Boosting Generalization in Diffusion-Based Neural Combinatorial Solver via Energy-guided Sampling

Haoyu LEI[1], Kaiwen Zhou[2], Yinchuan Li[2], Zhitang Chen[2], Farzan Farnia[1]

[1]Computer Science & Engineering Department, The Chinese University of Hong Kong, Hong Kong SAR, China
[2]Noah's Ark Lab, Huawei Technologies Co., Ltd., Shenzhen, China

Abstract

Diffusion-based Neural Combinatorial Optimization (NCO) has demonstrated effectiveness in solving NP-complete (NPC) problems by learning discrete diffusion models for solution generation, eliminating hand-crafted domain knowledge. Despite their success, existing NCO methods face significant challenges in both cross-scale and cross-problem generalization, and high training costs compared to traditional solvers. While recent studies have introduced training-free guidance approaches that leverage pre-defined guidance functions for zero-shot conditional generation, such methodologies have not been extensively explored in combinatorial optimization. To bridge this gap, we propose a general energy-guided sampling framework during inference time that enhances both the cross-scale and cross-problem generalization capabilities of diffusion-based NCO solvers without requiring additional training. Our experimental results demonstrate that a diffusion solver, trained exclusively on the Traveling Salesman Problem (TSP), can achieve competitive zero-shot solution generation on TSP variants, such as Prize Collecting TSP (PCTSP), through energy-guided sampling across different problem scales.

Introduction

Combinatorial optimization (CO) problems are fundamental challenges across numerous domains, from logistics and supply chain management to network design and resource allocation. While traditional exact solvers and heuristic methods have been widely studied, they often struggle with scalability and require significant domain expertise to design problem-specific algorithms (Arora 1998; Gonzalez 2007).

Recent advances in deep learning have sparked interest in Neural Combinatorial Optimization (NCO), which aims to learn reusable solving strategies directly from data, eliminating the need for hand-crafted heuristics (Bengio, Lodi, and Prouvost 2021). Among various deep learning approaches, diffusion-based models (Ho, Jain, and Abbeel 2020; Song et al. 2020) have emerged as a particularly promising direction for solving combinatorial optimization problems. These models have demonstrated remarkable capabilities in learning complex solution distributions by adapting discrete diffusion processes to graph structures. Recent works like (Sun and Yang 2023; Li et al. 2024) have achieved state-of-the-art performance on classical problems such as the Traveling Salesman Problem (TSP), showcasing the potential of diffusion-based approaches in combinatorial optimization.

However, the practical applicability of existing NCO approaches is limited by several critical challenges. First, current models suffer from cross-scale generalization, with performance degrading significantly when applied to larger problem instances than those seen during training, especially for transformer-based solvers (Khalil et al. 2017; Kool, Van Hoof, and Welling 2018). Second, these models show limited cross-problem transfer capabilities, struggling to adapt to problem variants with modified objectives or additional constraints. While several studies have attempted to enhance learning-based solvers' generalization through approaches such as training additional networks (Wang et al. 2024) and fine-tuning (Lin et al. 2024), these methods require substantial computational costs for training separate models for each problem type and scale.

In parallel, recent advances in diffusion models, particularly in computer vision, have demonstrated the effectiveness of training-free guidance approaches for enhancing conditional generation (Bansal et al. 2023; Chung et al. 2022; Yu et al. 2023). These approaches leverage predefined guidance functions or pre-trained networks to enable zero-shot conditional generation without additional training overhead. Inspired by these developments, we explore the adaptation of energy-based guidance to address the generalization challenges in combinatorial optimization.

In this work, we propose an energy-guided sampling framework that enhances the generalization capabilities of diffusion-based NCO solvers without requiring additional training costs. By incorporating problem-specific objectives and constraints during inference time, this approach enables zero-shot cross-problem transfer while maintaining solution feasibility. Through experiments on the TSP and its variant Prize Collecting TSP (PCTSP), we numerically demonstrate that our framework achieves effective zero-shot transfer from simpler to more complex problem variants while maintaining consistent performance across different problem scales. Our work represents a significant step toward more flexible and generalizable diffusion-based combinatorial optimization solvers, potentially reducing the need for problem-specific model training while maintaining compet-

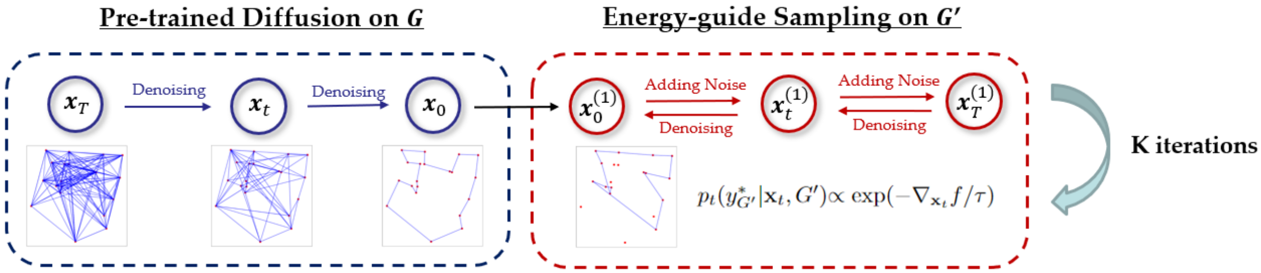

Figure 1: Overview of energy-guided sampling framework for achieving cross-problem generalization. Left: Pre-trained diffusion model performs denoising on original problem G (TSP). Right: Proposed energy-guided sampling on target problem G' (PCTSP) conducts K rewrite iterations of noise addition and guided denoising.

itive performance.

Related Works

Neural Network-based Combinatorial Solvers. Neural Combinatorial Optimization (NCO) approaches focus on leveraging neural networks to learn feasible solution distributions for combinatorial optimization problems (Bengio, Lodi, and Prouvost 2021; Zhang et al. 2023). Autoregressive construction solvers (Khalil et al. 2017; Kool, Van Hoof, and Welling 2018; Kwon et al. 2020; Kim, Park, and Park 2022; Hottung, Bhandari, and Tierney 2021) are built upon the success of transformer-based (Vaswani 2017) architectures in sequential generation tasks. Non-autoregressive construction solvers (Joshi, Laurent, and Bresson 2019; Fu, Qiu, and Zha 2021; Qiu, Sun, and Yang 2022; Wang et al. 2024; Sun and Yang 2023; Sanokowski, Hochreiter, and Lehner 2024) have also been proposed to learn high quality solution distributions.

Diffusion-based Generative Modeling. Recent advances in generative modeling have revolutionized various domains through diverse approaches, including Variational Autoencoders (VAE) (Kingma 2013), Generative Adversarial Networks (GAN) (Goodfellow et al. 2020), Diffusion models (Ho, Jain, and Abbeel 2020), and GFlowNet (Bengio et al. 2023). In particular, score-based diffusion models (Ho, Jain, and Abbeel 2020; Song et al. 2020; Sohl-Dickstein et al. 2015; Song and Ermon 2019; Dhariwal and Nichol 2021) have emerged as a powerful framework operating in continuous domains.

Beyond their state-of-the-art performance in traditional generative tasks, these models have shown remarkable potential in combinatorial optimization (CO). Pioneering work by (Sun and Yang 2023) established new state-of-the-art results for the Traveling Salesman Problem (TSP) by adapting discrete diffusion models (Austin et al. 2021) to graph structures. Building upon this foundation, (Li et al. 2024) enhanced the framework's performance through gradient search iterations during testing. (Sanokowski, Hochreiter, and Lehner 2024) proposed the first diffusion-based unsupervised learning framework.

Training-free Guidance for Diffusion Models. Conditional generation has emerged as a crucial component in real-world applications, enabling precise control over generated outputs. While traditional approaches such as classifier guidance (Dhariwal and Nichol 2021) and classifier-free guidance (Ho and Salimans 2022) have proven effective, they require substantial computational overhead due to additional training requirements for either the classifier or the diffusion model.

A promising alternative has emerged through training-free guidance methods (Bansal et al. 2023; Chung et al. 2022; Yu et al. 2023), which are guided by pre-trained networks or loss function. In the context of discrete diffusion models, this approach remained largely unexplored until (Li et al. 2024) pioneered the adaptation of loss-based guidance during inference, building upon the pre-trained discrete diffusion solver framework (Sun and Yang 2023). Our work demonstrates the significant potential of energy-guided sampling in enhancing the cross-problem generalization capabilities of diffusion-based NCO solvers.

Theoretical Analysis

Problem Formulation

Combinatorial optimization (CO) problems on graphs are fundamental to numerous real-world applications. Following recent advances (Sun and Yang 2023; Li et al. 2024), we address these problems by formalizing graph-based CO instances as follows.

We represent each problem instance as an undirected graph $G(V, E) \in \mathcal{G}$, where V and E denote the vertex and edge sets, respectively. This representation encompasses both vertex selection and edge selection problems, covering a broad spectrum of practical CO scenarios. For any instance $G \in \mathcal{G}$, we define a binary decision variable $\mathbf{x} \in \mathcal{X}_\mathcal{G}$, where $\mathcal{X}_\mathcal{G} = \{0, 1\}^N$ represents the feasible solution space. The optimization objective is to find the optimal solution $\mathbf{x}^*$ that minimizes a problem-specific objective function $f(\cdot; G) : \{0, 1\}^N \to \mathbb{R}$:

$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathcal{X}_\mathcal{G}}{\operatorname{argmin}} \ f(\mathbf{x}; G), \tag{1}$$

where the objective function decomposes into:

$$f(\mathbf{x}; G) = f_{\text{cost}}(\mathbf{x}; G) + \beta \cdot f_{\text{valid}}(\mathbf{x}; G). \tag{2}$$

Here $f_{\text{cost}}(\cdot; G)$ measures the solution quality, and $f_{\text{valid}}(\cdot; G)$ enforces problem-specific constraints through a penalty coefficient $\beta > 0$. The validity function returns 0 for feasible solutions and is strictly positive for infeasible ones.

As a concrete example, consider the classical Traveling Salesman Problem (TSP): given a complete graph G with edge weights, the objective is to find a minimum-weight Hamiltonian cycle. The decision variable $\mathbf{x} \in \{0,1\}^N$ encodes edge selections, where $f_{\text{cost}}(\cdot; G)$ measures the total tour length and $f_{\text{valid}}(\cdot; G)$ ensures the solution forms a valid Hamiltonian cycle. See (Sun and Yang 2023) for the detailed definition. For the Prize Collecting TSP (PCTSP) that we considered, each vertex has a prize $r_i > 0$ and penalty $p_i > 0$. The decision variables include edge selections $\mathbf{x}$ and vertex visits $\mathbf{y}$, where $f_{\text{cost}}(\mathbf{x}, \mathbf{y}; G) = \sum_{(i,j)\in E} d_{ij} x_{ij} + \sum_{i\in V\setminus S} p_i(1 - y_i) - \sum_{i \in V} r_i y_i$ balances the travel costs d_{ij}, the penalties for the unselected vertices $V \setminus S$ and the prizes, while $f_{\text{valid}}(\mathbf{x}, \mathbf{y}; G)$ ensures a valid tour through selected vertices.

Probabilistic Modeling for CO

To leverage recent advances in deep generative models, we reformulate the CO objective through an energy-based perspective (Lucas 2014). Specifically, we establish an energy function $\mathcal{E}(\cdot; G) := |y - f(\cdot; G)|$ that maps each solution to its corresponding energy state. This energy-based formulation naturally leads to a probabilistic framework through the Boltzmann distribution (LeCun et al. 2006):

$$p(y|\mathbf{x}; G) = \frac{\exp\big(-\frac{1}{\tau}\mathcal{E}(y, \mathbf{x}; G)\big)}{\mathcal{Z}}, \tag{3}$$

$$\text{where } \mathcal{Z} = \sum_{\mathbf{x}} \exp\Big(-\frac{1}{\tau}\mathcal{E}(y, \mathbf{x}; G)\Big), \tag{4}$$

where τ controls the temperature of the system and $\mathcal{Z}$ denotes the partition function that normalizes the distribution.

Recent works have demonstrated promising approaches to approximate this distribution using deep generative models by parameterizing a conditional distribution $p_\theta(\mathbf{x}|G)$ to minimize the energy function. Both supervised (Sun and Yang 2023; Li et al. 2024) and unsupervised (Sanokowski, Hochreiter, and Lehner 2024) learning paradigms have shown significant advances. Since our proposed training-free guidance mechanism is applicable to any pre-trained diffusion-based solver, we focus on the supervised learning framework in this work for ease of presentation.

Given a training set $\mathcal{G} = \{G_i\}_{i=1}^k$ of i.i.d. problem instances with their optimal solutions $\mathbf{x}$ and the corresponding optimal objective values y_G^*, we optimize the model parameters θ by maximizing the likelihood of the optimal solutions:

$$L(\theta) = \mathbb{E}_{G\sim\mathcal{G}}[-\log p_\theta(\mathbf{x}|y_G^*, G)]. \tag{5}$$

Discrete Diffusion Generation Modeling

We adopt a discrete diffusion framework to effectively sample optimal solutions from the learned distribution $p_\theta(\mathbf{x}|y^*, G)$. In contrast to continuous diffusion models that employ Gaussian noise, our discrete formulation is particularly well-suited for graph-based combinatorial optimization problems (Sun and Yang 2023; Li et al. 2024).

The diffusion process consists of two key components: a forward process that gradually corrupts the data, and a reverse process that learns to reconstruct the original distribution. The forward process $q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})$ maps clean data $\mathbf{x}_0 \sim q(\mathbf{x}_0|G)$ to a sequence of increasingly corrupted latent variables $\mathbf{x}_{1:T}$. The reverse process $p_\theta(\mathbf{x}_{0:T}|G) = p(\mathbf{x}_T)\prod_{t=1}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, G)$ learns to gradually denoise these latent variables to recover the original distribution. From a variational perspective, we optimize the model by minimizing an upper bound on the negative log-likelihood, where C is a constant:

$$\begin{aligned} L(\theta) &= \mathbb{E}_{G\sim\mathcal{G}}[-\log p_\theta(\mathbf{x}_0|G)] \\ &\leq \sum_{t=2}^T \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)}\Big[D_{KL}[q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)||p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, G)] \\ &\quad - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1, G)\Big] + C. \end{aligned} \tag{6}$$

For discrete state spaces, we define the forward process using a categorical distribution:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \text{Cat}(\mathbf{x}_t; \mathbf{p} = \widetilde{\mathbf{x}}_{t-1}\mathbf{Q}_t), \tag{7}$$

where $\widetilde{\mathbf{x}}_t \in \{0,1\}^{N\times 2}$ represents the one-hot encoding of $\mathbf{x}_t \in \{0,1\}^N$. The forward transition matrix $\mathbf{Q}_t$ is defined as:

$$\mathbf{Q}_t = \begin{bmatrix} (1-\beta_t) & \beta_t \\ \beta_t & (1-\beta_t) \end{bmatrix}, \quad \beta_t \in [0,1], \tag{8}$$

where $[\mathbf{Q}_t]_{ij}$ denotes the state transition probability from state i to state j. The t-step marginal distribution and posterior can be derived as:

$$\begin{aligned} q(\mathbf{x}_t|\mathbf{x}_0) &= \text{Cat}(\mathbf{x}_t; \mathbf{p} = \widetilde{\mathbf{x}}_0\overline{\mathbf{Q}}_t), \\ q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= \text{Cat}\left(\mathbf{x}_{t-1}; \mathbf{p} = \frac{\widetilde{\mathbf{x}}_t\mathbf{Q}_t^\top \odot \widetilde{\mathbf{x}}_0\overline{\mathbf{Q}}_{t-1}}{\widetilde{\mathbf{x}}_0\overline{\mathbf{Q}}_t\widetilde{\mathbf{x}}_t^\top}\right), \end{aligned} \tag{9}$$

where $\overline{\mathbf{Q}}_t = \mathbf{Q}_1\mathbf{Q}_2\ldots\mathbf{Q}_t$ and $\odot$ denotes element-wise multiplication.

To capture the structural properties of CO problems, we employ an anisotropic graph neural network architecture (Joshi, Laurent, and Bresson 2019). For a given instance G, the network learns to predict the clean data distribution $p_\theta(\widetilde{\mathbf{x}}_0|\mathbf{x}_t, G)$. Taking TSP as an example, where G encodes the 2D Euclidean coordinates of vertices, the network outputs a probability matrix $p_\theta(\widetilde{\mathbf{x}}_0|\mathbf{x}_t, G) \in [0,1]^{N\times 2}$. This matrix parameterizes N independent Bernoulli distributions, each corresponding to a binary decision variable in $\widetilde{\mathbf{x}}_0$. The reverse process during sampling follows:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, G) = \sum_{\widetilde{\mathbf{x}}_0} q(\mathbf{x}_{t-1}|\mathbf{x}_t, \widetilde{\mathbf{x}}_0)p_\theta(\widetilde{\mathbf{x}}_0|\mathbf{x}_t, G). \tag{10}$$

Energy-guided Sampling for Cross-problem

While training-free guidance has been extensively studied in computer vision (Bansal et al. 2023; Chung et al. 2022;

Yu et al. 2023), its application to combinatorial optimization problems has only recently emerged (Li et al. 2024). We extend this approach by introducing energy-based training-free guidance during inference, enabling flexible incorporation of additional constraints into pre-trained diffusion-based CO solvers and enhancing their cross-problem generalization capabilities.

Let $\mathcal{G}' = \{G'_i\}_{i=1}^n$ denote a set of test instances representing variants of the original training problems, such as problems with additional constraints or multiple objectives. For a new instance G' with its optimal solution pair $(\mathbf{x}, y^*_{G'})$, we need to estimate the new reverse process $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, y^*_{G'}, G')$ according to (10). Following the score estimation perspective of diffusion processes (Song et al. 2020), we decompose the conditional score function at time step t into two components:

$$\begin{aligned}\nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t|y^*_{G'}, G') = \nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t|G') \\ + \nabla_{\mathbf{x}_t} \log p_t(y^*_{G'}|\mathbf{x}_t, G').\end{aligned} \tag{11}$$

The first term $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t|G')$ is estimated using the pre-trained diffusion model. Since G' represents an extension of the original problem G with additional constraints or objectives, we assume that the solution distribution of $\mathbf{x}_t$ for G' is encompassed within the distribution learned for G. For instance, in the case of PCTSP, an optimal solution $\mathbf{x}_t$ can be viewed as an optimal TSP solution on the subgraph consisting of selected vertices. However, while this ensures solution feasibility, it does not guarantee optimality with respect to the modified objectives in G'. To address this limitation, we estimate the second term $\nabla_{\mathbf{x}_t} \log p_t(y^*_{G'}|\mathbf{x}_t, G')$ using an energy function that specifically accounts for the additional objectives and constraints of the variant problem:

$$\nabla_{\mathbf{x}_t} \log p_t(y^*_{G'}|\mathbf{x}_t, G') \propto -\nabla_{\mathbf{x}_t} \mathcal{E}(y^*_{G'}, \mathbf{x}_0(\mathbf{x}_t); G'), \tag{12}$$

where $\mathcal{E}(y^*_{G'}, \mathbf{x}_t; G') = |y^*_{G'} - f(\widetilde{\mathbf{x}}_0(\mathbf{x}_t); G')|$ measures the energy between the optimal value and the predicted solution. Here, $\widetilde{\mathbf{x}}_0(\mathbf{x}_t)$ represents the predicted clean sample from the current noisy state $\mathbf{x}_t$. To overcome this issue, we parameterize the model outputs as $[p_\theta(\widetilde{\mathbf{x}}_0|\mathbf{x}_t)]_i = (1-[\mathbb{E}\widetilde{\mathbf{x}}_0]_i, [\mathbb{E}\widetilde{\mathbf{x}}_0]_i)$, representing the logits of N independent Bernoulli samples, and estimate $\widetilde{\mathbf{x}}_0(\mathbf{x}_t) = \mathbb{E}_{\widetilde{\mathbf{x}}_0 \sim p_\theta(\widetilde{\mathbf{x}}_0|\mathbf{x}_t)}[\widetilde{\mathbf{x}}_0]$. Combining equations (11) and (12), we derive an energy-guided reverse sampling process:

$$\begin{aligned}&p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, y^*_{G'}, G') \propto p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t, G')p(y^*_{G'}|\mathbf{x}_t, G'), \\ &p(y^*_{G'}|\mathbf{x}_t, G') = \exp\left(-\frac{1}{\tau}\nabla_{\mathbf{x}_t} f\left(\widetilde{\mathbf{x}}_0(\mathbf{x}_t); G'\right)\right).\end{aligned} \tag{13}$$

Proposed Approach

Pre-trained Diffusion for Heatmap Generation. Building upon our theoretical analysis of graph-based discrete diffusion models, we employ these models to generate transition matrices for solving combinatorial optimization problems, where each matrix element encodes the selection probability of nodes or edges as a heatmap representation. The pre-trained diffusion model operates by progressively denoising a randomly perturbed graph structure to generate these probability heatmaps, followed by a greedy decoding strategy that constructs feasible solutions by iteratively selecting nodes or edges with the highest probabilities until reaching termination conditions (e.g., forming a complete cycle in TSP applications).

Algorithm 1: Energy-guided Diffusion Sampling for Cross-problem Transfer

Input:
- p_θ: Pre-trained diffusion model
- G': Target problem instance
- T: Number of diffusion steps
- τ: Energy guidance temperature
- K: Number of re-inference iterations

Output: Optimal solution $\mathbf{t}_K$ for problem instance G'

```
Initialize x_T with random binary values;
for k = 1 to K do
    Initialize x_T with previous best solution t_{k-1}
    for t = T to 1 do
        Compute p_θ(x_t | G') from pre-trained model;
        Compute energy gradient: ∇_{x_t} f(x̃_0(x_t); G');
        Compute p_t(y*_{G'} | x_t, G') ∝ exp(−∇_{x_t} f/τ);
        Compute guided posterior: p_θ(x_{t−1} | x_t, y*_{G'}, G') ∝
            p_θ(x_{t−1} | x_t, G') p_t(y*_{G'} | x_t, G');
        Update next state with Bernoulli Sampling:
            x_{t−1} ∼ Cat (x_{t−1}; p_θ(x_{t−1} | x_t, y*_{G'}, G'));
    end for
    Decode x_0 to the best solution t_k;
end for
return t_K
```

Energy-guided Sampling. We propose a conditional energy-guided sampling framework that enhances cross-scale and cross-problem generalization of pre-trained diffusion models without requiring additional training, leveraging DDIM (Song, Meng, and Ermon 2020) to accelerate the sampling process to 10 steps. Our initial experiments reveal that single-round sampling, despite utilizing energy function gradients for directional guidance, yields suboptimal cross-problem performance. To address this limitation, we adopt the rewrite technique from (Li et al. 2024), where we initialize each round by adding noise to the previous round's best solution and iteratively apply energy-guided sampling in Algo. 1. This iterative process establishes a natural trade-off between solution quality and computational efficiency, which we systematically analyze in our experimental evaluation.

Numerical Results

Dataset. We evaluate our approach on two classical NP-complete combinatorial optimization problems: the Traveling Salesman Problem (TSP) and its variant, the Prize Collecting Traveling Salesman Problem (PCTSP).

- **Traveling Salesman Problem (TSP)** requires finding the minimal-length Hamiltonian cycle in a complete graph, where the salesman must visit each city exactly once before returning to the starting point.

	Method	PCTSP-20		PCTSP-50		PCTSP-100				
		Gap ↓	Time ↓	Gap ↓	Time ↓	Gap ↓	Time ↓	Avg Gap ↓	Training-based	Training Time ↓
OR	Gurobi	**0.00%**	3.10s	—	—	—	—	—	—	—
OR	OR-Tools	2.13%	12.31s	4.85%	2.02m	10.33%	5.84m	5.77%	—	—
OR	ILS (C++)	1.07%	2.13s	**0.00%**	18.30s	**0.00%**	56.11s	0.36%	—	—
OR	ILS (Python 10x)*	63.23%	3.05s	148.05%	4.70s	209.78%	5.27s	140.35%	—	—
Auto-reg*	AM (Greedy)	2.88%	0.02s	17.95%	0.06s	29.24%	0.14s	16.69%	✓	3.5 days
Auto-reg*	AM (Sampling)	2.54%	2.43s	14.58%	7.08s	22.20%	15.13s	13.11%	✓	3.5 days
Auto-reg*	MDAM (Greedy)	11.76%	41.10s	24.73%	1.31m	30.07%	1.96m	22.19%	✓	4.3 days
Auto-reg*	MDAM (Beam Search)	5.88%	2.70m	18.81%	4.77m	26.09%	6.97m	16.93%	✓	4.3 days
Auto-reg*	AM (ASP)	12.05%	0.03s	10.34%	0.08s	**11.56%**	0.18s	11.32%	✓	1.1 days
Auto-reg*	AM-FT (Sampling)	**1.02%**	2.51s	14.11%	8.02s	25.19%	18.21s	13.44%	✓	4.9 days
Diff.	DIFUSCO (TSP)	19.21%	1.04s	18.61%	1.69s	43.42%	2.34s	27.08%	✗	1.5 days
Diff.	**DIF-guide (Ours)**	4.83%	4.58s	**7.58%**	6.37s	18.1%	10.79s	**10.17%**	✗	**0 days**

Table 1: Comprehensive evaluation of cross-scale generalization capabilities across different solver categories on PCTSP instances. Comparison includes exact solvers (Gurobi), OR-based heuristics (OR-Tools, ILS), autoregressive models (AM, MDAM, AM-ASP, AM-FT), and diffusion-based approaches (DIFUSCO, DIF-guide). Performance metrics include optimality gap, inference time, and training time. The best results marked with $*$ are reported from (Wang et al. 2024). The proposed DIF-guide achieves competitive performance while requiring no training.

- **Prize Collecting TSP (PCTSP)** (Balas 1989) extends the classical TSP by introducing node-specific prizes and penalties. The objective is to optimize a trade-off between minimizing tour length and unvisited node penalties while ensuring collected prizes exceed a predefined threshold. This formulation creates a more complex optimization landscape where node visitation decisions must balance multiple competing factors.

Method	PCTSP-20			PCTSP-50			PCTSP-100		
	Cost	Gap	Time	Cost	Gap	Time	Cost	Gap	Time
DIFUSCO	3.78	19.21%	**1.04s**	5.20	15.97%	**1.35s**	8.14	35.61%	**2.01s**
DIF-Guide (Ours)	**3.32**	**4.83%**	5.58s	**4.69**	**3.51%**	7.51s	**6.67**	**13.32%**	12.20s

Table 2: Zero-shot cross-problem transfer performance comparison between baseline DIFUSCO and our DIF-Guide approach on PCTSP instances. Results show solution cost, optimality gap, and computation time across three problem scales (20, 50, and 100 nodes). The energy-guided sampling uses 50 iterations with greedy decoding strategy.

Evaluation. Following (Kool, Van Hoof, and Welling 2018), we generate 1000 test instances for each problem scale: PCTSP-20, PCTSP-50, and PCTSP-100, where the numbers denote the node counts. We evaluate model performance using two primary metrics: average solution cost and optimality gap relative to the exact solution. Additionally, we measure computational efficiency through total training time and per-instance inference time.

Baselines. We compare our approach against multiple baseline categories: (1) Exact solver: Gurobi; (2) OR-based heuristics: OR-Tools and Iterated Local Search (ILS); (3) Learning-based methods: autoregressive models including AM (Kool, Van Hoof, and Welling 2018), MDAM (Xin et al. 2021), AM-ASP (Wang et al. 2024), and AM-FT (Lin et al. 2024), as well as the diffusion-based solver DIFUSCO (Sun and Yang 2023). Our proposed DIF-guide utilizes DIFUSCO's TSP-trained checkpoint with a PCTSP-specific energy function for energy-guided sampling. All experiments are conducted on a single Tesla V100 GPU.

First, we demonstrate the effectiveness of our energy-guided sampling framework in cross-problem transfer. Table 2 reveals substantial zero-shot performance improvement when applying our method to the pre-trained DIFUSCO model on TSP. The enhancements are consistent across all PCTSP scales, both in solution quality and optimality gap.

The cross-problem transfer capability of energy-guided sampling improves progressively through iterative rewriting rounds, as we initialize each round's noisy data $\mathbf{x}_T$ using the previous best solution. Figure 2 illustrates how solutions gradually adapt to new instances as rewrite rounds increase. While this iterative sampling increases inference time, our approach maintains computational efficiency and scalability compared to OR solvers while eliminating the need for problem-specific training.

Following (Wang et al. 2024), we evaluate our method's generalization capabilities across both problem scales and problem types. For cross-scale evaluation on PCTSP, we select the best-performing base model across all scales. Table 1 reveals that while existing autoregressive approaches struggle with cross-scale generalization due to their sequential generation scheme, our DIF-guide framework demonstrates competitive performance in cross-scale generalization, while maintaining zero-shot solution generation without any additional training. The framework's ability to adapt pre-trained diffusion models to new problem variants without additional training represents a significant advancement in CO applications.

Conclusions

In this work, we proposed an energy-guided sampling framework that enables zero-shot cross-problem generaliza-

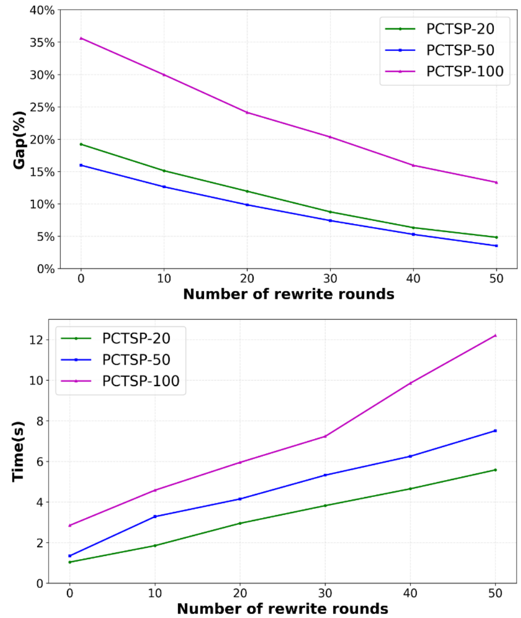

Figure 2: Trade-off between performance and inference time of energy-guided sampling with respect to rewriting rounds.

tion for diffusion-based combinatorial optimization solvers. By introducing an energy-based guidance mechanism during inference, our approach transfers pre-trained diffusion models to solve problem variants without additional training. Through numerical experiments on TSP and PCTSP, we demonstrated that our framework achieves competitive performance compared to other learning-based methods across different problem scales.

References

Arora, S. 1998. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *Journal of the ACM (JACM)*, 45(5): 753–782.

Austin, J.; Johnson, D. D.; Ho, J.; Tarlow, D.; and Van Den Berg, R. 2021. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34: 17981–17993.

Balas, E. 1989. The prize collecting traveling salesman problem. *Networks*, 19(6): 621–636.

Bansal, A.; Chu, H.-M.; Schwarzschild, A.; Sengupta, S.; Goldblum, M.; Geiping, J.; and Goldstein, T. 2023. Universal guidance for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 843–852.

Bengio, Y.; Lahlou, S.; Deleu, T.; Hu, E. J.; Tiwari, M.; and Bengio, E. 2023. Gflownet foundations. *The Journal of Machine Learning Research*, 24(1): 10006–10060.

Bengio, Y.; Lodi, A.; and Prouvost, A. 2021. Machine learning for combinatorial optimization: a methodological tour d'horizon. *European Journal of Operational Research*, 290(2): 405–421.

Chung, H.; Kim, J.; Mccann, M. T.; Klasky, M. L.; and Ye, J. C. 2022. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34: 8780–8794.

Fu, Z.-H.; Qiu, K.-B.; and Zha, H. 2021. Generalize a small pre-trained model to arbitrarily large tsp instances. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 7474–7482.

Gonzalez, T. F. 2007. *Handbook of approximation algorithms and metaheuristics*. Chapman and Hall/CRC.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11): 139–144.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Ho, J.; and Salimans, T. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Hottung, A.; Bhandari, B.; and Tierney, K. 2021. Learning a latent search space for routing problems using variational autoencoders. In *International Conference on Learning Representations*.

Joshi, C. K.; Laurent, T.; and Bresson, X. 2019. An efficient graph convolutional network technique for the travelling salesman problem. *arXiv preprint arXiv:1906.01227*.

Khalil, E.; Dai, H.; Zhang, Y.; Dilkina, B.; and Song, L. 2017. Learning combinatorial optimization algorithms over graphs. *Advances in neural information processing systems*, 30.

Kim, M.; Park, J.; and Park, J. 2022. Sym-nco: Leveraging symmetricity for neural combinatorial optimization. *Advances in Neural Information Processing Systems*, 35: 1936–1949.

Kingma, D. P. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kool, W.; Van Hoof, H.; and Welling, M. 2018. Attention, learn to solve routing problems! *arXiv preprint arXiv:1803.08475*.

Kwon, Y.-D.; Choo, J.; Kim, B.; Yoon, I.; Gwon, Y.; and Min, S. 2020. Pomo: Policy optimization with multiple optima for reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 21188–21198.

LeCun, Y.; Chopra, S.; Hadsell, R.; Ranzato, M.; Huang, F.; et al. 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Li, Y.; Guo, J.; Wang, R.; and Yan, J. 2024. From distribution learning in training to gradient search in testing for combinatorial optimization. *Advances in Neural Information Processing Systems*, 36.

Lin, Z.; Wu, Y.; Zhou, B.; Cao, Z.; Song, W.; Zhang, Y.; and Jayavelu, S. 2024. Cross-problem learning for solving vehicle routing problems. *arXiv preprint arXiv:2404.11677*.

Lucas, A. 2014. Ising formulations of many NP problems. *Frontiers in physics*, 2: 5.

Qiu, R.; Sun, Z.; and Yang, Y. 2022. Dimes: A differentiable meta solver for combinatorial optimization problems. *Advances in Neural Information Processing Systems*, 35: 25531–25546.

Sanokowski, S.; Hochreiter, S.; and Lehner, S. 2024. A Diffusion Model Framework for Unsupervised Neural Combinatorial Optimization. *arXiv preprint arXiv:2406.01661*.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Y.; and Ermon, S. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Sun, Z.; and Yang, Y. 2023. Difusco: Graph-based diffusion solvers for combinatorial optimization. *Advances in Neural Information Processing Systems*, 36: 3706–3731.

Vaswani, A. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Wang, C.; Yu, Z.; McAleer, S.; Yu, T.; and Yang, Y. 2024. Asp: Learn a universal neural solver! *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Xin, L.; Song, W.; Cao, Z.; and Zhang, J. 2021. Multi-decoder attention model with embedding glimpse for solving vehicle routing problems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 12042–12049.

Yu, J.; Wang, Y.; Zhao, C.; Ghanem, B.; and Zhang, J. 2023. Freedom: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 23174–23184.

Zhang, J.; Liu, C.; Li, X.; Zhen, H.-L.; Yuan, M.; Li, Y.; and Yan, J. 2023. A survey for solving mixed integer programming via machine learning. *Neurocomputing*, 519: 205–217.