SINKHORN-REPARAMETERIZED PRIMAL-DUAL OPTIMIZATION FOR SCALABLE QUADRATIC ASSIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Finding high-quality solutions to large-scale instances of the quadratic assignment problem (QAP) remains challenging under limited computational budgets. We approach QAP through an equivalent continuous formulation with entrywise integrality constraints over the doubly stochastic domain. Building on this formulation, we propose Sinkhorn-Reparameterized Primal-Dual Optimization (SRPD), which replaces explicit primal projection with a differentiable Sinkhorn parameterization. A finite sequence of Sinkhorn normalizations enables gradient-based primal search in an unconstrained latent space, while entrywise dual updates adapt the objective in response to integrality violations. SRPD then recovers feasible permutations through trajectory harvesting: search trajectories run in parallel on GPUs, and their soft assignments are greedily decoded throughout optimization to retain the best solution. For the finite Sinkhorn map and greedy decoder, we bound the decoding error and returned objective using integrality violations and soft-objective values, and identify sufficient conditions for exact penalization. Experiments on QAPLIB demonstrate competitive solution quality and runtime, with substantial improvements over the projected primal–dual baseline, reducing the average gap from 19.28% to 0.50%. On larger taiXXeYY instances, SRPD achieves substantially lower final gaps than the compared baselines under a common time budget.

1 INTRODUCTION

The Quadratic Assignment Problem (QAP) optimizes one-to-one assignments whose costs depend on pairwise interactions, with applications in facility layout (Elshafei, 1977), economic modeling (Heffley, 1980), and graph matching in computer vision (Caetano et al., 2009). Since its formulation by Koopmans & Beckmann (1957), QAP has remained a challenging combinatorial optimization problem: it is $\mathcal{NP}$-hard, admits strong hardness-of-approximation results (Sahni & Gonzalez, 1976), and has $n!$ feasible permutations. As exact methods become expensive with increasing dimension (Loiola et al., 2007), obtaining high-quality solutions within a finite computational budget remains a central challenge.

Existing approaches include discrete metaheuristics that explore permutations through neighborhood moves (Taillard, 1991b; Misevičius et al., 2024) and continuous methods that optimize over soft assignments (Vogelstein et al., 2015; Zaslavskiy et al., 2008). We adopt the continuous perspective by expressing permutation feasibility through doubly stochastic constraints and entrywise continuous equalities that enforce integrality. This equivalent formulation separates the coupled assignment structure from the integrality requirements and provides a basis for primal–dual optimization. Building on the binary optimization framework of Liu et al. (2025), primal updates optimize the current Lagrangian, while entrywise dual updates adjust the integrality terms according to their violations.

Applying this framework to QAP requires handling coupled row and column constraints beyond the coordinatewise box constraints of binary optimization. A direct implementation projects primal iterates onto the Birkhoff polytope, potentially reaching its boundary, where entropy-based integrality terms have singular derivatives. We propose **Sinkhorn-Reparameterized Primal-Dual Optimization (SRPD)**, which replaces explicit primal projection with a differentiable parameterization. A finite sequence of Sinkhorn normalizations maps unrestricted latent variables to strictly positive soft assignments. Differentiating through these normalizations enables gradient-based primal search in

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.

an unconstrained latent space, while retaining the entrywise dual updates. The resulting operations also support batched GPU execution.

To recover feasible permutations, SRPD uses trajectory harvesting: independently initialized searches run in parallel, their soft assignments are greedily decoded throughout optimization, and the best permutation is retained under the original QAP objective. Because finite Sinkhorn normalization need not produce exactly doubly stochastic assignments, our analysis explicitly accounts for the implemented map and decoder, connecting soft-objective values and integrality violations to the returned solution quality.

Our contributions are summarized as follows:

- **Sinkhorn-reparameterized primal–dual search.** We develop a scalable framework for QAPs that combines unconstrained latent primal updates, entrywise dual adaptation, and trajectory harvesting, effectively transforming the discrete QAP into a differentiable optimization process that is amenable to efficient GPU acceleration.
- **Recovery and objective guarantees.** We bound assignment residuals and greedy decoding errors for finite Sinkhorn outputs, derive objective bounds for the retained permutation, and establish sufficient conditions for exact penalization.
- **Empirical performance across scales.** On QAPLIB, SRPD reduces the reported average gap from 19.28% for the projected primal–dual baseline to 0.50%, with lower average runtime. On larger taiXXeYY instances, it achieves substantially lower final gaps than the compared baselines under a common time budget. Additional experiments demonstrate improvements from 2-opt refinement and effectiveness on graph-isomorphism benchmarks.

Figure 1 summarizes the continuous reformulation, Sinkhorn-reparameterized search, and trajectory harvesting stages.

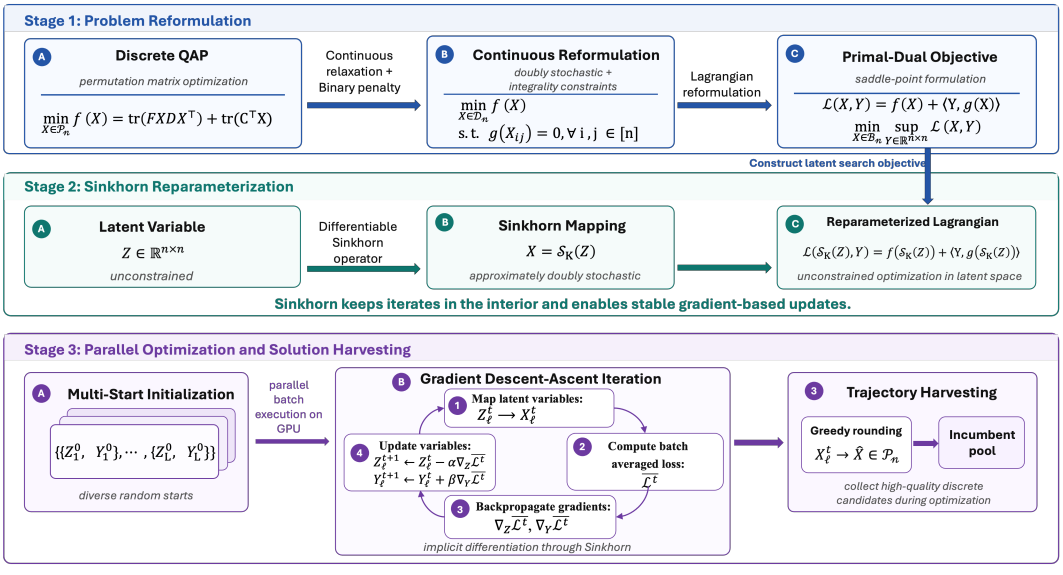

Figure 1: Overview of SRPD: problem reformulation, Sinkhorn reparameterization, and trajectory harvesting.

2 Related Work

2.1 Exact Methods and Discrete Search

Exact methods, including branch-and-bound, branch-and-cut, and mixed-integer optimization with solvers such as Gurobi (Gurobi Optimization, LLC, 2025), can certify global optimality, but their cost limits their use on large QAP instances (Loiola et al., 2007). Discrete metaheuristics instead

search for high-quality permutations through neighborhood exploration and local improvement. Representative methods include robust tabu search (Taillard, 1991b), breakout local search (Benlic & Hao, 2013), and memetic or hybrid evolutionary methods (Benlic & Hao, 2015a; Misevičius et al., 2024), which have shown strong performance on QAPLIB (Burkard et al., 1997).

2.2 Continuous Relaxation and Primal–Dual Optimization

Continuous methods optimize over domains larger than the permutation set. Semidefinite and related convex relaxations provide bounds at a substantial computational cost (Povh & Rendl, 2009; Fogel et al., 2013), while spectral methods use looser relaxations for efficient matching (Leordeanu & Hebert, 2005). Over the doubly stochastic relaxation, Frank–Wolfe methods use linear assignment oracles (Vogelstein et al., 2015), and path-following methods gradually deform the objective to guide the search toward discrete solutions (Zaslavskiy et al., 2008). The primal–dual binary optimization framework (PDBO) of Liu et al. (2025) is closely related to SRPD: it uses coordinatewise dual updates to adapt the integrality terms during optimization. However, its original formulation only considered coordinatewise box constraints and does not directly enforce the coupled row and column constraints required by QAP.

2.3 Sinkhorn and Learning-Based Assignment

Sinkhorn normalization constructs soft assignments through alternating row and column scaling (Sinkhorn, 1964) and is widely used in entropy-regularized optimal transport (Cuturi, 2013) and differentiable graph matching. Neural Graph Matching (Wang et al., 2021) incorporates differentiable assignment into a learned matching model, while BiQAP (Shi et al., 2025) learns modified QAP instances coupled with an entropically regularized inner solver. SAWT (Tan & Mu, 2024) learns iterative assignment improvements using solution-aware attention, and PLMA (Pan et al., 2026) combines a pretrained permutation model with MCMC finetuning on target instances.

3 Sinkhorn-Reparameterized Primal–Dual Optimization

SRPD organizes continuous optimization around the discrete candidates it produces. Integrality constraints give rise to an adaptive update rule inspired by first-order primal–dual methods, while a finite Sinkhorn map supplies the assignment representation for latent search. Independent trajectories are evaluated in batches, and decoded permutations are scored and retained throughout optimization. Throughout, $\langle \cdot, \cdot \rangle$ and $\| \cdot \|_F$ denote the Frobenius inner product and norm; $\| \cdot \|_1$ and $\| \cdot \|_\infty$ are entrywise norms.

3.1 QAP Relaxation and Integrality Functions

In this work, we focus on QAPs, formulated as in (1). This problem seeks a permutation that minimizes the total interaction cost between facilities and locations, formally stated as:

$$\min_{P \in \mathcal{P}_n} f(P) := \operatorname{tr}(FPDP^\top) + \operatorname{tr}(C^\top P), \tag{1}$$

where $F, D, C \in \mathbb{R}^{n \times n}$ specify pairwise and linear assignment costs, respectively. Without loss of generality, we assume D and F have zero diagonals. The feasible domain is the set of $n \times n$ permutation matrices, denoted by $\mathcal{P}_n$ and defined as:

$$\mathcal{P}_n \coloneqq \left\{ X \in \{0,1\}^{n \times n} \mid X\mathbf{1} = X^\top \mathbf{1} = \mathbf{1} \right\},$$

where $\mathbf{1}$ represents the column vector of all ones. The problem complexity grows drastically with size n, comprising a total of n^2 variables.

To mitigate the combinatorial complexity, a standard approach is to relax the discrete set $\mathcal{P}_n$ to its convex hull, the set of *doubly stochastic matrices* known as the Birkhoff Polytope:

$$\mathcal{B}_n = \operatorname{conv}(\mathcal{P}_n) = \{ X \in \mathbb{R}_+^{n \times n} : X\mathbf{1} = X^\top \mathbf{1} = \mathbf{1} \}.$$

To encode integrality, we use either the quadratic or binary-entropy function $g_{\text{quad}}(x) = x(x-1)$, $g_{\text{ent}}(x) = x \log x + (1-x)\log(1-x)$, with $0 \log 0 := 0$. Both are continuous on $[0,1]$, smooth

on $(0, 1)$, and satisfy $g(0) = g(1) = 0$ and $g(x) < 0$ for $x \in (0, 1)$. Thus, over $\mathcal{B}_n$, the constraints $g(X_{ij}) = 0$ characterize permutation matrices, giving the equivalent formulation (Appendix A.6)

$$\begin{aligned} \min_{X \in \mathcal{B}_n} \quad & f(X) \\ \text{s.t.} \quad & g(X_{ij}) = 0, \quad \forall i, j \in [n]. \end{aligned} \tag{2}$$

3.2 Lagrangian Reformulation and Primal–Dual Motivation

Introducing an entrywise multiplier $Y \in \mathbb{R}^{n \times n}$ for the integrality constraints gives

$$\mathcal{L}(X, Y) = f(X) + \langle Y, g(X) \rangle,$$

where $g(X)$ is applied entrywise. The supremum over unrestricted Y equals $f(X)$ when $g(X) = 0$ and $+\infty$ otherwise, so Eq. (2) can be written as

$$\min_{X \in \mathcal{B}_n} \sup_{Y \in \mathbb{R}^{n \times n}} \mathcal{L}(X, Y). \tag{3}$$

A direct first-order primal–dual scheme, following Liu et al. (2025), would use projected gradient descent in X and gradient ascent in Y, with stepsizes α and β:

$$\begin{aligned} X^{t+1} &\leftarrow \Pi_{\mathcal{B}_n} \left(X^t - \alpha \cdot \nabla_X \mathcal{L}(X^t, Y^t) \right) \\ Y^{t+1} &\leftarrow Y^t + \beta \cdot g(X^t). \end{aligned} \tag{4}$$

3.3 Sinkhorn-Reparameterized Primal Search

The projection step in (4) enforces feasibility in $\mathcal{B}_n$, but may place an iterate on its boundary. Consequently, boundary iterates require care in gradient evaluations. This motivates a smooth interior parameterization using unrestricted latent variables.

Exact Sinkhorn reparameterization. We represent the soft assignment as $X = \mathcal{S}(Z)$ using an unrestricted latent matrix $Z \in \mathbb{R}^{n \times n}$. Starting from the entrywise positive matrix $\exp(Z)$, we alternate the row and column normalizations

$$[T_r(M)]_{ij} = \frac{M_{ij}}{\sum_k M_{ik}}, \qquad [T_c(M)]_{ij} = \frac{M_{ij}}{\sum_k M_{kj}}.$$

These iterations converge for every finite Z (Sinkhorn, 1964), defining the exact Sinkhorn map $\mathcal{S}(Z) := \lim_{k \to \infty} (T_c T_r)^k (\exp(Z))$.

The map is smooth and produces strictly positive doubly stochastic assignments; its image covers the entire Birkhoff relative interior, with every permutation matrix lying in its closure (Appendix A.1). Thus, after a latent update, evaluating $X = \mathcal{S}(Z)$ enforces the assignment constraints through the representation. For $n \geq 2$, every output entry lies in $(0, 1)$, so both integrality functions have well-defined derivatives at each represented assignment.

This smooth interior representation supports gradient-based search in the latent space: for fixed Y, a sufficiently small gradient step in Z decreases the composite objective $\mathcal{L}(\mathcal{S}(Z), Y)$ whenever its latent gradient is nonzero. In addition, as $\mathcal{S}(Z)$ approaches a permutation, the Jacobian norm tends to zero, so the first-order response to any fixed latent direction vanishes. This describes the local sensitivity of the representation near permutation vertices. The differential characterization and these local conclusions are derived in Appendices A.2 and A.3.

Finite normalization for practical search. In practice, SRPD truncates the normalization to a fixed number $K \geq 1$ of row–column cycles:

$$\mathcal{S}_K(Z) := (T_c T_r)^K (\exp(Z)). \tag{5}$$

Each cycle consists of row–column reductions and entrywise rescaling, giving $\mathcal{O}(Kn^2)$ forward work and a fixed computation graph suited to batched GPU execution (Cuturi, 2013). For each fixed

finite Z, $\mathcal{S}_K(Z) \to \mathcal{S}(Z)$ as $K \to \infty$. At finite K, the output is strictly positive and column-stochastic, but may have a nonzero row residual, measured by

$$r_{\mathcal{B}}(X) := \|X\mathbf{1} - \mathbf{1}\|_1 + \|X^\top \mathbf{1} - \mathbf{1}\|_1.$$

Nevertheless, $\mathcal{S}_K(\log X) = X$ for every $X \in \mathrm{ri}(\mathcal{B}_n)$, so finite normalization retains all interior feasible assignments and permutation limits (Appendix A.1). The exact-scaling geometry above does not directly provide a finite-K guarantee; Section 4.1 instead analyzes the implemented map through its greedy decoding and objective transfer bounds.

Primal–dual updates with reparameterization. Using the finite Sinkhorn map as the assignment representation, we substitute $X = \mathcal{S}_K(Z)$ into the Lagrangian of Section 3.2 to obtain the composite search objective

$$\mathcal{L}_K(Z, Y) := \mathcal{L}(\mathcal{S}_K(Z), Y) = f(\mathcal{S}_K(Z)) + \langle Y, g(\mathcal{S}_K(Z))\rangle. \tag{6}$$

With $X^t = \mathcal{S}_K(Z^t)$, we perform unconstrained gradient descent in Z while retaining gradient ascent in Y based on the current integrality feedback:

$$\begin{aligned} Z^{t+1} &\leftarrow Z^t - \alpha \cdot \nabla_Z \mathcal{L}_K(Z^t, Y^t) \\ Y^{t+1} &\leftarrow Y^t + \beta \cdot g(X^t). \end{aligned} \tag{7}$$

The latent gradient is computed by backpropagating through the K normalization cycles with Y^t held fixed, and both updates are evaluated at (Z^t, Y^t). Appendix A.11 analyzes the latent update with fixed coefficients. The next section applies these updates across independent trajectories and harvests feasible permutations from their soft assignments.

3.4 Trajectory Harvesting

Optimization of the nonconvex relaxation can be sensitive to initialization. To broaden exploration and reduce reliance on a single starting point, SRPD runs L independently initialized trajectories. Each follows the updates in (7), with $X_t^\ell = \mathcal{S}_K(Z_t^\ell)$ for $\ell = 1, \ldots, L$. These updates are independent across trajectories and can be readily parallelized using batched GPU operations.

To extract discrete candidates throughout the search, we decode each X_t^ℓ at every iteration $t = 0, \ldots, T-1$ into $\widehat{P}_t^\ell = \mathrm{Decode}(X_t^\ell)$ using *global-maximum greedy decoding*: repeatedly match the row–column pair with the largest remaining entry and remove its row and column, breaking ties arbitrarily. After n matches, this yields a feasible permutation regardless of the Sinkhorn residual. We retain the permutation with the lowest QAP objective across all trajectories and iterations as $\widehat{P}_{\mathrm{best}}$. We call this overall process of parallel multistart search, greedy decoding, and retention of the best solution *trajectory harvesting*.

Harvesting requires decoding $L \cdot T$ soft assignments. Exact maximization of the linear matching score $\langle X, P\rangle$ over $\mathcal{P}_n$ would cost $\mathcal{O}(LTn^3)$ with standard assignment solvers (Jonker & Volgenant, 1987). Although greedy decoding need not maximize this score, an equivalent sort-and-scan implementation reduces the total decoding work to $\mathcal{O}(LTn^2 \log n)$, supporting frequent harvesting. Evaluating the resulting permutations under the QAP objective adds $\mathcal{O}(LTn^2)$ work.

3.5 Overall Algorithm

Algorithm 1 combines parallel primal–dual updates with trajectory harvesting. At each iteration, the trajectories form finite-Sinkhorn assignments, update their latent variables and entrywise coefficients, and decode these assignments into feasible permutations. Optional 2-opt refines all decoded candidates in batch by accepting pairwise assignment swaps that decrease f. The resulting candidates are scored under the original QAP objective, and the best candidate replaces the incumbent whenever it improves the stored objective value.

4 Theoretical Analysis

We relate finite-Sinkhorn assignments to the objective value of the permutation returned by SRPD. Section 4.1 bounds the greedy decoding error and the returned objective using the soft objective

Algorithm 1 Sinkhorn-Reparameterized Primal-Dual Optimization

Require: objective f; integrality function g; positive integers K, T, L (Sinkhorn cycles, iterations, trajectories); stepsizes $\alpha, \beta > 0$; initial dual value $\bar{Y} > 0$; optional 2-opt refinement.

Ensure: Best retained permutation $\widehat{P}_{\text{best}} \in \mathcal{P}_n$.

Independently sample $Z_{0,ij}^{\ell} \sim \mathcal{N}(0,1)$ for all $i, j \in \{1, \dots, n\}$ and $\ell \in \{1, \dots, L\}$.
$Y_0^{\ell} \leftarrow \bar{Y}\mathbf{1}\mathbf{1}^{\top}$ for $\ell = 1, \dots, L$; $(\widehat{P}_{\text{best}}, \hat{f}) \leftarrow (\text{null}, +\infty)$.
for $t = 0, \dots, T-1$ **do**
  for $\ell = 1, \dots, L$ **in parallel do**
    $X_t^{\ell} \leftarrow \mathcal{S}_K(Z_t^{\ell})$.
    $Z_{t+1}^{\ell} \leftarrow Z_t^{\ell} - \alpha \cdot \nabla_Z \mathcal{L}(\mathcal{S}_K(Z_t^{\ell}), Y_t^{\ell})$.
    $Y_{t+1}^{\ell} \leftarrow Y_t^{\ell} + \beta \cdot g(X_t^{\ell})$.
    $\widetilde{P}_t^{\ell} \leftarrow \text{Decode}(X_t^{\ell})$. # Global-maximum greedy decoding
    $\widetilde{P}_t^{\ell} \leftarrow \text{2-opt}(\widetilde{P}_t^{\ell})$. # Optional
    $q_t^{\ell} \leftarrow f(\widetilde{P}_t^{\ell})$.
  end for
  Select $\ell_t \in \arg\min_{1 \le \ell \le L} q_t^{\ell}$.
  if $q_t^{\ell_t} < \hat{f}$ **then**
    $(\widehat{P}_{\text{best}}, \hat{f}) \leftarrow (\widetilde{P}_t^{\ell_t}, q_t^{\ell_t})$.
  end if
end for
return $\widehat{P}_{\text{best}}$.

values and integrality violations of the inspected candidates. Section 4.2 establishes equality between the infimum of a sufficiently penalized latent objective and the QAP optimum, and bounds the objective gap of decoded solutions. Proofs appear in Appendix A.

4.1 Finite-Sinkhorn Recovery and Objective Bounds

Finite Sinkhorn normalization can leave a nonzero row residual. The following result bounds both this residual and the distance to the greedily decoded permutation in terms of integrality violation.

Write $h(x) := -g(x)$, $H_h(X) := \sum_{i,j} h(X_{ij})$, and $Q(X) := \sum_{i,j} X_{ij}(1 - X_{ij})$. A continuous function $h : [0,1] \to \mathbb{R}_+$ is *admissible* if $h(0) = h(1) = 0$, $h(x) > 0$ for $x \in (0,1)$, and $\kappa_h := \sup_{x \in (0,1)} x(1-x)/h(x) < \infty$. Both violations in Section 3.1 are admissible, and $Q(X) \le \kappa_h H_h(X)$ on $[0,1]^{n \times n}$. The following result controls the assignment residual and decoding error for every fixed $K \ge 1$.

Theorem 4.1 (Finite-Sinkhorn greedy recovery)**.** *Let h be admissible, Z be finite, $K \ge 1$, and $X = \mathcal{S}_K(Z)$. Write $P_G(X) := \text{Decode}(X)$. Then*

$$r_{\mathcal{B}}(X) \le \|X - P_G(X)\|_1 \le 4nQ(X) \le 4n\kappa_h H_h(X).$$

If $Q(X) < 1/2$, the column maxima occupy distinct rows, the greedy decoder selects these maxima, and the stronger bound $\|X - P_G(X)\|_1 \le 2Q(X)$ holds.

Define

$$M := \sup_{X \in [0,1]^{n \times n}} \|\nabla f(X)\|_{\infty}, \qquad C_h := 4n\kappa_h M.$$

The QAP objective is continuously differentiable on this compact cube, so $M < \infty$. Since the cube contains all finite-Sinkhorn outputs, the corresponding Lipschitz bound and Theorem 4.1 give

$$|f(P_G(X)) - f(X)| \le C_h H_h(X),$$

with the stronger bound $2MQ(X)$ when $Q(X) < 1/2$. Since the retained permutation has an objective value no greater than that of any decoded candidate,

$$f(\widehat{P}_{\text{best}}) \le \min_{\substack{1 \le \ell \le L \\ 0 \le t < T}} \{f(X_t^{\ell}) + C_h H_h(X_t^{\ell})\}. \tag{8}$$

Inequality (8) bounds the returned objective using the soft objective value and integrality violation of each inspected candidate. It holds for any finite collection of trajectory states, without assuming convergence or imposing sign restrictions on the adaptive coefficients. Optional 2-opt refinement preserves the bound because it does not increase candidate objectives (Appendix A.5).

Let $v^\star := \min_{P \in \mathcal{P}_n} f(P)$. If an inspected candidate X_t^ℓ satisfies both $f(X_t^\ell) \le v^\star + \delta$ and $Q(X_t^\ell) \le \eta < 1/2$, with $\delta, \eta \ge 0$, then

$$0 \le f(\widehat{P}_{\mathrm{best}}) - v^\star \le \delta + 2M\eta. \tag{9}$$

Here, δ bounds the soft objective's excess over $v^\star$, while $2M\eta$ bounds the objective discrepancy introduced by decoding.

4.2 Exact Penalization and Discrete Near-Optimality

The estimate in Eq. (9) requires control of both the soft objective and integrality violation. Sufficiently strong penalties allow both quantities to be controlled through the penalized objective. In particular, penalty coefficients exceeding C_h offset the possible objective increase allowed by the decoding bound. Write $\Lambda := -Y$ and $\rho(\Lambda) := \min_{i,j} \Lambda_{ij}$. For fixed Λ, the penalized latent objective is

$$J_\Lambda(Z) := f(\mathcal{S}_K(Z)) + \langle \Lambda, h(\mathcal{S}_K(Z)) \rangle.$$

The following theorem relates the infimum of J_Λ and the accuracy of its decoded solutions to the QAP optimum.

Theorem 4.2 (Finite-Sinkhorn exactness and decoded accuracy)**.** *Let $n \ge 2$ and h be admissible. Fix $K \ge 1$ and a finite coefficient matrix Λ with $\rho := \rho(\Lambda) > C_h$. For every finite Z, let $X = \mathcal{S}_K(Z)$. Then*

$$J_\Lambda(Z) \ge f(P_G(X)) + (\rho - C_h) H_h(X) \ge v^\star + (\rho - C_h) H_h(X).$$

Moreover, $\inf_Z J_\Lambda(Z) = v^\star$, and this infimum is not attained at any finite Z. If $J_\Lambda(Z) \le v^\star + \varepsilon$ for $\varepsilon > 0$, then

$$H_h(X) \le \frac{\varepsilon}{\rho - C_h}, \qquad 0 \le f(P_G(X)) - v^\star \le \varepsilon.$$

Under the stated penalty condition, the penalized latent problem has infimum $v^\star$ for every fixed $K \ge 1$, and an ε-optimal latent point decodes to a permutation with objective at most $v^\star + \varepsilon$. The infimum can be approached along a sequence of strictly positive assignments tending to an optimal permutation, as constructed in Appendix A.10. Supporting exact-penalty results over the Birkhoff polytope and explicit constants for the two violations appear in Appendices A.6–A.9. Fixed-coefficient stationarity and adaptive coefficient accumulation are analyzed separately in Appendices A.11 and A.12.

5 Experiments

We evaluate SRPD as a finite-budget solver that returns the best permutation encountered during its search. QAPLIB assesses solution quality on standard instances, while taiXXeYY examines performance on larger structured problems. We also evaluate SRPD on graph isomorphism (GI), where the task is to recover an adjacency-preserving permutation between isomorphic graphs. The QAP formulation, experimental setup, and full results are presented in Appendix C.

For QAPLIB and taiXXeYY, we report both `SRPD` and a variant incorporating 2-opt local search, denoted `SRPD+2-opt`, to evaluate the framework with and without discrete local refinement. In `SRPD+2-opt`, the permutations decoded at each iteration are refined in batch before candidate evaluation and incumbent selection. Experiments use a server equipped with an Intel Core i9-13900K CPU and an NVIDIA RTX 4090 GPU; SRPD is implemented in PyTorch. Dataset descriptions, hyperparameters, and baseline details are provided in Appendices B and D.

5.1 QAPLIB

We evaluate SRPD on QAPLIB instances with $12 \le n \le 256$ to assess solution quality on standard benchmarks and the benefit of local refinement (Burkard et al., 1997; ISA-QAP). We compare with

the exact solver Gurobi (Gurobi Optimization, LLC, 2025), the metaheuristics Ro-TS (Taillard, 1991b) and BMA (Benlic & Hao, 2015a), the primal-dual method PDBO (Liu et al., 2025), and the learning-based methods SAWT (Tan & Mu, 2024) and PLMA (Pan et al., 2026). Solution quality is measured by the relative gap to the best known solution (BKS) (Misevičius et al., 2024), $\mathrm{Gap} = (\mathcal{J} - \mathcal{J}_{\mathrm{BKS}})/\mathcal{J}_{\mathrm{BKS}} \times 100\%$, where $\mathcal{J}$ is the obtained objective value and lower gaps indicate better solutions. Table 1 reports family-level gaps and average runtimes; dataset descriptions and baseline settings are provided in Appendices B.1 and D.2.

Compared with PDBO, which uses the projection-based updates in Eq. (4), SRPD reduces the reported average gap from 19.28% to 0.50%, with lower gaps on all fifteen families. The average runtime also decreases from 0.93 to 0.41 seconds. These gains, achieved without 2-opt refinement, support the importance and effectiveness of Sinkhorn reparameterization within SRPD.

SRPD combines near-BKS solution quality with efficient search, attaining family-level gaps of at most 0.7% on thirteen of the fifteen families and the lowest reported average runtime of 0.41 seconds among the compared methods. The framework also integrates naturally with discrete local refinement, following the general strategy of coupling solution generation with local improvement used in PLMA (Pan et al., 2026). Adding 2-opt reduces the average gap from 0.50% to 0.12%, a 76% relative reduction, with all reported family-level gaps at most 0.6%. The refined variant maintains a subsecond average runtime (0.91 seconds), with an average gap within 0.10 percentage points of Ro-TS (0.06%), BMA (0.02%), and PLMA (0.03%).

Table 1: Relative gap to the BKS (%) across QAPLIB instance families. The last two rows report average gaps and runtimes (s); lower values are better.

Instance	Gurobi	Ro-TS	BMA	PDBO	SAWT	PLMA	SRPD	SRPD+2-opt
bur	0.5%	0.0%	0.0%	2.5%	4.0%	0.0%	0.0%	0.0%
chr	1.5%	0.6%	0.0%	91.5%	147.5%	0.0%	3.7%	0.6%
els	0.0%	0.0%	0.0%	38.1%	47.4%	0.0%	0.2%	0.0%
esc	8.6%	0.0%	0.0%	9.3%	43.4%	0.0%	0.1%	0.0%
had	0.2%	0.0%	0.0%	5.4%	5.2%	0.0%	0.0%	0.0%
kra	6.1%	0.0%	0.0%	15.1%	32.9%	0.0%	0.1%	0.0%
lip	6.9%	0.0%	0.0%	1.2%	1.4%	0.1%	0.3%	0.2%
nug	2.2%	0.0%	0.0%	8.7%	19.3%	0.0%	0.0%	0.0%
rou	1.3%	0.0%	0.0%	12.1%	15.1%	0.0%	0.0%	0.0%
scr	0.0%	0.0%	0.0%	23.6%	33.9%	0.0%	0.0%	0.0%
sko	14.0%	0.0%	0.0%	10.8%	16.2%	0.0%	0.7%	0.2%
ste	7.4%	0.0%	0.0%	36.9%	108.0%	0.0%	0.2%	0.0%
tai	10.5%	0.2%	0.2%	12.0%	34.7%	0.2%	1.2%	0.4%
tho	10.9%	0.0%	0.1%	15.2%	24.1%	0.0%	0.6%	0.2%
wil	7.5%	0.0%	0.0%	6.8%	9.5%	0.0%	0.5%	0.1%
Avg. Gap	5.17%	0.06%	0.02%	19.28%	36.17%	0.03%	0.50%	0.12%
Avg. Time (s)	82.73	2.74	3.04	0.93	14.72	0.95	0.41	0.91

5.2 TAIXXEYY

To assess performance on larger structured problems, we evaluate the first three taiXXeYY instances at each size $n \in \{125, 175, 343, 729\}$ (Drezner et al., 2005). We compare with Ro-TS (Taillard, 1991b), BMA (Benlic & Hao, 2015a), and PLMA (Pan et al., 2026), which achieved competitive solution quality in the QAPLIB experiments. All methods receive a time limit of $1,800$ seconds per instance. Dataset descriptions and baseline details are provided in Appendices B.2 and D.2.

Figure 2 examines the earlier search dynamics. PLMA often discovers better incumbents quickly, while SRPD initially requires more time to reach comparable solution quality. As search proceeds, however, SRPD steadily improves its incumbents, whereas the baseline trajectories generally show more limited progress after their initial gains. This difference is especially pronounced at $n = 343$ and $n = 729$, where both SRPD variants eventually attain substantially lower gaps than all three baselines within the displayed windows.

Table 2 reports final solution quality under the full $1,800$-second budget. SRPD obtains lower objective values than all three baselines on nine of the twelve instances, including all six instances with $n \in \{343, 729\}$. On these larger instances, its gaps to the BKS range from 3.23% to 6.70%, compared with 14.52% to 19.92% for BMA, which achieves the best baseline objective on each of

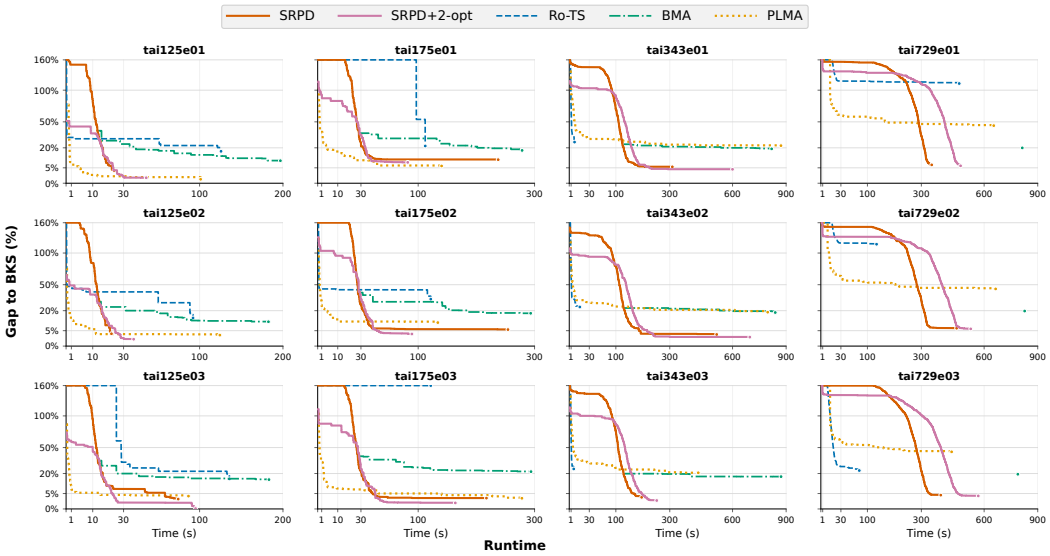

Figure 2: Incumbent gap to the BKS during the earlier search windows for `SRPD`, `SRPD+2-opt`, `Ro-TS`, `BMA`, and `PLMA` on taiXXeYY. Columns correspond to $n = 125, 175, 343, 729$, and rows to instances e01–e03. The displayed time ranges vary by problem size; final results under the full $1,800$-second time limit are reported in Table 2.

these instances. The refined variant, `SRPD+2-opt`, improves upon `SRPD` on all twelve instances and achieves the lowest objective among the compared methods on eleven.

Table 2: Best objective values obtained on taiXXeYY under a common $1,800$-second time limit per instance. Time records when each final incumbent was first found. Bold and underlined objectives denote the best and second-best results among the evaluated methods, respectively.

Instance	BKS	Ro-TS		BMA		PLMA		SRPD		SRPD+2-opt	
		Obj. ↓	Time (s) ↓	Obj. ↓	Time (s) ↓	Obj. ↓	Time (s) ↓	Obj. ↓	Time (s) ↓	Obj. ↓	Time (s) ↓
tai125e01	35,426	39,746	320	37,992	1,013	35,804	33	36,814	22	**35,744**	43
tai125e02	36,178	39,354	592	36,830	1,000	37,248	119	37,482	20	**36,672**	31
tai125e03	30,498	35,138	133	33,500	1,553	31,640	88	31,302	76	**30,534**	94
tai175e01	57,540	65,420	355	62,072	1,497	**60,694**	318	63,372	230	62,304	78
tai175e02	50,110	66,142	118	54,220	1,756	55,500	126	52,962	244	**51,798**	86
tai175e03	53,900	218,360	829	59,108	1,740	55,490	243	55,406	134	**54,490**	103
tai343e01	141,048	177,228	4	161,526	1,714	170,886	1,159	148,942	199	**147,068**	291
tai343e02	148,584	183,594	12	172,550	1,535	177,730	612	153,412	517	**151,692**	300
tai343e03	142,092	176,884	3	163,758	1,768	171,562	362	146,680	187	**144,866**	243
tai729e01	416,260	883,310	473	499,166	809	599,180	1,135	444,138	344	**443,204**	476
tai729e02	422,570	913,870	129	505,600	823	609,984	1,054	449,796	460	**447,958**	522
tai729e03	405,004	497,770	76	480,546	1,596	578,394	1,093	422,000	385	**420,562**	561

6 Conclusion

SRPD combines finite differentiable Sinkhorn reparameterization with unconstrained latent primal updates and entrywise dual updates for QAP. Trajectory harvesting uses batched multistart search on GPUs and greedy decoding throughout optimization to retain the best feasible permutation under the original QAP objective. Our recovery and objective bounds link the returned objective to soft-objective values and integrality violations along the search, complemented by an analysis of formulation consistency under appropriate fixed-penalty conditions. Experiments on QAP and graph-isomorphism benchmarks demonstrate the effectiveness of SRPD under finite computational budgets, with optional 2-opt refinement further improving QAP solution quality. Future work includes adaptive allocation of computation across trajectories and local-search neighborhoods tailored to problem structure.

AI USE STATEMENT

Generative AI tools were used to assist with drafting and editing the manuscript, refining mathematical arguments, and preparing LaTeX tables. All AI-assisted content was reviewed and revised by the authors. The authors take full responsibility for the accuracy and integrity of the final manuscript.

REFERENCES

Una Benlic and Jin-Kao Hao. Breakout local search for the quadratic assignment problem. *Applied Mathematics and Computation*, 219(9):4800–4815, 2013.

Una Benlic and Jin-Kao Hao. Memetic search for the quadratic assignment problem. *Expert Systems with Applications*, 42(1):584–595, 2015a.

Una Benlic and Jin-Kao Hao. Source code for memetic algorithm (BMA), 2015b. URL https://leria-info.univ-angers.fr/~jinkao.hao/BMA.html. Accessed: 2026-01-20.

Rainer E Burkard, Stefan E Karisch, and Franz Rendl. Qaplib–a quadratic assignment problem library. *Journal of Global optimization*, 10(4):391–403, 1997.

Tibério S Caetano, Julian J McAuley, Li Cheng, Quoc V Le, and Alex J Smola. Learning graph matching. *IEEE transactions on pattern analysis and machine intelligence*, 31(6):1048–1058, 2009.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Zvi Drezner, Peter M Hahn, and Éeric D Taillard. Recent advances for the quadratic assignment problem with special emphasis on instances that are difficult for meta-heuristic methods. *Annals of Operations research*, 139(1):65–94, 2005.

Alwalid N Elshafei. Hospital layout as a quadratic assignment problem. *Journal of the Operational Research Society*, 28(1):167–179, 1977.

Fajwel Fogel, Rodolphe Jenatton, Francis Bach, and Alexandre d'Aspremont. Convex relaxations for permutation problems. *Advances in neural information processing systems*, 26, 2013.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2025. URL https://www.gurobi.com.

Dennis R Heffley. Decomposition of the koopmans-beckmann problem. *Regional Science and Urban Economics*, 10(4):571–580, 1980.

Deborah Hendrych, Hannah Troppens, Mathieu Besançon, and Sebastian Pokutta. Convex mixed-integer optimization with frank–wolfe methods. *Mathematical Programming Computation*, 17 (4):731–757, 2025.

ISA-QAP. Isa-qap instances. GitHub repository, 2023. URL https://github.com/jeffrey-chr/ISA-QAP-Instances/tree/main.

Roy Jonker and Anton Volgenant. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340, 1987.

Tjalling C Koopmans and Martin Beckmann. Assignment problems and the location of economic activities. *Econometrica: journal of the Econometric Society*, pp. 53–76, 1957.

Marius Leordeanu and Martial Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pp. 1482–1489. IEEE, 2005.

Wenbo Liu, Akang Wang, Dun Ma, Hongyi Jiang, Jianghua Wu, and Wenguo Yang. Smoothing binary optimization: A primal-dual perspective. *arXiv preprint arXiv:2509.21064*, 2025.

Eliane Maria Loiola, Nair Maria Maia De Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. A survey for the quadratic assignment problem. *European journal of operational research*, 176(2):657–690, 2007.

Brendan McKay and Adolfo Piperno. nauty and Traces benchmark graph libraries. https://pallini.di.uniroma1.it/Graphs.html, 2025. Collections of benchmark graphs (DIMACS and dreadnaut formats); CURRENT VERSION: 2_9_3 (JAN 1, 2026).

Brendan D McKay and Adolfo Piperno. Practical graph isomorphism, ii. *Journal of symbolic computation*, 60:94–112, 2014.

Alfonsas Misevičius, Aleksandras Andrejevas, Armantas Ostreika, Dovilė Verenė, and Gintarė Žekienė. An improved hybrid genetic-hierarchical algorithm for the quadratic assignment problem. *Mathematics*, 12(23):3726, 2024.

Yicheng Pan, Ruisong Zhou, Haijun Zou, Tianyou Li, and Zaiwen Wen. Learning to solve the quadratic assignment problem with warm-started mcmc finetuning. *arXiv preprint arXiv:2604.20109*, 2026.

Janez Povh and Franz Rendl. Copositive and semidefinite relaxations of the quadratic assignment problem. *Discrete Optimization*, 6(3):231–241, 2009.

Sartaj Sahni and Teofilo Gonzalez. P-complete approximation problems. *Journal of the ACM (JACM)*, 23(3):555–565, 1976.

Liangliang Shi, Haoran Zhang, Shuheng Shen, Changhua Meng, Weiqiang Wang, and Junchi Yan. Biqap: Neural bi-level optimization-based framework for solving quadratic assignment problems. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 2550–2561, 2025.

Richard Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The annals of mathematical statistics*, 35(2):876–879, 1964.

Éric Taillard. Source code for robust tabu search (tabou_qap2.c), 1991a. URL `https://mistic.iict-heig-vd.ch/taillard/codes.dir/tabou_qap2.c`. Accessed: 2026-01-20.

Éric Taillard. Robust taboo search for the quadratic assignment problem. *Parallel computing*, 17 (4-5):443–455, 1991b.

Zhentao Tan and Yadong Mu. Learning solution-aware transformers for efficiently solving quadratic assignment problem. In *International Conference on Machine Learning*, pp. 47627–47648. PMLR, 2024.

Joshua T Vogelstein, John M Conroy, Vince Lyzinski, Louis J Podrazik, Steven G Kratzer, Eric T Harley, Donniell E Fishkind, R Jacob Vogelstein, and Carey E Priebe. Fast approximate quadratic programming for graph matching. *PLOS one*, 10(4):e0121002, 2015.

Runzhong Wang, Junchi Yan, and Xiaokang Yang. Neural graph matching network: Learning lawler's quadratic assignment problem with extension to hypergraph and multiple-graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5261–5279, 2021.

Mikhail Zaslavskiy, Francis Bach, and Jean-Philippe Vert. A path following algorithm for the graph matching problem. *IEEE transactions on pattern analysis and machine intelligence*, 31(12): 2227–2242, 2008.

A PROOFS FOR THE THEORETICAL RESULTS

The proofs cover the representation and geometry results in Section 3.3, followed by the recovery and formulation-consistency results in Section 4. Appendices A.1–A.3 establish coverage, the exact differential, and its consequences for latent gradient updates. Appendix A.4 proves finite-Sinkhorn recovery and the decoded objective bound; Appendix A.5 derives the harvested output guarantee. Appendices A.6–A.9 establish Birkhoff integrality bounds and exact penalization, and Appendix A.10 treats the finite latent formulation and its strong-penalty consequence for harvesting. Finally, Appendices A.11 and A.12 give the fixed-coefficient stationarity analysis and finite-time accounting for adaptive updates.

A.1 EXACT SINKHORN PARAMETERIZATION

Theorem A.1 (Entropic representation and coverage)**.** *For every $Z \in \mathbb{R}^{n\times n}$,*

$$\mathcal{S}(Z) = \arg\max_{X\in\mathcal{B}_n} \{\langle Z, X\rangle + \mathcal{H}(X)\}, \qquad \mathcal{S}(\mathbb{R}^{n\times n}) = \mathrm{ri}(\mathcal{B}_n),$$

where $\mathcal{H}(X) := -\sum_{i,j} X_{ij}\log X_{ij}$ with $0\log 0 := 0$.

The matrix entropy $\mathcal{H}$ characterizes the representation and is distinct from the binary-entropy integrality term in the search objective.

Proof. Consider

$$\max_{X\in\mathcal{B}_n} \left\{ \langle Z, X\rangle - \sum_{i,j} X_{ij}\log X_{ij} \right\}. \tag{10}$$

The objective is continuous on the compact set $\mathcal{B}_n$ and strictly concave, so it has a unique optimizer. To establish positivity, suppose an optimizer $\bar{X}$ has $m \geq 1$ zero entries and let $X_\epsilon = (1-\epsilon)\bar{X} + \epsilon\mathbf{1}\mathbf{1}^\top/n$. The linear objective changes by $\mathcal{O}(\epsilon)$. The entropy change at the positive entries is also $\mathcal{O}(\epsilon)$, whereas the zero entries contribute $(m/n)\epsilon\log(1/\epsilon) + \mathcal{O}(\epsilon)$. For sufficiently small $\epsilon > 0$, the total objective therefore increases, a contradiction. Thus the optimizer is strictly positive. Introduce Lagrange multipliers $a, b \in \mathbb{R}^n$ for the row and column constraints. The Lagrangian is

$$\mathcal{J}(X, a, b) = \langle Z, X\rangle - \sum_{i,j} X_{ij}\log X_{ij} + a^\top(X\mathbf{1} - \mathbf{1}) + b^\top(X^\top\mathbf{1} - \mathbf{1}).$$

The stationarity condition with respect to X_{ij} is

$$Z_{ij} - (\log X_{ij} + 1) + a_i + b_j = 0. \tag{11}$$

Therefore

$$X_{ij} = u_i \exp(Z_{ij}) v_j,$$

where the constant factor $\exp(-1)$ is absorbed into the positive scaling vectors u and v.

The row and column constraints require

$$X\mathbf{1} = \mathbf{1}, \qquad X^\top\mathbf{1} = \mathbf{1},$$

so u and v are exactly the Sinkhorn scaling factors of $\exp Z$. Hence the unique optimizer of (10) is

$$\mathcal{S}(Z) = \mathrm{Sinkhorn}\left(\exp Z\right).$$

It remains to characterize the image of the map. For every finite Z,

$$\exp Z > 0$$

entrywise, and Sinkhorn scaling preserves strict positivity. Thus

$$\mathcal{S}(Z) \in \mathrm{ri}(\mathcal{B}_n).$$

Conversely, take any

$$X \in \mathrm{ri}(\mathcal{B}_n).$$

Since $X_{ij} > 0$, define

$$Z = \log X$$

entrywise. Then

$$\exp Z = X.$$

Because X is already doubly stochastic, its Sinkhorn scaling is itself:

$$\mathcal{S}(Z) = X.$$

Therefore

$$\mathcal{S}(\mathbb{R}^{n\times n}) = \mathrm{ri}(\mathcal{B}_n).$$

□

Coverage under finite normalization. For any $X \in \mathrm{ri}(\mathcal{B}_n)$, the finite latent matrix $Z = \log X$ satisfies $\exp Z = X$. Since both row and column normalization leave X unchanged, $\mathcal{S}_K(\log X) = X$ for every $K \geq 1$. Thus $\mathrm{ri}(\mathcal{B}_n) \subseteq \mathcal{S}_K(\mathbb{R}^{n\times n})$. For any $P \in \mathcal{P}_n$ and $0 < \epsilon < 1$, let $X_\epsilon = (1-\epsilon)P + \epsilon \mathbf{1}\mathbf{1}^\top/n$. Then $X_\epsilon \in \mathrm{ri}(\mathcal{B}_n)$ and $\mathcal{S}_K(\log X_\epsilon) = X_\epsilon \to P$ as $\epsilon \downarrow 0$. Every permutation therefore belongs to the closure of the finite-map image. This inclusion does not assert that all finite-Sinkhorn outputs are doubly stochastic.

Finite-Sinkhorn consistency. For the finite map in Eq. (5), the matrix $\exp Z$ is strictly positive for every finite Z. By the classical Sinkhorn scaling theorem, alternating row and column normalizations converge to its unique doubly stochastic diagonal scaling. Therefore

$$\mathcal{S}_K(Z) \longrightarrow \mathcal{S}(Z) \qquad \text{as } K \to \infty.$$

Every finite iterate remains strictly positive, although it need not be exactly doubly stochastic after finitely many normalization cycles.

A.2 Differential of the Sinkhorn Map

We first establish a variational characterization of the differential.

Lemma A.2 (Differential of the exact Sinkhorn map)**.** *Let*

$$X = \mathcal{S}(Z),$$

and let $E \in \mathbb{R}^{n\times n}$. Define

$$\Delta := d\mathcal{S}(Z)[E].$$

Then

$$\Delta \in \mathcal{T},$$

where

$$\mathcal{T} = \left\{U : U\mathbf{1} = 0,\ U^\top \mathbf{1} = 0\right\}.$$

Moreover,

$$\Delta = \arg\min_{U\in\mathcal{T}} \left\{ \frac{1}{2}\sum_{i,j} \frac{U_{ij}^2}{X_{ij}} - \langle E, U\rangle \right\}. \tag{12}$$

Proof. We first justify differentiability. Remove one redundant marginal constraint, or equivalently fix the multiplier gauge by setting $b_n = 0$. The negative objective Hessian at a positive solution is $\mathrm{diag}(1/X_{ij})$, which is positive definite. Together with the full-row-rank marginal constraint matrix, it gives a nonsingular KKT Jacobian. The implicit function theorem therefore provides smooth local functions for the optimizer and the gauge-fixed multipliers as Z varies.

Define

$$X(s) = \mathcal{S}(Z + sE).$$

Since

$$X(s) \in \mathcal{B}_n,$$

we have

$$X(s)\mathbf{1} = \mathbf{1}, \qquad X(s)^\top \mathbf{1} = \mathbf{1}.$$

Differentiating at $s = 0$ gives

$$\Delta \mathbf{1} = 0, \qquad \Delta^\top \mathbf{1} = 0,$$

hence

$$\Delta \in \mathcal{T}.$$

From the KKT condition (11), there exist multipliers a, b satisfying

$$Z_{ij} - (\log X_{ij} + 1) + a_i + b_j = 0.$$

Under the perturbation $Z + sE$, write

$$X(s) = X + s\Delta + o(s),$$

$$a(s) = a + s\alpha + o(s), \qquad b(s) = b + s\gamma + o(s).$$

Differentiating the KKT condition at $s = 0$ gives

$$E_{ij} - \frac{\Delta_{ij}}{X_{ij}} + \alpha_i + \gamma_j = 0.$$

Equivalently,

$$\frac{\Delta_{ij}}{X_{ij}} - E_{ij} = \alpha_i + \gamma_j.$$

These equations, together with

$$\Delta \in \mathcal{T},$$

are precisely the KKT conditions for

$$\min_{U \in \mathcal{T}} \left\{ \frac{1}{2} \sum_{i,j} \frac{U_{ij}^2}{X_{ij}} - \langle E, U \rangle \right\}.$$

Since $X_{ij} > 0$, the quadratic objective is strictly convex on $\mathcal{T}$. Hence the minimizer is unique and equals Δ. □

A.3 Jacobian Damping Near Permutation Vertices

For $n \geq 2$, let $\mathcal{T} := \{U : U\mathbf{1} = 0,\ U^\top \mathbf{1} = 0\}$ be the tangent space of the Birkhoff polytope. For $X \in \mathrm{ri}(\mathcal{B}_n)$, define

$$\mu(X) := \min_{U \in \mathcal{T}, \|U\|_F = 1} \sum_{i,j} \frac{U_{ij}^2}{X_{ij}}.$$

The minimum exists because the unit sphere in $\mathcal{T}$ is compact, and it is strictly positive because X is entrywise positive.

Theorem A.3 (Jacobian damping near permutation vertices)**.** *Let $n \geq 2$ and $X = \mathcal{S}(Z)$. For every $E \in \mathbb{R}^{n \times n}$,*

$$\|d\mathcal{S}(Z)[E]\|_F \leq \frac{\|E\|_F}{\mu(X)}.$$

As $X \to P$ for any $P \in \mathcal{P}_n$, $\mu(X) \to \infty$ and $\|d\mathcal{S}(Z)\|_{\mathrm{op}} \to 0$.

Proof. Let

$$\Delta = d\mathcal{S}(Z)[E].$$

By Lemma A.2, the first-order optimality condition of (12) implies that, for every $U \in \mathcal{T}$,

$$\sum_{i,j} \frac{\Delta_{ij} U_{ij}}{X_{ij}} = \langle E, U \rangle.$$

Setting $U = \Delta$ gives

$$\sum_{i,j} \frac{\Delta_{ij}^2}{X_{ij}} = \langle E, \Delta \rangle \leq \|E\|_F \|\Delta\|_F. \tag{13}$$

By definition of $\mu(X)$,

$$\sum_{i,j} \frac{\Delta_{ij}^2}{X_{ij}} \geq \mu(X)\|\Delta\|_F^2.$$

Combining this inequality with (13) gives

$$\mu(X)\|\Delta\|_F^2 \leq \|E\|_F\|\Delta\|_F.$$

If $\Delta = 0$, the desired bound is immediate. Otherwise, dividing by $\|\Delta\|_F$ yields

$$\|\Delta\|_F \leq \frac{\|E\|_F}{\mu(X)}.$$

It remains to prove

$$\mu(X) \to \infty$$

as X approaches a permutation matrix. Suppose the contrary. Then there exist a sequence

$$X_k \to P, \qquad P \in \mathcal{P}_n,$$

a constant $M < \infty$, and matrices

$$U_k \in \mathcal{T}, \qquad \|U_k\|_F = 1,$$

such that

$$\sum_{i,j} \frac{(U_k)_{ij}^2}{(X_k)_{ij}} \leq M. \tag{14}$$

The unit sphere is compact, so after passing to a subsequence we may assume

$$U_k \to U, \qquad \|U\|_F = 1.$$

Consider an entry (i, j) for which

$$P_{ij} = 0.$$

Then

$$(X_k)_{ij} \to 0.$$

Because every term in (14) is nonnegative,

$$\frac{(U_k)_{ij}^2}{(X_k)_{ij}} \leq M.$$

Hence

$$|(U_k)_{ij}| \leq \sqrt{M(X_k)_{ij}} \to 0.$$

Thus

$$U_{ij} = 0 \qquad \text{whenever } P_{ij} = 0.$$

Therefore U can be supported only on the n entries at which $P_{ij} = 1$. Since $U \in \mathcal{T}$, every row sum and every column sum of U is zero. Each row of P contains exactly one supported position, so the row-sum constraints force every supported entry of U to be zero. Consequently

$$U = 0,$$

contradicting

$$\|U\|_F = 1.$$

Hence

$$\mu(X) \to \infty$$

as $X \to P$.

Finally,

$$\|d\mathcal{S}(Z)\|_{\text{op}} = \sup_{\|E\|_F=1} \|d\mathcal{S}(Z)[E]\|_F \leq \frac{1}{\mu(X)} \to 0.$$

□

Consequence for latent gradient updates. Fix Y and define $\phi_Y(Z) := \mathcal{L}(\mathcal{S}(Z), Y)$. At a finite Z, let $X = \mathcal{S}(Z)$, $G = \nabla_X \mathcal{L}(X, Y)$, and $J = d\mathcal{S}(Z)$. Both integrality functions are smooth at the strictly positive assignment X, whose entries lie in $(0, 1)$. The chain rule gives $\nabla\phi_Y(Z) = J^*G$. For the local latent descent path $Z(s) = Z - s\nabla\phi_Y(Z)$, set $X(s) = \mathcal{S}(Z(s))$. Differentiability yields

$$X(s) = X - sJJ^*G + o(s), \qquad U := X'(0) = -JJ^*G \in \mathcal{T}.$$

In particular, the adjoint identity gives

$$\frac{d}{ds}\mathcal{L}(X(s), Y)\bigg|_{s=0} = \langle G, U\rangle = -\langle J^*G, J^*G\rangle = -\|\nabla\phi_Y(Z)\|_F^2.$$

If the latent gradient is nonzero, sufficiently small positive steps therefore decrease the fixed-coefficient objective, and $X(s)$ remains in $\mathrm{ri}(\mathcal{B}_n)$ by construction. Since $\|J^*\|_{\mathrm{op}} = \|J\|_{\mathrm{op}}$, Theorem A.3 also implies

$$\|U\|_F \le \|J\|_{\mathrm{op}}\|J^*\|_{\mathrm{op}}\|G\|_F \le \frac{\|G\|_F}{\mu(X)^2}.$$

This is an attenuation bound relative to the assignment-gradient magnitude. Absolute vanishing of U as $X \to P$ follows, for example, when $\|G\|_F$ remains bounded; boundedness is not assumed for the binary-entropy gradient near the boundary. These local identities concern exact scaling and a fixed current coefficient matrix.

A.4 Greedy Recovery from Finite Sinkhorn Outputs

Throughout this subsection, $Z \in \mathbb{R}^{n\times n}$ has finite entries, $K \ge 1$, and $X = \mathcal{S}_K(Z)$. In particular, X is strictly positive and column stochastic. We first record the additional structure supplied by the final row–column normalization cycle.

Lemma A.4 (Final-cycle structure)**.** *There exist a strictly positive row-stochastic matrix R and positive scalars $c_1, \dots, c_n$ such that*

$$R_{ij} = c_j X_{ij}, \qquad \sum_j c_j X_{ij} = 1, \qquad \sum_i X_{ij} = 1. \tag{15}$$

If $m_j := \max_i X_{ij}$, then $c_j m_j \le 1$ for every j.

Proof. Let $R = T_r((T_cT_r)^{K-1}(\exp Z))$ be the matrix immediately before the final column normalization, and set $c_j = \sum_i R_{ij} > 0$. Then R is strictly positive and row stochastic, and $X_{ij} = R_{ij}/c_j$, which proves (15). For any $i_j \in \arg\max_i X_{ij}$, row stochasticity gives $c_j m_j = R_{i_j j} \le 1$. □

Proof of Theorem 4.1. Write $P_G = \mathrm{Decode}(X)$ for the global-maximum greedy decoder in Section 3.4. Define $q_j := \sum_i X_{ij}(1 - X_{ij})$ and $m_j := \max_i X_{ij}$. Since every column sums to one,

$$q_j = 1 - \sum_i X_{ij}^2 \ge 1 - m_j, \qquad E := \sum_j (1 - m_j) \le \sum_j q_j = Q(X). \tag{16}$$

Here the first inequality follows from $\sum_i X_{ij}^2 \le m_j \sum_i X_{ij} = m_j$.

Suppose first that $Q(X) < 1/2$. Then $m_j \ge 1 - q_j \ge 1 - Q(X) > 1/2$ for every column, so each column has a unique maximizer. We claim that these maximizers occupy distinct rows. Otherwise, at least one row r contains none of the n column maximizers. For this row, $X_{rj} \le 1 - m_j$ for every j. Using Lemma A.4 and Eq. (16) gives

$$1 = \sum_j c_j X_{rj} \le \sum_j \frac{X_{rj}}{m_j} \le \sum_j \frac{1 - m_j}{m_j} \le \frac{E}{1 - Q(X)} \le \frac{Q(X)}{1 - Q(X)} < 1,$$

a contradiction. The column maximizers therefore define a permutation. Its selected entries exceed $1/2$, whereas every other entry is smaller than $1/2$. Consequently, the first greedy selection belongs to this permutation. Removing its row and column leaves all other permutation entries available.

Induction shows that the decoder selects exactly these entries. Ties among maximizers in different columns may be resolved in any order, without changing the resulting permutation. It follows that

$$\|X - P_G\|_1 = 2\sum_j (1 - m_j) = 2E \le 2Q(X).$$

For the global bound, column stochasticity and permutation feasibility imply $\|X - P_G\|_1 \le \|X\|_1 + \|P_G\|_1 = 2n$. If $Q(X) \ge 1/2$, this gives $\|X - P_G\|_1 \le 2n \le 4nQ(X)$; if $Q(X) < 1/2$, the local bound already gives $\|X - P_G\|_1 \le 2Q(X) \le 4nQ(X)$. Admissibility yields $x(1-x) \le \kappa_h h(x)$ on $[0,1]$, and hence

$$\|X - P_G\|_1 \le 4nQ(X) \le 4n\kappa_h H_h(X).$$

The constant $4n$ is a sufficient global bound; no sharpness is claimed.

Finally, $X^\top \mathbf{1} = \mathbf{1}$ and $P_G \mathbf{1} = \mathbf{1}$, so

$$r_{\mathcal{B}}(X) = \|(X - P_G)\mathbf{1}\|_1 \le \|X - P_G\|_1.$$

This proves all the assertions. □

In particular, any sequence of finite Sinkhorn outputs for which $H_h(X^t) \to 0$ also satisfies $r_{\mathcal{B}}(X^t) \to 0$ and $\mathrm{dist}_1(X^t, \mathcal{P}_n) \to 0$. Every accumulation point is therefore a permutation. This consequence uses the normalization structure in Lemma A.4; column stochasticity alone would not suffice.

Finite Sinkhorn outputs can have row sums different from one. To compare their objective values with decoded permutations, we use $M := \sup_{A \in [0,1]^{n\times n}} \|\nabla f(A)\|_\infty$, which is finite because the QAP objective is continuously differentiable on a compact set. For any $A, B \in [0,1]^{n\times n}$, the line segment between them remains in the same cube, and the fundamental theorem of calculus gives

$$\begin{aligned} |f(A) - f(B)| &= \left| \int_0^1 \langle \nabla f(B + s(A-B)), A - B \rangle \, ds \right| \\ &\le M \|A - B\|_1. \end{aligned}$$

Consequently, Theorem 4.1 implies, with $C_h := 4n\kappa_h M$,

$$|f(\mathrm{Decode}(X)) - f(X)| \le C_h H_h(X), \qquad X = \mathcal{S}_K(Z). \tag{17}$$

When $Q(X) < 1/2$, the local recovery bound improves the right-hand side to $2\kappa_h M H_h(X)$.

A.5 Objective Guarantees for Trajectory Harvesting

Proposition A.5 (Harvested objective guarantee)**.** *For L trajectories of $T \ge 1$ iterations,*

$$f(\widehat{P}_{\mathrm{best}}) \le \min_{\substack{1 \le \ell \le L \\ 0 \le t < T}} \{ f(X_t^\ell) + C_h H_h(X_t^\ell) \}.$$

Proof of Proposition A.5. For every trajectory $1 \le \ell \le L$ and iteration $0 \le t < T$, let $\widehat{P}_t^\ell = \mathrm{Decode}(X_t^\ell)$. Trajectory harvesting retains the best objective encountered, and optional 2-opt refinement can only decrease it. Thus, by Eq. (17),

$$f(\widehat{P}_{\mathrm{best}}) \le f(\widehat{P}_t^\ell) \le f(X_t^\ell) + C_h H_h(X_t^\ell).$$

Taking the minimum over all candidate indices proves

$$f(\widehat{P}_{\mathrm{best}}) \le \min_{\substack{1 \le \ell \le L \\ 0 \le t < T}} \{ f(X_t^\ell) + C_h H_h(X_t^\ell) \}.$$

□

The bound applies to any finite collection of candidates and does not require stochastic independence of the trajectories.

Local accuracy consequence. Suppose an inspected candidate X_t^ℓ satisfies $f(X_t^\ell) \le v^\star + \delta$ and $Q(X_t^\ell) \le \eta < 1/2$, with $\delta, \eta \ge 0$. The local bound in Theorem 4.1 and the Lipschitz comparison above imply

$$\begin{aligned} f(\widehat{P}_{\mathrm{best}}) &\le f(P_G(X_t^\ell)) \\ &\le f(X_t^\ell) + M\|X_t^\ell - P_G(X_t^\ell)\|_1 \\ &\le v^\star + \delta + 2M\eta. \end{aligned}$$

Feasibility of $\widehat{P}_{\mathrm{best}}$ gives $f(\widehat{P}_{\mathrm{best}}) \ge v^\star$, proving Eq. (9).

A.6 Integrality Characterization and Birkhoff Error Bound

We begin with the basic characterization of permutation matrices induced by an admissible integrality violation.

Lemma A.6 (Exact integrality characterization)**.** *Let $h : [0,1] \to \mathbb{R}_+$ satisfy*

$$h(0) = h(1) = 0, \qquad h(x) > 0 \quad \textit{for all } x \in (0,1).$$

Then, for every $X \in \mathcal{B}_n$,

$$H_h(X) := \sum_{i,j} h(X_{ij}) = 0 \quad \Longleftrightarrow \quad X \in \mathcal{P}_n.$$

Proof. Since $X \in \mathcal{B}_n$, every entry satisfies

$$0 \le X_{ij} \le 1.$$

Because h is nonnegative,

$$H_h(X) = \sum_{i,j} h(X_{ij}) \ge 0.$$

If $X \in \mathcal{P}_n$, then every entry of X is either 0 or 1. Since $h(0) = h(1) = 0$, it follows immediately that

$$H_h(X) = 0.$$

Conversely, suppose $H_h(X) = 0$. Every summand is nonnegative, and therefore

$$h(X_{ij}) = 0 \qquad \forall i, j.$$

By assumption, the only zeros of h on $[0,1]$ are 0 and 1. Hence

$$X_{ij} \in \{0, 1\} \qquad \forall i, j.$$

Together with

$$X\mathbf{1} = \mathbf{1}, \qquad X^\top \mathbf{1} = \mathbf{1},$$

this implies that every row and every column contains exactly one unit entry. Thus X is a permutation matrix:

$$X \in \mathcal{P}_n.$$

□

The following error bound quantifies proximity to a permutation.

Lemma A.7 (Birkhoff integrality error bound)**.** *Let h be admissible. For every $X \in \mathcal{B}_n$, there exists $P_X \in \mathcal{P}_n$ such that*

$$\|X - P_X\|_1 \le 2Q(X) \le 2\kappa_h H_h(X), \qquad \mathrm{dist}_F^2(X, \mathcal{P}_n) \le Q(X) \le \kappa_h H_h(X).$$

Proof of Lemma A.7. Fix $X \in \mathcal{B}_n$. By the Birkhoff–von Neumann theorem, there exists a decomposition

$$X = \sum_{r=1}^{m} \theta_r P_r, \qquad P_r \in \mathcal{P}_n, \qquad \theta_r \ge 0, \qquad \sum_{r=1}^{m} \theta_r = 1.$$

Taking the inner product with X gives

$$\sum_{r=1}^{m} \theta_r \langle X, P_r \rangle = \left\langle X, \sum_{r=1}^{m} \theta_r P_r \right\rangle = \langle X, X \rangle = \|X\|_F^2.$$

Hence at least one permutation in the decomposition, denoted by P_X, satisfies

$$\langle X, P_X \rangle \geq \|X\|_F^2. \tag{18}$$

For any permutation matrix P, we have

$$\|X - P\|_1 = \sum_{P_{ij}=1} (1 - X_{ij}) + \sum_{P_{ij}=0} X_{ij}.$$

Since both X and P have total mass n,

$$\sum_{P_{ij}=1} (1 - X_{ij}) = n - \langle X, P \rangle,$$

and

$$\sum_{P_{ij}=0} X_{ij} = n - \langle X, P \rangle.$$

Therefore

$$\|X - P\|_1 = 2\left(n - \langle X, P \rangle\right). \tag{19}$$

Applying (19) to P_X and using (18),

$$\|X - P_X\|_1 \leq 2\left(n - \|X\|_F^2\right) = 2\sum_{i,j} X_{ij}(1 - X_{ij}),$$

where we used

$$\sum_{i,j} X_{ij} = n.$$

For the Frobenius distance,

$$\begin{aligned} \|X - P_X\|_F^2 &= \|X\|_F^2 + \|P_X\|_F^2 - 2\langle X, P_X \rangle \\ &\leq \|X\|_F^2 + n - 2\|X\|_F^2 \\ &= n - \|X\|_F^2 \\ &= \sum_{i,j} X_{ij}(1 - X_{ij}). \end{aligned}$$

Since $P_X \in \mathcal{P}_n$,

$$\mathrm{dist}_F^2(X, \mathcal{P}_n) \leq \|X - P_X\|_F^2.$$

Finally, by the definition

$$\kappa_h = \sup_{x \in (0,1)} \frac{x(1-x)}{h(x)},$$

we have

$$x(1 - x) \leq \kappa_h h(x), \qquad x \in [0, 1],$$

where the endpoint cases follow by continuity. Summing entrywise gives

$$\sum_{i,j} X_{ij}(1 - X_{ij}) \leq \kappa_h H_h(X).$$

Combining the preceding inequalities proves

$$\|X - P_X\|_1 \leq 2\kappa_h H_h(X)$$

and

$$\mathrm{dist}_F^2(X, \mathcal{P}_n) \leq \kappa_h H_h(X).$$

□

A.7 Scalar Exact Penalization

We first record the Lipschitz property used in the proof. Define

$$L_1 := \sup_{X \in \mathcal{B}_n} \|\nabla f(X)\|_\infty.$$

Because $\mathcal{B}_n$ is compact and f is continuously differentiable, $L_1 < \infty$. Moreover, for all $X, Y \in \mathcal{B}_n$,

$$|f(X) - f(Y)| \le L_1 \|X - Y\|_1. \tag{20}$$

The following scalar result will be used to establish entrywise exactness in Appendix A.9.

Lemma A.8 (Scalar finite exactness)**.** *Let h be admissible and $v^\star := \min_{P \in \mathcal{P}_n} f(P)$. For every $X \in \mathcal{B}_n$ and $\rho \ge 0$,*

$$f(X) + \rho H_h(X) \ge v^\star + (\rho - 2\kappa_h L_1) H_h(X). \tag{21}$$

If $\rho > 2\kappa_h L_1$, the global minimizers of $f(X) + \rho H_h(X)$ over $\mathcal{B}_n$ coincide with the globally optimal QAP permutations.

Proof of Lemma A.8. Fix $X \in \mathcal{B}_n$, and let $P_X \in \mathcal{P}_n$ be supplied by Lemma A.7. By (20),

$$f(X) \ge f(P_X) - L_1 \|X - P_X\|_1.$$

Since $v^\star = \min_{P \in \mathcal{P}_n} f(P)$, we have $f(P_X) \ge v^\star$. Using Lemma A.7, $\|X - P_X\|_1 \le 2\kappa_h H_h(X)$, and therefore

$$f(X) \ge v^\star - 2\kappa_h L_1 H_h(X).$$

Adding $\rho H_h(X)$ to both sides gives

$$f(X) + \rho H_h(X) \ge v^\star + (\rho - 2\kappa_h L_1)\, H_h(X),$$

which proves (21).

Now suppose $\rho > 2\kappa_h L_1$. Let $P^\star$ be a globally optimal QAP solution. Because $P^\star$ is a permutation,

$$H_h(P^\star) = 0,$$

and hence

$$f(P^\star) + \rho H_h(P^\star) = v^\star.$$

Thus the minimum value of the penalized problem is at most $v^\star$.

On the other hand, the lower bound above gives $f(X) + \rho H_h(X) \ge v^\star$ for every $X \in \mathcal{B}_n$. Consequently, the minimum penalized value is exactly $v^\star$.

Let $X^\star$ be any global minimizer of the penalized problem. Then

$$v^\star = f(X^\star) + \rho H_h(X^\star) \ge v^\star + (\rho - 2\kappa_h L_1)\, H_h(X^\star).$$

Since $\rho - 2\kappa_h L_1 > 0$ and $H_h(X^\star) \ge 0$, we must have $H_h(X^\star) = 0$. By Lemma A.6, $X^\star \in \mathcal{P}_n$. Moreover, $f(X^\star) = v^\star$, so $X^\star$ is a globally optimal solution of the original QAP.

Conversely, every globally optimal QAP solution $P^\star$ satisfies $H_h(P^\star) = 0$ and therefore attains the penalized value $v^\star$. Hence the two global minimizer sets coincide. □

A.8 Explicit Constants for the Two Integrality Functions

For the quadratic violation

$$h_{\mathrm{quad}}(x) = x(1 - x),$$

we immediately have

$$\frac{x(1-x)}{h_{\mathrm{quad}}(x)} = 1$$

for all $x \in (0, 1)$. Therefore

$$\kappa_{\mathrm{quad}} = 1.$$

For completeness, we next establish the sharp constant for the binary-entropy violation.

Lemma A.9 (Binary-entropy domination). *Let* $h_{\mathrm{ent}}(x) = -x\log x - (1-x)\log(1-x)$. *Then*

$$h_{\mathrm{ent}}(x) \geq 4(\log 2)x(1-x), \qquad x \in [0,1], \tag{22}$$

and the constant is sharp. Consequently, $\kappa_{\mathrm{ent}} = \frac{1}{4\log 2}$.

Proof. The claim is symmetric under $x \mapsto 1-x$, so write

$$x = \frac{1+s}{2}, \qquad s \in [0,1].$$

Then

$$x(1-x) = \frac{1-s^2}{4}.$$

Define

$$G(s) := h_{\mathrm{ent}}\left(\frac{1+s}{2}\right) - (\log 2)(1-s^2).$$

The desired inequality is equivalent to

$$G(s) \geq 0, \qquad s \in [0,1].$$

Direct calculation gives

$$G(0) = G(1) = 0,$$

and

$$G'(s) = 2(\log 2)s - \operatorname{arctanh}(s),$$

where

$$\operatorname{arctanh}(s) = \frac{1}{2}\log\frac{1+s}{1-s}.$$

Differentiating once more,

$$G''(s) = 2\log 2 - \frac{1}{1-s^2}.$$

The function $1/(1-s^2)$ is strictly increasing on $[0,1)$. Moreover,

$$G''(0) = 2\log 2 - 1 > 0,$$

whereas

$$G''(s) \to -\infty \qquad \text{as } s \to 1^-.$$

Hence G'' has exactly one zero on $(0,1)$. Therefore G' first increases and then decreases.

Since

$$G'(0) = 0$$

and

$$G'(s) \to -\infty \qquad \text{as } s \to 1^-,$$

there exists a unique $s^\star \in (0,1)$ such that

$$G'(s) > 0 \quad \text{for } s \in (0, s^\star),$$

and

$$G'(s) < 0 \quad \text{for } s \in (s^\star, 1).$$

Thus G increases from $G(0) = 0$ and subsequently decreases to $G(1) = 0$, implying

$$G(s) \geq 0 \qquad \forall s \in [0,1].$$

This proves (22).

Equality holds at $x = 1/2$, because

$$h_{\mathrm{ent}}(1/2) = \log 2$$

and

$$4(\log 2)(1/2)(1/2) = \log 2.$$

Hence

$$\sup_{x\in(0,1)} \frac{x(1-x)}{h_{\mathrm{ent}}(x)} = \frac{1}{4\log 2}.$$

□

A.9 Entrywise Exactness

We extend the scalar exact-penalty guarantee to entrywise coefficients.

Theorem A.10 (Entrywise finite exactness)**.** *Let h be admissible and $\Lambda \in \mathbb{R}_+^{n\times n}$. If $\rho(\Lambda) > 2\kappa_h L_1$, the global minimizers of $f(X) + \langle \Lambda, h(X)\rangle$ over $\mathcal{B}_n$ coincide with the globally optimal QAP permutations.*

Proof of Theorem A.10. Let $\rho = \rho(\Lambda)$ and $v^\star = \min_{P\in\mathcal{P}_n} f(P)$. For every $X \in \mathcal{B}_n$, nonnegativity of h gives

$$\begin{aligned} f(X) + \langle \Lambda, h(X)\rangle &\geq f(X) + \rho H_h(X) \\ &\geq v^\star + (\rho - 2\kappa_h L_1) H_h(X), \end{aligned}$$

where the second inequality follows from Lemma A.8. Under the hypothesis $\rho > 2\kappa_h L_1$, every fractional X has penalized value strictly greater than $v^\star$, whereas every globally optimal QAP permutation has zero violation and attains $v^\star$. Thus the minimum penalized value is $v^\star$, and every global minimizer must be a permutation with objective value $v^\star$. Conversely, every globally optimal QAP permutation attains the minimum penalized value, proving equality of the two solution sets. □

For $h_{\mathrm{quad}}(x) = x(1-x)$, $\kappa_{\mathrm{quad}} = 1$; for $h_{\mathrm{ent}}(x) = -x\log x - (1-x)\log(1-x)$, $\kappa_{\mathrm{ent}} = (4\log 2)^{-1}$ (Appendix A.8). Thus the sufficient entrywise thresholds on the Birkhoff polytope are $2L_1$ and $L_1/(2\log 2)$, respectively. For finite-Sinkhorn decoding, the corresponding constants $C_h = 4n\kappa_h M$ are $4nM$ and $nM/\log 2$.

A.10 Exact Penalization in the Finite Sinkhorn Latent Space

The objective comparison in Eq. (17) allows us to relate the penalized latent objective to the QAP optimum.

Proof of Theorem 4.2. Fix a finite coefficient matrix Λ with $\rho := \rho(\Lambda) > C_h$, and write $X = \mathcal{S}_K(Z)$, $P_G = \mathrm{Decode}(X)$, and $v^\star = \min_{P\in\mathcal{P}_n} f(P)$. Nonnegativity of h and Eq. (17) yield

$$\begin{aligned} J_\Lambda(Z) &= f(X) + \langle \Lambda, h(X)\rangle \\ &\geq f(P_G) - C_h H_h(X) + \rho H_h(X) \\ &= f(P_G) + (\rho - C_h) H_h(X) \\ &\geq v^\star + (\rho - C_h) H_h(X). \end{aligned}$$

Thus $\inf_Z J_\Lambda(Z) \geq v^\star$.

For the reverse inequality, choose an optimal permutation $P^\star$ and let $U = \mathbf{1}\mathbf{1}^\top/n$. For $0 < \delta < 1$, set $X_\delta = (1-\delta)P^\star + \delta U$ and $Z_\delta = \log X_\delta$, with the logarithm applied entrywise. The matrix X_δ is strictly positive and doubly stochastic, so both normalization operators leave it unchanged. For every fixed $K \geq 1$, we therefore have $\mathcal{S}_K(Z_\delta) = X_\delta$. As $\delta \downarrow 0$, continuity of f and h, together with $h(0) = h(1) = 0$, gives

$$J_\Lambda(Z_\delta) = f(X_\delta) + \langle \Lambda, h(X_\delta)\rangle \longrightarrow f(P^\star) = v^\star.$$

It follows that $\inf_Z J_\Lambda(Z) = v^\star$.

For finite Z, strict positivity and column stochasticity, with $n \geq 2$, imply $0 < X_{ij} < 1$. Hence $H_h(X) > 0$, and the first bound above gives $J_\Lambda(Z) > v^\star$ because $\rho > C_h$. The infimum is consequently not attained at any finite latent matrix.

Finally, suppose $J_\Lambda(Z) \leq v^\star + \varepsilon$. Combining this inequality with the lower bound gives

$$H_h(X) \leq \frac{\varepsilon}{\rho - C_h}, \qquad v^\star \leq f(P_G) \leq J_\Lambda(Z) - (\rho - C_h) H_h(X) \leq v^\star + \varepsilon.$$

This proves both the violation bound and the decoded objective gap. □

Harvested-output consequence of Theorem 4.2. Suppose that some candidate satisfies $\rho(\Lambda_t^\ell) > C_h$ and $J_{\Lambda_t^\ell}(Z_t^\ell) \le v^\star + \varepsilon$. Applying Theorem 4.2 with the coefficient matrix fixed at this particular iteration yields $f(\widehat{P}_t^\ell) \le v^\star + \varepsilon$. Since all decoded candidates are permutations,

$$v^\star \le f(\widehat{P}_{\mathrm{best}}) \le f(\widehat{P}_t^\ell) \le v^\star + \varepsilon.$$

The coefficient matrix may vary at all other iterations: only the conditions at the selected candidate are used. □

The near-optimality premise involves the unknown value $v^\star$; it is a theoretical error-transfer condition rather than a computable stopping certificate in the absence of an accompanying lower bound.

A.11 Fixed-Penalty Finite-Sinkhorn Stationarity

Fix $Y \in \mathbb{R}^{n\times n}$ and $K \ge 1$, and set $\phi_{Y,K}(Z) := f(\mathcal{S}_K(Z)) + \langle Y, g(\mathcal{S}_K(Z))\rangle$. This objective is lower bounded because f and g are continuous on the compact cube and Y is fixed. The following result describes primal updates with the coefficients held fixed.

Theorem A.11 (Fixed-penalty first-order stationarity)**.** *Suppose $\phi_{Y,K}$ is lower bounded and has an L_ϕ-Lipschitz continuous gradient on a set containing the iterates and the line segments between consecutive iterates, with $L_\phi > 0$. For the updates $Z^{t+1} = Z^t - \alpha\nabla\phi_{Y,K}(Z^t), \qquad 0 < \alpha < 2/L_\phi$, we have $\sum_{t=0}^{\infty} \|\nabla\phi_{Y,K}(Z^t)\|_F^2 < \infty$,*

$$\min_{0\le t<T} \|\nabla\phi_{Y,K}(Z^t)\|_F^2 \le \frac{\phi_{Y,K}(Z^0) - \inf_Z \phi_{Y,K}(Z)}{\alpha(1 - L_\phi\alpha/2)T}, \qquad T \ge 1.$$

If $\{Z^t\}$ is bounded, every accumulation point is a first-order stationary point of $\phi_{Y,K}$.

Proof of Theorem A.11. For brevity, write

$$\phi(Z) := \phi_{Y,K}(Z)$$

and let L_ϕ denote the Lipschitz constant of its gradient on a set containing the line segments between successive iterates.

The descent lemma gives

$$\phi(Z^{t+1}) \le \phi(Z^t) + \left\langle \nabla\phi(Z^t), Z^{t+1} - Z^t \right\rangle + \frac{L_\phi}{2}\|Z^{t+1} - Z^t\|_F^2.$$

Using

$$Z^{t+1} - Z^t = -\alpha\nabla\phi(Z^t),$$

we obtain

$$\begin{aligned}\phi(Z^{t+1}) &\le \phi(Z^t) - \alpha\|\nabla\phi(Z^t)\|_F^2 \\ &\quad + \frac{L_\phi\alpha^2}{2}\|\nabla\phi(Z^t)\|_F^2 \\ &= \phi(Z^t) - \alpha\left(1 - \frac{L_\phi\alpha}{2}\right)\|\nabla\phi(Z^t)\|_F^2. \end{aligned} \tag{23}$$

Because

$$0 < \alpha < \frac{2}{L_\phi},$$

the coefficient

$$c_\alpha := \alpha\left(1 - \frac{L_\phi\alpha}{2}\right)$$

is strictly positive.

Summing (23) from $t = 0$ to $T - 1$ yields

$$c_\alpha \sum_{t=0}^{T-1} \|\nabla\phi(Z^t)\|_F^2 \le \phi(Z^0) - \phi(Z^T).$$

Since ϕ is lower bounded,

$$\phi(Z^T) \geq \inf_Z \phi(Z),$$

and therefore

$$\sum_{t=0}^{T-1} \|\nabla\phi(Z^t)\|_F^2 \leq \frac{\phi(Z^0) - \inf_Z \phi(Z)}{\alpha\left(1 - \frac{L_\phi \alpha}{2}\right)}. \tag{24}$$

Letting $T \to \infty$ proves

$$\sum_{t=0}^{\infty} \|\nabla\phi(Z^t)\|_F^2 < \infty.$$

Furthermore,

$$T \min_{0\leq t<T} \|\nabla\phi(Z^t)\|_F^2 \leq \sum_{t=0}^{T-1} \|\nabla\phi(Z^t)\|_F^2.$$

Combining this with (24) gives

$$\min_{0\leq t<T} \|\nabla\phi(Z^t)\|_F^2 \leq \frac{\phi(Z^0) - \inf_Z \phi(Z)}{\alpha\left(1 - \frac{L_\phi \alpha}{2}\right) T}.$$

Finally, suppose $\{Z^t\}$ is bounded. It therefore has accumulation points. Let

$$Z^{t_k} \to \bar{Z}.$$

Since the squared gradient norms are summable,

$$\|\nabla\phi(Z^t)\|_F \to 0.$$

By continuity of $\nabla\phi$,

$$\nabla\phi(\bar{Z}) = \lim_{k\to\infty} \nabla\phi(Z^{t_k}) = 0.$$

Hence every accumulation point is a first-order stationary point. □

Remark on the entropy integrality function. For

$$h_{\text{ent}}(x) = -x\log x - (1-x)\log(1-x),$$

we have

$$h'_{\text{ent}}(x) = \log\frac{1-x}{x},$$

which becomes singular as $x \to 0$ or $x \to 1$. Accordingly, the Lipschitz-gradient assumption in Theorem A.11 is a local condition on the realized finite-Sinkhorn trajectory. It is satisfied, for example, on any bounded latent region whose finite-Sinkhorn images remain uniformly separated from 0 and 1.

A.12 Finite-Time Accumulation of Adaptive Coefficients

For the adaptive coefficients used by SRPD, the sign convention $\Lambda = -Y$ gives $\Lambda^{t+1} = \Lambda^t + \beta h(X^t)$. The coefficients record accumulated integrality violations, including when Λ^0 is negative because the algorithm initializes Y^0 positively. Define the total coefficient increase $A_T := \sum_{i,j}(\Lambda_{ij}^T - \Lambda_{ij}^0)$. The following finite-time relations require no boundedness assumption on the coefficients.

Proposition A.12 (Finite-time violation accounting)**.** *For $\beta > 0$ and $T \geq 1$, the adaptive recursion satisfies*

$$A_T = \beta \sum_{t=0}^{T-1} H_h(X^t), \qquad \min_{0\leq t<T} H_h(X^t) \leq \frac{A_T}{\beta T}.$$

If $X^t = \mathcal{S}_K(Z^t)$ with finite Z^t and $K \geq 1$, then $\min_{0\leq t<T} \|X^t - P_G(X^t)\|_1 \leq \frac{4n\kappa_h A_T}{\beta T}$.

Proof of Proposition A.12. The recursion $\Lambda^{t+1} = \Lambda^t + \beta h(X^t)$ implies, for each coordinate and each $T \geq 1$,

$$\Lambda_{ij}^T - \Lambda_{ij}^0 = \beta \sum_{t=0}^{T-1} h(X_{ij}^t). \tag{25}$$

Summing over i, j gives

$$A_T := \sum_{i,j} (\Lambda_{ij}^T - \Lambda_{ij}^0) = \beta \sum_{t=0}^{T-1} H_h(X^t).$$

The minimum is no larger than the average, so

$$\min_{0 \leq t < T} H_h(X^t) \leq \frac{1}{T} \sum_{t=0}^{T-1} H_h(X^t) = \frac{A_T}{\beta T}.$$

If $X^t = \mathcal{S}_K(Z^t)$, take $t_T \in \arg\min_{0 \leq t < T} H_h(X^t)$ and apply Theorem 4.1 to obtain

$$\begin{aligned} \min_{0 \leq t < T} \|X^t - \mathrm{Decode}(X^t)\|_1 &\leq \|X^{t_T} - \mathrm{Decode}(X^{t_T})\|_1 \\ &\leq 4n\kappa_h H_h(X^{t_T}) \leq \frac{4n\kappa_h A_T}{\beta T}. \end{aligned}$$

No sign condition on Λ^0 is needed. □

The coordinatewise identity also quantifies how persistent fractionality increases a penalty. Fix $0 < \delta < 1/2$ and define $m_\delta := \min_{x \in [\delta, 1-\delta]} h(x) > 0$, where positivity follows from continuity and admissibility. Let $N_{ij,T}(\delta) := \#\{0 \leq t < T : X_{ij}^t \in [\delta, 1-\delta]\}$. At each counted iteration, $h(X_{ij}^t) \geq m_\delta$; all other terms are nonnegative. Eq. (25) therefore gives

$$\Lambda_{ij}^T - \Lambda_{ij}^0 \geq \beta m_\delta N_{ij,T}(\delta).$$

In particular, a coordinate that visits this interval infinitely often has $\Lambda_{ij}^T \to +\infty$.

For completeness, an additional assumption $A_T = o(T)$ yields a best-candidate asymptotic consequence. With t_T selected as above, Proposition A.12 implies $H_h(X^{t_T}) \to 0$. For finite Sinkhorn outputs, Theorem 4.1 then gives

$$\|X^{t_T} - \mathrm{Decode}(X^{t_T})\|_1 \to 0, \qquad r_{\mathcal{B}}(X^{t_T}) \to 0.$$

Thus every accumulation point of these selected candidates is a permutation. The condition $A_T = o(T)$ is not implied by the recursion alone, and the conclusion concerns the selected candidates, not the last iterate. No bounded-dual assumption is used in the finite-time identities or bounds above.

B Experimental Datasets

This section describes the datasets used in Sections 5.1–5.2 and the graph-isomorphism experiments in Appendix C. Solver settings and hyperparameters are reported separately in Appendix D.

B.1 QAPLIB

QAPLIB (Burkard et al., 1997; ISA-QAP) contains application-derived instances, including circuit-board design and facility layout problems, as well as randomly generated instances. Our evaluation covers assignment sizes from $n = 12$ to $n = 256$. Results are grouped by the instance-family prefixes listed in Table 1, such as "bur", "chr", and "tai".

B.2 taiXXeYY

The taiXXeYY instances (Drezner et al., 2005) are structured QAP benchmarks, where XX specifies the assignment dimension and YY identifies the instance. Their recursive construction embeds

uniform submatrices in a larger block-diagonal structure. Small positive off-diagonal flows and large finite off-diagonal distances prevent trivial decomposition into independent blocks. This structure creates barriers for transposition-based local search, since exchanging facility blocks can require intermediate solutions with worse objective values.

We evaluate the first three instances, "e01"–"e03", at each dimension $n \in \{125, 175, 343, 729\}$. The resulting twelve instances are listed in Table 2; the larger dimensions assess scalability beyond the sizes used in the QAPLIB comparison.

B.3 Graph Isomorphism

We use four graph families from the `nauty and Traces` benchmark collection (McKay & Piperno, 2025). The evaluation set has the following composition, as summarized in Table 3:

- **Paley**: 53 instances with 5–491 vertices.
- **Lattice**: 27 instances with 16–900 vertices.
- **Latin**: 19 instances with 4–400 vertices.
- **cfi**: 16 instances with 200–500 vertices.

For each graph, we construct an isomorphic counterpart by applying a random permutation to its vertex labels. The task is to recover an adjacency-preserving bijection between the original and permuted graphs. Appendix C presents the residual objective, its QAP reformulation, and the experimental results.

C Graph Isomorphism Experiments

We evaluate graph isomorphism on four graph families from the `nauty and Traces` benchmark collection (McKay & Piperno, 2025), with up to 900 vertices (Appendix B.3). Given adjacency matrices $A_1, A_2 \in \mathbb{R}^{n\times n}$, we seek a permutation minimizing

$$\min_{P\in\mathcal{P}_n} \|PA_1 - A_2P\|_F^2. \tag{26}$$

The graphs are isomorphic if and only if the minimum is zero. For any $P \in \mathcal{P}_n$, expanding Eq. (26) gives

$$\begin{aligned}\|PA_1 - A_2P\|_F^2 &= \operatorname{tr}(A_1^\top P^\top PA_1) + \operatorname{tr}(P^\top A_2^\top A_2P)\\ &\quad - \operatorname{tr}(A_1^\top P^\top A_2P) - \operatorname{tr}(P^\top A_2^\top PA_1)\\ &= \|A_1\|_F^2 + \|A_2\|_F^2 - 2\operatorname{tr}(P^\top A_2^\top PA_1).\end{aligned}$$

The last equality uses $P^\top P = PP^\top = I$ and the equality of the two cross terms under transposition of the trace. Removing the constant term and dividing by two yields the equivalent QAP

$$\min_{P\in\mathcal{P}_n} f_{\mathrm{GI}}(P), \qquad f_{\mathrm{GI}}(P) := -\operatorname{tr}(P^\top A_2^\top PA_1). \tag{27}$$

By cyclic invariance of the trace, this objective has the form in Section 3.1 with $F = -A_2^\top$, $D = A_1$, and $C = 0$. This reduction is exact over permutation matrices; SRPD applies its soft-assignment parameterization to the resulting QAP objective.

Writing $v^\star_{\mathrm{GI}}$ for the optimal value of Eq. (27), the original minimum squared residual is $\|A_1\|_F^2 + \|A_2\|_F^2 + 2v^\star_{\mathrm{GI}}$. Consequently, the two graphs are isomorphic if and only if

$$v^\star_{\mathrm{GI}} = -\frac{1}{2}\big(\|A_1\|_F^2 + \|A_2\|_F^2\big).$$

We compare `SRPD` with the optimization-based baselines `Gurobi-Feas` (Gurobi Optimization, LLC, 2025) and `Boscia` (Hendrych et al., 2025), and report the specialized solver `Nauty` (McKay & Piperno, 2014) as a reference. Formulations and stopping rules are detailed in Appendix D.2.

A strict time limit of 3600 seconds is imposed for all methods.

Table 3: Graph-isomorphism results under a 3600-second time limit. For each family, we report the total instance count, the number solved by each method, and its mean runtime over successfully solved instances.

Graph (# Nodes)	# Ins.	Gurobi-Feas		Boscia		SRPD		Nauty	
		# Solved ↑	Time (s) ↓	# Solved ↑	Time (s) ↓	# Solved ↑	Time (s) ↓	# Solved ↑	Time (s) ↓
Paley (5 – 491)	53	14	77	27	498	**52**	8	53	0.007
Lattice (16 – 900)	27	26	134	7	676	**27**	67	27	0.02
Latin (4 – 400)	19	**18**	113	10	229	12	3	19	0.003
cfi (200 – 500)	16	1	6	0	–	**9**	28	16	0.1

Table 3 reports the number of instances solved within the time limit and the mean runtime over each method's successful instances. Across the 115 instances, `SRPD` solves 100, compared with 59 for `Gurobi-Feas` and 44 for `Boscia`. Among these optimization-based methods, `SRPD` solves the most instances on Paley ($52/53$), Lattice ($27/27$), and cfi ($9/16$). On Latin, `Gurobi-Feas` solves more instances ($18/19$) than `SRPD` ($12/19$) and `Boscia` ($10/19$).

The reported mean runtimes for `SRPD` on its successful instances are 8, 67, 3, and 28 seconds on Paley, Lattice, Latin, and cfi, respectively. The specialized `Nauty` solver solves all 115 instances, with family-average runtimes of at most 0.1 seconds. SRPD therefore provides greater aggregate instance coverage than the two optimization-based baselines, while `Nauty` remains the most effective solver for these graph-isomorphism benchmarks.

D Experimental Configurations

D.1 SRPD Hyperparameters

Table 4 reports the SRPD hyperparameters for each task. Here n is the assignment dimension, L is the number of parallel trajectories, K is the number of Sinkhorn normalization cycles, α and β are the primal and dual stepsizes, $\bar{Y}$ is the initial value of every entry of the dual matrix, and T is the iteration budget. GI denotes graph isomorphism. We use $K = 15$ row–column normalization cycles in all experiments.

We use the quadratic integrality function g_{quad} for QAPLIB, and the binary-entropy function g_{ent} for taiXXeYY and GI, as defined in Section 3.1.

A scale-dependent comparison motivates the entropy choice on larger instances. In a diffuse assignment, entries have a representative scale of $1/n$. Writing $h = -g$, expansion at this scale gives

$$h_{\mathrm{quad}}(1/n) = \frac{n-1}{n^2}, \qquad h_{\mathrm{ent}}(1/n) = \frac{\log n + 1}{n} + \mathcal{O}(n^{-2}).$$

Since Eq. (7) gives $|Y_{ij}^{t+1} - Y_{ij}^{t}| = \beta h(X_{ij}^{t})$, the entropy-based coefficient update at $X_{ij}^{t} = 1/n$ is asymptotically $\log n + 1$ times the quadratic update at the same dual stepsize. This relative amplification partially offsets the dilution of entrywise integrality feedback as the assignment dimension grows, providing a rationale for using entropy on larger tasks under the fixed-stepsize protocol below. This comparison characterizes coefficient-update magnitudes at the same diffuse state; by itself, it does not predict the effective primal search direction, optimization speed, or final discrete solution quality.

The stepsizes are fixed at $\alpha = \beta = 0.02$ across all experiments, without extensive stepsize tuning. The batch size L is chosen primarily according to available memory and computational resources, with fewer parallel trajectories for larger assignments. For taiXXeYY and GI, the initial dual value $\bar{Y}$ is calibrated on smaller instances and then applied to larger instances.

For taiXXeYY, we use an iteration budget of $T = 5000$. With $\bar{Y} = 150$, $\beta = 0.02$, and $h_{\mathrm{ent}}(x) \le \log 2$, Eq. (7) gives $Y_{ij}^{t} \ge 150 - 0.02t \log 2 \ge 150 - 100 \log 2 > 0$ for $0 \le t \le T$. These runs therefore use entropy regularization with adaptively decreasing positive coefficients. Feasible permutations are obtained by greedy decoding and selected through trajectory harvesting. Theorem 4.2 characterizes the distinct regime of sufficiently negative coefficients.

Table 4: Hyperparameter settings for different tasks.

Task	n	L	K	$g(x)$	α	β	$\bar{Y}$	T
QAPLIB	$n \leq 200$	2000	15	g_{quad}	0.02	0.02	1	1000
	$200 < n \leq 500$	1000						
	$n > 500$	200						
taiXXeYY	$n \leq 200$	2000	15	g_{ent}	0.02	0.02	150	5000
	$200 < n \leq 500$	1000						
	$n > 500$	200						
GI	$n \leq 200$	2000	15	g_{ent}	0.02	0.02	3	2000
	$200 < n \leq 500$	1000						
	$n > 500$	200						

D.2 Baseline Methods and Solver Settings

We summarize the comparison methods used in Sections 5.1–5.2 and Appendix C.

QAPLIB and taiXXeYY. The six methods below are compared on QAPLIB. For taiXXeYY, we compare with `Ro-TS`, `BMA`, and `PLMA`, using a $1,800$-second time limit per instance for all methods.

- `Gurobi` (Gurobi Optimization, LLC, 2025) is an exact optimization solver. For QAPLIB, we use version 12.0.3 with 8 threads and a time limit of 100 seconds.
- `Ro-TS` (Taillard, 1991b) is a tabu-search heuristic with dynamic tabu tenure to balance intensification and diversification. The authors provide source code (Taillard, 1991a).
- `BMA` (Benlic & Hao, 2015a) is a memetic algorithm combining population-based exploration with local improvement of offspring solutions. The authors provide source code (Benlic & Hao, 2015b).
- `PDBO` (Liu et al., 2025) reformulates binary optimization as a continuous minimax problem and applies simultaneous projected primal gradient descent and dual gradient ascent. Its coordinate-wise coefficients adapt according to integrality violations. For QAP, these updates take the form in Eq. (4), providing a direct baseline for SRPD.
- `SAWT` (Tan & Mu, 2024) is a reinforcement-learning method that uses a solution-aware Transformer to iteratively refine assignments without constructing a full association graph.
- `PLMA` (Pan et al., 2026) combines a pretrained permutation model with warm-started Markov chain Monte Carlo finetuning on test instances. An additive energy-based model supports sampling over permutations, while cross-graph attention represents facility–location interactions. Sampled permutations are refined through a 2-swap local improvement map.

Graph isomorphism. The two optimization-based baselines stop when an isomorphism is found; `Nauty` serves as a specialized reference solver.

- `Gurobi-Feas` formulates graph isomorphism as a mixed-integer linear feasibility problem solved by `Gurobi` (Gurobi Optimization, LLC, 2025). It terminates as soon as a valid isomorphism is found.
- `Boscia` (Hendrych et al., 2025) is a branch-and-bound framework for mixed-integer optimization whose node relaxations are solved using Frank–Wolfe methods. It terminates when an integer solution with objective value 0 is found.
- `Nauty` (McKay & Piperno, 2014) is a specialized graph-isomorphism tool based on canonical labeling and automorphism computation. It is reported as a reference rather than an optimization-based competitor.

All methods in the graph-isomorphism experiments are subject to the 3600-second time limit specified in Appendix C.