Convex Compositional Reasoning Models

Meir Roketlishvili[1] **Semyon Semenov**[1] **Maksim Bobrin**[2,3] **Viktor Kovalchuk**[1] **Albert Baichorov**[1]
Abduragim Shtanchaev[1] **Fakhri Karray**[1] **Dmitry V. Dylov**[2,3] **Martin Takáč**[1] **Arip Asadulaev**[1]

[1] Mohamed bin Zayed University of Artificial Intelligence
[2] Applied AI Institute, Computational Imaging Lab
[3] AXXX

Meir.Roketlishvili@mbzuai.ac.ae

Abstract

Compositional energy-based models can generalize to larger combinatorial reasoning problems by reusing a learned factor energy across many local constraints. In our paper, we show that a key bottleneck in compositional reasoning is not composition itself, but the non-convex geometry of the learned energy landscape. To solve this problem, we introduce Convex Compositional Energy Minimization (CCEM), a framework that parameterizes each factor with an input-convex neural network and optimizes the composed energy over a tight convex relaxation of the feasible set. Because convexity is preserved under summation, the global relaxed objective remains convex, enabling deterministic projected first-order optimization. CCEM is trained in two stages: factor-level contrastive learning to shape local energy basins, followed by end-to-end refinement through an unrolled projected solver. Our experiments show that our models trained on small subproblems or a single problem size transfer to larger instances without retraining. The official implementation of CCEM is publicly available at https://github.com/Meir-Rocket/CCEM.

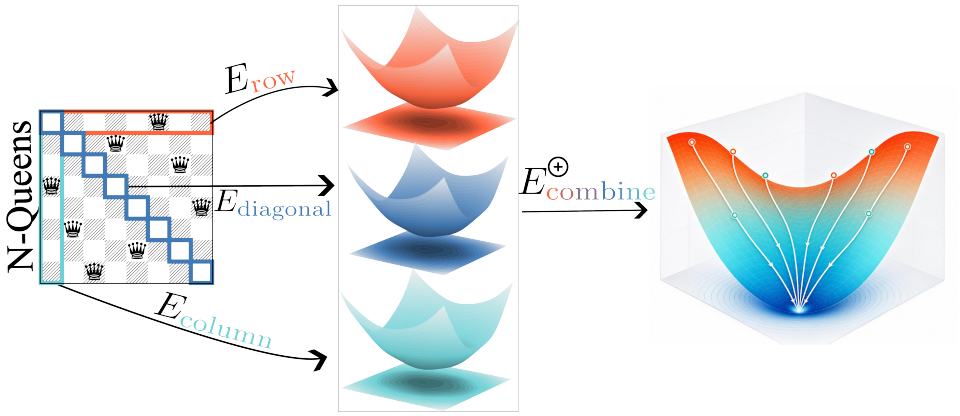


Figure 1: Our method composes convex factors into a smooth globally convex landscape, enabling efficient deterministic reasoning.

1 Introduction

Many reasoning problems are naturally compositional. A candidate solution is valid only when it satisfies many local constraints: every row, column, and diagonal in tabular data, every edge in graph coloring, or every clause in a satisfiability formula. This structure suggests a simple route to

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

generalization: learn an energy model for a small local constraint, reuse it across all constraints in a larger instance, and solve the full problem by minimizing the sum of the local energies. Compositional energy-based models follow this principle. Given an instance x and a candidate relaxed solution y, they define $E_\theta(x, y) = \sum_{k=1}^{K} w_k f_\theta(y_{S_k}; c_k)$, where each factor f_θ scores a local scope S_k under context c_k. The same factor network can therefore be trained on small subproblems and applied to larger instances simply by adding more terms to the sum.

However, composition also creates the main optimization difficulty. If the factor energy is an unconstrained neural network, then the composed objective is generally non-convex. Even when each local factor appears well behaved in isolation, their sum can introduce **spurious local minima** (see proposition 3.1). *As the number of constraints grows, the optimizer must search a larger and more rugged energy landscape.* Prior work addresses this issue with powerful sampling procedures, such as long diffusion chains or Parallel Energy Minimization (PEM), which use many particles and noise injection to escape poor local basins [17]. These methods can be effective, but they treat the symptoms of a non-convex composed landscape rather than the cause.

In this paper, we take a different approach. *We argue that the difficulty is not inherent to compositional reasoning, but to the parameterization of the local energy factors.* If each factor is convex in the decision variable, and the factors are composed using nonnegative weights, then **the full relaxed energy remains convex**. Adding more constraints may make the optimization problem larger or more ill-conditioned, but it cannot create new nonglobal local minima. This changes the role of inference: instead of relying on particle-based exploration to escape a rugged landscape, we can minimize a globally well-behaved relaxed objective with projected first-order methods (Theorem 3.4).

We introduce *Convex Compositional Energy Minimization* (CCEM), a framework for compositional **reasoning with convex factor energies**. Each local factor is parameterized by an input-convex or partially input-convex neural network (ICNN) [1], so that it remains convex in the candidate solution variables while still conditioning flexibly on the problem context. At test time, the global energy is obtained by summing all local factors and is optimized over a tight convex relaxation of the feasible set, such as the Birkhoff polytope for N-Queens or simplex-based relaxations for coloring. The resulting inference procedure is deterministic, differentiable, and compatible with other solvers.

Our training pipeline has two stages. First, we train the factor energy using local contrastive supervision, assigning low energy to locally valid patterns and higher energy to invalid alternatives. Second, we refine the composed energy end-to-end by backpropagating through an unrolled projected solver. This refinement aligns the energy with the behavior of the actual inference procedure while preserving the convex structure of the model. Unlike diffusion-based compositional EBMs, CCEM does not need to learn a denoising trajectory in ambient Euclidean space; the optimization is performed directly on the convex relaxation where feasibility is enforced by projection.

The central benefit of convexity is theoretical as well as practical. We show that interactions among unrestricted non-convex factors can leave stable but globally incorrect minima in the composed objective. In contrast, convex composition rules out nonglobal local minima over the relaxation. We explicitly separate this continuous guarantee from discrete recovery: integrality and valid decoding are not consequences of convexity alone. A decoder-aware energy margin gives a sufficient condition under which finite-accuracy relaxed optimization transfers to a valid discrete answer. This condition is instance-dependent, and we make no worst-case polynomial-time claim for the NP-hard tasks considered here. We make the following **contributions**:

- We identify interactions among non-convex factor geometries as a source of stable, globally incorrect minima in compositional EBMs, separating this issue from the choice of sampler.
- We propose a novel compositional reasoning framework that uses ICNN factor energies so that nonnegative factor summation preserves convexity of the relaxed global objective.
- We develop a projected optimization pipeline for training and inference directly on tight convex relaxations, replacing diffusion sampling with deterministic first-order minimization.
- We distinguish relaxed optimization from discrete recovery and give decoder-aware margin conditions that combine approximation, optimization, and decoding errors, together with the corresponding complexity boundary.
- We evaluate the method on N-Queens, graph coloring, and 3-SAT, reasoning benchmarks, showing that convex compositional energies can match or improve particle-based compositional EBMs while using a simpler inference procedure.

2 Preliminaries

2.1 Reasoning as compositional energy minimization

A useful way to view reasoning is as search over possible answers. Given a problem instance x, the model defines an energy function $E_\theta(x, y)$ over candidate solutions y, where valid solutions should have low energy and invalid ones higher energy. Solving the problem then means finding

$$\hat{y} = \arg\min_y E_\theta(x, y).$$

This differs from the usual end-to-end reasoning setup, where a model directly maps inputs to outputs and often learns heuristics tied to the training distribution. In the energy-based view, inference is an optimization process, so harder problems can be given more computation at test time. Compositional reasoning adds the idea that many reasoning problems are made of smaller constraints. For example, N-Queens combines row, column, and diagonal constraints; graph coloring combines edge constraints; and SAT combines clauses. Instead of learning a separate model for each full problem size, we learn a local energy for one constraint and reuse it across the larger instance. Thus, regular reasoning treats the problem more as one object, while compositional reasoning builds the solution by combining reusable pieces. Following Oarga and Du [17], if feasibility factorizes into K local constraints, we define the global energy as

$$y \in \mathcal{S}(x) \iff \bigwedge_{k=1}^{K} \mathcal{C}_k(y_{\mathrm{sc}(k)}, c_k), \qquad E_\theta(x, y) = \sum_{k=1}^{K} w_k \, f_\theta\big(y_{\mathrm{sc}(k)};\, c_k\big)\,. \tag{1}$$

Here, $\mathrm{sc}(k)$ are the variables involved in the k-th constraint, c_k is its context, and the same factor network f_θ is reused for all factors. In this sense, the model learns the local rule once and applies it many times. Larger problems are handled by adding more energy terms, rather than retraining.

The benefit is that learning is separated from problem size. The difficulty is that summing many learned factors can make the energy landscape harder to optimize, especially when the factors are unconstrained neural networks. This is why prior work relies on sampling methods such as Parallel Energy Minimization, while our approach focuses on making the composed landscape better behaved by choosing factor energies whose structure is preserved under summation.

2.2 Input Convex Neural Networks.

ICNNs [1] parameterize a scalar function that is convex in (a subset of) its inputs by combining non-negative weights with non-decreasing convex activations. They have been used for control [5], optimal transport [11, 16], and domain adaptation [2, 3]. To our knowledge, ICNNs have not previously been used as the per-factor energy of a compositional reasoning model: prior compositional EBMs deliberately use unrestricted MLP scores so that the composed landscape can be *shaped* by additional losses, whereas we exploit the closure of convexity under non-negative sums to keep the composed objective tractable by construction.

3 Convex Compositional Energy Minimization

Our main claim is that an important obstruction in unconstrained compositional EBMs is the geometry of the composed objective, rather than only the choice of sampler. When local factors are unrestricted and non-convex, minima that are legitimate for individual factors can remain stable but become globally incorrect after the factors interact in the sum. We do not claim that summation creates non-convexity from convex factors; the counterexample instead shows how non-convex factor geometries can produce a bad composed landscape.

Proposition 3.1 (Stable invalid minima from unrestricted factors)**.** *There exist a convex relaxation $\mathcal{Y} \subset \mathbb{R}^d$, smooth non-convex factors f_k, and a composed energy (1) such that E has a global minimizer $y^\star \in \mathcal{Y}$ and a distinct strict local minimizer $\bar{y} \in \mathcal{Y}$ with*

$$E(y^\star) = \min_{y \in \mathcal{Y}} E(y), \qquad \nabla E(\bar{y}) = 0, \qquad \nabla^2 E(\bar{y}) \succ 0, \qquad E(\bar{y}) > E(y^\star).$$

Moreover, for some stepsize $\eta > 0$, gradient descent $y_{t+1} = y_t - \eta \nabla E(y_t)$ converges to $\bar{y}$ from all initializations in an open neighborhood of $\bar{y}$.

Proposition 3.1 motivates imposing convexity at the factor level. Since nonnegative sums of convex functions remain convex, factor composition cannot introduce nonglobal local minima over a convex relaxation. This statement concerns the continuous optimization problem. It does not by itself imply that a relaxed minimizer is integral or that its decoded discrete assignment is valid.

For instance x, let $\mathcal{Y}_x$ be the compact convex relaxation, $\mathcal{Z}_x$ the discrete output space, $\mathcal{V}_x \subseteq \mathcal{Z}_x$ the valid discrete solutions, and $D_x : \mathcal{Y}_x \to \mathcal{Z}_x$ the test-time decoder. Define the decoder-valid region and its robust complement by

$$\mathcal{A}_x = \{y \in \mathcal{Y}_x : D_x(y) \in \mathcal{V}_x\}, \qquad \mathcal{B}_x = \overline{\mathcal{Y}_x \setminus \mathcal{A}_x}. \tag{2}$$

The closure places argmax and threshold tie boundaries in $\mathcal{B}_x$, since decoding there is not robust. For the learned energy, write

$$E_x^\star = \min_{y \in \mathcal{Y}_x} E_\theta(x, y), \qquad \mathcal{M}_x = \arg\min_{y \in \mathcal{Y}_x} E_\theta(x, y), \tag{3}$$

and define its decoder-aware margin

$$\Gamma_x^D(E_\theta) = \min_{y \in \mathcal{B}_x} \big(E_\theta(x, y) - E_x^\star\big). \tag{4}$$

Lemma 3.2 (Separation of relaxed and discrete guarantees)**.** *If $E_\theta(x, \cdot)$ is continuous and convex on compact $\mathcal{Y}_x$, then $\Gamma_x^D(E_\theta) > 0$ if and only if $\mathcal{M}_x \cap \mathcal{B}_x = \varnothing$. Moreover, every $\widehat{y}$ with*

$$E_\theta(x, \widehat{y}) - E_x^\star < \Gamma_x^D(E_\theta)$$

decodes validly: $D_x(\widehat{y}) \in \mathcal{V}_x$.

Thus integrality is sufficient but not necessary: a fractional minimizer is harmless when it lies strictly inside a valid decoder cell. Conversely, a globally optimized convex relaxation can still fail if its minimizer lies in an invalid cell or on a decoding boundary. The following conditional transfer result also includes model approximation and finite solver error.

Theorem 3.3 (Decoder-aware certificate transfer)**.** *Let G_x be a continuous convex reference energy whose decoder-aware margin over $\mathcal{B}_x$ is*

$$\gamma_{\mathrm{dec}} = \min_{y \in \mathcal{B}_x} G_x(y) - \min_{y \in \mathcal{Y}_x} G_x(y) > 0.$$

Suppose $\sup_{y \in \mathcal{Y}_x} |E_\theta(x, y) - G_x(y)| \le \Delta$, and the solver returns $\widehat{y}$ satisfying

$$E_\theta(x, \widehat{y}) - \min_{y \in \mathcal{Y}_x} E_\theta(x, y) \le \delta.$$

If $2\Delta + \delta < \gamma_{\mathrm{dec}}$, then $D_x(\widehat{y}) \in \mathcal{V}_x$.

For the compositional model, factorwise errors ε_k imply $\Delta \le \sum_k w_k \varepsilon_k$. Theorem 3.3 is deliberately conditional: it does not establish that efficiently constructible certificates with positive margins exist for arbitrary N-Queens, graph-coloring, or 3-SAT instances. Indeed, a uniform inverse-polynomial-margin construction together with polynomial-time convex optimization would yield a polynomial-time algorithm for the corresponding NP-hard search problem. Appendix A.1 gives the proofs, this complexity boundary, and directly checkable margin certificates. For row-wise argmax, a valid unique minimizer $y^\star$ with minimum top-two gap

$$\Delta_{\arg} = \min_i \left(y^\star_{i,j_i^\star} - \max_{j \neq j_i^\star} y^\star_{i,j} \right) > 0 \tag{5}$$

has a neighborhood of radius $\Delta_{\arg}/2$ that decodes identically. If the energy additionally has quadratic growth $E_\theta(x, y) - E_x^\star \ge \frac{\mu}{2}\|y - y^\star\|_2^2$, then $E_\theta(x, \widehat{y}) - E_x^\star < \mu \Delta_{\arg}^2/8$ guarantees the same valid decoding. This separates optimizer accuracy from decoder robustness.

Once the distinction above is explicit, standard convex analysis gives a useful training criterion: a first-order condition at a target relaxed solution certifies global optimality of that target for the relaxation.

Theorem 3.4 (First-order supervision is sufficient)**.** *Let $\mathcal{Y} \subset \mathbb{R}^d$ be convex, and let $E_\theta(x, \cdot)$ be convex and differentiable. If $y^\star \in \mathcal{Y}$ satisfies*

$$\langle \nabla_y E_\theta(x, y^\star),\, y - y^\star \rangle \ge 0 \qquad \forall y \in \mathcal{Y},$$

then

$$y^\star \in \arg\min_{y \in \mathcal{Y}} E_\theta(x, y).$$

The result is the first-order convexity inequality; its proof and the standard projected-gradient convergence rates are included in Appendix A. Auxiliary energy-shaping losses can still improve training empirically, as in Eq. (9), but they are not needed to turn a satisfied first-order condition into global optimality of the relaxed convex objective. Discrete correctness still requires the additional decoder compatibility captured by Lemma 3.2.

The assumed first-order condition therefore implies $E_\theta(x, y) \geq E_\theta(x, y^\star)$ for every feasible y. In contrast, for a non-convex energy, the same local condition would certify only stationarity or local optimality, not global optimality. This gives the main theoretical reason why convex compositional EBMs retain the benefits of energy-based inference while avoiding auxiliary landscape-shaping losses. The model still learns an energy over candidate solutions and performs inference by minimizing that energy, but convexity makes the relaxed inference problem globally well behaved.

Corollary 3.5 (Convergence of convex inference)**.** *Assume $E_\theta(x, \cdot)$ is convex and L-smooth on $\mathcal{Y}$, and let $y^\star \in \arg\min_{y\in\mathcal{Y}} E_\theta(x, y)$. Projected gradient descent with step size $1/L$,*

$$y_{t+1} = \Pi_{\mathcal{Y}}\left(y_t - \frac{1}{L}\nabla_y E_\theta(x, y_t)\right),$$

satisfies

$$E_\theta(x, y_T) - E_\theta(x, y^\star) \leq \frac{L\|y_0 - y^\star\|_2^2}{2T}.$$

If $E_\theta(x, \cdot)$ is additionally μ-strongly convex, then

$$\|y_T - y^\star\|_2^2 \leq \left(1 - \frac{\mu}{L}\right)^T \|y_0 - y^\star\|_2^2.$$

Together, Theorem 3.4 and Corollary 3.5 show that convex compositional energies change the role of training. Instead of learning an unconstrained landscape and then correcting it with auxiliary energy-shaping losses, the model learns inside a function class whose geometry makes local optimality certificates global. Additional regularization may still be useful empirically, but it is not required to rule out spurious minima in the relaxed composed energy.

4 Algorithm

In our algorithm, we use *Partially Input Convex Neural Network*, so that $f_\theta(y; c)$ is convex in the scope variable y for every context c. Instead of training $\nabla_y E_\theta$ to match Gaussian noise through a denoising-diffusion objective and sampling with PEM in $\mathbb{R}^n$, we define E_θ directly on the convex hull of the feasible set—the simplex/Birkhoff projection for N-Queens and the corresponding simplex relaxation for graph coloring—and minimize it with a projected first-order solver. The convex factor energy has no nonglobal local minima in $\mathcal{Y}$, so explicit particle resampling is not needed to escape local basins. We instantiate f_θ as a PICNN, which interleaves a non-convex context pathway with a convex scope pathway:

$$\begin{aligned} u_{\ell+1} &= \sigma\big(\widetilde{W}_\ell^u u_\ell + \widetilde{b}_\ell^u\big), \\ z_{\ell+1} &= \sigma\Big(W_\ell^z\big(z_\ell \odot \mathrm{softplus}(W_\ell^{zu} u_\ell)\big) + W_\ell^y\big(y \odot W_\ell^{yu} u_\ell\big) + W_\ell^u u_\ell + b_\ell\Big). \end{aligned} \tag{6}$$

Here σ is a non-decreasing convex activation, W_ℓ^z has non-negative entries, implemented as $W_\ell^z = \mathrm{softplus}(\widetilde{W}_\ell^z)$, and the gates $\mathrm{softplus}(W_\ell^{zu} u_\ell)$ keep the z-pathway non-negative. Therefore $f_\theta(\cdot; u)$ is convex in y by composition. The context encodes all factor-specific information.

Factor-level contrastive pretraining. Oarga and Du [17] shape the energy with a diffusion-MSE loss and a noise-corrupted contrastive loss. Because our factor scope is small and the convex hull is known, **we can drop the diffusion regression** and use a purely contrastive InfoNCE objective on clean samples:

$$\mathcal{L}_{\mathrm{NCE}}(\theta) = -\log \frac{\exp\big(-f_\theta(y^+; c)\big)}{\exp\big(-f_\theta(y^+; c)\big) + \sum_{j=1}^J \exp\big(-f_\theta(y_j^-; c)\big)} + \lambda_{\mathrm{e}} f_\theta(y^+; c). \tag{7}$$

Positives are vertices of the convex hull, such as one-hot row, column, or box vectors and zero or one-hot diagonal vectors. Negatives are sampled outside the hull using factor-specific proposals:

under-satisfied, overloaded, two-peak, and renormalized mixtures for N-Queens. This plays the role of the contrastive shaping loss in prior work, but acts at factor granularity before global composition.

Composed-energy refinement through an unrolled solver. Oarga and Du [17] refine the composed score by regressing $\nabla_y \sum_k E_\theta^k$. We instead refine E_θ by backpropagating through a differentiable projected solver on the relaxation. Given ground-truth solutions $\{x_i^\star\}$, we run T projected updates:

$$y_{t+1} = \Pi_{\mathcal{Y}}\big(y_t - \eta_t A_t \nabla_y E_\theta(x, y_t)\big), \qquad t = 0, \dots, T-1, \tag{8}$$

where A_t is either the identity, giving PGD, or the Adam preconditioner, giving projected Adam. The step size η_t follows a cosine schedule from η_0 to η_T. We add a small uniform tie-breaking field $\xi \sim \mathcal{U}([-1,1])^n$ scaled by λ_{tb} to break permutation symmetry between equal-energy minima. This affine perturbation preserves convexity and selects a representative from a degenerate relaxed minimizer set; it does not restore a multimodal energy landscape or guarantee discrete validity. With S random feasible warm-starts, we score each candidate under `no_grad`, pick the lowest-energy candidate, and backpropagate only through the solver trajectory from that initialization. The refinement loss combines regression, an energy-margin term, and hard-negative mining over the solver trajectory:

$$\begin{aligned}\mathcal{L}_{\text{board}} = \alpha\|\hat{x} - x^\star\|_2^2 + \beta\,\text{softplus}\big(E_\theta(x^\star) - E_\theta(\hat{x}) + \rho\big) \\ + \gamma\,\frac{1}{|\mathcal{H}|}\sum_{\tilde{x}\in\mathcal{H}} \text{softplus}\big(E_\theta(x^\star) - E_\theta(\tilde{x}) + \rho_{\text{h}}\big).\end{aligned} \tag{9}$$

Here $\mathcal{H}$ contains the last H solver iterates, the warm-start candidates, and a random convex mixture between the warm-start and the target. The hard-negative term plays the role of a board-level contrastive loss and prevents E_θ from collapsing onto trivial solutions of the regression term.

Inference. At test time, we run (8) from S random feasible starts, score each final point by E_θ, and select the lowest-energy candidate. The task-specific decoder is row-wise argmax for N-Queens, blockwise argmax for graph coloring, and thresholding for 3-SAT. $S = 1$ is the genuine single-trajectory convex-inference mode. Multiple starts are an optional finite-budget device: a convex objective can have a non-unique or fractional minimizer set, and truncated trajectories can approach near-equal-energy relaxed points that decode differently. Starts therefore improve relaxed-to-discrete recovery; they do not serve to escape nonglobal local minima. For comparison, *Parallel Energy Minimization* (PEM) [17, Alg. 1] can be used by replacing (8) with a P-particle ensemble that resamples using $w_i^{(t)} \propto \exp\big(-E_\theta(y_i^{(t)})\big)$ and re-injects scheduled Gaussian noise. PEM is the only inference mode in which our solver becomes equivalent to the sampler of [17].

5 Related Work

Reasoning as energy minimization. Casting reasoning as the search for a low-energy configuration of a learned scalar field has a long history in structured prediction [12] and has recently been revived as a paradigm for combinatorial problem solving. Most directly related to us, [17] compose subproblem energies at inference time and introduce Parallel Energy Minimization (PEM), a sequential-Monte-Carlo-style sampler that maintains a population of particles to escape local minima of the composed landscape. Our work shares the compositional construction but removes the source of those local minima at the parameterization level: by enforcing convexity in the decision variable per factor, the composed energy stays convex regardless of how many factors are stacked.

Neural combinatorial optimization. Graph Neural Networks are the dominant tool for learning-based combinatorial optimization, including direct supervised solvers for TSP and MIS [8], reinforcement-learning approaches [4], GFlowNets [22], discrete diffusion solvers [14, 20], and dedicated neural SAT solvers [15, 19]. Compositional EBMs side-step this by re-using a small per-factor model across arbitrarily many constraints. We keep this advantage while making inference deterministic and convex.

Iterative computation for reasoning. Recurrent and program-style architectures iteratively refine partial solutions [7, 9, 18], and neural SAT solvers refine truth assignments via belief propagation [15] or message passing [19]. Our projected solver in (8) is itself an iterative refinement procedure, but the iteration is performed on a continuous relaxation under a learned convex objective rather than on the discrete state space, and the number of iterations can be increased at test time to handle larger instances without retraining.

6 Experiments

We replicate a small compositional benchmark using convex relaxations to validate the optimization stability. We train a per-constraint input-convex energy on subproblem data only, compose the full energy at test time, and run convex inference. We specifically benchmark against the original framework's N-Queens formulation, which decomposes the board into row, column, and diagonal energy functions trained on a single eight-queens instance, Graph Coloring formulation, which decomposes graph node-wise and edge-wise and the 3-SAT formulation, which composes models trained to generate valid assignments for single clauses generated in the hard phase transition region.

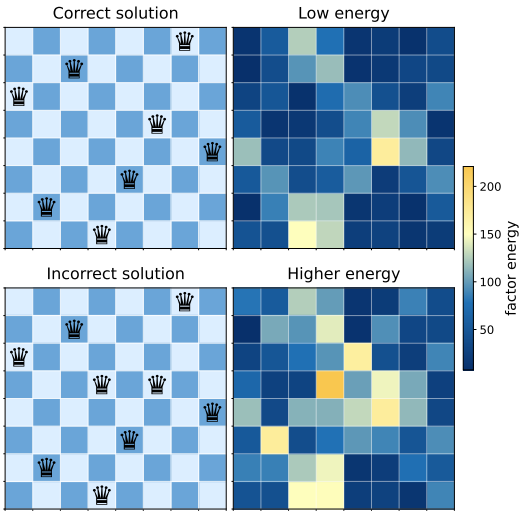


Figure 2: Energy map for 8-Queens. A correct solution is assigned low energy, while adding an extra queen produces higher local energy.

We study the tradeoff between convexity strength and expressivity by moving from fully convex networks to partially convex hybrids. We hypothesize our convex formulation will match the baseline's near-perfect correct instance rates on the logical reasoning tasks while fundamentally addressing its limitations in achieving fully optimal solutions without conflicting edges in the highly constrained graph coloring environments by eliminating the local minima that plague particle-based samplers.

We address four questions: **(Q1)** does enforcing convexity in the scope variable improve over the unconstrained MLP score network? **(Q2)** does projecting onto the exact convex hull of the feasible set replace the need for a long diffusion chain? **(Q3)** how do projected (multi-start) Adam and PEM compare as inference samplers on the *same* learned energy? **(Q4)** does compositional generalisation to larger instances carry over to the convex factor energy?

6.1 Tasks and protocols

N-Queens. We train on a single 8×8 instance, decompose it row-wise, column-wise and diagonal-wise, and re-use f_θ across rows, columns and (anti-)diagonals at test time. We evaluate by sampling 100 boards, decoding greedily, and reporting: number of correct instances (out of 100), average number of queens placed, and average row/column/diagonal conflicts. For generalisation we additionally report results on $N \in \{4, 5, 6, 7, 9, 10\}$.

Graph coloring. We train on a single edge-level subtask: generating a valid coloring for a pair of adjacent nodes given the available colors. The full graph energy is composed edge-wise. We evaluate on the COLOR benchmark and random Erdos–Renyi, Holme–Kim, random regular expander, Paley, and complete graphs, covering both 20–40 and 80–100 nodes. We report both the number of conflicting edges and the exact valid-coloring rate after blockwise argmax.

3-SAT. Each local factor represents one clause. During factor pretraining, complementation is used only as a local parameterization of the single falsifying assignment. At composition and formula-level evaluation, the original literal polarities are retained consistently; the method does not search for one assignment that simultaneously violates every original clause and then negate it. We train on formulas with 10–20 variables near the phase transition, using $4.258n$ clauses, and test on 100 formulas with 20 variables and 91 clauses. Thresholding produces the discrete assignment.

6.2 Baselines

For N-Queens we reproduce the comparison table of [17] and add our PICNN factor energy. The non-EBM baselines (GFlowNets [22], DIFUSCO [20], Fast T2T [14]) take the N-Queens problem encoded as a Maximum Independent Set. The EBM baseline of [17] (**EBM-Diff**, P=1024, PEM) shares the compositional construction with us, but uses an unconstrained score network trained



Figure 3: Optimized samples across projected Adam timesteps for 8-Queens. Intermediate columns show relaxed boards before decoding, and the final column shows the greedy-decoded board. Increasing the number of starts improves finite-budget relaxed-to-discrete recovery.

Table 1: 8-Queens evaluation (100 sampled boards). Baselines are reproduced from [17, Tab. 1]; CCEM uses $T = 50$ and $S = 64$.

Model	Type	Correct ↑	Queens placed ↑
LWD	RL+S	22	$7.10_{\pm 0.57}$
GFlowNets	UL+S	14	$6.93_{\pm 0.59}$
DIFUSCO (T=50)	SL+S	17	$6.94_{\pm 0.65}$
Fast T2T (T_S=1, T_G=1)	SL+S	21	$6.82_{\pm 0.88}$
Fast T2T (T_S=5, T_G=5, GS)	SL+GS	41	$7.38_{\pm 0.54}$
EBM-Diff (P=1024, PEM)	SL+PEM	97	$7.97_{\pm 0.17}$
CCEM (ours)	SL+Adam	**100**	**8.00**$_{\pm 0.00}$

with combination of diffusion and contrastive losses. For Graph Coloring we compare against GNN-GCP [13], canonical Graph Neural Networks (GCN [10] and GAT [21]), RL guided by Neural Algorithmic Reasoners (XLVIN [6]) and EBM baseline [17] (P=128).

6.3 Implementation details

Whereas [17] runs E_θ on $\mathbb{R}^n$ and relies on the diffusion process to deliver near-binary samples, we optimise on the smallest tight convex set that contains every feasible configuration, projected by alternating row/column simplex sweeps $\mathcal{Y} = \mathcal{B}_N = \{X \in [0,1]^{N\times N} \mid \sum_i X_{ij} = \sum_j X_{ij} = 1\}$. Diagonal factors live on the relaxed simplex $\{y \in [0,1]^d \mid \sum_i y_i \le 1\}$. We denote the projector $\Pi_{\mathcal{Y}}$. Crucially, $\Pi_{\mathcal{Y}}$ is differentiable almost everywhere, which lets gradients flow through the solver.

The factor PICNN has 3 hidden layers of width 256 with softplus (N-Queens, graph coloring) or ReLU (3-SAT). Phase 1 is trained for 1000 epochs of AdamW at learning rate 10^{-3}, weight decay 10^{-4}, and J negatives per positive ($J = 4$ for N-Queens, 7 for 3-SAT, and 5 for graph coloring). Phase 2 is trained for 300 epochs with cosine step decay $0.1 \to 0.02$ (train) and $0.05 \to 0.005$ (evaluation), $S = 4$ starts in training, and up to $S = 64$ in evaluation. We use $T = 50$ evaluation steps for N-Queens and $T = 140$ for graph coloring and 3-SAT. The hard-negative count is $H = 12$, the loss weights are $(\alpha, \beta, \gamma) = (1, 0.25, 0.25)$, the margins are $\rho = \rho_{\mathrm{h}} = 0.1$, and $\lambda_{\mathrm{tb}} = 10^{-4}$. We use simplex projection and a single GPU. Appendix C provides hardware-specific timing and energy-evaluation counts.

Table 2: Graph Coloring evaluation (100 sampled tasks).

Distribution	GCN	GAT	XLVIN	GNN-GCP	EBM (P=128)	CCEM (Ours)
Erdos Renyi	46.80 ± 20.47	34.00 ± 11.55	25.00 ± 7.81	15.20 ± 4.32	8.60 ± 4.82	**2.20 ± 2.14**
Erdos Renyi 2	151.60 ± 12.09	130.20 ± 11.47	93.80 ± 31.12	53.80 ± 8.34	29.20 ± 8.05	**4.60 ± 2.94**
Holme Kim	74.00 ± 14.74	51.20 ± 10.03	29.00 ± 7.75	13.20 ± 7.46	10.60 ± 2.70	**5.00 ± 3.41**
Holme Kim 2	408.00 ± 26.40	253.20 ± 50.71	182.60 ± 24.73	55.20 ± 12.63	59.00 ± 3.74	**14.00 ± 1.26**
Regular Expander	87.60 ± 22.58	58.60 ± 12.44	29.00 ± 7.75	15.40 ± 6.65	11.00 ± 4.89	**4.20 ± 3.43**
Regular Expander 2	144.80 ± 6.90	118.80 ± 12.59	112.60 ± 10.97	141.60 ± 69.47	37.20 ± 4.71	**14.40 ± 3.56**
Paley	285.00 ± 117.70	239.20 ± 159.43	151.80 ± 92.13	91.20 ± 63.14	34.80 ± 20.27	**1.80 ± 3.60**
Complete	46.00 ± 15.04	46.00 ± 15.04	34.80 ± 16.42	30.00 ± 2.54	3.40 ± 1.14	**0.00 ± 0.00**

6.4 Results

Convexity vs. MLP score. Table 1, Table 2, and Table 11 compare CCEM with compositional and task-specific baselines. CCEM solves all sampled 8-Queens boards at its largest budget and reduces graph-coloring conflicts across every tested distribution. The revised 3-SAT formulation improves clause satisfaction from the submitted CCEM value 0.9665 to 0.9959 at $S = 64$, substantially narrowing the gap to the EBM+PEM value 0.9985.

Figure 4: Phase-2 loss ablation on 8-Queens (100 sampled boards, projected Adam, S=64).

MSE (α)	Rank (β)	Hard (γ)	Correct
✓			100
	✓	✓	100
✓	✓		100
✓		✓	100
✓	✓	✓	100

Tight relaxation replaces the diffusion chain. [17] need T=100 diffusion steps and a P=1024 particle ensemble to lift Gaussian noise to the binary solution manifold. With the Birkhoff projector $\Pi_{\mathcal{B}_N}$ applied at every solver step, the relaxation is already tight, and we match their reported accuracy with T=50 projected-Adam steps and S=64 starts (Table 1). The EBM-Diff row is the result reported by Oarga and Du [17]; our reproduction using the authors' public code solved 75/100 boards.

Compositional generalisation to larger N. We train once on N=8, evaluate on $N \in \{7, 8, 9, 10\}$ by re-summing f_θ over the rows, columns and diagonals of the larger board, and sweep the inference budget. Because f_θ conditions on the scope mask and active length, the same network generalises to all four sizes without retraining. Increasing S for projected Adam and P (particles) for PEM both monotonically improve correct-instance counts; PEM with P=128 matches projected Adam with S=32 on N=10. For CCEM, this gain reflects finite-step optimization and decoder selection among near-equal relaxed candidates, not escape from nonglobal local minima.

The conflict-count gains translate into higher exact discrete success rates: CCEM improves over EBM+PEM on six of the eight graph families and ties on the two hardest zero-success cases (Table 10). In particular, it solves 29/100 Erdos–Renyi, 59/100 Paley, and 100/100 complete-graph instances.

Loss ablation. We ablate the Phase 2 loss components and observe that all variants remain surprisingly strong. This suggests that the main gain comes not from a single loss term, but from the compositional energy structure itself: row, column, and diagonal factors already impose a strong inductive bias. With S=64 starts, all variants solve all sampled instances, indicating that the inference procedure can reliably recover good solutions once enough starts are used.

Sharpness diagnostics. For every relaxed heatmap we report row entropy, the top-2 row gap, the fraction of rows with $\max_j X_{ij} > 0.9$, and row/col sum-to-one errors after projection. The PICNN energy concentrates on a single column per row (gap $\approx$ 1, entropy $\approx$ 0, sum errors $< 10^{-3}$), confirming that the convex factor energy yields a *decisive* basin on the Birkhoff polytope rather than a smeared heatmap that requires temperature-style decoding.

Relaxation-to-decoding diagnostics. To attribute failures, we separate solver accuracy, relaxation looseness, and decoding. For a final iterate y_T, we use the projected residual

$$r_{\text{opt}}(y_T) = \eta^{-1} \| y_T - \Pi_{\mathcal{Y}_x}(y_T - \eta \nabla E_x(y_T)) \|_2,$$

fractionality $\phi(y_T) = 1 - m^{-1} \sum_{i=1}^{m} \max_j y_{T,ij}$, the top-two gap, and the pre/post-decoding energy increase $\Delta E_{\text{dec}} = E_x(D_x(y_T)) - E_x(y_T)$. We pair these with relaxed and exact constraint violations. A large residual indicates finite-optimization error; a small residual with a large relaxed-to-discrete

gap indicates a loose relaxation; and a small residual and relaxation gap but large decoder regret or a small top-two gap indicates decoding instability. On small instances, we compare $E^\star_{\mathcal{Y}} = \min_{\mathcal{Y}_x} E_x$, $E^\star_{\mathcal{Z}} = \min_{\mathcal{Z}_x} E_x$, and $E^\star_{\mathcal{V}} = \min_{\mathcal{V}_x} E_x$ to distinguish relaxation, model, and decoder gaps.

7 Limitations and Broader Impacts

Compositional energy minimization adds constraints by summing energies, and input-convex architectures keep the resulting decision landscape free of new nonconvex traps. Prior work shows strong generalization, including SAT instances with hundreds of clauses after training only on single clauses, and unseen Paley or random regular expander graphs. In our view, composition turns generalization into a convex program: harder tasks mainly require more solver precision, not escape from pathological geometry. The tradeoff is that convexity can be too restrictive, producing flat regions, fractional relaxations, or poor fits to multimodal discrete solution spaces.

8 Conclusion

We introduced Convex Compositional Energy Minimization (CCEM), a framework for compositional reasoning that replaces unconstrained factor energies with input-convex factor models and performs inference over tight convex relaxations. By preserving convexity under nonnegative factor composition, CCEM removes a central source of spurious local minima and enables deterministic projected first-order optimization in place of long diffusion or particle-based sampling procedures. Our theory guarantees global optimization of the relaxed objective and gives additional decoder-aware conditions for discrete recovery, without claiming that convexity alone solves the combinatorial problem. Experiments on three structured reasoning tasks show transfer from small training instances to larger problems, improved graph-coloring validity, revised competitive 3-SAT performance, and substantially lower inference budgets than particle-based composition. These results suggest that controlling factor-level geometry is a promising direction for efficient neural reasoning when the relaxation and decoder are well aligned.

References

[1] Brandon Amos, Lei Xu, and J Zico Kolter. Input convex neural networks. In *International conference on machine learning*, pages 146–155. PMLR, 2017.

[2] Arip Asadulaev. Connecting convex energy-based inference and optimal transport for domain adaptation. In *Energy Based Models Workshop-ICLR 2021*.

[3] Arip Asadulaev, Vitaly Shutov, Alexander Korotin, Alexander Panfilov, Vladislava Kontsevaya, and Andrey Filchenkov. A minimalist approach for domain adaptation with optimal transport. In Sarath Chandar, Razvan Pascanu, Hanie Sedghi, and Doina Precup, editors, *Proceedings of The 2nd Conference on Lifelong Learning Agents*, volume 232 of *Proceedings of Machine Learning Research*, pages 1009–1024. PMLR, 22–25 Aug 2023. URL https://proceedings.mlr.press/v232/asadulaev23a.html.

[4] Irwan Bello, Hieu Pham, Quoc V Le, Mohammad Norouzi, and Samy Bengio. Neural combinatorial optimization with reinforcement learning. *arXiv preprint arXiv:1611.09940*, 2016.

[5] Yize Chen, Yuanyuan Shi, and Baosen Zhang. Optimal control via neural networks: A convex approach. *arXiv preprint arXiv:1805.11835*, 2018.

[6] Andreea Deac, Petar Veličković, Ognjen Milinković, Pierre-Luc Bacon, Jian Tang, and Mladen Nikolić. Xlvin: executed latent value iteration nets. *arXiv preprint arXiv:2010.13146*, 2020.

[7] Alex Graves, Greg Wayne, and Ivo Danihelka. Neural turing machines. *arXiv preprint arXiv:1410.5401*, 2014.

[8] Chaitanya K Joshi, Thomas Laurent, and Xavier Bresson. An efficient graph convolutional network technique for the travelling salesman problem. *arXiv preprint arXiv:1906.01227*, 2019.

[9] Łukasz Kaiser and Ilya Sutskever. Neural gpus learn algorithms. *arXiv preprint arXiv:1511.08228*, 2015.

[10] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[11] Alexander Korotin, Lingxiao Li, Aude Genevay, Justin M Solomon, Alexander Filippov, and Evgeny Burnaev. Do neural optimal transport solvers work? a continuous wasserstein-2 benchmark. *Advances in neural information processing systems*, 34:14593–14605, 2021.

[12] Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[13] Henrique Lemos, Marcelo Prates, Pedro Avelar, and Luis Lamb. Graph colouring meets deep learning: Effective graph neural network models for combinatorial problems. In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 879–885. IEEE, 2019.

[14] Yang Li, Jinpei Guo, Runzhong Wang, Hongyuan Zha, and Junchi Yan. Fast t2t: Optimization consistency speeds up diffusion-based training-to-testing solving for combinatorial optimization. *Advances in Neural Information Processing Systems*, 37:30179–30206, 2024.

[15] Zhaoyu Li and Xujie Si. Nsnet: A general neural probabilistic framework for satisfiability problems. *Advances in Neural Information Processing Systems*, 35:25573–25585, 2022.

[16] Ashok Makkuva, Amirhossein Taghvaei, Sewoong Oh, and Jason Lee. Optimal transport mapping via input convex neural networks. In *International Conference on Machine Learning*, pages 6672–6681. PMLR, 2020.

[17] Alexandru Oarga and Yilun Du. Generalizable reasoning through compositional energy minimization. *arXiv preprint arXiv:2510.20607*, 2025.

[18] Avi Schwarzschild, Eitan Borgnia, Arjun Gupta, Furong Huang, Uzi Vishkin, Micah Goldblum, and Tom Goldstein. Can you learn an algorithm? generalizing from easy to hard problems with recurrent networks. *Advances in Neural Information Processing Systems*, 34:6695–6706, 2021.

[19] Daniel Selsam, Matthew Lamm, Benedikt Bünz, Percy Liang, Leonardo de Moura, and David L Dill. Learning a sat solver from single-bit supervision. *arXiv preprint arXiv:1802.03685*, 2018.

[20] Zhiqing Sun and Yiming Yang. Difusco: Graph-based diffusion solvers for combinatorial optimization. *Advances in neural information processing systems*, 36:3706–3731, 2023.

[21] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017.

[22] Dinghuai Zhang, Nikolay Malkin, Zhen Liu, Alexandra Volokhova, Aaron Courville, and Yoshua Bengio. Generative flow networks for discrete probabilistic modeling. In *International Conference on Machine Learning*, pages 26412–26428. PMLR, 2022.

A Proofs

Proposition A.1 (Non-convex factor composition admits spurious stable minima)**.** *Let $\mathcal{Y} \subset \mathbb{R}^d$ be a convex relaxation of a discrete reasoning problem, and let a compositional energy be*

$$E(y) = \sum_{k=1}^{K} f_k(y_{S_k}),$$

where each factor $f_k : \mathbb{R}^{|S_k|} \to \mathbb{R}$ is twice continuously differentiable and $S_k \subseteq \{1, \dots, d\}$ denotes the scope of factor k. Assume the factors are not constrained to be convex in their scope variables. Then there exist smooth factor energies $\{f_k\}_{k=1}^K$ and a feasible global solution $y^\star \in \mathcal{Y}$ such that:

1. *$y^\star$ is a global minimizer of the composed energy:*
$$E(y^\star) = \min_{y \in \mathcal{Y}} E(y).$$
2. *The composed energy also contains a point $\bar{y} \in \mathcal{Y}$, with $\bar{y} \neq y^\star$, satisfying*
$$\nabla E(\bar{y}) = 0, \qquad \nabla^2 E(\bar{y}) \succ 0.$$
3. *Therefore, $\bar{y}$ is a strict local minimum of E, but not a global minimizer:*
$$E(\bar{y}) > E(y^\star).$$

Moreover, for gradient descent

$$y_{t+1} = y_t - \eta \nabla E(y_t),$$

there exists a stepsize $\eta > 0$ and an open neighborhood $U \subset \mathcal{Y}$ of $\bar{y}$ such that every initialization $y_0 \in U$ converges to $\bar{y}$. Hence failure from such initializations is caused by the geometry of the composed energy landscape, rather than by the absence of a sufficiently expressive local inference procedure.

Proof. It suffices to give a one-dimensional construction. Let

$$\mathcal{Y} = [-2, 2],$$

and define two factor energies

$$f_1(y) = (y^2 - 1)^2, \qquad f_2(y) = \alpha(y - 1)^2,$$

where $\alpha > 0$ is a small constant. The composed energy is

$$E_\alpha(y) = f_1(y) + f_2(y) = (y^2 - 1)^2 + \alpha(y - 1)^2.$$

The point

$$y^\star = 1$$

is a global minimizer because

$$E_\alpha(1) = 0,$$

and both terms in E_α are nonnegative.

Now consider the behavior near $y = -1$. When $\alpha = 0$, we have

$$E_0(y) = (y^2 - 1)^2,$$

and $y = -1$ is a strict local minimum because

$$E_0'(-1) = 0, \qquad E_0''(-1) = 8 > 0.$$

Since E_α is a smooth perturbation of E_0, the implicit function theorem implies that for all sufficiently small $\alpha > 0$, there exists a critical point $\bar{y}_\alpha$ near -1 such that

$$E_\alpha'(\bar{y}_\alpha) = 0.$$

Furthermore, by continuity of the second derivative,

$$E''_\alpha(\bar{y}_\alpha) > 0$$

for all sufficiently small $\alpha > 0$. Therefore, $\bar{y}_\alpha$ is a strict local minimum of E_α.

However,

$$\bar{y}_\alpha \neq 1.$$

Since $E_\alpha(1) = 0$ is the global minimum and $E_\alpha(y) > 0$ for all $y \neq 1$, we have

$$E_\alpha(\bar{y}_\alpha) > E_\alpha(1).$$

Thus $\bar{y}_\alpha$ is a strict spurious local minimum of the composed energy.

Finally, because

$$E''_\alpha(\bar{y}_\alpha) > 0,$$

there exists an open neighborhood U of $\bar{y}_\alpha$ on which E_α is strongly convex and has Lipschitz-continuous gradient. For a sufficiently small stepsize $\eta > 0$, gradient descent initialized in U remains in U and converges to the unique local minimizer $\bar{y}_\alpha$. This proves the claim. □

A.1 Decoder-aware relaxed-to-discrete guarantees

Proof of Lemma 3.2. The set $\mathcal{B}_x$ is closed and is therefore compact as a subset of compact $\mathcal{Y}_x$. Continuity implies that the minimum in Eq. (4) is attained. It equals zero exactly when a global minimizer of $E_\theta(x, \cdot)$ belongs to $\mathcal{B}_x$, proving the first claim. If $\widehat{y}$ decoded invalidly, then $\widehat{y} \in \mathcal{Y}_x \setminus \mathcal{A}_x \subseteq \mathcal{B}_x$ and therefore

$$E_\theta(x, \widehat{y}) - E^\star_x \geq \Gamma^D_x(E_\theta),$$

which contradicts the assumed strict inequality. □

Proof of Theorem 3.3. Let $G^\star = \min_{\mathcal{Y}_x} G_x$ and $E^\star = \min_{\mathcal{Y}_x} E_\theta(x, \cdot)$. Uniform approximation gives $E^\star \leq G^\star + \Delta$. For every $y \in \mathcal{B}_x$,

$$E_\theta(x, y) \geq G_x(y) - \Delta \geq G^\star + \gamma_{\text{dec}} - \Delta \geq E^\star + \gamma_{\text{dec}} - 2\Delta.$$

If $\widehat{y} \in \mathcal{B}_x$, the left-hand side is at least $\gamma_{\text{dec}} - 2\Delta$ above $E^\star$, whereas solver accuracy bounds it by δ. The condition $2\Delta + \delta < \gamma_{\text{dec}}$ rules this out. Hence $\widehat{y} \notin \mathcal{B}_x$ and its decoding is valid. For a compositional approximation, the triangle inequality gives $\Delta \leq \sum_k w_k \varepsilon_k$. □

Theorem 3.3 is a transfer statement, not an existence theorem. For example, if a valid assignment z_x were already known, $G_x(y) = \|y - z_x\|_2^2$ would have a unique valid minimizer and a positive decoder margin. Constructing this energy requires knowing the answer, however; similarly, describing the convex hull of all valid assignments can encode the entire combinatorial problem.

Proposition A.2 (Complexity boundary for uniform convex certificates)**.** *Consider a family of feasible instances of an NP-hard search problem. Suppose a polynomial-time procedure constructs from every instance x a tractable convex relaxation $\mathcal{Y}_x$, a convex energy G_x, and a polynomial-time decoder D_x with decoder-aware margin at least $1/p(|x|)$ for a polynomial p. Suppose also that the convex problem can be solved in polynomial time to error less than $1/p(|x|)$. Then the associated NP-hard search problem can be solved in polynomial time.*

Proof. Construct $(\mathcal{Y}_x, G_x, D_x)$, optimize G_x below its guaranteed margin, and apply D_x. Lemma 3.2 makes the decoded output a valid solution. Thus CCEM makes no claim that uniformly constructible, inverse-polynomial-margin certificates exist for all instances of 3-SAT or graph coloring; its guarantees are conditional and instance dependent. □

For product-simplex relaxations, a useful sufficient condition can be checked at a valid discrete vertex.

Theorem A.3 (Strict margin implies exact argmax decoding)**.** *Let $\mathcal{Y}_x = \prod_i \Delta_q$, let $z^\star_i = e_{c_i}$ be a valid one-hot solution, and let $E_\theta(x, \cdot)$ be convex and differentiable. Define*

$$m = \min_i \min_{j \neq c_i} \left[\frac{\partial E_\theta(x, z^\star)}{\partial y_{ij}} - \frac{\partial E_\theta(x, z^\star)}{\partial y_{i,c_i}} \right].$$

If $m > 0$, then $z^\star$ is the unique relaxed minimizer. Moreover, every y whose blockwise argmax differs from $z^\star$ satisfies

$$E_\theta(x, y) - E_\theta(x, z^\star) \geq m/2.$$

Consequently, any $\widehat{y}$ with optimization error below $m/2$ decodes exactly to $z^\star$.

Proof. Let $g = \nabla_y E_\theta(x, z^\star)$. By convexity,

$$E_\theta(x, y) - E_\theta(x, z^\star) \geq \sum_i \sum_{j \neq c_i} y_{ij}(g_{ij} - g_{i,c_i}) \geq m \sum_i (1 - y_{i,c_i}).$$

The bound is positive for every $y \neq z^\star$, so the minimizer is unique. If argmax differs in block i, then $y_{i,c_i} \leq 1/2$ and the bound is at least $m/2$. □

The unknown optimization error can be upper-bounded by the Frank–Wolfe gap

$$g_{\mathrm{FW}}(\widehat{y}) = \max_{s \in \mathcal{Y}_x} \langle \nabla E_\theta(x, \widehat{y}), \widehat{y} - s \rangle.$$

Therefore $g_{\mathrm{FW}}(\widehat{y}) < m/2$ is a directly checkable certificate of exact decoding under Theorem A.3.

The trained-model margin in Eq. (4) can also be bounded directly. For the argmax and threshold decoders used in our tasks, $\mathcal{B}_x$ is a finite union of polyhedral violation regions $\mathcal{C}_{x,q}$. A witness q is a same-color edge for graph coloring, a conflicting queen pair for N-Queens, or a clause whose decoded literals are all false for 3-SAT. Each witness is imposed by linear inequalities. Hence

$$E^\star_{x,\mathrm{bad}} = \min_q \min_{y \in \mathcal{C}_{x,q}} E_\theta(x, y)$$

is a collection of convex problems. For any candidate $\widetilde{y}_q \in \mathcal{C}_{x,q}$, convexity gives the lower bound

$$L_q = E_\theta(x, \widetilde{y}_q) + \min_{z \in \mathcal{C}_{x,q}} \langle \nabla E_\theta(x, \widetilde{y}_q), z - \widetilde{y}_q \rangle \leq \min_{y \in \mathcal{C}_{x,q}} E_\theta(x, y).$$

For any feasible relaxed point $\widetilde{y} \in \mathcal{Y}_x$, the condition $\min_q L_q - E_\theta(x, \widetilde{y}) > 0$ certifies a positive decoder-aware margin. The inner problem defining L_q is linear over the original relaxation plus decoder inequalities; a valid subgradient gives the same bound for a nondifferentiable PICNN.

Theorem A.4 (First-order supervision suffices for convex energy minima)**.** *Let $\mathcal{Y} \subset \mathbb{R}^d$ be a nonempty compact convex relaxation of the solution space, and let*

$$E_\theta(x, y)$$

be an energy function that is convex and differentiable in y for every fixed instance x. Let $y^\star \in \mathcal{Y}$ be a correct solution for instance x.

Suppose that training enforces the first-order optimality condition

$$\langle \nabla_y E_\theta(x, y^\star),\, y - y^\star \rangle \geq 0 \qquad \textit{for all } y \in \mathcal{Y}.$$

Then $y^\star$ is a global minimizer of the energy:

$$y^\star \in \arg\min_{y \in \mathcal{Y}} E_\theta(x, y).$$

Thus, for convex energy models, an additional contrastive landscape-shaping loss is not needed to make the correct solution a global minimizer. It is sufficient to enforce the convex first-order optimality condition at the correct relaxed solution. Discrete validity additionally requires the decoder-aware condition in Lemma 3.2.

Proof. Fix an instance x. Since $E_\theta(x, \cdot)$ is convex and differentiable on $\mathcal{Y}$, the first-order convexity inequality gives, for every $y \in \mathcal{Y}$,

$$E_\theta(x, y) \geq E_\theta(x, y^\star) + \langle \nabla_y E_\theta(x, y^\star),\, y - y^\star \rangle .$$

By the assumed first-order optimality condition,

$$\langle \nabla_y E_\theta(x, y^\star),\, y - y^\star \rangle \geq 0 \qquad \text{for all } y \in \mathcal{Y}.$$

Therefore,

$$E_\theta(x, y) \geq E_\theta(x, y^\star) \qquad \text{for all } y \in \mathcal{Y}.$$

Hence $y^\star$ is a global minimizer:

$$y^\star \in \arg\min_{y \in \mathcal{Y}} E_\theta(x, y).$$

□

Table 3: N-Queens valid decoded boards (%, mean ± standard deviation) across inference starts. Each entry summarizes evaluations on 100 boards.

N	$S=1$	$S=2$	$S=4$	$S=8$	$S=16$	$S=32$	$S=64$
4	28.00 ± 3.29	56.00 ± 4.02	79.00 ± 3.68	98.00 ± 1.99	100.00 ± 0.58	100.00 ± 0.00	100.00 ± 0.00
5	47.00 ± 5.69	79.00 ± 4.18	99.00 ± 2.20	100.00 ± 0.63	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
6	4.00 ± 1.79	9.00 ± 3.03	12.00 ± 4.31	30.00 ± 4.72	48.00 ± 4.47	79.00 ± 4.23	96.00 ± 1.97
7	14.00 ± 3.74	31.00 ± 4.80	49.00 ± 5.43	80.00 ± 4.29	98.00 ± 1.97	100.00 ± 0.45	100.00 ± 0.00
8	13.00 ± 2.38	19.00 ± 3.11	31.00 ± 4.70	51.00 ± 5.22	81.00 ± 3.87	98.00 ± 2.09	100.00 ± 0.39
9	6.00 ± 2.31	11.00 ± 3.44	23.00 ± 4.45	48.00 ± 3.94	73.00 ± 4.47	95.00 ± 2.65	100.00 ± 0.79
10	4.00 ± 1.82	7.00 ± 2.15	12.00 ± 3.12	20.00 ± 3.85	40.00 ± 4.88	58.00 ± 5.46	82.00 ± 3.62

Corollary A.5 (Approximate first-order supervision gives approximate optimality)**.** *Under the assumptions of Theorem A.4, suppose the first-order condition is satisfied up to error $\varepsilon \geq 0$, meaning*

$$\langle \nabla_y E_\theta(x, y^\star),\, y - y^\star \rangle \geq -\varepsilon \qquad \text{for all } y \in \mathcal{Y}.$$

Then $y^\star$ is ε-globally optimal:

$$E_\theta(x, y^\star) \leq \min_{y \in \mathcal{Y}} E_\theta(x, y) + \varepsilon.$$

Proof. By convexity, for every $y \in \mathcal{Y}$,

$$E_\theta(x, y) \geq E_\theta(x, y^\star) + \langle \nabla_y E_\theta(x, y^\star),\, y - y^\star \rangle .$$

Using the approximate first-order condition,

$$E_\theta(x, y) \geq E_\theta(x, y^\star) - \varepsilon \qquad \text{for all } y \in \mathcal{Y}.$$

Equivalently,

$$E_\theta(x, y^\star) \leq E_\theta(x, y) + \varepsilon \qquad \text{for all } y \in \mathcal{Y}.$$

Taking the minimum over $y \in \mathcal{Y}$ gives

$$E_\theta(x, y^\star) \leq \min_{y \in \mathcal{Y}} E_\theta(x, y) + \varepsilon.$$

□

Corollary A.6 (Convergence of projected convex inference)**.** *Let $E_\theta(x, \cdot)$ be convex and L-smooth on a closed convex set $\mathcal{Y}$. Let $y^\star \in \arg\min_{y \in \mathcal{Y}} E_\theta(x, y)$, and run projected gradient descent with stepsize $\eta = 1/L$:*

$$y_{t+1} = \Pi_{\mathcal{Y}} \left(y_t - \frac{1}{L} \nabla_y E_\theta(x, y_t) \right).$$

Then after T iterations,

$$E_\theta(x, y_T) - E_\theta(x, y^\star) \leq \frac{L \|y_0 - y^\star\|_2^2}{2T}.$$

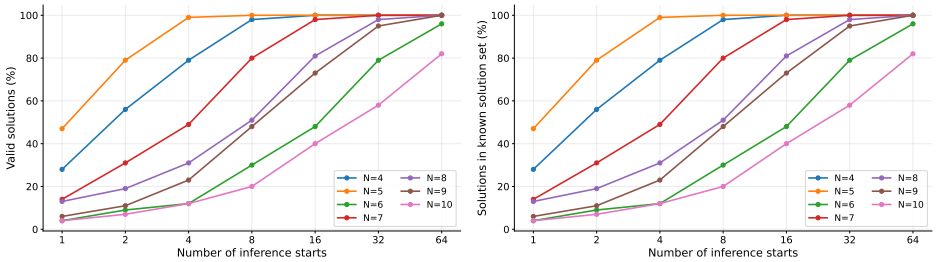
If, in addition, $E_\theta(x, \cdot)$ is μ-strongly convex on $\mathcal{Y}$, then projected gradient descent converges linearly:

$$\|y_T - y^\star\|_2^2 \leq \left(1 - \frac{\mu}{L}\right)^T \|y_0 - y^\star\|_2^2.$$

B Ablations

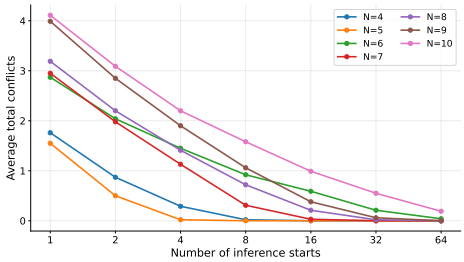
C Computational Details

All runs use a single GPU with no distributed training. The end-to-end method comparison below was measured on an NVIDIA RTX A6000; the loss-component ablation in Table 8 was measured separately on an RTX A4000. The main model is a three-layer PICNN with hidden width 256, followed by composed-energy refinement through an unrolled projected solver. Phase 1 uses 1000 AdamW epochs and Phase 2 uses 300 epochs, with $S = 4$ starts during refinement and up to $S = 64$ at evaluation.

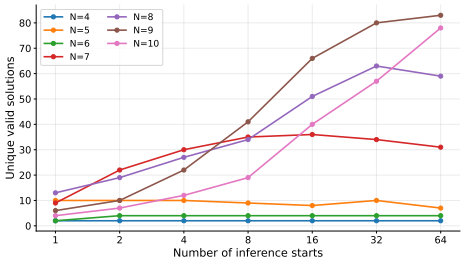


(a) Valid solutions.

(b) Known solution rate.

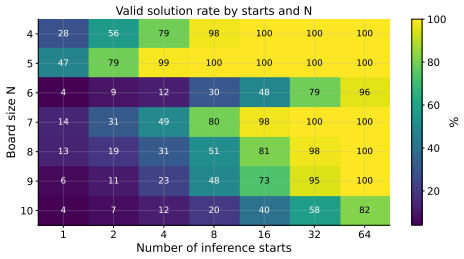


(c) Average conflicts.

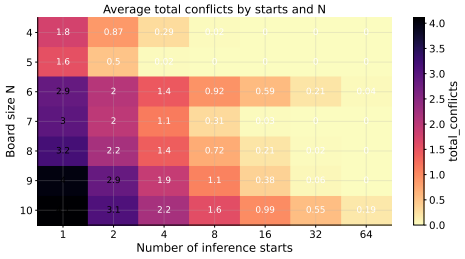


(d) Unique valid solutions.

Figure 5: Inference-start ablation for N-Queens. Increasing the number of independent starts improves the probability of finding a valid decoded board, reduces the average number of conflicts, and increases the diversity of valid solutions.



(a) Valid solution rate.



(b) Average conflicts.

Figure 6: Heatmap view of the inference-start ablation across board sizes. Larger boards require more starts to obtain comparable validity, while increasing the number of starts consistently reduces the average number of conflicts.

A CCEM trajectory performs one projected gradient update per step and one additional energy-only evaluation for final candidate selection. PEM also evaluates energy at every step for importance weighting and resampling. Thus a CCEM trajectory uses 51 energy forwards for N-Queens and 141 for graph coloring or 3-SAT, versus 201 for EBM+PEM.

The largest reported CCEM budget uses 64 independent trajectories, whereas EBM+PEM uses 1,024 particles for N-Queens and 3-SAT and 128 for graph coloring. CCEM therefore reduces both the work within a trajectory and the number of parallel trajectories used for the reported solution quality.

Table 8 reports wall-clock runtimes for the N-Queens Phase 2 loss ablation on the same hardware. The timings show that adding the ranking and hard-negative terms introduces only a small overhead relative to the regression-only objective. The MSE-only configuration required 1:53:35, corresponding to 6.82 seconds per iteration, while the full objective with MSE, ranking, and hard-negative terms required 1:55:47, corresponding to 6.95 seconds per iteration. Thus, the full board-level objective increases per-iteration time by approximately 1.9% compared with the MSE-only setting. The most expensive ablated configuration was the Rank+Hard objective without the MSE term, which took 2:00:07, or 7.21 seconds per iteration.

Table 4: N-Queens average decoded conflicts (mean $\pm$ standard deviation) across inference starts.

N	$S = 1$	$S = 2$	$S = 4$	$S = 8$	$S = 16$	$S = 32$	$S = 64$
4	1.80 ± 0.13	0.87 ± 0.09	0.29 ± 0.05	0.02 ± 0.02	0.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00
5	1.60 ± 0.18	0.50 ± 0.10	0.02 ± 0.05	0.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
6	2.90 ± 0.11	2.00 ± 0.11	1.40 ± 0.10	0.92 ± 0.08	0.59 ± 0.05	0.21 ± 0.04	0.04 ± 0.02
7	3.00 ± 0.19	2.00 ± 0.15	1.10 ± 0.13	0.31 ± 0.08	0.03 ± 0.03	0.00 ± 0.00	0.00 ± 0.00
8	3.20 ± 0.20	2.20 ± 0.15	1.40 ± 0.12	0.72 ± 0.10	0.21 ± 0.05	0.02 ± 0.02	0.00 ± 0.00
9	4.00 ± 0.17	2.90 ± 0.14	1.90 ± 0.13	1.10 ± 0.10	0.38 ± 0.08	0.06 ± 0.03	0.00 ± 0.01
10	4.10 ± 0.18	3.10 ± 0.13	2.20 ± 0.11	1.60 ± 0.10	0.99 ± 0.08	0.55 ± 0.07	0.19 ± 0.04

Table 5: End-to-end wall-clock comparison on one NVIDIA RTX A6000 (mean $\pm$ standard deviation).

Method	Task	Train (min)	Eval. (s)
EBM+PEM	N-Queens	258 ± 15	340 ± 25
EBM+PEM	3-SAT	619 ± 36	2000 ± 30
EBM+PEM	Graph coloring	155 ± 9	180 ± 15
CCEM	N-Queens	115 ± 14	60 ± 30
CCEM	3-SAT	61 ± 4.6	60 ± 5
CCEM	Graph coloring	15 ± 2.3	30 ± 3

The total wall-clock time for the four N-Queens ablation runs in Table 8 was approximately 7 hours and 44 minutes. These measurements are intended to characterize the cost of the proposed loss components under a fixed implementation and hardware setup, rather than to provide a fully hardware-independent benchmark. In practice, the dominant computational cost comes from backpropagating through the unrolled projected solver; the additional energy-margin and hard-negative terms require extra energy evaluations but do not change the asymptotic structure of the training loop. At inference time, compute scales linearly with the number of projected solver steps and the number of warm starts or particles used by the sampler.

Table 6: Energy evaluations per trajectory. A gradient evaluation includes one energy forward followed by differentiation.

Task	Method	T	Energy-only	Total forwards
N-Queens	CCEM	50	1	51
N-Queens	EBM+PEM	100	101	201
Graph coloring	CCEM	140	1	141
Graph coloring	EBM+PEM	100	101	201
3-SAT	CCEM	140	1	141
3-SAT	EBM+PEM	100	101	201

Table 7: Full inference-budget comparison. Counts include all starts or particles.

Task	Method	T	Starts S/particles P	Energy-only	Gradient evals.	Total forwards
N-Queens	CCEM	50	64	64	3,200	3,264
N-Queens	EBM+PEM	100	1,024	103,424	102,400	205,824
Graph coloring	CCEM	140	64	64	8,960	9,024
Graph coloring	EBM+PEM	100	128	12,928	12,800	25,728
3-SAT	CCEM	140	64	64	8,960	9,024
3-SAT	EBM+PEM	100	1,024	103,424	102,400	205,824

Table 8: Runtime comparison across loss components for the N-Queens Phase 2 refinement experiment. Times are wall-clock durations on a single NVIDIA RTX A4000 GPU.

MSE (α)	Rank (β)	Hard (γ)	Time	Avg. sec/iter
✓			1:53:35	6.82
	✓	✓	2:00:07	7.21
✓	✓		1:54:26	6.87
✓	✓	✓	1:55:47	6.95

D Additional experiments

For Graph Coloring evaluation we used parameters from Table 9.

Table 9: Graph Coloring problem distribution parameters.

Distribution	V	E	d	χ
Erdos Renyi	[20, 39]	[29, 76]	0.12	[3, 4]
Erdos Renyi 2	[81, 99]	[193, 225]	0.05	[3, 4]
Holme Kim	[22, 34]	[56, 92]	0.26	[4, 4]
Holme Kim 2	[86, 100]	[398, 469]	0.10	[5, 6]
Regular Expander	[21, 40]	[63, 120]	0.22	[4, 4]
Regular Expander 2	[86, 100]	[184, 200]	0.23	[3, 3]
Paley	[19, 37]	[171, 465]	0.80	[6, 10]
Complete	[8, 12]	[36, 66]	1.00	[8, 12]

Table 10: Exact valid colorings after discrete decoding (out of 100). The EBM+PEM values are from our evaluation of the authors' public implementation.

Distribution	EBM+PEM	CCEM (ours)
Erdos Renyi	11	**29**
Erdos Renyi 2	0	**4**
Holme Kim	0	**3**
Holme Kim 2	0	0
Regular Expander	7	**20**
Regular Expander 2	0	0
Paley	32	**59**
Complete	57	**100**

For 3-SAT, each clause has seven satisfying local assignments and one falsifying assignment. Complementation is used only to simplify local-factor pretraining; original literal polarities are preserved when clause factors are composed and when the full formula is evaluated. This corrects the overly compressed submitted description, which could be read as searching for an assignment that violates every original clause and then negating it. Training uses random 3-SAT instances with 10–20 variables near the phase transition ($4.258n$ clauses). Evaluation uses 100 formulas with 20 variables and 91 clauses.

Table 11: 3-SAT evaluation on 100 sampled formulas. Fully solved is a percentage of formulas; clause error is reported in percentage points.

Model	Type	Satisfied clauses	Fully solved (%)	Clause error
GCN	SL	0.9617 ± 0.0264	–	–
DGL	SL+TS	0.9520 ± 0.0330	–	–
DIFUSCO ($T = 50$)	SL+S	0.9734 ± 0.0156	–	–
Fast T2T ($T_S = 1, T_G = 1$)	SL+S	0.9749 ± 0.0210	–	–
Fast T2T ($T_S = 5, T_G = 5$)	SL+S	0.9760 ± 0.0273	–	–
NeuroSAT ($T = 50$)	SL+IR	0.9661 ± 0.0185	–	–
NeuroSAT ($T = 500$)	SL+IR	0.9742 ± 0.0154	–	–
NSNet ($T = 50$)	SL+BP	0.9845 ± 0.0272	–	–
NSNet ($T = 500$)	SL+BP	0.9856 ± 0.0266	–	–
EBM ($P = 1024$)	SL+PEM	0.9985 ± 0.0048	–	–
CCEM ($S = 1$)	SL+ProjAdam	0.9953 ± 0.0131	66.04 ± 0.98	0.4673 ± 0.0131
CCEM ($S = 2$)	SL+ProjAdam	0.9955 ± 0.0096	67.02 ± 0.66	0.4519 ± 0.0096
CCEM ($S = 4$)	SL+ProjAdam	0.9956 ± 0.0094	67.54 ± 0.65	0.4435 ± 0.0094
CCEM ($S = 8$)	SL+ProjAdam	0.9956 ± 0.0115	67.94 ± 0.76	0.4381 ± 0.0115
CCEM ($S = 16$)	SL+ProjAdam	0.9957 ± 0.0123	68.40 ± 0.72	0.4315 ± 0.0123
CCEM ($S = 32$)	SL+ProjAdam	0.9958 ± 0.0151	69.01 ± 0.87	0.4211 ± 0.0151
CCEM ($S = 64$)	SL+ProjAdam	0.9959 ± 0.0122	69.65 ± 0.73	0.4108 ± 0.0122

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the main contributions of CCEM, including convex factor energies, projected optimization over convex relaxations, theoretical guarantees, and empirical evaluation. These claims are supported by the theoretical results in Section 3 and Appendix A, as well as the experiments in Section 6.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in Section 7, including the restrictiveness of convexity, possible fractional relaxed solutions that decode poorly, underfitting of highly multimodal discrete landscapes, flat energy regions, and the mismatch between continuous convex minima and discrete optima.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper states assumptions for its theoretical results in Section 3 and provides proofs in Appendix A for the non-convex spurious-minima result, convex certificate sufficiency, first-order optimality sufficiency, approximate optimality, and projected convex inference convergence.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper describes the tasks, evaluation protocols, model architecture, training stages, solver settings, hyperparameters, inference budgets, and optimization details in Sections 4 and 6. Some implementation details are also summarized in Appendix B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper describes experimental settings and indicates that implementation and reproducibility details are relevant, please see supplementary materials.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 6 specifies the N-Queens and 3-SAT protocols, evaluation sizes, number of sampled instances, architecture width and depth, optimizer, learning rates, training epochs, negative sampling counts, projected-solver steps, warm starts, loss weights, margins, and compute setting.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All experiments, corresponding empirical tables showcasing performance, report standard metrics used in existing literature (Mean and standard deviation).

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experiments run on a single GPU NVIDIA RTX A4000. All details for each experiment are provided in the corresponding Appendix Section C.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper studies algorithmic methods for compositional reasoning on synthetic or standard combinatorial benchmarks and does not report human-subject experiments, sensitive data collection, or deployment on high-risk populations.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Section 7 discusses broader impacts and limitations of compositional energy minimization, including improved scalable reasoning through convex optimization as well as risks from relaxation mismatch and failures on discrete combinatorial problems.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk models such as pretrained language models or image generators, nor does it introduce scraped datasets with apparent misuse risk.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper cites all of the original works, providing sufficient references.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [No]

Justification: The paper proposes a new approach for iterative reasoning by modifying energy functions as ICNNs. Current manuscript does not include any new benchmarks, datasets.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or research with human subjects, so IRB approval or equivalent review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method is based on convex compositional energy minimization using ICNN/PICNN factor energies and projected optimization; the manuscript does not indicate that LLMs are used as an important, original, or non-standard component of the research method.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.