability*, the fraction of independent runs that reach the target, alongside *speed*, since a method that often fails can look fast on the few runs it solves. Conventions, detailed results and calibration are in Appendix E, and the code is available (Anonymous, 2026). One caveat applies throughout: Theorem 3.4 is a vanishing-noise statement, while the analytic benchmarks run at a noise level at which the proven cap is negligible. Where the experiments agree with the theory we therefore claim consistency, not confirmation; the small-noise regime is tested in Appendix E.5.1.

4.1 Double-well and quadruple-well benchmarks

The one-dimensional double well and the two-dimensional quadruple well each have a shallow basin, where every run starts, and a deeper target basin, separated by a barrier whose height we control.

Softmin reaches the target in every run at every barrier height on both landscapes, across barriers spanning three orders of magnitude (Figure 1, top). Every baseline eventually fails: Simulated Annealing matches Softmin while the barrier is low and then collapses until no run succeeds, Langevin dynamics fails much earlier, and CBO and PSO almost never reach the target. This is the pattern Theorem 3.4 describes, where a capped effective barrier keeps a method escaping while methods governed by the true height fail once it is large. For the reason above we read it as consistency only, and an ablation (Appendix E.5.4) points to the Gibbs weighting of the ensemble as the more plausible source of the advantage.

Where Simulated Annealing still succeeds, it is faster than Softmin (Figure 5). A practitioner cannot know in advance which runs will fail, however, so we also charge each method for its failed runs, as is standard in black-box benchmarking (Hansen et al., 2021). The comparison then becomes a crossover (Figure 1, bottom). Simulated Annealing is cheaper on low barriers, but beyond a moderate barrier its cost grows without bound, because its cooling schedule freezes the dynamics after a fixed time whatever the barrier. Over the whole sweep Softmin is more than an order of magnitude cheaper (Appendix E.4.2).

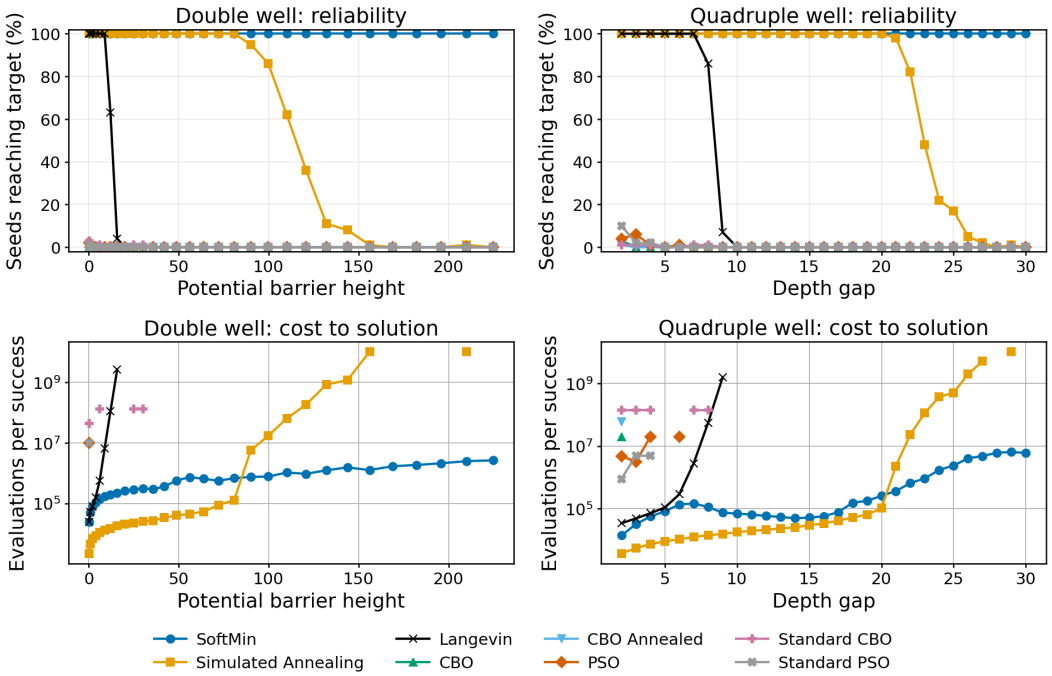

Figure 1: Double well (left, against barrier height) and quadruple well (right, against the depth gap between the two wells). Top: fraction of the 100 runs reaching the target. Bottom: expected cost of a success in function and gradient evaluations, counting failed runs; a gap means no run succeeded. Time to the target over successful runs is in Figure 5; conventions in Appendix E.4.1.

4.2 Lennard-Jones cluster benchmark

Lennard-Jones clusters of N atoms (Jones, 1924) are a standard, physically motivated test whose number of local minima grows exponentially with N. We use clusters of 13 to 75 atoms, up to 225 dimensions, with best known energies from the Cambridge Cluster Database (Wales & Doye, 1997). As is standard, each final configuration is relaxed to its nearest local minimum, and we score its energy gap to the best known minimum, since beyond the smallest cluster almost no method reaches a strict target. Softmin's gap stays small and nearly constant as the cluster grows, while every baseline's grows roughly in proportion to N (Figure 2), so its advantage widens with dimension instead of narrowing (Appendix E.4.3).

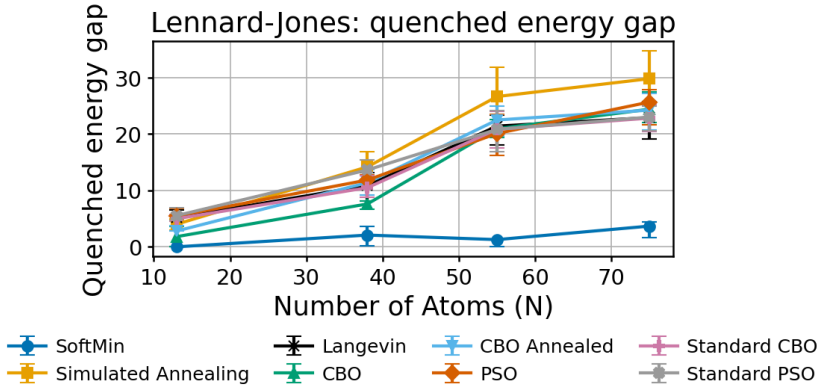

Figure 2: Lennard-Jones clusters of 13 to 75 atoms: energy gap between the best configuration each method finds, after relaxation to its nearest local minimum, and the best known minimum.

4.3 SCALING WITH ENSEMBLE SIZE

Every benchmark above uses 100 particles. Simulated Annealing with n particles is n independent runs, so a larger ensemble may help it through restarts alone. We therefore ask whether Softmin's coupling achieves something that restarts do not, by varying the ensemble from 5 to 400 particles on the quadruple well at three depth gaps.

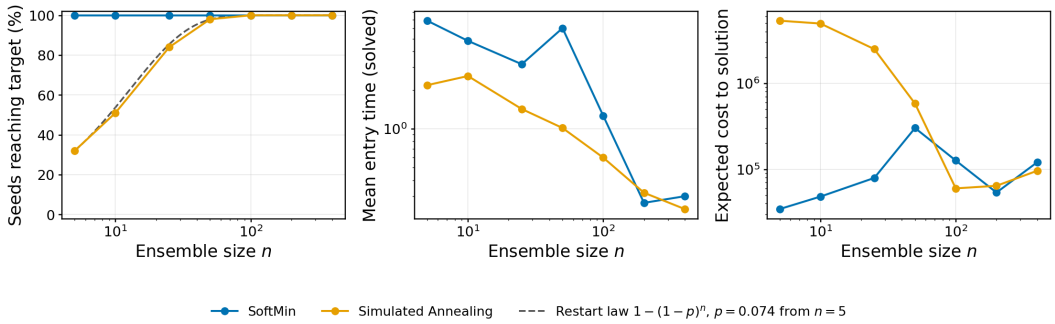

Figure 3: Softmin and Simulated Annealing on the quadruple well at an intermediate depth gap, with 5 to 400 particles. Left: fraction of runs reaching the target; the dashed line is what independent restarts predict from the smallest ensemble alone. Center: time to the target over successful runs. Right: expected cost of a success, counting failed runs. Other difficulties in Appendix E.4.5.

Softmin reaches the target in every run at every ensemble size and difficulty, even with 5 particles. Simulated Annealing always succeeds at the smallest depth gap and is never rescued at the largest. At the intermediate one, its reliability grows exactly as independent restarts predict from the smallest ensemble (Figure 3, left), so its gain from the ensemble is restarts and nothing more. Langevin dynamics sharpens the point: with one particle Softmin's pre-factor is identically one and its update is exactly Langevin's, which fails at all three depth gaps even with 100 independent particles (Figure 1). Five Softmin particles thus differ from five copies of a failing method only in their interaction. This matches Theorem 3.4: restarts multiply a success probability without changing how fast it decays with the barrier, whereas a capped barrier should need them at neither difficulty.

Counting failed runs, Softmin is cheapest with its smallest ensemble, while Simulated Annealing's cost falls as extra restarts buy reliability. At the smallest depth gap Simulated Annealing is cheaper at every size. At the intermediate one Softmin is far cheaper with small ensembles, but from 100 particles on the two cost about the same, so brute force can overcome this advantage (Section 5.1). At the largest, Simulated Annealing's cost is effectively unbounded. An unexplained non-monotonicity in Softmin's time to the target at the largest gap is analyzed in Appendix E.4.5.

4.4 HIGH-DIMENSIONAL PROBLEMS: ACKLEY AND A TRANSFORMER

The generalized Ackley function places a dense lattice of shallow minima on one smooth funnel, and is our synthetic high-dimensional test (10 to 90 dimensions, Appendix E.4.4). It is where Softmin does worst: PSO is the fastest method at every dimension by a wide margin, and Softmin fails in 10 and 20 dimensions. From 40 dimensions on it reaches the target in every run, and it is the only method outside the PSO family to become reliable. On a landscape whose difficulty lies in the number of minima rather than their depth, Softmin thus delivers reliability but not speed.

As a real network, we train a small transformer on modular addition, a standard testbed for small transformers (Power et al., 2022; Nanda et al., 2023) also used to compare optimizers (Tveit et al., 2025). Depth varies from 2 to 50 layers, or from about 10^5 to 2.5×10^6 parameters. We add Adam (Kingma & Ba, 2015) and SGD with momentum to Simulated Annealing and Langevin dynamics. Every method uses 10 particles and must bring the training loss below 0.10 (Appendix E.4.6).

Every method reaches the target from every starting point at every depth, so this benchmark does not separate the methods on reliability, and depth did not make training harder. It measures speed at scale (Figure 4). Softmin's step count barely changes with depth. It needs about a quarter of Simulated Annealing's steps and fewer than Langevin dynamics at every depth, while SGD goes

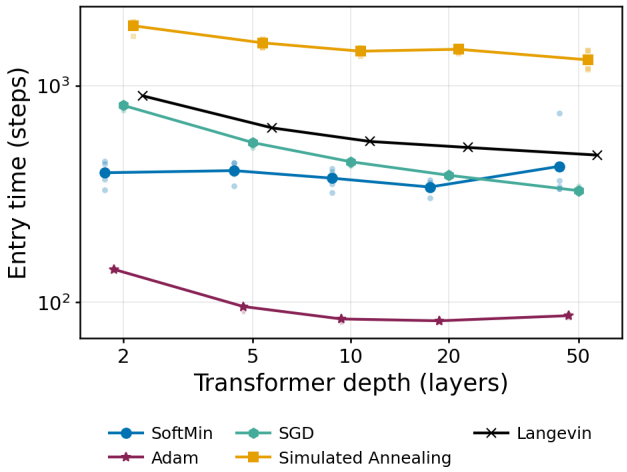

Figure 4: Transformer training at depths from 2 to 50 layers, five seeds per method and depth. Steps needed to reach the target training loss, as the mean over seeds (line) and for each seed (faint points); markers are offset horizontally so that they do not overlap.

from twice as slow as Softmin to matching it at 50 layers, and Adam is three to four and a half times faster throughout. We read this as evidence that Softmin carries over to networks with millions of parameters, not of an advantage there: when every method reaches the target from every start, a mechanism for escaping poor basins has nothing to act on.

5 Conclusion

We introduced a swarm method driven by the Softmin energy and proved two guarantees for it. Without noise, the ensemble's best value converges at least as fast as gradient flow under a Polyak–Łojasiewicz condition, for any temperature schedule. At fixed temperature, the interaction caps the quasipotential barrier between adjacent basins at order $1/\beta$, independently of the true barrier height. Empirically, Softmin's case is reliability. It is the only method that keeps reaching the target as barriers grow, with as few as five particles and at a fraction of Simulated Annealing's cost over the sweep, and its gap on Lennard-Jones clusters stays flat with dimension. It is not the fastest method on Ackley or on transformer training.

5.1 Limitations and future work

The barrier cap bounds a quasipotential through one explicit channel. It becomes a statement about exit times only under metastability hypotheses we do not verify, and the exact transition exponent remains open. The experiments run at a noise level where the cap is negligible, and an ablation finds that at $\beta = 10$ the sign-reversal mechanism is not what carries the advantage, though it appears to dominate at $\beta = 2$ as barriers grow (Appendix E.5.4). The results therefore show the behavior the theory describes, not that its mechanism produces it. The advantage is also bounded: PSO is faster on Ackley, restarts recover Simulated Annealing's reliability at a moderate depth gap once $n \geq 100$, and our network benchmark contains no reliability test, with Adam several times faster. Open directions are a mean-field analysis of the $n \to \infty$ limit, adaptive temperature schedules, and network-training problems on which runs do fail. Appendix F discusses these points in full.

AI use statement

In this work, we used generative AI tools to generate the dataset for the transformer experiments, to assist in the writing of proofs, and to check the validity of algebraic formulae, to implement methods other than Softmin. We did not use generative AI tools to develop theoretical models or conceptual frameworks, to provide critical ingredients for proving mathematical claims, to propose or refine hypotheses, to design or provide feedback on the research methodology or experiments, to assist

with translation, to support qualitative or thematic data analysis, or to interpret results; cleaning or reformatting datasets is not applicable to this work. Additionally, we used generative AI tools to create and edit scientific figures and plotting code, and to draft and edit parts of the paper to improve readability and fit the page limit. We have reviewed all AI-assisted work: all authors checked every proof and derivation, the generated dataset was verified against the definition of modular addition, and the implemented methods where verifyed against literature. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Reproducibility statement

All theoretical results are stated with their assumptions in Sections 2–3, with complete proofs in Appendix C and the large-deviation background in Appendix B. The discretized algorithm is given exactly as run in Algorithm 1. Appendix E lists every landscape, hyperparameter, initialization, step budget and seed count, together with the calibration procedure applied identically to Softmin and to every baseline. Anonymized source code for all experiments and figures is available (Anonymous, 2026).

References

Kwangjun Ahn, Xiang Cheng, Minhak Song, Chulhee Yun, Ali Jadbabaie, and Suvrit Sra. Linear attention is (maybe) all you need (to understand transformer optimization). In *International Conference on Learning Representations (ICLR)*, 2024.

O. Deniz Akyildiz, Francesca Romana Crucinio, Mark Girolami, Tim Johnston, and Sotirios Sabanis. Interacting particle langevin algorithm for maximum marginal likelihood estimation. *ESAIM: Probability and Statistics*, 29:243–280, 2025. doi: 10.1051/ps/2025005.

Anonymous. Softmin energy optimization repository. `https://anonymous.4open.science/r/softmin-iclr-A976/`, 2026. Accessed: 2026-08-31.

Sunith Bandaru and Kalyanmoy Deb. Metaheuristic techniques. In *Decision sciences*, pp. 709–766. CRC Press, 2016.

Nils Berglund. Kramers' law: Validity, derivations and generalisations. *arXiv preprint arXiv:1106.5799*, 2011.

Dimitri P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, USA, 2nd edition, 1999.

Jérôme Bolte, Laurent Miclo, and Stéphane Villeneuve. Swarm gradient dynamics for global optimization: the mean-field limit case. *Mathematical Programming*, 205(1):661–701, 2024.

J. A. Carrillo, Y.-P. Choi, C. Totzeck, and O. Tse. An analytical framework for consensus-based global optimization method. *Mathematical Models and Methods in Applied Sciences*, 28(06): 1037–1066, 2018.

José A Carrillo, Shi Jin, Lei Li, and Yuhua Zhu. A consensus-based global optimization method for high dimensional machine learning problems. *ESAIM: Control, Optimisation and Calculus of Variations*, 27:S5, 2021.

Maurice Clerc and James Kennedy. The particle swarm – explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73, 2002.

W. T. Coffey, Y. P. Kalmykov, and J. T. Waldron. *The Langevin Equation: With Applications to Stochastic Problems in Physics, Chemistry, and Electrical Engineering*. World Scientific, Singapore, 2nd edition, 2004.

Paula Cordero Encinar, Francesca Romana Crucinio, and Omer Deniz Akyildiz. Proximal interacting particle langevin algorithms. In *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pp. 1220–1265. PMLR, 2025.

Marco Dorigo and Thomas Stützle. *Ant Colony Optimization*. MIT Press, Cambridge, MA, USA, 2004.

Benjamin Fehrman, Benjamin Gess, and Arnulf Jentzen. Convergence rates for the stochastic gradient descent method for non-convex objective functions. *Journal of Machine Learning Research*, 21 (136):1–48, 2020.

Massimo Fornasier, Timo Klock, and Konstantin Riedl. Consensus-based optimization methods converge globally. *SIAM Journal on Optimization*, 34(3):2973–3004, 2024.

Mark I. Freidlin and Alexander D. Wentzell. *Random Perturbations of Dynamical Systems*. Springer, 3 edition, 2012.

Saul B. Gelfand and Sanjoy K. Mitter. On sampling methods and annealing algorithms. Technical Report LIDS-P-2008, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA, USA, December 1990.

David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Boston, MA, USA, 1989.

Nikolaus Hansen, Anne Auger, Raymond Ros, Olaf Mersmann, Tea Tušar, and Dimo Brockhoff. COCO: a platform for comparing continuous optimizers in a black-box setting. *Optimization Methods and Software*, 36(1):114–144, 2021.

Kashif Hussain, Mohd Najib Mohd Salleh, Shi Cheng, and Yuhui Shi. Metaheuristic research: a comprehensive survey. *Artificial intelligence review*, 52(4):2191–2233, 2019.

J. E. Jones. On the determination of molecular fields. ii. from the equation of state of a gas. *Proceedings of the Royal Society of London A*, 106(738):463–477, 1924. doi: 10.1098/rspa.1924.0082.

James Kennedy and Russell Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks*, volume 4, pp. 1942–1948. ieee, 1995.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Scott Kirkpatrick, C. Daniel Gelatt, and Mario P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

Shalima Binta Manir and Anamika Paul Rupa. A systematic empirical study of grokking: Depth, architecture, activation, and regularization. *arXiv preprint arXiv:2603.25009*, 2026.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International conference on machine learning*, pp. 6820–6829. PMLR, 2020.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*, 2023.

Mathew M Noel. A new gradient based particle swarm optimization algorithm for accurate computation of global minimum. *Applied Soft Computing*, 12(1):353–359, 2012.

René Pinnau, Claudia Totzeck, Oliver Tse, and Stephan Martin. A consensus-based model for global optimization and its mean-field limit. *Mathematical Models and Methods in Applied Sciences*, 27 (01):183–204, 2017.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: A nonasymptotic analysis. In *Proceedings of the 2017 Conference on Learning Theory*, volume 65 of *Proceedings of Machine Learning Research*, pp. 1674–1703. PMLR, 2017.

Kanchan Rajwar, Kusum Deep, and Swagatam Das. An exhaustive review of the metaheuristic algorithms for search and optimization: taxonomy, applications, and open challenges. *Artificial Intelligence Review*, 56(11):13187–13257, 2023. doi: 10.1007/s10462-023-10470-y.

Konstantin Riedl, Timo Klock, Carina Geldhauser, and Massimo Fornasier. Gradient is all you need? *arXiv preprint arXiv:2306.09778*, 2023.

Yuhui Shi and Russell Eberhart. A modified particle swarm optimizer. In *1998 IEEE International Conference on Evolutionary Computation Proceedings*, pp. 69–73. IEEE, 1998.

Zhao Song, Ron Parr, and Lawrence Carin. Revisiting the softmax bellman operator: New benefits and new perspective. In *International conference on machine learning*, pp. 5916–5925. PMLR, 2019.

Wenpin Tang, Yuhang Wu, and Xun Yu Zhou. Discrete-time simulated annealing: A convergence analysis via the Eyring–Kramers law. *Numerical Algebra, Control and Optimization*, 14(4): 778–794, 2024. doi: 10.3934/naco.2024015.

Amund Tveit, Bjørn Remseth, and Arve Skogvold. Muon optimizer accelerates grokking. *arXiv preprint arXiv:2504.16041*, 2025.

David J. Wales and Jonathan P. K. Doye. Global optimization by basin-hopping and the lowest energy structures of lennard-jones clusters containing up to 110 atoms. *The Journal of Physical Chemistry A*, 101(28):5111–5116, 1997. doi: 10.1021/jp970984n.

Wenlong Wang, Jonathan Machta, and Helmut G. Katzgraber. Population annealing: Theory and application in spin glasses. *Physical Review E*, 92(6), December 2015. ISSN 1550-2376. doi: 10.1103/physreve.92.063307. URL http://dx.doi.org/10.1103/PhysRevE.92.063307.

Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 681–688, 2011.

Mitchell Wortsman, Peter J. Liu, Lechao Xiao, Katie Everett, Alex Alemi, Ben Adlam, John D. Co-Reyes, Izzeddin Gur, Abhishek Kumar, Roman Novak, Jeffrey Pennington, Jascha Sohl-dickstein, Kelvin Xu, Jaehoon Lee, Justin Gilmer, and Simon Kornblith. Small-scale proxies for large-scale transformer training instabilities. *arXiv preprint arXiv:2309.14322*, 2023.

Xin-She Yang. *Nature-inspired optimization algorithms*. Academic Press, 2020.

A Extended Related Work

Metaheuristic optimization methods Metaheuristic optimization methods include Consensus-Based Optimization (CBO) (Carrillo et al., 2018; Fornasier et al., 2024; Pinnau et al., 2017; Carrillo et al., 2021), Genetic Algorithms (Goldberg, 1989), Ant Colony Optimization (Dorigo & Stützle, 2004), and Particle Swarm Optimization (PSO) (Kennedy & Eberhart, 1995; Noel, 2012). Traditional particle-swarm approaches typically operate without explicit gradient information, relying instead on position-based updates to coordinate particles toward promising regions. When reliable gradient information is available, leveraging it enables convergence rates comparable to standard stochastic gradient descent, which gradient-agnostic metaheuristics cannot achieve.

Recent gradient-based swarm methods (Riedl et al., 2023; Bolte et al., 2024; Carrillo et al., 2021; Noel, 2012) have made progress by incorporating gradient information into multi-particle dynamics. However, our approach differs fundamentally in its mechanism for inducing exploration and escaping barriers. Rather than consensus-based averaging or general gradient-weighted potentials, we employ the Softmin energy functional to explicitly create a minimizing/maximizing partition of the particle ensemble: particles below the Softmin threshold are drawn toward local minima via gradient descent, while particles above the threshold are repelled toward maxima. This automatic role assignment (Theorem 2.1) simultaneously reduces effective potential barriers (Theorem 3.4) and forces suboptimal particles into exploratory regimes without requiring manual basin-switching heuristics.

To the best of our knowledge, no other metaheuristic employs the Softmin energy functional, computed as the Softmin of function values at particle positions, as the sole source of inter-particle interaction. This design enables provable escape from suboptimal local minima with a barrier bounded independently of the true barrier height (Theorem 3.4, Section 3), together with a deterministic guarantee that the ensemble's best value never lags ordinary gradient flow under a Polyak–Łojasiewicz condition (Theorem 2.4, Section 2.2). Section 4 compares these predictions with experiments, reporting both where the observed behavior is consistent with the mechanism and where it is not.

Langevin-based sampling and interacting-particle methods Stochastic Gradient Langevin Dynamics (SGLD) (Welling & Teh, 2011) and its non-asymptotic analysis for non-convex optimization (Raginsky et al., 2017) study a single diffusion whose drift is the gradient of the objective itself, with guarantees governed by the spectral gap of the associated Gibbs measure; when multiple particles are used, interaction typically enters only through averaging over independent runs. Interacting Particle Langevin Dynamics (IPLD) (Akyildiz et al., 2025; Cordero Encinar et al., 2025) instead couples an ensemble of particles through their empirical measure, using this mean-field interaction to build unbiased or stabilized estimators of the marginal likelihood gradient in latent-variable models. Our construction is closer in spirit to IPLD, in that it also relies on genuine ensemble-level coupling rather than independent copies, but the interaction itself is different in kind: instead of pooling particles through an empirical average, the Softmin energy J_β couples particles through their relative energies, producing a state-dependent, sign-changing effective learning rate $A_k(\mathbf{x})$ that deterministically assigns each particle a minimizing or maximizing role (Theorem 2.1). It is this competitive, temperature-sharpened coupling, rather than mean-field averaging, that produces the barrier-capping mechanism of Theorem 3.4, and it is aimed directly at global optimization of f rather than at marginal-likelihood estimation.

Connection to Statistical Physics and Functional Design Our approach shares conceptual roots with population-based statistical physics methods such as Population Annealing (PA) (Wang et al., 2015), which also employs a Gibbs-weighted ensemble to locate global ground states. However, while PA relies on discrete demographic selection, duplicating low-energy particles and eliminating high-energy ones prior to local Markov Chain Monte Carlo steps, our method maintains a fixed population size and drives exploration through continuous dynamical repulsion. This active exploration mechanism differentiates our Softmin energy functional from the standard log-sum-exp (or free-energy) functional, $F_\beta(\mathbf{x}) = -\beta^{-1} \log \sum e^{-\beta f(x_i)}$. The gradient of the log-sum-exp formulation, $\nabla_{x_k} F_\beta(\mathbf{x}) = w_k \nabla f(x_k)$, strictly preserves gradient descent, causing sub-optimal particles with small weights to simply stagnate. In contrast, the gradient of our proposed Softmin functional, $J_\beta(\mathbf{x}) = \sum w_i f(x_i)$, natively incorporates the sign-changing effective learning rate, $A_k(\mathbf{x})$. Consequently, rather than discarding trapped particles, the geometry of the Softmin functional actively

destabilizes their local minima, forcing high-energy particles to hill-climb and escape via the energy barrier-flattening mechanism formalized in our theoretical analysis.

Softmin energy Theoretical work on the Gibbs Softmin (or Boltzmann Softmin) has primarily focused on its properties as a smooth approximation to the minimum operator (Bertsekas, 1999), convergence of softmax policy gradients in reinforcement learning (Mei et al., 2020), and its role in exploration–exploitation trade-offs via temperature scheduling (Song et al., 2019). To our knowledge, this is the first work employing J_β as an optimization objective for non-convex global search. This is conceptually distinct from swarm-gradient methods that anneal a mean-field particle density toward the Dirac mass at the global minimizer (Bolte et al., 2024): rather than acting on a population density, the Softmin energy couples particles directly through the relative ranking of their objective values. The effective barrier reduction via particle cooperation, analyzed in Theorem 3.4, is previously unstudied.

B Freidlin–Wentzell Large Deviation Theory

This appendix summarizes the background large deviation theory used by Theorem 3.4 and Corollary 3.5. Consider a small-noise gradient diffusion on $\mathbb{R}^N$,

$$\mathrm{d}\mathbf{X}_t = -\nabla U(\mathbf{X}_t)\mathrm{d}t + \sqrt{2\sigma}\mathrm{d}B_t,$$

for a potential $U \in C^2(\mathbb{R}^N)$ and standard N-dimensional Brownian motion $(B_t)_{t\geq 0}$; in our application $N = nd$ and $U = nJ_\beta$. As $\sigma \downarrow 0$, sample paths of this diffusion concentrate around trajectories of the noiseless flow $\dot{\mathbf{X}}_t = -\nabla U(\mathbf{X}_t)$, and rare deviations from this flow are governed by the Freidlin–Wentzell large deviation principle (Freidlin & Wentzell, 2012). For an absolutely continuous path $\phi : [0, T] \to \mathbb{R}^N$, the Freidlin–Wentzell action is

$$I_{0T}(\phi) := \frac{1}{4}\int_0^T \left\|\dot{\phi}(t) + \nabla U(\phi(t))\right\|^2 \mathrm{d}t, \tag{18}$$

and the quasipotential between a point $x \in \mathbb{R}^N$ and a closed set $A \subset \mathbb{R}^N$ is

$$V(x, A) := \inf_{T>0} \inf_{\substack{\phi(0)=x \\ \phi(T)\in A}} I_{0T}(\phi). \tag{19}$$

Intuitively, $I_{0T}(\phi)$ penalizes deviation from the noiseless flow, vanishing exactly along trajectories of $\dot{\phi} = -\nabla U(\phi)$; $V(x, A)$ is the cheapest such penalty over all paths from x reaching A, and quantifies how atypical (large-deviation-improbable) it is for the diffusion to travel from x to A without the help of noise. The infimum in (19) need not be attained by any single finite-time path: it is generally only approximated arbitrarily well, e.g. by following the time-reversed noiseless flow toward x and then the forward flow toward A, taking arbitrarily long to approach the relevant critical points. This is the sense in which the proof of Theorem 3.4 below computes exactly the minimal action along one specific channel of paths; the quasipotential itself is not computed, but bounded from above by that value, since it is an infimum over all paths.

Under the standard hypotheses of Freidlin–Wentzell exit theory (Freidlin & Wentzell, 2012) – in particular that D is a bounded domain with $x \in D$, that x is the unique asymptotically stable equilibrium of the noiseless flow in D, and that ∂D contains no other equilibria – the exit time τ_D^σ of the diffusion from D, started at x, satisfies

$$\lim_{\sigma\downarrow 0} \sigma \log \mathbb{E}_x \tau_D^\sigma = V(x, \partial D). \tag{20}$$

For a gradient drift, the action of any path admits the exact decomposition

$$I_{0T}(\phi) = U(\phi(T)) - U(\phi(0)) + \frac{1}{4}\int_0^T \left\|\dot{\phi}(t) - \nabla U(\phi(t))\right\|^2 \mathrm{d}t, \tag{21}$$

obtained by expanding the square in (18) and completing it against $\mathrm{d}U(\phi(t))/\mathrm{d}t = \langle \nabla U(\phi(t)), \dot{\phi}(t)\rangle$. Consequently, any segment of ϕ that follows the noiseless flow *downhill* (i.e. $\dot{\phi} = -\nabla U(\phi)$) contributes zero action, while a segment moving *uphill* against the flow (i.e. $\dot{\phi} = \nabla U(\phi)$) costs exactly the potential increase along that segment. This identity is what allows the proof of Theorem 3.4 to compute exactly the minimal action along a specific channel of admissible paths, and hence to bound the quasipotential from above.

C TECHNICAL PROOFS

Here we will collect some of the more technical proofs of the lemmas stated in the main body.

C.1 PROOFS OF SECTION 2

Lemma C.1 (Component gradient and Hessian at critical configurations)**.** *Let* $w_k = e^{-\beta f(x_k)}(\sum_j e^{-\beta f(x_j)})^{-1}$ *denote the Gibbs weight of particle* x_k*, and let* $A_k(\mathbf{x}) = nw_k\left[1-\beta(f(x_k)-J_\beta(\mathbf{x}))\right]$ *be the dynamic pre-factor. Then*

$$\nabla_{x_k}(nJ_\beta(\mathbf{x})) = A_k(\mathbf{x})\nabla f(x_k), \qquad \sum_{k=1}^{n} A_k(\mathbf{x}) = n. \tag{22}$$

Moreover, for $k \neq \ell$ *the mixed Hessian block is the outer product*

$$\frac{\partial^2(nJ_\beta(\mathbf{x}))}{\partial x_\ell \partial x_k} = \nabla f(x_k)\big(\nabla_{x_\ell} A_k(\mathbf{x})\big)^\top, \tag{23}$$

which vanishes whenever $\nabla f(x_k) = 0$*. Consequently, if* $\mathbf{x}$ *is such that* $\nabla f(x_k) = 0$ *for every* k*, then* $\nabla(nJ_\beta(\mathbf{x})) = 0$ *and the Hessian is block diagonal,*

$$\nabla^2(nJ_\beta(\mathbf{x})) = \mathrm{blockdiag}\big(A_1(\mathbf{x})\nabla^2 f(x_1), \ldots, A_n(\mathbf{x})\nabla^2 f(x_n)\big). \tag{24}$$

Finally, if $A_k(\mathbf{x}) = 0$ *then* $\frac{\partial^2(nJ_\beta(\mathbf{x}))}{\partial x_k^2} = -n\beta w_k \nabla f(x_k)\nabla f(x_k)^\top$.

Proof. We compute the derivatives using the chain rule and the properties of the Gibbs weights. First, we establish the gradient of the weight w_i with respect to x_k:

$$\nabla_{x_k} w_i = \begin{cases} \beta w_i w_k \nabla f(x_k) & \text{if } i \neq k, \\ -\beta w_k(1-w_k)\nabla f(x_k) & \text{if } i = k. \end{cases}$$

For the first derivative of $nJ_\beta(\mathbf{x}) = n\sum_i w_i f(x_i)$, an identical computation to the main text gives $\nabla_{x_k}(nJ_\beta(\mathbf{x})) = nw_k[1-\beta(f(x_k)-J_\beta(\mathbf{x}))]\nabla f(x_k) = A_k(\mathbf{x})\nabla f(x_k)$; summing over k and using $\sum_k w_k = 1$, $\sum_k w_k f(x_k) = J_\beta(\mathbf{x})$, gives $\sum_k A_k(\mathbf{x}) = n[1-\beta(J_\beta(\mathbf{x})-J_\beta(\mathbf{x}))] = n$.

For the mixed second derivative ($k \neq \ell$), we differentiate $\nabla_{x_k}(nJ_\beta(\mathbf{x})) = A_k(\mathbf{x})\nabla f(x_k)$ with respect to x_ℓ. Because $\nabla f(x_k)$ is independent of x_ℓ, the Jacobian matrix is the outer product $\nabla f(x_k)(\nabla_{x_\ell} A_k(\mathbf{x}))^\top$. Substituting $\nabla_{x_\ell} w_k = \beta w_k w_\ell \nabla f(x_\ell)$ and $\nabla_{x_\ell} J_\beta(\mathbf{x}) = \frac{1}{n}A_\ell(\mathbf{x})\nabla f(x_\ell)$,

$$\begin{aligned}\nabla_{x_\ell} A_k(\mathbf{x}) &= n(\nabla_{x_\ell} w_k)\left[1-\beta(f(x_k)-J_\beta(\mathbf{x}))\right] + nw_k\beta\left(\nabla_{x_\ell} J_\beta(\mathbf{x})\right) \\ &= \beta w_\ell\big(nw_k[1-\beta(f(x_k)-J_\beta(\mathbf{x}))]\big)\nabla f(x_\ell) + \beta w_k A_\ell(\mathbf{x})\nabla f(x_\ell) \\ &= \beta\left[w_\ell A_k(\mathbf{x}) + w_k A_\ell(\mathbf{x})\right]\nabla f(x_\ell),\end{aligned}$$

which gives (23) and vanishes whenever $\nabla f(x_k) = 0$ (or symmetrically $\nabla f(x_\ell) = 0$). If $\nabla f(x_k) = 0$ for every k, every off-diagonal block vanishes and (22) gives $\nabla(nJ_\beta(\mathbf{x})) = 0$; the diagonal block computed below then reduces to $A_k(\mathbf{x})\nabla^2 f(x_k)$, giving the block-diagonal form (24).

Finally, for the diagonal block ($k = \ell$), applying the product rule to $\nabla_{x_k}(nJ_\beta(\mathbf{x})) = A_k(\mathbf{x})\nabla f(x_k)$ includes the local Hessian $\nabla^2 f(x_k)$:

$$\frac{\partial^2(nJ_\beta(\mathbf{x}))}{\partial x_k^2} = A_k(\mathbf{x})\nabla^2 f(x_k) + \nabla f(x_k)\big(\nabla_{x_k} A_k(\mathbf{x})\big)^\top.$$

Using $\nabla_{x_k} w_k = -\beta w_k(1-w_k)\nabla f(x_k)$ and $\nabla_{x_k} J_\beta(\mathbf{x}) = \frac{1}{n}A_k(\mathbf{x})\nabla f(x_k)$, a computation identical in structure to the mixed case gives

$$\nabla_{x_k} A_k(\mathbf{x}) = -\beta\left[nw_k + A_k(\mathbf{x}) - 2w_k A_k(\mathbf{x})\right]\nabla f(x_k),$$

so that, substituting back,

$$\frac{\partial^2(nJ_\beta(\mathbf{x}))}{\partial x_k^2} = A_k(\mathbf{x})\nabla^2 f(x_k) - \beta\left[nw_k + A_k(\mathbf{x}) - 2w_k A_k(\mathbf{x})\right]\nabla f(x_k)\nabla f(x_k)^\top,$$

which reduces to $-n\beta w_k \nabla f(x_k)\nabla f(x_k)^\top$ when $A_k(\mathbf{x}) = 0$. □

Proof of Theorem 2.1. A locally asymptotically stable equilibrium of the gradient flow (3) is an isolated local minimum of nJ_β: indeed $\frac{d}{dt}(nJ_\beta(\mathbf{x}(t))) = -\|\nabla(nJ_\beta)\|^2 \le 0$, so if points arbitrarily close to $\bar{\mathbf{x}}$ had strictly smaller value the trajectories starting there could not converge to $\bar{\mathbf{x}}$; and a non-isolated equilibrium admits nearby constant trajectories that likewise do not converge to $\bar{\mathbf{x}}$.

Fix k and freeze the other particles at $\bar{x}_j$, $j \neq k$. Since nJ_β depends on x_k only through $r = f(x_k)$, we may write

$$nJ_\beta(\bar{x}_1, \dots, x_k, \dots, \bar{x}_n) = g(f(x_k)), \qquad g(r) := n\,\frac{re^{-\beta r} + C_1}{e^{-\beta r} + C_0},$$

with $C_0 := \sum_{j\neq k} e^{-\beta f(\bar{x}_j)} > 0$ and $C_1 := \sum_{j\neq k} f(\bar{x}_j)e^{-\beta f(\bar{x}_j)}$. By the chain rule and Lemma C.1, wherever $\nabla f(x_k) \neq 0$ one has $g'(f(x_k)) = A_k(\mathbf{x})$; by continuity of both sides in $f(x_k)$, $g'(f(\bar{x}_k)) = A_k(\bar{\mathbf{x}})$. Writing $g'(r) = n\omega(r)[1 - \beta(r - g(r)/n)]$ with $\omega(r) = e^{-\beta r}/(e^{-\beta r} + C_0) \in (0, 1)$, a direct computation shows that at any critical point r_0 of g (where the bracket vanishes), $g''(r_0) = -n\beta\omega(r_0) < 0$.

Step 1: $A_k(\bar{\mathbf{x}}) \neq 0$. Suppose $A_k(\bar{\mathbf{x}}) = 0$. By the above, $r_0 = f(\bar{x}_k)$ is a critical point of g, and it is a strict local maximum. If f is non-constant near $\bar{x}_k$, there are points y arbitrarily close to $\bar{x}_k$ with $f(y)$ close to but different from r_0, whence $nJ_\beta(\dots, y, \dots) = g(f(y)) < g(r_0) = nJ_\beta(\bar{\mathbf{x}})$, contradicting local minimality. If f is constant near $\bar{x}_k$, moving x_k within that level set leaves every objective value, Gibbs weight, and gradient unchanged, producing a continuum of equilibria arbitrarily close to $\bar{\mathbf{x}}$ and contradicting isolation. Hence $A_k(\bar{\mathbf{x}}) \neq 0$.

Step 2: $\nabla f(\bar{x}_k) = 0$. Since $A_k(\bar{\mathbf{x}}) \neq 0$, $g'(f(\bar{x}_k)) \neq 0$, so g is strictly monotone near $f(\bar{x}_k)$. If $\nabla f(\bar{x}_k) \neq 0$, then f takes values on both sides of $f(\bar{x}_k)$ arbitrarily near $\bar{x}_k$; strict monotonicity of g then yields nearby configurations with $nJ_\beta < nJ_\beta(\bar{\mathbf{x}})$, contradicting local minimality. Hence $\nabla f(\bar{x}_k) = 0$.

Step 3: classification. If $A_k(\bar{\mathbf{x}}) > 0$, then g is strictly increasing near $f(\bar{x}_k)$, so local minimality of nJ_β forces $f(x_k) \ge f(\bar{x}_k)$ for x_k near $\bar{x}_k$; thus $\bar{x}_k$ is a local minimum of f. If $A_k(\bar{\mathbf{x}}) < 0$, the symmetric argument shows $\bar{x}_k$ is a local maximum. Because $nw_k > 0$, the sign of $A_k(\bar{\mathbf{x}})$ is that of $1 - \beta(f(\bar{x}_k) - J_\beta(\bar{\mathbf{x}}))$, giving the stated threshold. Finally, $\sum_k A_k(\bar{\mathbf{x}}) = n$ forces some $A_k(\bar{\mathbf{x}}) > 0$, so at least one particle is at a local minimum of f. □

Proof of Theorem 2.4. Fix t and let i attain the minimum, $f(x_i(t)) = m(t)$. Since $f(x_i(t)) \le f(x_j(t))$ for all j, we have $e^{-\beta_t f(x_j(t))} \le e^{-\beta_t f(x_i(t))}$, hence $\sum_j e^{-\beta_t f(x_j(t))} \le ne^{-\beta_t f(x_i(t))}$ and therefore $nw_i(t) \ge 1$. As $J_{\beta_t}(\mathbf{x}(t))$ is a convex combination of the values $f(x_j(t))$, it satisfies $J_{\beta_t}(\mathbf{x}(t)) \ge f(x_i(t))$, so $1 - \beta_t(f(x_i(t)) - J_{\beta_t}(\mathbf{x}(t))) \ge 1$. Multiplying,

$$A_i(\mathbf{x}(t)) \ge 1 \qquad \text{for every } i \text{ attaining the minimum at } t. \tag{25}$$

Along such a particle,

$$\frac{d}{dt} f(x_i(t)) = \langle \nabla f(x_i(t)), \dot{x}_i(t) \rangle = -A_i(\mathbf{x}(t))\|\nabla f(x_i(t))\|^2 \le -\|\nabla f(x_i(t))\|^2.$$

Consider $m = \min_k g_k$ with $g_k(t) = f(x_k(t))$. Since β_t is only assumed measurable, each x_k is a Carathéodory solution and hence absolutely continuous; as f has locally Lipschitz gradient, each g_k is absolutely continuous on compact time intervals, and a pointwise minimum of finitely many absolutely continuous functions is absolutely continuous. Therefore m is absolutely continuous, and m together with every g_k is differentiable at a.e. t. At any such t, choose i attaining the minimum; the function $g_i - m \ge 0$ is then differentiable at t and vanishes there, so it has zero derivative at t, giving $m'(t) = g_i'(t)$. Combined with the previous display,

$$m'(t) = -A_i(\mathbf{x}(t))\|\nabla f(x_i(t))\|^2 \le -\|\nabla f(x_i(t))\|^2 \le 0 \qquad \text{for a.e. } t.$$

Integrating, m is nonincreasing, so every minimizing particle stays in $S_0 = \{x : f(x) \le m(0)\}$, where the Polyak–Łojasiewicz condition (Definition 2.3) applies:

$$m'(t) \le -\|\nabla f(x_i(t))\|^2 \le -2\mu\big(f(x_i(t)) - f_\star\big) = -2\mu\big(m(t) - f_\star\big).$$

Setting $y(t) := e^{2\mu t}(m(t) - f_\star)$, which is absolutely continuous, we get $y'(t) = e^{2\mu t}[m'(t) + 2\mu(m(t) - f_\star)] \le 0$ for a.e. t, so $y(t) \le y(0)$, i.e.

$$m(t) - f_\star \le e^{-2\mu t}\big(m(0) - f_\star\big), \qquad t \ge 0. \qquad \square$$

Remark C.2 (Global existence). Theorem 2.4 presupposes a global solution. Since m is nonincreasing, every particle attaining the running minimum stays in the compact sublevel set S_0 when f is coercive; the theorem controls these particles only. Maximizing particles ($A_k(\mathbf{x}(t)) < 0$) are not controlled by this estimate, but they are damped strongly enough that blow-up is easy to exclude. Write $u_k(t) := f(x_k(t)) - m(t) \geq 0$. From $\sum_j e^{-\beta_t f(x_j)} \geq e^{-\beta_t m}$ we get $w_k \leq e^{-\beta_t u_k}$, and hence $\beta_t\big(J_{\beta_t}(\mathbf{x}) - m\big) = \beta_t \sum_j w_j u_j \leq \beta_t \sum_j u_j e^{-\beta_t u_j} \leq n/e$. Therefore

$$|A_k(\mathbf{x}(t))| \;\leq\; n\big(1 + \beta_t u_k + n/e\big)e^{-\beta_t u_k} \;\leq\; n\big(1 + n/e\big),$$

using $(1+v)e^{-v} \leq 1$ and $e^{-v} \leq 1$ for $v \geq 0$. The pre-factor is thus bounded uniformly and decays exponentially in the particle's energy gap, so for coercive f whose gradient grows at most polynomially in f the drift stays bounded along trajectories and global solutions exist. We nonetheless state the theorem for a given global solution, since its conclusion concerns only the minimizing particles.

C.2 Proofs of Section 3

Remark C.3 (Metastability of the initial state). At the consensus configuration $\bar{\mathbf{x}} = (z_-, \dots, z_-)$, every Gibbs weight equals $1/n$, so $A_k(\bar{\mathbf{x}}) \equiv 1$ for every k. By Corollary 2.2, using $f''(z_-) > 0$, $\bar{\mathbf{x}}$ is therefore a strict local minimum of nJ_β and a locally asymptotically stable equilibrium of (3), i.e. genuinely metastable for the noisy dynamics (2) at small σ.

Proof of Lemma 3.2. Since z_- is a critical point of f, every frozen coordinate satisfies $\dot{x}_j = -A_j(\Gamma(x))f'(z_-) = 0$ by (3), so $\{\Gamma(x) : x \in \mathbb{R}\}$ is invariant. Differentiating (9) along the channel gives $W'(x) = \frac{d}{dx}\big[nJ_\beta(\Gamma(x))\big] = u'(r(x))r'(x) = u'(r(x))f'(x)$; applying the chain rule directly to nJ_β and (4) gives the same derivative as $A_1(\Gamma(x))f'(x)$. Hence $W'(x) = A_1(\Gamma(x))f'(x)$, matching the active coordinate's drift $-A_1(\Gamma(x))f'(x)$ in (3), and in particular reproving $A_1(\Gamma(x)) = u'(r(x))$, made explicit in closed form in Lemma 3.3 below. □

Proof of Lemma 3.3. Write $r := r(x)$. Factoring $e^{-\beta f(z_-)}$ out of the Gibbs weights at $\Gamma(x)$,

$$w_1(\Gamma(x)) = \frac{e^{-\beta r}}{n - 1 + e^{-\beta r}}, \qquad w_j(\Gamma(x)) = \frac{1}{n - 1 + e^{-\beta r}} \quad (j \geq 2),$$

whence $J_\beta(\Gamma(x)) - f(z_-) = re^{-\beta r}/(n - 1 + e^{-\beta r}) = u(r)/n$, which is (9). Therefore

$$f(x) - J_\beta(\Gamma(x)) = r - \frac{re^{-\beta r}}{n - 1 + e^{-\beta r}} = \frac{(n-1)r}{n - 1 + e^{-\beta r}},$$

and substituting into $A_1(\Gamma(x)) = nw_1(\Gamma(x))[1 - \beta(f(x) - J_\beta(\Gamma(x)))]$ gives

$$A_1(\Gamma(x)) = \frac{ne^{-\beta r}}{n - 1 + e^{-\beta r}} \cdot \frac{n - 1 + e^{-\beta r} - \beta(n-1)r}{n - 1 + e^{-\beta r}} = \frac{ne^{-\beta r}q(r)}{(n - 1 + e^{-\beta r})^2}.$$

Differentiating (9) directly yields the same expression for $u'(r)$, proving (10), consistently with the identity $A_1(\Gamma(x)) = u'(r(x))$ already established in Lemma 3.2. In particular $A_1(\Gamma(x)) = 0$ if and only if $q(r) = 0$, which by the displayed formula for $f(x) - J_\beta$ is equivalent to $f(x) - J_\beta(\Gamma(x)) = 1/\beta$, i.e. to the vanishing of the Softmin threshold gap of the active particle.

Since $q'(r) = -\beta(n-1) - \beta e^{-\beta r} < 0$ for all $r \in \mathbb{R}$, $q(0) = n > 0$, and $q(r) \to -\infty$ as $r \to \infty$, q has a unique zero $r_c > 0$. From $q(1/\beta) = e^{-1} > 0$ we get $r_c > 1/\beta$; and $q(r_c) = 0$ reads $(n-1)(\beta r_c - 1) = e^{-\beta r_c} < 1$, giving $\beta r_c < n/(n-1)$. This is (11). As $ne^{-\beta r}/(n-1+e^{-\beta r})^2 > 0$, the sign of u' is the sign of q, so u is strictly increasing on $(-\infty, r_c]$ and strictly decreasing on $[r_c, \infty)$. □

Proof of Theorem 3.4. Part 1. For $r \leq 0$ we have $u(r) \leq 0$. For $r > 0$, since $n - 1 + e^{-\beta r} > n - 1$,

$$u(r) < \frac{n}{n-1} re^{-\beta r} \leq \frac{n}{n-1} \max_{\rho \geq 0} \rho e^{-\beta\rho} = \frac{n}{e(n-1)\beta},$$

the maximum being attained at $\rho = 1/\beta$. Hence $u_c = \max_{\mathbb{R}} u < n/(e(n-1)\beta)$, which is (13); the bound on r_c is (11).

Part 2. A channel path has the form $\phi(t) = \Gamma(\psi(t))$; the frozen coordinates stay at the critical point z_-, so their drift vanishes and, with $W(x) := u(r(x))$, the action reduces to $I_{0T} = \frac{1}{4}\int_0^T |\dot\psi + W'(\psi)|^2 \mathrm{d}t$. The one-dimensional version of (21) is

$$I_{0T} = W(\psi(T)) - W(\psi(0)) + \frac{1}{4}\int_0^T |\dot\psi - W'(\psi)|^2 \mathrm{d}t.$$

By (7), r is a strictly increasing bijection from $[z_-, s]$ onto $[0, h]$, so there is a unique $x_m \in (z_-, s]$ with $r(x_m) = \min\{h, r_c\}$. On $[z_-, x_m]$ we have $r(x) \le r_c$, so by Lemma 3.3 W is strictly increasing there, with total rise $W(x_m) - W(z_-) = u_h$; on $[x_m, s]$ we have $r(x) \ge r_c$, so W is nonincreasing. Every continuous channel path from z_- to s passes through x_m, and applying the displayed decomposition to the sub-path up to the first hitting time of x_m bounds the action from below by u_h. This value is attained in the quasipotential sense by following the reversed deterministic flow from z_- up to x_m (cost exactly u_h) and then the forward flow toward s at zero additional cost, with the standard finite-time approximations near the critical points. Therefore the channel quasipotential equals u_h, and since V_β is an infimum over the larger class of all paths in $\mathbb{R}^n$,

$$V_\beta(\Gamma(z_-), \Gamma(s)) \le u_h.$$

Because $u(r)/r = ne^{-\beta r}/(n - 1 + e^{-\beta r}) < 1$ for $r > 0$, we have $u_h < \min\{h, r_c\} \le h$, which gives (14).

Part 3. Assume $r_c < h$. At $\Gamma(s)$ every particle sits at a critical point of f (either s or z_-). For the active particle, (10) gives $A_1(\Gamma(s)) = u'(h) < 0$ because $h > r_c$, and $f''(s) < 0$ since s is a nondegenerate maximum. For a frozen particle at z_-,

$$A_j(\Gamma(s)) = n w_j(\Gamma(s))\big[1 + \beta\big(J_\beta(\Gamma(s)) - f(z_-)\big)\big] > 0$$

by (9), since $J_\beta(\Gamma(s)) - f(z_-) = u(h)/n > 0$, and $f''(z_-) > 0$. Hence Corollary 2.2 applies and $\Gamma(s)$ is a strict local minimum of nJ_β. On the z_- side, r decreases strictly from h through r_c as x moves from s toward z_- (recall $x_m \in (z_-, s)$ is where $r(x_m) = r_c$, well-defined since $r_c < h$), so the channel well at $\Gamma(s)$ has barrier $u_c - u(h)$ on this side unconditionally — this uses only $r_c < h$, not ρ_+.

The z_+ side depends on ρ_+. On (s, z_+), r decreases strictly from h to ρ_+ by (7)–(8).

If $\rho_+ < r_c$, then r crosses r_c on this interval, so there is a unique ξ_+ with $r(\xi_+) = r_c$. Between s and ξ_+ the reduced potential rises from $u(h)$ to u_c; the gradient-action argument of Part 2 gives channel action $u_c - u(h)$, and since $0 < u(h) < u_c$ we obtain (16), so the well has matching barrier heights on both sides. Finally, for $x \in (\xi_+, z_+)$ we have $r(x) < r_c$, hence $A_1(\Gamma(x)) = u'(r(x)) > 0$ by (10), while $f'(x) < 0$; therefore the active coordinate of the deterministic channel flow satisfies $\dot x = -A_1(\Gamma(x))f'(x) > 0$, driving the particle toward z_+. Concatenating the admissible path of Part 2 (from $\Gamma(z_-)$ to $\Gamma(s)$, action u_h) with this one (from $\Gamma(s)$ to $\Gamma(\xi_+)$, action $u_c - u(h)$) gives an admissible path from $\Gamma(z_-)$ to $\Gamma(\xi_+)$ of total action $u_h + (u_c - u(h))$, proving (17).

If instead $\rho_+ \ge r_c$, then $r(x) \in [r_c, h)$ for every $x \in (s, z_+)$: the interval lies entirely on the decreasing side of u, where u is strictly decreasing in r while r is strictly decreasing in x, so $u(r(x))$ is strictly increasing in x throughout (s, z_+), from $u(h)$ to $u(\rho_+)$. No interior peak occurs, so no point ξ_+ with $r(\xi_+) = r_c$ exists in (s, z_+): the channel potential rises monotonically to $\Gamma(z_+)$ with no intermediate barrier to cross, and the total rise $u(\rho_+) - u(h)$ is at most the z_--side barrier $u_c - u(h)$, since $u(\rho_+) \le u_c$. In this regime $A_1(\Gamma(z_+)) = u'(\rho_+) \le 0$ while $f''(z_+) > 0$, so by Theorem 2.1 $\Gamma(z_+)$ is not itself a stable equilibrium, and no post-saddle well forms on this side.

If instead $h \le r_c$, then $r(x) \le h \le r_c$ for x near s, with equality only at $x = s$, and the strict monotonicity of u on $(-\infty, r_c]$ shows that $x \mapsto u(r(x))$ has a strict local maximum at s. Hence $\Gamma(s)$ is not a local minimum of nJ_β. □

D Discussion of the Barrier-Flattening Theorem

This appendix collects the discussion of Theorem 3.4 that the main text condenses: the low-barrier regime and the comparison with independent Langevin dynamics in full, the scope of the theorem, and the extension to higher dimensions together with the inverse-temperature schedule.

Remark D.1 (Low-barrier regime: quasipotential reduction without sign reversal). By (10), $A_1(\Gamma(x)) > 0$ precisely when $r(x) < r_c$. Hence $u_h = u(h)$, i.e. $\min\{h, r_c\} = h$, exactly in the regime $h \le r_c$, and there the drift of the active particle is minimizing along the entire crossing: the sign-reversal mechanism is never activated, the crossing is made entirely against the drift, and by Theorem 3.4(3) no Softmin-induced well appears at s. The quasipotential reduction is nonetheless strict, since $u(r)/r = ne^{-\beta r}/(n-1+e^{-\beta r}) < 1$ for $r > 0$ gives $u(h) < h$. In this regime the gain comes purely from the Gibbs-weight damping of the leading particle's contribution to nJ_β, not from a sign reversal. By (11) the regime is $h \lesssim \beta^{-1}$; the sign-reversal mechanism is what handles barriers beyond that scale.

Remark D.2 (Interpretation and scope). Part 1 shows that, along this channel, an arbitrarily high physical barrier h is flattened to a level of order β^{-1}: as h grows the active particle receives exponentially small Gibbs weight, so the barrier saturates rather than growing with h. Part 2 gives a strict full-system *upper* bound on the exit exponent through an explicit admissible path; because V_β is an infimum over all paths in $\mathbb{R}^n$, the channel construction can only over-estimate it. It is a bound on the Freidlin–Wentzell quasipotential itself, established by exhibiting an admissible path; it does not yet assert anything about $\mathbb{E}\tau^\sigma$, for which see Corollary 3.5. Part 3 addresses the post-saddle obstruction: a high barrier turns the physical maximum s into a Softmin-induced well. When $\rho_+ < r_c$ this well has equal-height barriers, with depth $u_c - u(h)$ strictly below the initial barrier u_c, so it is not a worse obstruction than the one already crossed, giving the composite two-stage bound (17); when $\rho_+ \ge r_c$ the z_+ side presents no intermediate barrier at all, and its monotone rise $u(\rho_+) - u(h)$ is itself bounded by $u_c - u(h)$, so this case is only more favorable. We emphasize that (17) bounds a quasipotential, for which the two costs do add. If $\Gamma(s)$ were itself metastable in the sense of Corollary 3.5, so that the two crossings are traversed as successive escapes each restarting the clock, the resulting exit-time exponent would instead be governed by the larger of them, $\max\{u_h, u_c - u(h)\} \le u_c$, since the dominant crossing would set the asymptotic rate; we do not establish that here. The theorem does *not* identify the exact transition exponent between the two basins, which would require ruling out returns to the initial basin and competing collective paths in $\mathbb{R}^n$.

Remark D.3 (Comparison with independent Langevin dynamics). For a single overdamped Langevin particle in the potential f started at z_-, the classical small-noise exit exponent from the basin of z_- is the physical barrier $h = f(s) - f(z_-)$. For n *independent* copies, the first of the n exit times has the same exponential scale h: taking a minimum over n i.i.d. variables with common exponential rate h/σ changes only the multiplicative pre-factor, by a factor $1/n$, not the exponent, as $\sigma \downarrow 0$.

By (14), the channel construction gives $V_\beta(\Gamma(z_-), \Gamma(s)) \le u_h < h$ for the joint Softmin diffusion, in *every* regime of h (see Remark D.1). This is a comparison of an *upper bound on a quasipotential* for the interacting system against the *exact* exponent h of the noninteracting one; it shows the admissible channel gives a strictly lower quasipotential bound than independent escape, not that the true joint exit exponent is smaller. The stronger, exit-*time* statement that the interaction cannot slow escape holds only under the hypotheses of Corollary 3.5, which are not verified in this paper.

Remark D.4 (Higher dimensions, temperature, and scale). The channel identity of Part 1 extends algebraically to $d > 1$: at a critical point s of f the active Hessian block is $u'(h)\nabla^2 f(s)$, so if s is an index-one saddle and $u'(h) < 0$ the induced local minimum is a genuinely one-dimensional phenomenon, and transverse unstable directions may open cheaper escape routes. Although Theorem 2.4 holds for an arbitrary schedule β_t, and Section 3 fixed β purely for tractability of the exit-time analysis, a natural question is whether annealing β helps in practice. Regarding temperature, the fixed-β bound (13) decreases like β^{-1}, favoring large β for barrier reduction; but large β also suppresses the drift of non-leading particles through their Gibbs weights, since a particle trailing the ensemble minimum by a gap Δ is weighted by $e^{-\beta\Delta}$. Both limits degrade the mechanism: as $\beta \to \infty$ the ensemble degenerates to leader-only drift, while as $\beta \to 0$ the Softmin coupling degenerates toward independent noisy gradient descent and (13)'s bound itself diverges. A decaying (or, symmetrically, an increasing) schedule β_t is a natural candidate for balancing these two effects over time; however, we found empirically that neither improves on a well-chosen *fixed* β on the analytic benchmarks considered here. Both a geometric decay from a large starting β and a geometric growth from a small one consistently reach the target *more slowly* than a fixed β tuned once via the diagnostic of Appendix E, with the gap widening at harder barrier heights, and an aggressive growth rate can drive β to numerical overflow before the schedule ever reaches a useful range. We report this as a negative empirical result rather than infer a general claim about annealing from it: a rigorous

large-deviation treatment of β_t, and of whether some other schedule shape or landscape can realize the exploration/refinement trade-off this remark motivates, remain open.

E Appendix: Experimental Details and Hyperparameters

To ensure full reproducibility of the numerical experiments presented in Section 4, this appendix details the hyperparameters and initialization schemes used across all benchmark landscapes. All continuous-time dynamics were discretized using the Euler-Maruyama method. The source code used to execute these experiments is publicly available at (Anonymous, 2026).

Algorithm 1 states the discretized method exactly as run.

Algorithm 1 Softmin energy dynamics: Euler–Maruyama discretization of (2).

Require: objective f and gradient ∇f; ensemble size n; inverse temperature β; step size η; noise intensity σ; box B with coordinatewise projection Π_B; horizon T

draw $x_1, \dots, x_n$ i.i.d. uniformly on the initialization region (Table 1)
for $t = 1, \dots, T$ **do**
  $f_k \leftarrow f(x_k)$ for every k, and $f_* \leftarrow \min_j f_j$
  $w_k \leftarrow e^{-\beta(f_k - f_*)} \big/ \sum_j e^{-\beta(f_j - f_*)}$ ▷ shifted for numerical stability
  $J \leftarrow \sum_k w_k f_k$ ▷ Softmin energy (1)
  $A_k(\mathbf{x}) \leftarrow n w_k \big[1 - \beta(f_k - J)\big]$ for every k ▷ pre-factor (4)
  $x_k \leftarrow \Pi_B\big(x_k - \eta\, A_k(\mathbf{x}) \nabla f(x_k) + \sqrt{2\sigma\eta}\,\xi_k\big),\ \xi_k \sim \mathcal{N}(0, I_d)$ i.i.d.
end for
return $\arg\min_{x_k} f(x_k)$

E.1 Hardware and Computational Resources

All numerical experiments were executed on an academic scientific-computing cluster. To fully leverage the vectorized nature of the simulations and evaluate independent trials in parallel, jobs were strictly allocated to a single compute node at a time. The cluster nodes utilized for these experiments include:

- **High-Capacity Nodes:** AMD EPYC 7763 processors featuring 128 physical cores (64 cores per socket, 2 sockets, 256 logical threads) and 2 TB of RAM. Each also hosts four NVIDIA A40 GPUs (48 GB), one of which was allocated per job for the transformer benchmark (Section 4.4), the only experiment run on a GPU.
- **Standard Nodes:** Intel Xeon CPU E5-2650 v4 @ 2.20GHz featuring 24 physical cores (12 cores per socket, 2 sockets) and 256 GB of RAM. These nodes were specifically utilized to parallelize the independent runs of the experiments using Python's `multiprocessing` framework.

E.2 General Experimental Settings

Table 1 summarizes the shared parameters utilized across all algorithms for each specific landscape. Within each benchmark, the swarm size and maximum step limit are the same for every algorithm, to provide a fair baseline for evaluating the topological escape mechanisms. The domain bounds listed below are enforced by clipping each coordinate back into the box after every discrete update, identically for every algorithm. Domain bounds for the 1D and 2D wells are derived directly from each landscape's own critical points rather than set by hand: the domain extends $K = 5$ natural harmonic fluctuation-lengthscales ($1/\sqrt{f''}$ at the relevant minimum, or the softest Hessian eigendirection in 2D) beyond the outermost minimum – the same role a 5σ truncation plays for a locally-Gaussian process, giving negligible boundary-contamination probability without an oversized domain. A fixed absolute margin was checked and found to represent anywhere from ~ 1 to ~ 5.5 lengthscales across the swept difficulty range, an uncalibrated $> 5\times$ inconsistency this formula removes. The quadruple well is swept over $A \in \{2, \dots, 30\}$ (not 1): at $A = 1$ the softest Hessian eigenvalue at the well vanishes exactly, so the landscape does not yet have a genuine local minimum in that direction.

Table 1: General settings and landscape configurations.

Parameter	Value / Description
Swarm Size (n)	100 (1D/2D wells, Ackley, Lennard-Jones); swept over $\{5, \ldots, 400\}$ in the ensemble-size study (Section 4.3); 10 for the transformer, where each step costs n full forward/backward passes
Independent Runs	100 (1D/2D wells, Ackley, ensemble-size study), 30 (Lennard-Jones), 5 (transformer); additional confirmation runs in Appendices E.4.5 and E.4.6
Max Step Limit	10^7 (1D/2D wells, Ackley, ensemble-size study), 10^5 (Lennard-Jones), 2×10^3 (transformer)
Gradient Clipping	Per-atom norm 1 on the Lennard-Jones term; per-particle norm 10 (transformer); none otherwise; same for every algorithm
1D Double Well	$f(x) = x^4 - \delta x^2 + 0.05x$, swept over $\delta \in \{1, \ldots, 30\}$ (asymmetric: the linear tilt breaks the left/right symmetry, giving a genuinely suboptimal starting basin – see Appendix E.3)
Initialization	Uniformly sampled around the shallower (suboptimal) minimum z_-
Domain Bounds	$[z_- - 5/\sqrt{f''(z_-)},\ z_+ + 5/\sqrt{f''(z_+)}]$, $f''(x) = 12x^2 - 2\delta$ ($z_\pm$: the two minima)
2D Quadruple Well	$f(x, y) = x^4 + y^4 - A(x^2 + y^2) + xy$, swept over $A \in \{2, \ldots, 30\}$; the depth gap between the two wells equals A exactly
Initialization	Uniformly sampled around the shallower diagonal minimum (ϱ_-, ϱ_-) (genuinely suboptimal starting basin); target is the deeper anti-diagonal minimum $(\varrho_+, -\varrho_+)$ – see Appendix E.3
Domain Bounds	$[-(\varrho_+ + m),\ \varrho_+ + m]^2$, $\varrho_- = \frac{1}{2}\sqrt{2A-1}$, $\varrho_+ = \frac{1}{2}\sqrt{2A+1}$ ($\geq \varrho_-$), $m = 5/\sqrt{\lambda_{\min}}$, $\lambda_{\min} = \min\big(f_{xx}(\varrho_-, \varrho_-) - \lvert f_{xy}(\varrho_-, \varrho_-)\rvert,\ f_{xx}(\varrho_+, -\varrho_+) - \lvert f_{xy}(\varrho_+, -\varrho_+)\rvert\big)$
High-Dimensional Ackley	$f(x) = -a\exp(-b\lVert x\rVert/\sqrt{d}) - \exp(d^{-1}\sum_i \cos(cx_i)) + a + e$, swept over $d \in \{10, 20, \ldots, 90\}$. Both terms are per-coordinate averages, so $f = 0$ at the origin for every d; target $f \leq 1$
Function Parameters	$a = 20.0$, $b = 0.2$, $c = 2\pi$
Initialization	Uniformly sampled in $[-10, 10]^d$
Domain Bounds	$[-10, 10]^d$
Lennard-Jones Cluster	$E(\mathbf{x}) = \sum_{i<j} 4(r_{ij}^{-12} - r_{ij}^{-6}) + E_{\text{conf}}(\mathbf{x})$ in reduced units ($\epsilon = r_0 = 1$), $N \in \{13, 38, 55, 75\}$ atoms ($d = 3N$)
Confinement term	$E_{\text{conf}}(\mathbf{x}) = \kappa \sum_i \max(0, \lVert x_i\rVert - R_c)^2$ with $\kappa = 10$, origin-centered: it prevents atoms evaporating into zero-gradient regions under the noise, pins the cluster by removing the translational zero mode, and vanishes identically on the reference global minimum
Initialization	Uniform-in-ball, rejecting pairs closer than 0.9
Domain Bounds	$[-(R_c + 2),\ R_c + 2]^d$, $R_c = 1.5N^{1/3} + 1.5$
Transformer (modular addition)	Training loss of a decoder-only transformer on $(a+b) \bmod p$, $p = 53$, presented as $[a, b, =]$ and predicted from the final position; cross-entropy over a fixed 80% subset of all p^2 pairs, evaluated FULL-BATCH so gradients are exact. Depth swept over $\{2, 5, 10, 20, 50\}$
Architecture	Pre-LN, $d_{\text{model}} = 64$, 2 heads, MLP ratio 4, learned positional embeddings; d_{model} and head count held fixed across the sweep so depth is the only architectural variable
Optimization Dimension	d = flattened parameter count, 1.07×10^5 (depth 2) to 2.51×10^6 (depth 50)
Initialization	n independent PyTorch default initializations, shared across all methods
Target	Training loss ≤ 0.10 (chance level $\log p \approx 3.97$)
Domain Bounds	$[-10, 10]^d$ (never binding in practice)

E.3 HYPERPARAMETER SELECTION METHODOLOGY

Each of the 100 independent runs (Table 1) is indexed by an integer seed $\in \{0, \ldots, 99\}$, and every worker seeds NumPy's global RNG once via `np.random.seed(seed*1000)` before drawing that run's initial swarm and simulating every algorithm on it. This seed indexing is reused identically at every sweep point (every δ, A, or d) rather than redrawn independently per point: `seed=7` at $\delta = 5$ and `seed=7` at $\delta = 25$ consume the same underlying draw from the same seeded RNG state, remapped only by that point's own initialization bounds. Consequently the 100-run samples at different sweep points are correlated, not independent, and the quartile error bars in Figure 5 should be interpreted as within-seed variability across the ensemble rather than as independent per-point confidence intervals.

Two conventions govern every entry-time figure in this paper and are worth stating explicitly, since both affect how the curves should be read. First, a run that never reaches the target returns the timeout sentinel (max steps $\times\, \eta$) rather than a genuine entry time, so each plotted point is the mean over only those seeds whose logged status is `target_reached`. A method's plotted curve is therefore conditional on success, and must be read together with its reached-fraction: where a baseline is faster than Softmin at a difficulty it only clears in a fraction of seeds, that mean is taken over the easier subset of the shared seed set, not over a comparable sample. Second, a point whose reached-fraction falls below 50% is dropped from the plot entirely, to avoid a handful of lucky seeds rendering as a confident-looking point; points between 50% and 90% are plotted but annotated with the exact fraction. A curve that simply stops therefore marks the difficulty at which that method became unreliable, not the end of the sweep.

Per-algorithm settings are collected in Tables 3–9. Learning rates are unified to $\eta = 0.001$ across every benchmark and algorithm (the Theorem 3.4 verification drivers excepted, which use their own, separately-motivated small-noise/small-step-size regime). The Softmin inverse temperature β is selected per benchmark family via a two-stage, code-audited procedure (`src/hyperparameter_diagnostics.py`), evaluated at a representative *middle*-of-sweep setting rather than the hardest one, so the choice is not fit to the one condition the sweep is meant to stress-test: (1) an initial screen measures the distribution of $\beta(f_i - J_\beta)$ across the ensemble, flagging candidate values where trailing particles are not uniformly driven into the exponentially-suppressed regime; (2) because that screen alone favors values that are too small – it only measures whether every particle receives rescue-drift, not what the reported first-entry-time metric rewards, namely the single fastest particle reaching the target, which benefits from a *concentrated* ensemble via the leading particle's pre-factor approaching n – the candidates are then validated directly by measuring actual entry time. Across every benchmark tested (1D double well, 2D quadruple well, high-dimensional Ackley, and a spot check on Lennard-Jones), this second step confirmed $\beta = 10$ as at or near the empirical optimum, matching the value used throughout; going lower is consistently slower or fails to reach the target at all, and going higher does not help (and is markedly worse on the quadruple well). The same direct-validation procedure was applied to Simulated Annealing's geometric cooling rate, and to comparing β held fixed against both a decaying and an increasing schedule – see Remark D.4 for the latter result. The Lennard-Jones benchmark additionally clips the gradient's Lennard-Jones term (never the confinement term, whose restoring force must scale freely to remain able to pull back an atom that has drifted outside the container) to a per-atom norm of 1.0 before the swarm's discrete update, applied identically to every algorithm; this clip is not applied to the L-BFGS-B local polish ("quench") of each method's final configuration, which uses the true, unclipped gradient throughout, since a quasi-Newton method's curvature estimate is only valid for the true gradient. The transformer benchmark likewise applies a per-particle gradient-norm clip, at 10, again identically for every algorithm; that value is selected in Table 7 and is loose enough to reproduce the unclipped loss exactly, so it bounds a pathological step without shaping the dynamics.

Baseline calibration. Simulated Annealing, CBO, and PSO were each given the same diagnostic-guided procedure as Softmin's β and, where a comparable parameterization exists, the same candidate budget (Table 2), rather than literature-standard values adopted without checking. The three cases are reported below; two of them end in negative results, which we report as such.

Simulated Annealing. Seven candidate cooling rates were derived from target cooling *timescales* $t^\star \in \{10^3, 3{\times}10^3, 10^4, 3{\times}10^4, 10^5, 3{\times}10^5, 10^6\}$ – the number of steps to reach 5% of the initial

temperature, rate $= 0.05^{1/t^\star}$ – rather than hand-picked decimals, and validated by direct entry-time measurement in the same way as β; the rate 0.99995 was confirmed on both wells. The initial temperature T_0 is inherently problem-scale-dependent, so no universal literature default exists, and it was calibrated separately. Escape time is a degenerate criterion for T_0 in this continuous, always-accept diffusion, since larger T_0 is monotonically faster with no interior optimum and eventually indistinguishable from bounded random search; we therefore used a Kramers/Arrhenius criterion instead. Five candidates $T_0 = c\,h$ with $c \in \{0.25, 0.5, 1, 2, 4\}$, where h is the double well's mid-sweep barrier height, were scored by the closed-form escape factor $\exp(-h/T_0)$, and we selected the value placing that factor in the *low* band $[0.01, 0.05]$: a genuine, non-trivial barrier, rather than the near-unity band a literal reading of the discrete-Metropolis initial-acceptance-ratio heuristic (Kirkpatrick et al., 1983) would suggest, which would leave the barrier thermodynamically negligible from $t = 0$ and trivialize precisely the obstacle this baseline is meant to face. The selected $T_0 = h/4$ reached the target within a few hundred steps at the calibration midpoint and transferred to the quadruple well. It nonetheless fails at the hardest points of the double-well sweep ($h \gtrsim 156$). That failure is a property of the cooling schedule rather than of the step budget: with early stopping disabled and a horizon of $1{,}500{,}000$ steps, fifteen times the original probe, the best value found is unchanged from step $10{,}000$ onward at both $h = 156.1$ and $h = 224.8$, consistent with the geometric cooling driving the temperature to ≈ 0 well before step $200{,}000$ and freezing the dynamics. Applying the same criterion to Ackley's own barrier scale was inconclusive even at a five-times-longer diagnostic budget, so Ackley and Lennard-Jones retain $T_0 = 10$.

CBO. The same 11-value β grid used for Softmin was swept, together with two annealed-schedule variants (growth-only, matching CBO's own literature convention rather than Softmin's decaying schedule). Every candidate times out through consensus collapse on the double well, the quadruple well, *and* Ackley, at the identical step regardless of which β was used: direct testing across $\beta \in \{0.01, 0.02, 0.1, 1.0, 10.0\}$ produced position collapse at the same step on every benchmark tested. The collapse is therefore a β-independent structural property of vanilla CBO's self-shrinking anisotropic diffusion, not a symptom of a poorly chosen β. Since the choice is empirically immaterial, we report $\beta = 0.01$, which lies in CBO's literature-typical range.

PSO. PSO has no single temperature-like scalar to sweep, so six named, literature-grounded (w, c_1, c_2) triples were tested instead: the canonical Clerc–Kennedy constriction coefficients (Clerc & Kennedy, 2002); the original Kennedy–Eberhart rule (Kennedy & Eberhart, 1995); both endpoints of the Shi–Eberhart linearly-decreasing-inertia schedule (Shi & Eberhart, 1998); a social-weighted variant biased toward the swarm's global best; and a deliberately non-contractive stress-test triple. All six were pre-filtered by a finite/at-bound stability probe. On the double well and quadruple well, PSO shows the same negative result as CBO. On Ackley, by contrast, the literature-standard constriction coefficients themselves *fail*, timing out through consensus collapse at every dimension probed ($d = 50, 70, 90$), while two of the six alternatives succeed at all three; the faster of those, $(w, c_1, c_2) = (0.4, 2.0, 2.0)$, was adopted for Ackley and is used both in Section E.4.4 and for the Ackley-d=50 ablation in Appendix E.5.4. A spot check found it also improves the Lennard-Jones quenched result, so it is used there too (Table 1).

Table 2: Calibration budget: candidate values/triples tested, per benchmark family, for each algorithm's diagnostic-guided tuning pass. PSO's lower total reflects having no schedule mechanism to test (see text), not less scrutiny. SA's "schedule variants" column is its T_0 candidates, evaluated once at the double-well midpoint (not per benchmark, see text) rather than a decaying/growing schedule.

Algorithm	**Fixed candidates**	**Schedule/T_0 variants**	**Total per benchmark**
Softmin	11	2	13
SA	7	5	12
CBO	11	2	13
PSO	6	0	6

Table 3: Hyperparameters for Softmin energy dynamics.

Parameter	Value
Learning Rate (η)	0.001 (analytic benchmarks), 0.1 (transformer), 0.01 (Thm. 3.4 verification)
Adaptive Learning Rate (Adagrad)	Disabled (False)
Noise Intensity (σ)	1.0 (analytic benchmarks), 10^{-5} (transformer), 0.05 (Thm. 3.4 verification)
Noise Distribution	Normal
Fixed Inverse Temperature (β)	10.0 (all benchmarks, including the transformer), 2.0 (Thm. 3.4 verification and the $\beta = 2$ ablation)

Table 4: Hyperparameters for Simulated Annealing (SA). T_0 differs by benchmark family: the wells' value is Kramers-factor calibrated and validated (see Appendix E.3); Ackley and Lennard-Jones retain $T_0 = 10$, since the same calibration attempt was inconclusive there, and the transformer uses the same value; T_0 has no problem-independent literature standard.

Parameter	Value
Learning Rate (η)	0.001 (analytic benchmarks), 0.1 (transformer)
Base Noise Variance Factor	$2 \times \sigma \times \eta$
Initial Temperature (T_0)	14.0283 (1D/2D wells), 10.0 (Ackley, Lennard-Jones, transformer)
Cooling Schedule	Geometric
Geometric Cooling Rate	0.99995 (1D/2D wells, re-validated via a 7-candidate timescale sweep), 0.9999 (Ackley, Lennard-Jones, transformer)

Table 5: Hyperparameters for overdamped Langevin dynamics. On every benchmark it uses the same step size and noise level as Softmin, with no schedule and no interaction; with a single particle, Softmin's update coincides with it exactly.

Parameter	Value
Learning Rate (η)	0.001 (analytic benchmarks), 0.1 (transformer)
Noise Intensity (σ)	1.0 (analytic benchmarks), 10^{-5} (transformer)
Update	$x \leftarrow x - \eta \nabla f(x) + \sqrt{2\sigma\eta}\,\xi,\ \xi \sim \mathcal{N}(0, I_d)$

Table 6: Hyperparameters for the deterministic first-order baselines, used only on the transformer benchmark (Section 4.4). Both are run at their literature-standard defaults rather than tuned for this benchmark, which is the counterpart of the carried-over, not-re-tuned convention applied to every other method there. Both are also run as $n = 10$ *independent* chains, so they receive the same restart budget and the same per-step cost as the Softmin ensemble they are compared against.

Parameter	Adam	SGD
Learning Rate (η)	0.001	0.01
Momentum	$\beta_1 = 0.9$, $\beta_2 = 0.999$	0.9 (heavy-ball)
Numerical Stability (ε)	10^{-8}	—
Injected Noise	None (deterministic)	None (deterministic)

Table 7: Diagnostic-guided calibration for the transformer benchmark (`src/calibrate_transformer.py`), run at the mid-sweep depth 10 with a 4-particle ensemble, the same convention every other benchmark family uses. Losses are the best reached within the stated budget; chance level is $\log p = 3.970$, so a value at or above it means the method has not learned anything at all. Two constraints bind rather than one. The learning rate is the dominant one: $\eta = 10^{-3}$ does not move the iterate at any clip. The gradient clip binds as well, which a first pass on a 1-layer model had missed – at this depth the per-particle gradient norm averages 0.9311 and reaches 1.0341, so clipping at 1 costs more than an order of magnitude in final loss, while clipping at 10 reproduces the unclipped result exactly. The selected configuration is $\eta = 0.1$, clip 10, $\beta = 10$, $\sigma = 10^{-5}$; see Appendix E.4.6 for why β and σ are not set to this sweep's nominal optima, $\beta = 100$ and $\sigma = 0$, at which the noise-level stage was run. The selected configuration is used unchanged at every depth of the sweep.

Stage	**Candidate**	**Best loss**
Learning rate and clip (400 steps)		
	$\eta = 10^{-3}$ (carried over), any clip	3.9996
	$\eta = 10^{-2}$, any clip	3.9645
	$\eta = 0.1$, clip 1.0	1.1648
	$\eta = 0.1$, clip 10 or none (adopted)	**0.0754**
Inverse temperature (800 steps)		
	$\beta = 0.1$	0.0157
	$\beta = 1$	0.0042
	$\beta = 10$ (shared value, adopted)	**0.0028**
	$\beta = 100$	0.0024
Noise level (800 steps, $\beta = 100$)		
	$\sigma = 0$ (degenerate, see text)	0.0024
	$\sigma = 10^{-5}$ (adopted)	**0.0065**
	$\sigma = 10^{-4}$	0.0485
	$\sigma = 10^{-3}$	3.9537
	$\sigma = 10^{-2}$	4.0800
Reference	Adam, $\eta = 10^{-3}$, 800 steps	0.0010

Table 8: Hyperparameters for Gradient-Enhanced Particle Swarm Optimization (PSO). The literature-standard constriction coefficients were found, by direct escape-time testing, to fail on Ackley specifically (all dimensions tested); a diagnostic-guided alternative is used there instead (see Appendix E.3).

Parameter	**Value**
Gradient Step Learning Rate (η)	0.001
Inertia Weight (w)	0.729 (1D/2D wells), 0.4 (Ackley, Lennard-Jones)
Cognitive Coefficient (c_1)	1.49445 (1D/2D wells), 2.0 (Ackley, Lennard-Jones)
Social Coefficient (c_2)	1.49445 (1D/2D wells), 2.0 (Ackley, Lennard-Jones)

Table 9: Hyperparameters for Consensus-Based Optimization (CBO). $\beta = 0.01$, CBO's literature-typical range (unlike PSO's constriction coefficients, CBO has no single canonical numeric β, but 0.01–0.02 is the commonly cited scale) . The value is immaterial here: an 11-value sweep including $\beta = 0.01$, plus two annealed-schedule variants, found consensus collapse at the identical step regardless of β on all three main benchmarks – a β-independent structural negative finding, see Appendix E.3.

Parameter	**Value**
Learning Rate (Δt)	0.001
Consensus Weight (λ)	1.0
Exploration Noise (σ_{CBO})	$\sqrt{2}$
Inverse Temperature (β)	0.01

E.4 Experimental Details

This appendix gives, for each benchmark of Section 4, the precise setup, reporting conventions and detailed numbers behind the summary in the main text.

E.4.1 Protocol and Reporting Conventions

We evaluate the capacity of Softmin dynamics to escape local minima and outrun sub-optimal basins by comparing fixed-β ($\beta = 10$) Softmin against seven baselines: Simulated Annealing (SA) (Gelfand & Mitter, 1990); overdamped Langevin dynamics (Welling & Teh, 2011), a control that shares Softmin's step size and noise level but has no interaction; Gradient-Aware Consensus-Based Optimization (CBO) (Carrillo et al., 2021), with and without upward β-annealing; Gradient-Enhanced Particle Swarm Optimization (PSO) (Noel, 2012); and the canonical, gradient-free formulations of both CBO (Carrillo et al., 2018; Pinnau et al., 2017) and PSO (Kennedy & Eberhart, 1995). The ensemble-size sweep compares Softmin with Simulated Annealing alone, and the transformer benchmark with Simulated Annealing, Langevin dynamics, Adam and SGD. The continuous dynamics are discretized using the Euler–Maruyama scheme with step size η, and every entry time we report is in continuous-time units, that is (number of steps) $\times\, \eta$, except on the transformer benchmark, where the methods use different step sizes and entry time is reported in steps. Entry time is measured against a value-space target: for the wells, 0.15 of that benchmark's own depth gap above the target well's value, so the tolerance scales with the sweep instead of being fixed in absolute terms; for Lennard-Jones, a tolerance of 10^{-2} above the literature global minimum; for Ackley, the absolute value $f \leq 1$; and for the transformer, a training loss of 0.10. Each reported entry-time mean is taken over those seeds that actually reached the target, and any sweep point at which fewer than half the seeds reach it is omitted from the entry-time panels rather than plotted from a small surviving sample. Because that convention makes an unreliable method disappear rather than degrade, every entry-time result is accompanied by the reached-fraction at every swept point, with no cutoff and no masking, as a reliability panel or, for Ackley, in the caption of Figure 6 and the text of Appendix E.4.4, so a method's speed is never shown without its reliability. Landscape formulations, hyperparameters, initialization bounds, hardware specifications, and real per-particle gradient/objective evaluation counts and wall-clock time are detailed in Appendix E; the source code is available at (Anonymous, 2026).

One caveat frames everything that follows. Theorem 3.4 is a $\sigma \downarrow 0$ statement, whereas the analytic benchmarks below run at $\sigma = 1$ (the transformer benchmark uses $\sigma = 10^{-5}$, see Appendix E.4.6): at $\beta = 10$ and $n = 100$ the bound of (13) is 0.0372 and the channel maximum it bounds is $u_c \approx 0.0370$, so $u_c/\sigma \approx 0.04$ and the reduced landscape along the channel is essentially flat relative to the noise. The large-deviation exponent therefore carries little information at the noise level used here, and these experiments should be read as testing whether the Softmin coupling helps in a practically relevant regime, not as a numerical confirmation of the exponent. The small-noise regime the theorem does describe is probed separately in Appendix E.5.1, where σ is swept toward 0.

E.4.2 Double-Well and Quadruple-Well Benchmarks

Both well families are swept through the quadratic coefficient of their defining quartic (Table 1): δ for the 1D double well $f(x) = x^4 - \delta x^2 + 0.05x$, and A for the 2D quadruple well $f(x, y) = x^4 + y^4 - A(x^2 + y^2) + xy$. We plot each sweep against the quantity that actually sets its difficulty rather than against the raw coefficient. For the double well this is the barrier height $h = f(s) - f(z_-)$, which grows like $\delta^2/4$, from $h = 0.2$ at $\delta = 1$ to $h = 224.8$ at $\delta = 30$; for the quadruple well it is the depth gap between the shallow diagonal minimum and the deep anti-diagonal one, which equals A exactly (Appendix E.3).

We first check that simulations on a 1D double well, at the small noise the theorem requires, are consistent with the large deviation scaling of Theorem 3.4; results are in Appendix E.5 (Figures 8 and 9), alongside the ascent-descent mechanism ablation (Appendix E.5.4), the per-particle pre-factor illustration (Figure 10), and the full cost comparison (Table 12).

Figure 1 (top left) reports reliability across barrier heights h. Softmin reaches the target in all 100 seeds at every h tested, across a sweep spanning a factor of about 1100 in barrier height ($h = 0.2$ to $h = 224.8$). Every baseline eventually fails. SA holds at 100% while the barrier is small, then degrades as it grows: 86% at $h = 99.8$, 1% at $h = 156.1$, and 0% at $h = 224.8$. Langevin fails far earlier, already at 0% by $h = 24.9$ having succeeded at $h = 6.2$. CBO and PSO, gradient-aware and

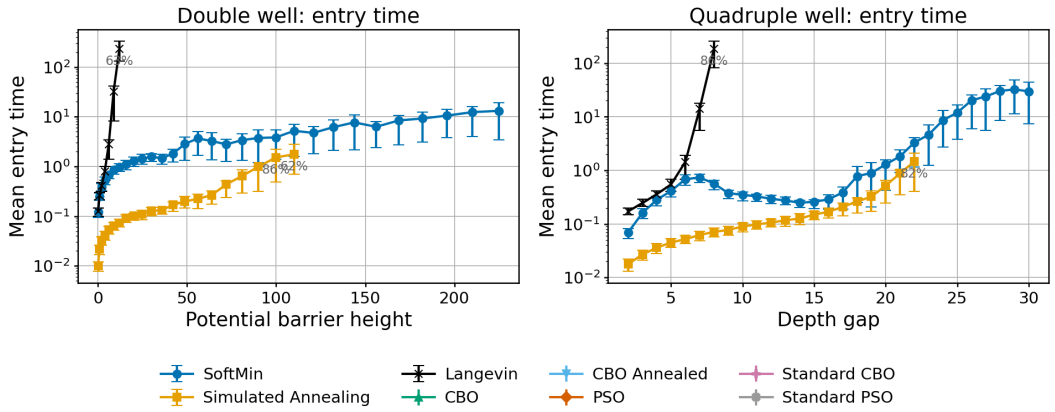

Figure 5: Mean first-entry time into the target on the double well (left, against barrier height) and the quadruple well (right, against the depth gap), in continuous-time units (steps $\times$ η), *conditional on success*: each point is the mean over only those seeds that reached the target, with error bars spanning their interquartile range; a point below a 50% reached-fraction is omitted, and one between 50% and 90% is annotated with the exact fraction. It must therefore be read against the reliability panels of Figure 1: where a baseline lies below Softmin it is faster only on the subset of seeds it solved, and a curve that stops marks unreliability rather than a missing run. The same 100 seeds are reused at every sweep point (Appendix E.3), so the bars reflect within-seed variability.

canonical alike, reach the target in at most 3% of seeds at any barrier height tested, far below the 50% floor a point must clear to be plotted in Figure 5 (Appendix E.3), and so never appear there.

This separation is the qualitative signature Theorem 3.4 describes. A method whose quasipotential barrier is capped at order $1/\beta$ *independently of the true barrier height* should keep escaping as h grows, while a method governed by the true exponent h – SA and Langevin both – should fail once h is large enough, which is the pattern the top row shows. Softmin's entry time (Figure 5, left) quantifies the same behavior from the other side: its mean grows from 0.12 at $h = 0.2$ to 13.2 at $h = 224.8$, a factor of about 110 against that factor of about 1100 in barrier height, that is, far more slowly than the barrier itself, whereas the escape time of a method governed by the true barrier grows exponentially in it. We put this no more strongly than consistency, for the reason given at the start of Section 4: at $\sigma = 1$ the proven exponent is far too small to be the operative effect, and the ablation of Appendix E.5.4 finds the Gibbs weighting the more plausible source of the advantage at $\beta = 10$. What the theory predicts and the data show is the same *shape*, not an established causal link.

SA is genuinely faster than Softmin wherever it still succeeds, and Figure 5 should not be read as saying otherwise: at the mid-sweep point $h = 56.1$ SA reaches the target in a mean 0.22 against Softmin's 3.67. That comparison is conditional on success, however, and averages over only the seeds each method solved. Table 10 prices the same runs without conditioning.

Figure 1 (right column) extends this to the 2D quadruple well, whose `particle_init` and target sit at different depths, with a depth gap equal to A exactly (Appendix E.3, Table 1). The same pattern holds and sharpens. Softmin again reaches the target at every depth gap tested (2 through 30, 100% of seeds throughout), while SA's reached-fraction falls to 17% by $A = 25$ and 0% by $A = 30$, and Langevin fails already by $A = 10$ (0%, having succeeded at $A = 5$). CBO and PSO reach the target in at most 3% of seeds at every depth gap tested here; an independent, algorithm-agnostic stagnation check, logged per run (Appendix E.3), confirms this is a genuine stall rather than an unreached timeout. As on the double well, SA is faster where it still succeeds (0.15 mean entry time at $A = 15$ against Softmin's 0.26), on the seeds it solved.

Cost to solution. Entry time counts steps, not the price of a solution. A practitioner cannot know in advance which instances a method will solve, so the cost that matters is the expected cost of obtaining a success, including the runs that produce none. We therefore report the expected running time (ERT) standard in black-box optimization benchmarking (Hansen et al., 2021): the total cost over all 100 seeds divided by the number of successes, which under a restart model is the expected cost of the first

success, and is undefined (infinite) where a method never succeeds. Table 10 gives it in wall-clock seconds along the double-well sweep. The bottom row of Figure 1 gives it at every sweep point of both benchmarks and for every method, measured instead in objective plus gradient evaluations: this unit does not depend on the hardware, and unlike gradient evaluations alone it also charges the canonical, gradient-free CBO and PSO, which use none. A gap marks a point at which no seed succeeded, so that the expected cost is infinite, and an isolated marker a point at which only a few did; the CBO and PSO markers at low barriers, and Simulated Annealing's single marker near $h = 210$, are of this kind. In this unit Simulated Annealing's expected cost first exceeds Softmin's at $h = 90.1$ on the double well, consistent with the crossover between $h = 56.1$ and $h = 99.8$ in the table, and at a depth gap of 21 on the quadruple well.

Table 10: Cost to solution on the 1D double well: expected running time (ERT) in wall-clock seconds, that is, total wall-clock over all 100 seeds divided by the number that reached the target, with the reached-fraction alongside. Measured on the sweep actually run, not a synthetic probe; both methods use $n = 100$ particles and $\eta = 0.001$, so per-step cost is identical and the difference is entirely in steps taken. SA is an order of magnitude cheaper than Softmin while the barrier is small, and the two cross between $h = 56.1$ and $h = 99.8$; past the crossover SA's cost diverges, reaching roughly $7{,}600\times$ Softmin's at $h = 156.1$ before becoming unbounded at $h = 224.8$, where no seed succeeds.

	Softmin		**Simulated Annealing**	
Barrier h	**ERT (s)**	**Reached**	**ERT (s)**	**Reached**
6.2	2.6	100%	0.2	100%
24.9	5.5	100%	0.4	100%
56.1	13.0	100%	0.8	100%
99.8	14.4	100%	302.3	86%
156.1	23.5	100%	179,233.7	1%
224.8	49.2	100%	∞	0%

Read this way the trade-off is a crossover, not a ranking. Where the barrier is small SA is the cheaper method outright, by roughly an order of magnitude, and we do not claim otherwise; averaged over only the runs it solves, across both well sweeps, SA costs 0.79 s against Softmin's 11.68 s. What changes past the crossover is not that SA becomes slower but that its cost stops being bounded at all: its geometric cooling drives the temperature to ≈ 0 after a fixed number of steps regardless of the barrier actually faced, so it freezes rather than slows, and additional budget does not recover it (Appendix E.3 verifies this directly at $1{,}500{,}000$ steps, fifteen times the original probe, with the best value unchanged from step $10{,}000$ onward). Softmin's ERT meanwhile grows by a factor of 19 between $h = 6.2$ and $h = 224.8$, a factor of 36 in barrier height, and by a factor of about 110 in evaluations over the full sweep (Figure 1), which spans a factor of about 1100. Aggregated over the full double-well and quadruple-well sweep the same asymmetry appears as a single number: Softmin's mean wall-clock is 11.68 s against SA's 351.57 s and Langevin's 1033.74 s, a $30\times$ and $88\times$ difference, rising to $42\times$ and $384\times$ in ERT (Table 12). Softmin's three wall-clock figures coincide in the same comparison because it never times out.

E.4.3 Lennard-Jones Cluster Benchmark

Lennard-Jones clusters are our first benchmark for scaling with dimension: N atoms interact through the pairwise potential $V(r) = 4\epsilon\big[(r_0/r)^{12} - (r_0/r)^6\big]$ (Jones, 1924), giving a search space of dimension $d = 3N$ whose number of local minima grows exponentially in N. Here r is the interatomic distance and r_0 the length scale, conventionally written σ; we rename it because σ denotes the noise intensity throughout this paper. We work in reduced units, $\epsilon = r_0 = 1$. We use the $N \in \{13, 38, 55, 75\}$ clusters (d up to 225) from the Cambridge Cluster Database (Wales & Doye, 1997), whose lowest known energies serve as the literature global minimum against which every method is scored. Following the standard basin-hopping convention of separating global search from local relaxation (Wales & Doye, 1997), each method's final configuration is locally polished, or *quenched*, via L-BFGS-B to its nearest local minimum before scoring; we report this quenched configuration's energy gap above the literature global minimum, rather than the raw configuration's, so the comparison reflects basin-finding rather than residual local-convergence noise. The strict hit-time target used for the double-well and quadruple-well benchmarks above is rarely reached by any

method (Softmin included) beyond the smallest cluster size, once $N \geq 38$, so the quenched-energy gap is the informative comparison at this scale.

Figure 2 reports the quenched-energy gap. Softmin's gap stays small and essentially flat across the full range tested (from 0 at $N = 13$ to 3.7 at $N = 75$), while every baseline's gap grows roughly linearly with N, from 1.8–5.6 at $N = 13$ to 23–30 at $N = 75$. Softmin's advantage over every baseline *widens*, not narrows, as the cluster grows: the opposite of the ordinary curse-of-dimensionality intuition.

E.4.4 High-Dimensional Ackley Benchmark

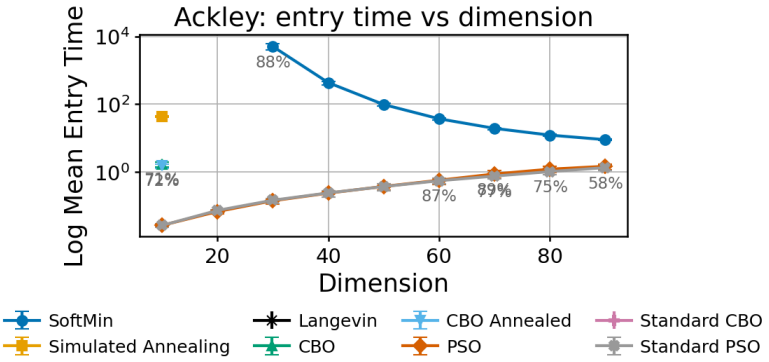

Figure 6: Generalized Ackley function in 10 to 90 dimensions: time to reach the target, averaged over successful runs, with the conventions of Appendix E.4.1. Softmin is missing in 10 and 20 dimensions, where it never reaches the target; Langevin dynamics and canonical CBO are missing everywhere for the same reason.

Our final benchmark is the generalized Ackley function in $d \in \{10, 20, \ldots, 90\}$ dimensions, which superimposes a dense lattice of shallow local minima, generated by a cosine term, on a single smooth radial funnel. Both of its terms are per-coordinate averages, so its global minimum is 0 at the origin for every d and the hit-time target can be the fixed absolute value $f \leq 1$ with no dimension-dependent rescaling. This is the benchmark on which Softmin does worst, and Figure 6 reports it in full.

Particle Swarm Optimization is the fastest method here at every dimension, by a wide margin. PSO reaches the target in a mean 0.03 at $d = 10$, rising gently to 1.47 at $d = 90$, while Softmin does not reach it at all within the step budget at $d = 10$ or $d = 20$ (0% of seeds) and is far slower where it does succeed: 5055 at $d = 30$, 97.9 at $d = 50$, 8.98 at $d = 90$. Simulated Annealing succeeds only at $d = 10$; CBO degrades from 71% reached at $d = 10$ to 5% at $d = 30$; Langevin and canonical CBO never reach the target at any dimension tested.

What separates the methods here, however, is not access to the gradient. The canonical gradient-free PSO is statistically indistinguishable from the gradient-enhanced form at every dimension, the two means agreeing to within 15% throughout (0.026 vs. 0.027 at $d = 10$, 1.35 vs. 1.47 at $d = 90$): at $\eta = 0.001$ the gradient step contributes essentially nothing beside PSO's own velocity update. The split is instead between the PSO family and everything else. Outside that family, Softmin is the only method that becomes reliable as the dimension grows, reaching the target in every run from $d = 40$ on after failing entirely at $d = 10$ and $d = 20$: Simulated Annealing drops to 0% at $d = 20$, both gradient-aware CBO variants fall below the 50% reliability floor there and reach 0% by $d = 50$, and Langevin and canonical CBO never succeed at any dimension tested. Ackley therefore discriminates search dynamics rather than gradient information, and is a weak test of the question this paper is about.

Read against dimension, moreover, the trend runs in Softmin's favor and tracks the theory. The barrier out of the global basin along a single coordinate direction, $\max_{t \in [0,1]} f(te_1) - f(0)$, falls monotonically as d grows – from 1.23 at $d = 10$ to 0.42 at $d = 90$ – because both terms of Ackley are per-coordinate averages. Softmin's entry time falls with it, and the gap to PSO closes by nearly four orders of magnitude across the sweep, from a factor of about 3.6×10^4 at $d = 30$ to a factor

of 6 at $d = 90$. These barriers remain well above the switching level $r_c = 0.100$ of Lemma 3.3 at $\beta = 10$, by a factor of 4 to 12, so the sign-reversal mechanism is active throughout, and Softmin's improvement is consistent with the barrier flattening of Theorem 3.4 rather than with the mechanism lapsing. We put this no more strongly than consistency: at the $\sigma = 1$ used here the proven exponent is too small to be the operative effect, as noted at the start of Section 4, and agreement between two monotone trends is not on its own evidence of a causal link. Reliability moves the same way: Softmin's reached-fraction rises to 100% from $d = 40$ onward, so its $d = 10$ and $d = 20$ failures are plausibly the continuation of that trend against a fixed budget rather than a separate effect, though we did not confirm this by raising the budget, and at $d = 90$ Softmin is the most reliable method in the comparison at 100%, against 96% for gradient-enhanced PSO, 58% for the canonical form, and 0% for everything else.

None of this overturns the headline: PSO is faster than Softmin at every dimension tested, and we do not claim otherwise. What Ackley establishes is the boundary of the claim: on a landscape whose difficulty is carried by a dense lattice of shallow minima on a smooth global funnel, and on which gradient information demonstrably buys nothing, barrier flattening delivers reliability but not speed. We do not identify what does drive PSO's advantage here. That its velocity-based attraction toward the swarm's global best is well suited to a single smooth funnel is a natural conjecture, but we have not tested it and nothing in our experiments distinguishes it from competing explanations; we record it as an open question rather than a finding. Softmin's β was fixed at the mid-sweep dimension $d = 50$ by the same pre-committed protocol applied to every benchmark family and every baseline (Appendix E.3), and never re-tuned per dimension; the $d = 10$ and $d = 20$ results are therefore genuinely out-of-sample. We note finally that the PSO coefficients used here are not the literature-standard constriction values, which themselves fail at every Ackley dimension probed ($d = 50, 70, 90$); they were selected by the same diagnostic-guided procedure applied to every baseline (Appendix E.3).

E.4.5 Ensemble-Size Sweep

Every benchmark above fixes the swarm at $n = 100$ (Table 1), so the reliability result is so far a statement at a single ensemble size. This section varies n directly, on the 2D quadruple well, at three depth gaps chosen to span Simulated Annealing's collapse at $n = 100$: $A = 10$ and $A = 20$, where it reaches the target in every seed, and $A = 30$, where it reaches it in none.

The comparison is deliberately restricted to Softmin and Simulated Annealing, and is sharper than it first appears. Every method in this paper is run as an ensemble, so Simulated Annealing at ensemble size n is n *independent* annealed chains: increasing n hands it n restarts of a single-chain method, and its reliability may well improve with n for that reason alone. The question this sweep asks is therefore not whether an ensemble helps, which it does for both methods, but whether Softmin's *coupling* buys anything beyond embarrassingly-parallel restarts at matched cost. Both methods perform the same $O(n)$ work per step, so per-particle evaluation counts are directly comparable at every n. Hyperparameters are held at the values calibrated at $n = 100$ for this benchmark and are not re-tuned per n, so $n = 100$ is the in-sample point and every other ensemble size is out-of-sample for both methods symmetrically.

Reading Figure 3. Ensemble-size sweep on the 2D quadruple well at depth gap $A = 20$, n from 5 to 400 (log axis), 100 seeds per point. This is the one difficulty at which the two methods separate as n varies: at $A = 10$ both reach every seed at every n, and at $A = 30$ Simulated Annealing reaches essentially none at any n, so neither discriminates. Both are shown in Figure 7. Left: reliability, the fraction of seeds reaching the value-space target, plotted at every n with no cutoff; the dashed line is the independent-restart law $1 - (1 - p)^n$ with the single per-chain probability $p = 0.074$ fitted from $n = 5$ alone and every larger ensemble extrapolated, not fitted (Table 11). Center and right are deliberately two different accountings of cost, and the contrast between them is the point. Center: mean first-entry time over the seeds that reached the target *only*, on which Simulated Annealing lies below Softmin throughout – where it works, it is genuinely the faster method, and we do not claim otherwise. Right: expected cost to solution (ERT), the total per-particle gradient work over *all* seeds divided by the number of successes, which charges each method for the runs it never solves. On that accounting the order reverses below $n = 100$: at $n = 5$ Simulated Annealing's conditional cost is 1.1×10^4 against an ERT of 5.3×10^6, because 68% of its runs fail and contribute nothing to the conditional average.

Softmin reaches the target in all 100 seeds at every one of the 21 (difficulty, ensemble size) combinations, including at $n = 5$, a twentieth of the ensemble used everywhere else in this paper. Simulated Annealing splits sharply by difficulty, and the way it splits is the result of this section. At $A = 10$ it also reaches every seed at every n. At $A = 20$ its reliability climbs with n exactly as independent restarts predict: fitting a single per-chain success probability from the smallest ensemble alone, $p = 1 - (1 - 0.32)^{1/5} = 0.074$, the restart law $1 - (1 - p)^n$ predicts 0.54, 0.85, 0.98 and 0.9996 at $n = 10, 25, 50, 100$ against the measured 0.51, 0.84, 0.98 and 1.00, and an exact binomial test rejects the law at no swept point (smallest p-value 0.62). The coupling in Softmin's interaction is what the ensemble buys that restarts do not: for $n = 5$ *independent* chains to reach every seed as Softmin does would require a per-chain probability of at least 0.49, against the 0.074 measured here. The comparison with Langevin dynamics sharpens this further. With $n = 1$ the pre-factor is identically 1, so Softmin's update reduces exactly to that of Langevin dynamics with the same η and σ (Table 5), which on the quadruple well reaches the target in no seed from $A = 10$ on even with $n = 100$ independent particles (Appendix E.4.2). Five Softmin particles thus differ from five independent copies of a method that fails at all three depth gaps only in their interaction. At $A = 30$ the same law, fitted from the single success observed across 400×100 chains, gives $p \approx 2.5 \times 10^{-5}$, some 3000 times smaller than at $A = 20$, and no ensemble size tested recovers it: Simulated Annealing reaches 0% of seeds up to $n = 200$ and 1% at $n = 400$, while Softmin is at 100% from $n = 5$ on.

Table 11: Simulated Annealing's reliability at $A = 20$ against the independent-restart law $1-(1-p)^n$, with the single per-chain probability $p = 0.074$ fitted from $n = 5$ alone and every larger ensemble predicted, not fitted. Fractions of 100 seeds. The last row is the two-sided exact binomial p-value for the observed count under the predicted rate; none rejects at the 5% level. Softmin reaches 100% at every n at this difficulty and is omitted.

Ensemble size n	5	10	25	50	100	200	400
Observed	0.32	0.51	0.84	0.98	1.00	1.00	1.00
Restart law	(fit)	0.54	0.85	0.98	1.00	1.00	1.00
Binomial p	—	0.62	0.67	1.00	1.00	1.00	1.00

That contrast is the signature Theorem 3.4 predicts. Restarts multiply a per-chain success probability without changing its exponent, so a mechanism whose escape rate is governed by the true barrier h can be rescued by more chains at a moderate barrier and cannot be at a large one, while a mechanism whose effective barrier is capped independently of h should not need them at either. We put this no more strongly than consistency, for the reasons given at the start of Section 4: at $\sigma = 1$ the proven cap is too small to be the operative effect, and an agreement of this kind does not on its own establish the mechanism.

The ensemble is not free, and the two methods spend it on opposite things. Both perform the same $O(n)$ work per step, so per-particle gradient counts are directly comparable between them at every n; charging each method for its failures as well as its successes, as the expected running time of Section 4.1 does – the quantity plotted in the right panels of Figures 3 and 7 – Softmin is cheapest at its *smallest* ensemble at every difficulty (1.5×10^4, 3.4×10^4 and 5.4×10^4 evaluations at $A = 10, 20, 30$ with $n = 5$), because it does not need the extra particles for reliability and pays for each one in gradient evaluations. Simulated Annealing's expected cost instead *falls* with n wherever its reliability is still climbing, from 5.3×10^6 at $n = 5$ to 6.0×10^4 at $n = 100$ for $A = 20$, because there the restarts are buying reliability it otherwise lacks. The resulting comparison is difficulty-dependent in the same way as Section 4.1: at $A = 10$ Simulated Annealing is cheaper at every n, by a factor of about 11 at $n = 5$; at $A = 20$ Softmin is some 155 times cheaper at $n = 5$, and once Simulated Annealing's reliability saturates at $n \geq 100$ the two sit within about a factor of two of each other in either direction; at $A = 30$ Softmin's cheapest configuration costs 5.4×10^4 evaluations while Simulated Annealing's expected cost is infinite up to $n = 200$ and 2.0×10^{10} at $n = 400$. None of this contradicts the center panels of the two figures, where Simulated Annealing lies below Softmin at every point of the sweep: that is cost on solved runs alone, and the reversal here is precisely the price of the seeds it does not solve. Which of the two accountings is the relevant one depends on whether a practitioner can tell in advance that a given instance will be solved; when they cannot, it is the expected cost that they pay.

Two qualifications belong with this result. The first is that Simulated Annealing's disadvantage at $A = 20$ is recoverable by brute force: by $n = 100$ it reaches every seed, and from there the two expected costs are within about a factor of two of each other in either direction (Simulated Annealing's is lower at $n = 100$ and $n = 400$, and higher at $n = 200$), so what the sweep establishes at this difficulty is a gap at small ensembles and not a permanent one (Section 5.1). The second concerns Softmin's own entry time at $A = 30$, which is not monotone in n. The median over solved seeds falls from 9.43 at $n = 25$ to 1.18 at $n = 50$ and rises again to 21.31 at $n = 100$, where the distribution also carries a heavy right tail (24% of solved seeds above $t = 50$) that the mean in Figure 7 reports as a spike. Because a whole-distribution shift of that size over 100 seeds could still be an unlucky draw, we re-ran the three affected ensemble sizes on a *disjoint* set of 100 seeds with nothing else changed (`src/confirm_particle_anomaly.py`). It reproduces: the fresh medians are 11.74, 1.18 and 23.70, keeping the same shape across n, and a Mann–Whitney test does not distinguish the two seed batches at any of the three points ($p = 0.37, 0.56, 0.51$). The effect is therefore a property of the ensemble size at this difficulty rather than a sampling accident, and we have no explanation for it; we report it because it qualifies the cost panels, and note that reliability is untouched – Softmin reaches every seed at every ensemble size swept, so the result this section is about does not rest on it.

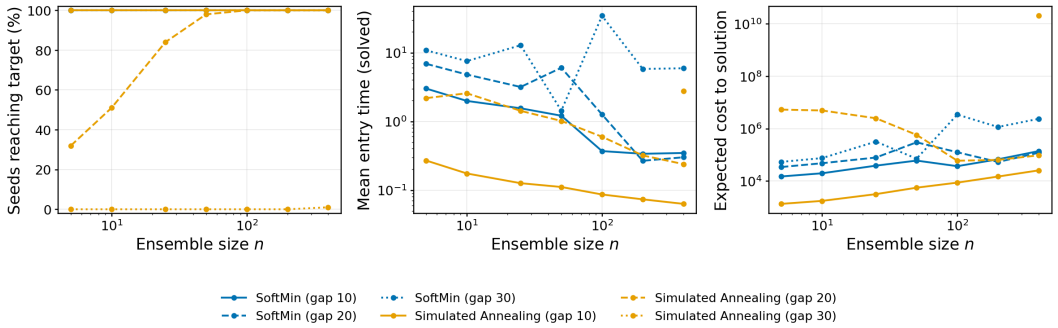

Figure 7: The ensemble-size sweep of Section 4.3 at all three depth gaps (linestyle) rather than the $A = 20$ slice of Figure 3, for Softmin and Simulated Annealing (color); panels as in that figure. The two omitted difficulties are reported here because neither discriminates between the methods as n varies, not because they are unfavorable: at $A = 10$ both reach every seed at every ensemble size, and at $A = 30$ Softmin reaches every seed at every ensemble size while Simulated Annealing reaches none up to $n = 200$ and 1% at $n = 400$. In the right panel a missing point is an infinite ERT, i.e. a configuration in which no seed ever solved the problem: Simulated Annealing at $A = 30$ appears only as the single isolated marker at $n = 400$, at 2.0×10^{10} evaluations, which is the cost of its one success in 400×100 chains. The center and right panels also show Softmin's entry time at $A = 30$ (dotted blue) to be non-monotone in n, with a pronounced spike at $n = 100$; this is discussed in Appendix E.4.5 and is a property of the mean, which these panels report for consistency with every other benchmark in the paper.

All difficulties.

E.4.6 Transformer Benchmark

This appendix gives the setup of the transformer benchmark of Section 4.4 and the detailed analysis behind its summary.

Task. The task is modular addition, $(a + b) \bmod p$ with $p = 53$, presented to the model as the sequence $[a, b, =]$ and predicted from the final position. This is the standard controlled testbed for small transformers (Power et al., 2022; Nanda et al., 2023), and, more directly relevant here, it is already established as an instrument for *comparing optimizers* on transformer training (Tveit et al., 2025), with its depth axis swept within the same task family (Manir & Rupa, 2026). Transformer training is a genuinely hard non-convex problem rather than a stand-in for one (Ahn et al., 2024), and small transformers are a recognized proxy for optimization difficulties that appear at scale (Wortsman

et al., 2023). Two practical properties matter for our purposes: the full dataset is small enough to take *exact, full-batch* gradients, so the comparison keeps the exact-gradient setting used throughout this paper rather than introducing minibatch noise as a new confound; and the whole benchmark is reproducible from a seed with no data download.

Depth sweep and baselines. We sweep depth $\in \{2, 5, 10, 20, 50\}$ with d_{model} and the head count held fixed, so depth is the only architectural variable; the flattened parameter count d is the optimization dimension, from 1.07×10^5 at depth 2 to 2.51×10^6 at depth 50, some five to six orders of magnitude above the well benchmarks and three to four above the largest Lennard-Jones cluster. The range is deliberately wide. A shallower $\{1, 2, 4, 6\}$ sweep did not produce a monotone difficulty gradient, with Adam solving depth 2 faster than depth 1, so we extended depth by an order of magnitude to test whether a deeper model would; as reported below, it does not. A feasibility probe (`src/probe_depth_feasibility.py`) confirms the deep end is genuinely trainable rather than a foregone failure: the measured gradient norm stays flat between 0.79 and 0.93 from depth 5 to depth 50, neither exploding nor vanishing, and Adam drives the loss well below chance at every depth tested. Alongside the swarm baselines we add Adam (Kingma & Ba, 2015) and SGD with momentum, without which a neural-network comparison would not be a comparison. Crucially, these are run as n *independent* chains exactly like every other baseline, so they receive the same $n = 10$ restarts and the same per-step budget the Softmin ensemble consumes: the comparison is cost-matched rather than one-run-against-ten. All n particles are evaluated in a single batched forward and backward pass, since a Softmin step costs one objective and one gradient evaluation per particle. Entry time is measured against a training-loss target of 0.10, as for Lennard-Jones, and reliability is again the fraction of seeds reaching it. Consensus-Based and Particle Swarm Optimization are omitted here: they reach the target in at most 3% of seeds at every difficulty on both well benchmarks, so sweeping them across five depths would spend a substantial part of the budget re-measuring a flat zero.

Calibration. Three settings do not transfer to weight space and are calibrated for this benchmark family by the same diagnostic-guided procedure applied to every other family (Appendix E.3, Table 7), exactly as Simulated Annealing's T_0 and Particle Swarm Optimization's coefficients already are. The first is the learning rate. At $\eta = 10^{-3}$, the value used on the analytic benchmarks, a step moves each coordinate by some 1.4×10^{-6} against weights of scale 10^{-1}, and the loss does not leave its initial value within any feasible budget: after 400 steps it sits at 3.9996, *above* the chance level $\log p = 3.970$, at every gradient-clipping value tested. Adam is exempt by construction, since it rescales per coordinate; that is precisely why it must not be the only method able to move, and $\eta = 0.1$ is selected instead. It remains stable at the deepest sweep point: the sweep runs the adopted configuration unchanged at every depth, and every run reaches the target at depth 50 as at every other depth. The second is the gradient clip, which a first calibration pass on a 1-layer model had found inactive and which does not stay inactive as depth grows: at depth 10 the per-particle gradient norm averages 0.9311 but reaches 1.0341, so a clip of 1 does bind, and it costs a great deal – at $\eta = 0.1$ and 400 steps, clipping at 1 reaches only 1.1648 against 0.0754 at 10. We therefore raise the clip to 10, which reproduces the unclipped loss to four decimals and so bounds a pathological step without shaping the dynamics. The third is the noise level. The per-step perturbation has standard deviation $\sqrt{2\sigma\eta}$, so at $\eta = 0.1$ the analytic benchmarks' $\sigma = 1$ would inject about 0.45 per coordinate, against weights of scale 10^{-1}. Training breaks down long before that: already $\sigma = 10^{-3}$ (a perturbation of about 0.014) leaves the loss at 3.95, at chance, and $\sigma = 10^{-2}$ (about 0.045) at 4.08, above it. The sweep's own optimum is $\sigma = 0$, which we deliberately do not adopt: it would reduce Simulated Annealing and Langevin dynamics to one and the same deterministic gradient descent, collapsing two of the four baselines into duplicates of each other. We use $\sigma = 10^{-5}$, the smallest value that keeps every method's dynamics distinct, at a cost well inside the target margin.

The inverse temperature is *not* re-tuned: $\beta = 10$ is carried over from every other benchmark. The sweep is decisive about its order of magnitude, since $\beta = 0.1$ and $\beta = 1$ do not learn here at all, but $\beta = 10$ and $\beta = 100$ both clear the target comfortably, so the shared value is kept rather than the nominal per-benchmark optimum. Adam and SGD use their literature-standard defaults, the corresponding convention for them.

Reliability and depth. Every method reaches the target on every seed at every depth: all 125 runs succeed. This benchmark therefore does not discriminate on reliability, and we draw no reliability conclusion from it. Nor did depth produce the difficulty gradient the sweep was designed around.

Every baseline reaches the target *sooner* in the deepest model than in the shallowest: from depth 2 to depth 50 the mean entry time falls by a factor of 2.48 for SGD, 1.88 for Langevin dynamics, 1.64 for Adam, and 1.44 for Simulated Annealing (Figure 4). Two explanations are plausible, and we tested neither: the training set is fixed at $2{,}247$ examples, which a larger model may simply fit more easily; and the model is a pre-normalized residual architecture, a design intended to keep added depth from degrading optimization. Either way, at this target the landscape is benign for every method tested, and what the benchmark measures is speed and dimensional scaling, not the escape from poor basins that the theory addresses.

Speed and cost. Measured in steps, Softmin's entry time is the least sensitive to depth of the five methods: its median stays between 340 and 410 steps at every depth, a spread of $1.21\times$ against $1.49\times$ to $2.49\times$ for the others. Its position relative to the baselines follows from that. It is faster than Simulated Annealing at every depth, by a factor of 3.8 to 4.9 in median entry time, and faster than Langevin dynamics, by 1.4 to 2.3. Against SGD its margin is 2.1 at depth 2 and narrows as SGD speeds up with depth, to 1.13 at depth 20, until SGD is marginally ahead at depth 50 (median 328 steps against 342; on the mean, which the slow run described below inflates, 327 against 423). Adam is the fastest method at every depth, by a factor of 2.8 at depth 2 and 4.0 to 4.5 from depth 5 onward. Wall-clock time is less favorable to Softmin than step counts. The model evaluations are identical in number across methods, but Softmin's ensemble update is the most expensive per step, at 2.5 s against 2.0 s for Adam and 1.8 s for SGD at depth 50. SGD is therefore faster than Softmin in wall-clock from depth 5 onward, and Softmin's wall-clock margin over Langevin dynamics vanishes at depth 50, though it stays 2.7 to 4.1 times faster than Simulated Annealing.

Slow runs. One Softmin run is an outlier. At depth 50, seed 2 reaches the target at step 743, against 333 to 363 for the other four seeds, while no other method is unusual on that seed. To tell a rare tail from a fluke, we reran Softmin alone for 25 seeds at depths 20 and 50. The ten seeds shared with the sweep reproduce its entry times and final losses exactly, so the slow run is a property of its seed and not of hardware nondeterminism. The slow runs recur: at depth 50, five of the 25 seeds take between 527 and 743 steps, separated by a clear gap from the other twenty, which all finish within 403; at depth 20, one seed of 25 takes 551 steps, against at most 385 for the rest. The two rates do not differ significantly (Fisher's exact test, $p = 0.19$). Nothing in the loss by step 250 singles these seeds out: at depth 50 they rank 5th, 6th, 19th, 21st and 24th of 25, and our checkpoints are too sparse to tell whether they plateau or descend more slowly. None comes close to failing, since the slowest uses 743 of the $2{,}000$ allowed steps, and the medians reported above are robust to them: over 25 seeds Softmin's median is 345 steps at depth 50 and 360 at depth 20. We ran the additional seeds for Softmin only, so we cannot say whether the other methods have a comparable tail: at a rate of one run in five, five seeds would show none of it a third of the time.

E.5 Further Experimental Results

E.5.1 Theory Verification

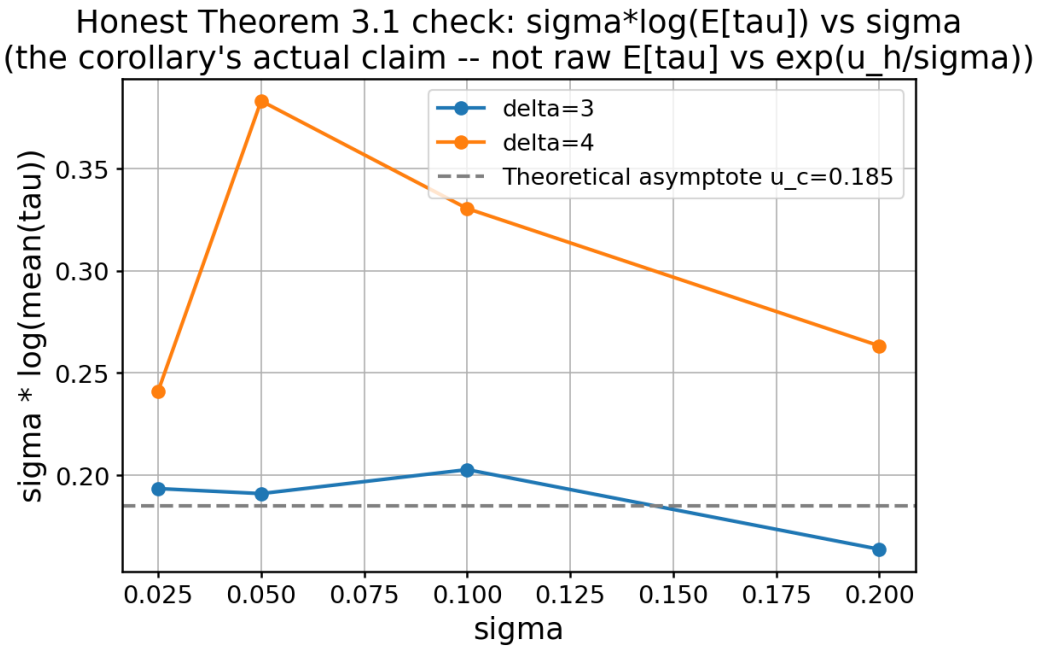

Figure 8: $\sigma \log(\text{mean } \tau)$ vs. σ at $\delta = 3, 4$, i.e. barrier heights $h = 2.19$ and $h = 3.93$ (both already in the saturated $h > r_c$ regime), against the theoretical asymptote $u_c = u(r_c, \beta, n)$. This is the quantity Corollary 3.5 would constrain – $\lim_{\sigma \to 0} \sigma \log \mathbb{E}[\tau] = V_\beta \leq u_h$, and $u_h = u_c$ in this saturated regime – under the metastable-domain hypotheses that corollary assumes and that we do not verify here; the plot is therefore a consistency check, not a test of a proven identity. It is nonetheless the right quantity to plot, unlike a direct overlay of raw $\mathbb{E}[\tau]$ against $\exp(u_c/\sigma)$ at one fixed, non-vanishing σ, which conflates the proven log-asymptotic rate with an uncontrolled, h-dependent sub-exponential pre-factor. Both series trend toward u_c as σ shrinks, consistent with the corollary; neither is monotonic all the way down, which is expected finite-σ behavior, not a violation.

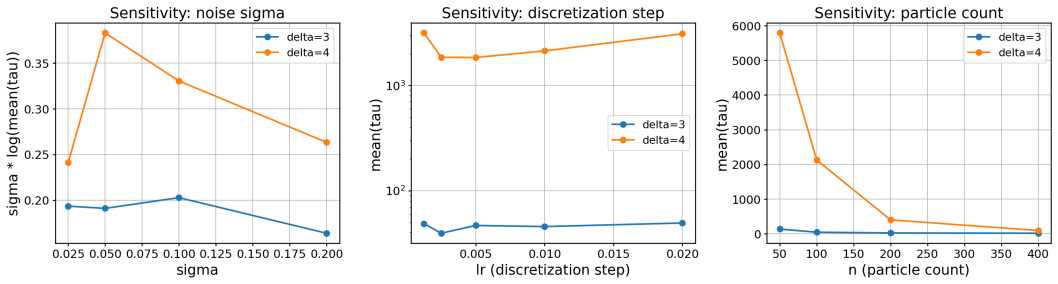

Figure 9: Sensitivity of the same corollary check to the discretization step η and particle count n, at $\delta = 3, 4$ ($h = 2.19, 3.93$). $\sigma \log(\text{mean } \tau)$ (left) reproduces Figure 8; mean τ vs. η (center) should flatten as $\eta \to 0$ if the discretization is close to its continuous-time limit; mean τ vs. n (right) should only ever improve with more particles.

E.5.2 COMPUTATIONAL COST COMPARISON

Table 12: Mean per-particle gradient/objective evaluation counts and wall-clock time, averaged over the full double-well/quadruple-well sweep actually run (not a short synthetic probe). Counts are per-particle (n_particles $\times$ steps), not per vectorized call. Because a run that never escapes contributes its full step budget to the average, SA's and Langevin's large counts partly reflect how often they exhaust the budget without reaching the target (Figure 1), not a per-step cost difference – every gradient-based method performs the same $O(n)$ work per step. Standard CBO/PSO have zero gradient evaluations by construction (gradient-free); Standard CBO's nonzero objective-evaluation count above PSO's reflects its consensus-weight computation, which re-evaluates the objective every step. The three wall-clock columns price the same runs three ways and must be read together. *All runs* is the plain mean over all 5,900 runs, and is the column the timeout caveat above applies to. *Solved* averages over just those seeds whose logged status is `target_reached`, removing that confound entirely, and on it Simulated Annealing beats Softmin by roughly $15\times$: where SA works it genuinely is the cheaper method, and Table 10 locates the barrier height at which that stops holding. *ERT* is the expected running time (Hansen et al., 2021): total wall-clock over all seeds divided by the number of successes, which under a restart model is the expected cost of the first success and so charges each method for its own failures. ERT is the column that reconciles the other two, and the only one that does not presuppose knowing in advance which instances a method will solve. Reached-fractions are pooled over both well sweeps and so sit below any individual sweep point's.

			Wall-clock (s)			
Method	**Grad. evals**	**Obj. evals**	**All runs**	**Solved**	**ERT**	**Reached**
Softmin	502,150	502,150	11.68	11.68	11.68	100.0%
Simulated Annealing	14,149,330	14,149,330	351.57	0.79	485.21	72.5%
Langevin	42,907,470	42,907,470	1033.74	68.44	4484.61	23.1%
CBO	110,514	110,514	2.79	0.66	5491.64	0.1%
CBO Annealed	110,807	110,807	2.79	1.01	16,449.79	0.0%
PSO	99,868	99,868	2.43	0.04	1022.99	0.2%
Standard CBO	0	1,363,282	18.08	2.40	9699.14	0.2%
Standard PSO	0	99,852	1.31	0.02	514.56	0.3%

E.5.3 Ascent-Descent Mechanism Illustration

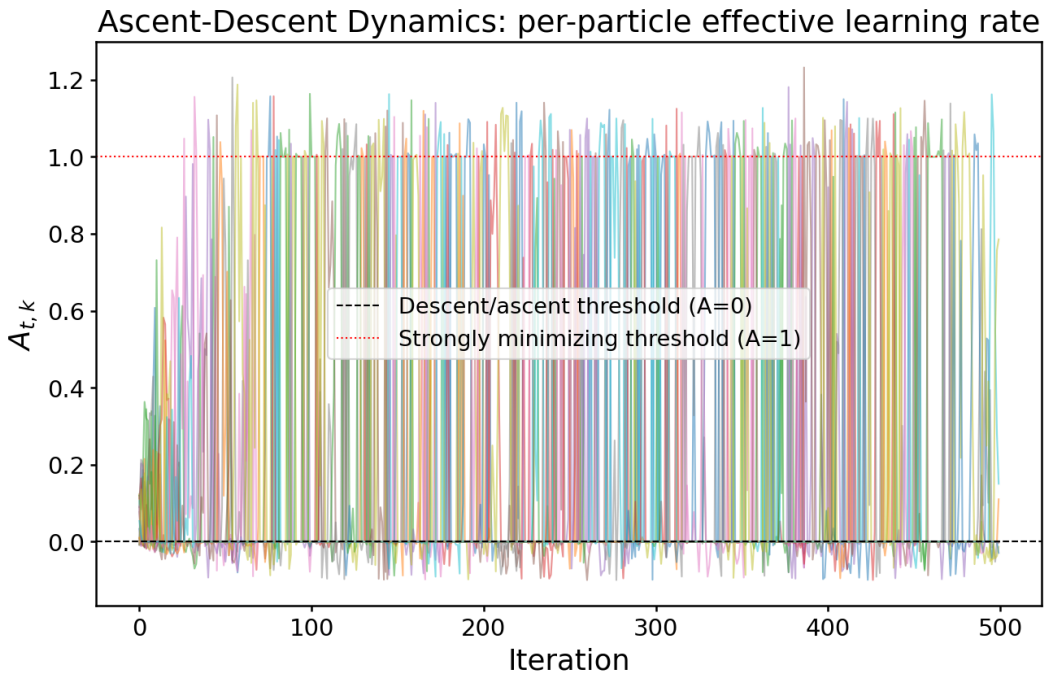

Figure 10: Per-particle effective learning rate $A_k(\mathbf{x}_t)$ over a single Softmin run on the 1D double well ($\delta = 4$, i.e. $h = 3.93$; 20 particles). Particles with $A_k(\mathbf{x}_t) > 0$ descend (more strongly so above the $A_k(\mathbf{x}_t) = 1$ line); particles with $A_k(\mathbf{x}_t) < 0$ ascend, the mechanism the ablation study below tests directly.

E.5.4 Ablation Study: Ascent-Descent Mechanism

We isolate the contribution of the sign-changing pre-factor $A_k(\mathbf{x})$ itself by comparing the real (*signed*) mechanism against two ablations: *positive part*, $(A_k(\mathbf{x}))_+$, which freezes any particle the real mechanism would have pushed uphill instead of letting it ascend; and *absolute value*, $|A_k(\mathbf{x})|$, which forces every particle to always descend, at whatever magnitude the real mechanism would have applied. All three variants share the same Gibbs/softmax weighting $n \cdot w_k$ that scales $A_k(\mathbf{x})$ in the first place – only the sign treatment on top of it differs.

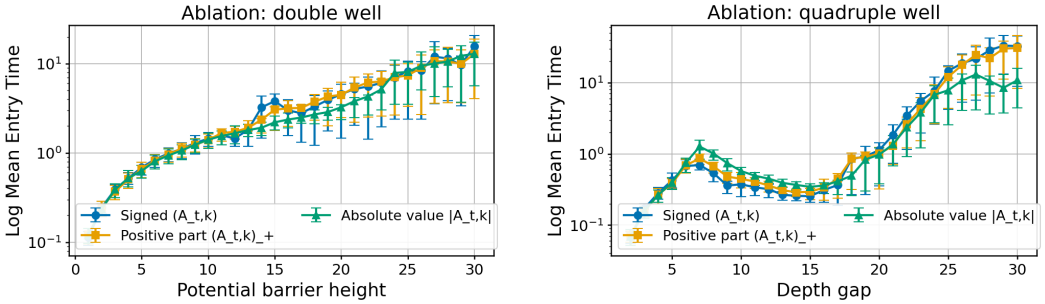

Figure 11: Ascent-descent mechanism ablation on the two swept well benchmarks (mean entry time over the seeds that reached the target, band spanning their interquartile range, vs. difficulty; same value-space target and status-based masking as Figure 5).

The result is not the clean "signed wins, the ablations degrade" story the sign-flipping mechanism might suggest. On the two swept well benchmarks (Figure 11), there is no consistent winner:

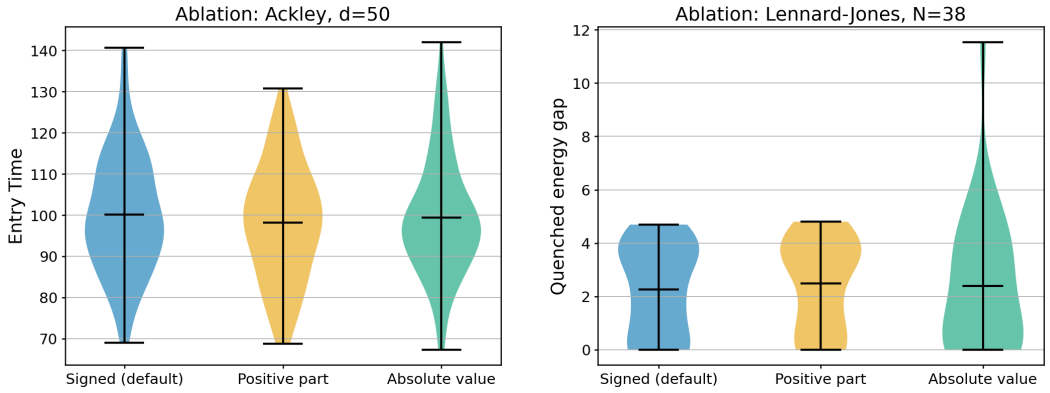

Figure 12: Ascent-descent mechanism ablation at two fixed, hard settings with no natural sweep axis (Ackley $d = 50$, 100 seeds; Lennard-Jones $N = 38$, 30 seeds), shown as per-seed distributions (violin plots) rather than a single mean-and-interquartile-range point per mode, since the three modes turn out to substantially overlap here.

at the easier settings signed is competitive or best (e.g. quadruple well $A = 15$: signed 0.25, positive part 0.29, absolute 0.34), but at harder settings absolute is clearly faster – double well $\delta = 15$ ($h = 56.1$): signed 3.81, positive part 3.06, absolute 2.20; quadruple well $A = 30$: signed 32.69, positive part 30.70, absolute 10.82, roughly $3\times$ faster than either alternative that still preserves or freezes the sign. At the two fixed, hard settings (Figure 12) the three modes are close to indistinguishable: Ackley $d = 50$ entry time (mean $100.1/98.2/99.4$ for signed/positive part/absolute, all 100% reached) and Lennard-Jones $N = 38$ quenched-energy gap ($2.26/2.48/2.40$) show heavily overlapping distributions.

Taken together, these four benchmarks suggest that preserving the sign of $A_k(\mathbf{x})$ – letting trailing particles genuinely ascend rather than merely stall or be redirected downhill – is not what is doing most of the practical work in this noise regime ($\sigma = 1$). The Gibbs/softmax weighting $n \cdot w_k$, identical across all three variants, already strongly suppresses trailing particles' contribution regardless of the sign convention applied on top of it; that shared attenuation is the more plausible source of the empirical advantage over the baselines on the well and Lennard-Jones benchmarks of Section 4. The sign-flipping ascent mechanism remains what Theorem 3.4 is stated and proved for, and is a $\sigma \to 0$ asymptotic statement – these benchmarks run at a fixed, non-vanishing σ, and the ablation results here are consistent with the sign convention mattering more as σ shrinks than it does at the noise level tested.

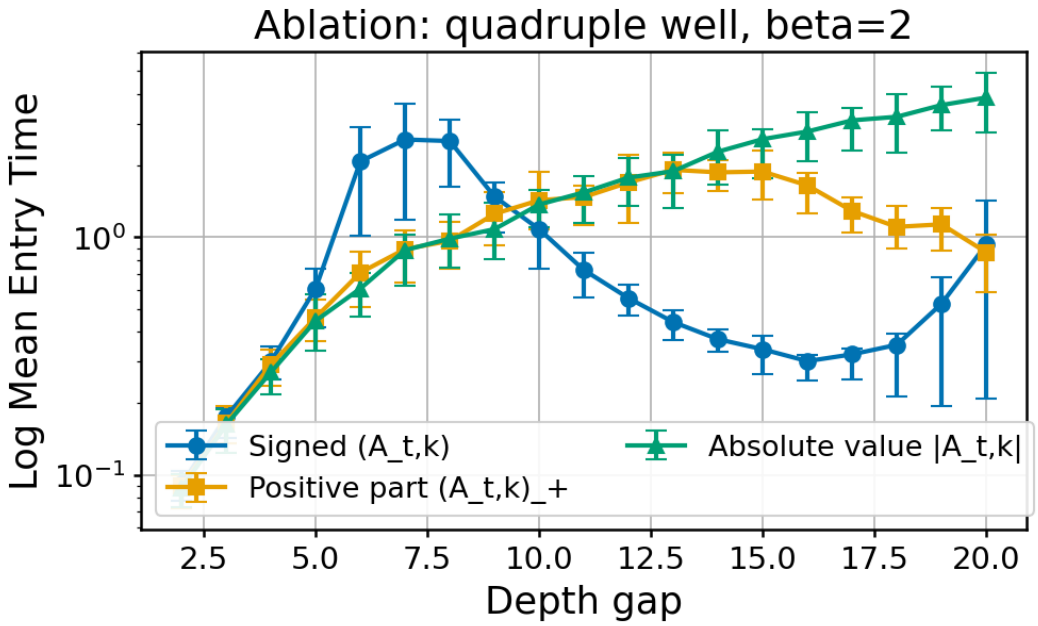

Figure 13: Ascent-descent mechanism ablation on the 2D quadruple well at a lower inverse temperature ($\beta = 2$, vs. $\beta = 10$ above; 50 seeds per point, $A = 2, \ldots, 20$), where the Gibbs weight $n \cdot w_k$ no longer concentrates sharply enough to mask the sign convention's effect.

The near-parity at $\beta = 10$ raises a natural question: is it specific to that aggressive a weighting? A sharply concentrated Gibbs weight already suppresses trailing particles' contribution regardless of the sign convention applied on top of it, by the argument above, so a less concentrated weighting should let the sign convention's own effect show through more directly. Repeating the quadruple-well ablation at $\beta = 2$ (Figure 13) is consistent with this: at the easier settings absolute and positive part are faster than signed, sometimes markedly so (depth gap 7: signed 2.58, positive part 0.89, absolute 0.88), but this reverses once the barrier grows past a depth gap of about 10 – from depth gap 13 through 19 signed is consistently and substantially faster (e.g. depth gap 15: signed 0.34, positive part 1.89, absolute 2.59; depth gap 19: signed 0.52, positive part 1.13, absolute 3.60, nearly $7\times$ faster than absolute). At this lower temperature, preserving the sign of $A_k(\mathbf{x})$ – letting trailing particles genuinely ascend rather than merely stall or be redirected downhill – appears to become the dominant factor as the barrier increases: in this sweep, as the escape gets harder, the variant that lets trailing particles actively climb out is the one that keeps up, which is consistent with the sign-flipping mechanism Theorem 3.4 is stated and proved for.

F Extended Limitations and Future Work

Our theoretical analysis provides rigorous guarantees in the small-noise regime where the Freidlin–Wentzell Large Deviation Principles apply. While the bound in Theorem 3.4 is not sharp – in particular u_c and u_h are only upper-estimated by an explicit channel, not computed exactly for general f – a tighter characterization of the true quasipotential, together with intermediate-noise asymptotics, would sharpen the picture. Our equilibrium characterization (Theorem 2.1) requires no convexity assumption on f whatsoever; the ensemble-descent estimate (Theorem 2.4) guarantees the running minimum is nonincreasing regardless, but its exponential rate of convergence specifically to the global minimum requires a Polyak–Łojasiewicz condition on the relevant sublevel set, as discussed after its statement. The exit-time consequence of Theorem 3.4 holds unconditionally only at the level of the quasipotential, and becomes an actual statement about $\mathbb{E}\tau^\sigma$ only under the unverified metastable-domain hypotheses of Corollary 3.5; identifying the exact transition exponent between adjacent basins, and a rigorous large-deviation treatment of the annealed schedule β_t, remain open.

A second gap is empirical, between what the theory describes and what the benchmarks measure. The analytic benchmarks run at $\sigma = 1$, where the cap of (13) evaluates to $u_c \approx 0.037$ and so is negligible against the noise; the ablation of Appendix E.5.4 correspondingly finds that at $\beta = 10$ the sign-reversal mechanism the theory is built on is not what produces the observed advantage, with the Gibbs weighting shared by all three ablation variants the more plausible source. The mechanism does appear to become dominant at the lower temperature $\beta = 2$ as barriers grow, consistently with the theory; but establishing that the proven mechanism drives the headline results would require either running the benchmarks down toward the small-noise regime or a control that removes the drift entirely, neither of which we do here. The Ackley results of Section E.4.4 bound the claim from the other side. There the sign-reversal mechanism is active – the landscape's barriers sit well above r_c – and Softmin's performance improves with dimension as those barriers shrink, yet PSO is faster at every dimension, on a benchmark where gradient information demonstrably contributes nothing. Barrier flattening is therefore not sufficient for competitive speed on a landscape whose difficulty is carried by the density of minima rather than by their depth. Why PSO in particular does so well here we leave open: we did not isolate the responsible mechanism, and the comparison identifies the limitation without explaining it. The ensemble-size sweep of Section 4.3 bounds the reliability claim from a third side. At the moderate depth gap $A = 20$, Simulated Annealing's reliability is recovered in full by restarts alone once $n \geq 100$, and from there its expected cost to solution sits within about a factor of two of Softmin's; the advantage we report at the paper's default $n = 100$ is therefore specific to difficulties at which restarts have stopped working, and is not a uniform property of the method at every barrier height. The transformer benchmark of Section 4.4 bounds it from a fourth side: every method reached the target on every seed at every depth, so our experiments contain no test of reliability on a neural-network objective, and on that benchmark Adam is several times faster than Softmin, while SGD catches up with it in steps as depth grows and is faster in wall-clock time from depth 5 on.

Future work should address several important directions. First, extending convergence proofs and explicit convergence rates beyond the Polyak–Łojasiewicz setting, particularly to objectives with multiple disconnected basins, would significantly broaden applicability. Second, analyzing the mean-field limit as $n \to \infty$ could provide insights into the scaling of the method with ensemble size and potentially yield dimension-free convergence rates; Section 4.3 measures that scaling empirically over $5 \leq n \leq 400$ and finds reliability already saturated at the bottom of that range, which such an analysis would need to explain. Third, although Theorem 2.4 guarantees the ensemble-descent bound for an arbitrary schedule β_t, our experiments (Remark D.4) found that neither a decaying nor an increasing schedule improved on a well-chosen fixed β on the benchmarks considered, contrary to the original motivation for annealing β over time; understanding whether this is a property of these specific landscapes or a more general phenomenon, and whether an adaptive schedule driven by the ensemble's own state (rather than a fixed function of t) can do better, is an open, concrete direction. Fourth, our neural-network benchmark turned out to be one on which every method succeeds; whether Softmin's reliability advantage carries over to network-training problems on which runs do fail, where it could separate from Adam, remains untested. Finally, tighter estimates for the exponential rate in the small noise regime can sharpen the theoretical bound to match experimental evidence more closely.