Neural Assortment Optimization

Zhen Yang* Jiayou Liang† Zhi Wang‡ Yang An§ Rui Gao* Shuang Li◇

Abstract

Assortment optimization selects a subset of items to maximize expected revenue under a discrete choice model and is widely used in revenue management and online platforms. Its combinatorial nature creates a practical tension among generality, scalability, and provable guarantees: model-specific algorithms can be strong when their structural assumptions hold, but are hard to adapt across choice models; broadly applicable heuristics and learning-based methods scale to large instances but often come without near-optimality guarantees. We propose *Neural Assortment Optimization* (NAO), a choice-model-agnostic optimization framework that only requires an oracle that evaluates the expected revenue of a candidate assortment. NAO follows an "extend-particle search-round" pipeline: it constructs a tight continuous extension of the discrete objective, then optimizes a population of interacting particles using noisy subgradient descent, and finally rounds candidates back to discrete assortments. The particle objective admits a simple neural-network interpretation: each particle can be viewed as a hidden neuron, and an entropic-risk pooling layer emphasizes the best-performing candidates during optimization. We establish global convergence and near-optimality guarantees under mild conditions, and extend the method to cardinality constraints via a capacity-aware rolling-window construction. Experiments on challenging Mixed ulti-nomial Logit and Nested Logit benchmarks show that NAO achieves near-optimal revenue with strong computational efficiency, consistently outperforming competitive heuristics and neural baselines and scaling well to large instances.

1 Introduction

Assortment optimization is the problem of selecting a subset of items to offer to a customer in order to maximize an objective such as expected revenue, engagement, or welfare. It is a fundamental decision problem in revenue management [2, 5, 11, 17, 64], and is widely used in online retail, digital advertising, and recommendation systems [30, 54], where a platform must decide which products (or content) to display from a large catalog. In these settings, customer behavior is commonly described by a discrete choice model [63], which specifies, for any offered assortment, the probability that each offered item is chosen (as well as the probability of no purchase). The resulting expected-revenue function is then a set function defined over subsets of items. As platforms increasingly rely on data-driven decision making and operate at ever larger scales, the ability to compute high-quality assortments quickly and reliably has become a basic computational requirement.

From an optimization standpoint, assortment optimization is challenging because the decision is combinatorial while the objective is defined on discrete sets. This makes the problem inherently non-differentiable with respect to the decision variables, limiting the direct use of the gradient-based optimization methods that have become standard in large-scale continuous problems. The computational burden becomes more pronounced when the choice model moves beyond simple forms in order to capture richer substitution patterns

*The University of Texas at Austin; †The University of Chicago; ‡The University of Toronto; §The University of Toronto; ◇The Chinese University of Hong Kong, Shenzhen. Email addresses: zhen.yang@mccombs.utexas.edu; jiayouliang@uchicago.edu; zhiss.wang@utoronto.ca; alanmilleran@gmail.com; rui.gao@mccombs.utexas.edu; lishuang@cuhk.edu.cn.

and customer heterogeneity. In such cases, one often faces a tension between using a more realistic model and maintaining computational tractability at the scale of modern applications.

A large literature has developed effective methods in important special cases, but existing approaches often face trade-offs among generality, guarantees, and scalability. One class of methods is model-specific (e.g., Blanchet et al. [6], Désir et al. [17], Gallego et al. [24], Kunnumkal [37], Talluri & Van Ryzin [61]): these approaches exploit the special structure of a particular choice model to obtain exact or carefully designed approximate algorithms, and can be highly effective when their assumptions match the application. Their main limitation is portability: when the choice model changes, the optimization method often requires substantial new development. A second class of methods is heuristic and is designed primarily for scalability, including revenue-ordered policies [5] and local-search procedures [34]. These methods are broadly applicable and can perform well in practice, but their solution quality can be sensitive to the instance structure, and they often provide limited worst-case guarantees. A third, more recent, class of methods uses first-order optimization and learning-based components [10, 28, 41] to search directly over assortments. These methods can be attractive for large-scale instances and have shown promising empirical performance, but in many cases, the available theory provides only partial guidance on solution quality.

This paper proposes a framework that is designed to reduce these trade-offs. Our goal is to develop a general approach that is *choice-model agnostic*, admits *near-optimal guarantees* on the quality of the returned assortment, and remains *efficient and scalable* on practical instances. The model-agnostic goal is important in revenue management practice: choice models differ across applications and are often fit and updated in a modular way. Requiring a specialized optimizer for each model family can therefore be a major barrier to adoption. Motivated by this, we adopt an oracle viewpoint: the optimization method only needs the ability to evaluate the expected revenue of a candidate assortment under the chosen choice model, for example through closed-form evaluation or simulation. Under this interface, the same optimization procedure can be applied across different choice models without relying on model-specific reformulations.

Our method, *Neural Assortment Optimization* (`NAO`), follows a three-stage pipeline that can be summarized as *"extend–particle search–round"*. The first stage constructs a continuous extension of the discrete objective. The second stage performs a particle-based first-order search over the continuous domain, allowing the method to explore multiple candidate solutions in parallel and to explore the nonconvex landscape. The third stage maps the continuous solutions back to discrete assortments through a simple rounding rule and returns the best recovered set. This procedure has a natural neural-network interpretation, where each particle can be viewed as a neuron in a wide neural network.

In the extension stage, we use the Lovász extension, a standard construction in combinatorial optimization that maps a set function to a continuous function defined on the unit hypercube. Among many possible continuous surrogates, this choice has two features that are central to our setting. First, it is exact in optimal value: optimizing the continuous extension attains the same best value as optimizing the original discrete objective over all assortments. This makes the extension a principled formulation rather than a loose relaxation. Second, the extension is piecewise affine, respects the potentially local convexity of the original objective, and supports efficient gradient-based oracles built from marginal changes of the original discrete objective along a nested sequence of assortments. This enables scalable subgradient-based optimization without requiring analytic gradients of the underlying choice model.

The particle search stage addresses a central difficulty of assortment optimization under general choice models: even with an exact continuous extension, the resulting objective is typically nonconvex. In such landscapes, a single run of first-order methods can be sensitive to initialization and may converge to solutions that are far from globally optimal. `NAO` mitigates this by maintaining a system of candidate particles and optimizing their entropic risk, which is a smooth objective that places more weight on better-performing particles while retaining diversity across them. The particles are updated jointly, with better-performing particles exerting greater influence on the search direction. The algorithm also uses injected noise and weight decay in the updates to encourage exploration and improve stability, and enforce feasibility through simple

truncation. This particle-based design is naturally parallelizable and is well-suited for large-scale instances.

The rounding stage produces the final discrete assortment. Each continuous particle induces an ordering of items and therefore a nested family of assortments obtained by taking the top items under that ordering. NAO rounds a candidate by searching over this family and selecting the best assortment according to the original revenue objective. This rounding rule is computationally attractive because it reuses the same structure generated during the evaluation of the continuous surrogate. More importantly, it comes with a strong guarantee: the discrete objective value of the rounded assortment is no worse than the value of the continuous Lovász surrogate at the candidate point. As a result, guarantees established for the continuous optimization carry over directly to the final discrete assortment, without introducing an additional rounding error term, which is a common complication in relaxation-based methods.

Many applications also impose a capacity limit due to limited shelf space or limited display slots. NAO includes a capacity-aware variant of the extension that evaluates only feasible assortments while retaining the main advantages of the unconstrained construction. The main idea is to replace the single nested sequence with a rolling sequence of feasible windows along the sorted order. This keeps the surrogate tied to feasible assortments and maintains informative gradient feedback for items that are not currently near the top of the ranking. With this modification, the same particle search and rounding steps apply with minimal change.

A central feature of NAO is that it is supported by end-to-end performance guarantees that do not rely on a particular choice model family. Under general conditions, we establish its global convergence and near-optimality guarantees. The bounds identify how the solution quality improves with computational time and the size of the neural network, and how it depends on the regularization hyperparameters. This combination of model-agnostic design and near-optimality guarantees differentiates NAO from both model-specific approaches and heuristic or learning-based optimizers.

We complement the theory with an extensive empirical evaluation on challenging benchmark instances under Mixed Multi-nomial Logit and Nested Logit models [28], in both unconstrained and capacity-constrained settings. Across these benchmarks, NAO consistently achieves very small optimality gaps and strong computational efficiency relative to competitive heuristics and neural baselines. The results also highlight that the method scales well to larger instances, supported by efficient gradient updates and the parallelizable particle-based search.

Taken together, this paper contributes a general-purpose assortment optimizer that can be used as a drop-in tool across a wide range of discrete choice models and comes with both theoretical and empirical evidence of strong performance.

Literature Review

Assortment optimization with specific choice models. A large body of research has focused on developing efficient algorithms under particular parametric choice models. For instance, assortment optimization under the MNL model has been extensively studied, leading to an optimal revenue-order policy [61] with its robustness being justified in [57]. Extensions to variants have also been studied, not limited to nested logit models [16, 23, 37], random parameter logit [58], Markov Chain (MC) choice model without constraints [6], Mallows model [17], and neural-network choice models that account for assortment effects [3, 67]. However, assortment optimization becomes computationally intractable in many other common settings. Notably, even the unconstrained case of a mixture of just two MNL models [7] or the constrained problem under the MC model [18] is NP-hard. In such cases, the literature has largely resorted to heuristic methods specific to each choice model structure.

Continuous and asymptotic assortment formulations. A related stream uses continuous product space, asymptotic regimes, or otherwise structured formulations to make assortment optimization more tractable [1, 26, 52]. These works are complementary to ours: rather than attacking the original combinatorial problem directly, they exploit continuum or limiting structure. In contrast, our framework starts from a finite discrete

assortment problem, constructs an exact Lovász-extension-based continuous surrogate on the same finite ground set, and uses rounding to recover a discrete assortment with no additional rounding loss.

Heuristics for general choice models. Beyond model-specific approaches, another line of work seeks to design heuristics that apply under general or even arbitrary choice models, which is the setting our work addresses. These include but are not limited to revenue-ordered assortments [5], local search strategies [25, 34], and independent-demand relaxation [24], which provide tractable solutions but either have rationality assumptions or lack rigorous global convergence guarantees.

First-order optimization using machine learning. Complementing these efforts, emerging machine learning-based approaches directly optimize assortments [10, 28, 41] via gradient-descent solvers, demonstrating strong empirical scalability, though such methods generally lack theoretical guarantees regarding solution quality for generic choice models. Our framework also relates to the learning problem of neural set functions [35, 50], as the objective of assortment optimization is a set function. Recent research [64] has also drawn connections between assortment optimization and subset selection. The subset selection or sampling approaches have three promising directions: score function estimator, pathwise gradient estimator, and relaxed sampling [68]. Our approach is most relevant to the third direction.

While submodular maximization for subset selection admits strong performance guarantees [33, 64, 70], these results do not extend to general revenue functions [64]. The Lovász extension is classical in submodular optimization, but its role in assortment optimization has been limited because assortment revenue functions are generally not submodular. Existing uses are mostly model-specific or heuristic, such as projected-gradient methods for joint assortment-inventory planning [27], approximation algorithms that exploit connections to submodular maximization [4], and mixed-integer conic reformulations using Lovász-type extenstions [29]. Notably, Goyal et al. [27] point out that efficiently recovering high-quality integral solutions from Lovász-based fractional solutions is nontrivial. In contrast, our framework uses the Lovász extension as a general continuous surrogate and pairs it with a chain-rounding rule that transfers surrogate guarantees to discrete assortments without an additional rounding loss.

Compared with the above literature, our work provides a general-purpose near-optimal solution framework that scales to large consideration sets and applies flexibly across choice models. We note that there is also a literature on online and dynamic assortment optimization (e.g., [2, 11, 42]), whereas this paper focuses on the offline setting with one-shot assortment planning.

2 Assortment Optimization

We study the standard *assortment optimization* problem. A firm has a universe of n products $[n] = \{1, \dots, n\}$ and must decide which subset (assortment) $S \subseteq [n]$ to offer to a customer. Upon seeing S, the customer either purchases one product $j \in S$ or chooses the *no-purchase* option (denoted by 0). If product j is purchased, the firm earns revenue (or margin) $r_j \in [0, \bar{r}]$; the no-purchase option yields zero revenue.

Customer choice is modeled through choice probabilities $\mathbb{P}(j \mid S)$ for $j \in S \cup \{0\}$, determined by an underlying discrete choice model. The expected revenue from offering S is

$$\max_{S \subseteq [n]} r(S), \quad \text{where} \quad r(S) := \sum_{j \in S} r_j \cdot \mathbb{P}(j \mid S). \tag{1}$$

We emphasize that our algorithm treats $r(S)$ (equivalently, $\mathbb{P}(\cdot \mid S)$) as an *oracle*: given an offered set S, we can evaluate its expected revenue under a specified choice model. Our method is model-agnostic and can be applied to any choice model for which such an oracle is available.

We briefly list several prominent families used in revenue management and marketing.

- *Attraction models / Multinomial Logit (MNL).* Each product j has an attraction parameter $v_j > 0$ and

the no-purchase option has $v_0 > 0$. The choice probability under the Luce (MNL) model is

$$\mathbb{P}(j \mid S) = \frac{v_j}{v_0 + \sum_{i \in S} v_i}, \qquad j \in S,$$

with $\mathbb{P}(0 \mid S) = \frac{v_0}{v_0 + \sum_{i \in S} v_i}$. A common parameterization is $v_j = e^{u_j}$ for utilities $u_j \in \mathbb{R}$ [45, 47].

- *Nested Logit (NL).* To capture correlations and stronger substitution among similar products, items are partitioned into G nests (groups) $\{V_g\}_{g=1}^{G}$ [15, 69]. Each nest g has a dissimilarity parameter $\gamma_g > 0$, a nest-level no-purchase attraction $v_0^g > 0$, and product-level attractions $\{v_{gk}\}_{k \in V_g}$. In addition, there is a global no-purchase attraction $v_0 > 0$. When $\gamma_g \in (0, 1]$, products within the nest are substitutes; when $\gamma_g > 1$, they exhibit complementarity. The choice process follows a two-level structure: the customer first selects a nest (or the global no-purchase option), then chooses a product within that nest (or the nest-level no-purchase option). For an offered set S, let $S_g := S \cap V_g$. Define the inclusive value of nest g as

  $$V_g(S) := v_0^g + \sum_{k \in S_g} v_{gk},$$

  which aggregates the attractions of all offered products in nest g together with the nest-level no-purchase term v_0^g. The probability that the customer chooses product $j \in S_g$ is

  $$\mathbb{P}(j \mid S) = \frac{v_{gj}}{V_g(S)} \cdot \frac{V_g(S)^{\gamma_g}}{v_0 + \sum_{h=1}^{G} V_h(S)^{\gamma_h}},$$

  and the probability of the global no-purchase option is $\mathbb{P}(0 \mid S) = \frac{v_0}{v_0 + \sum_{h=1}^{G} V_h(S)^{\gamma_h}}$.

- *Mixed MNL (MMNL).* To model customer heterogeneity, the mixed logit model assumes the arriving customer is drawn from one of C classes [48]. Segment c occurs with probability α_c (where $\sum_{c=1}^{C} \alpha_c = 1$) and follows an MNL model with parameters $\{v_j^c\}_{j \in V}$ and outside option v_0^c. The aggregate choice probability is the mixture

  $$\mathbb{P}(j \mid S) = \sum_{c=1}^{C} \alpha_c \frac{v_j^c}{v_0^c + \sum_{i \in S} v_i^c}, \qquad j \in S.$$

To align with the minimization convention used in our optimization development, we define the *loss* $f(S) := -r(S)$, so that (1) is equivalently

$$\min_{S \subseteq [n]} \; f(S). \tag{2}$$

Many applications also impose a *capacity constraint* (e.g., limited shelf space or limited display slots), leading to

$$\min_{S \subseteq [n],\ |S| \le K} \; f(S) \tag{3}$$

for a given capacity K. Under standard choice models, offering nothing yields zero revenue, i.e., $r(\varnothing) = 0$, and hence $f(\varnothing) = 0$.

The tractability of (2) and (3) depends on the choice model. Under MNL, the problem admits efficient solution methods and the optimal assortment can be found in polynomial time [61], although tractability can already break down under more general capacity constraints [19]. However, once the model allows richer substitution and correlation patterns (e.g., general nested logit or mixed logit), the optimization problem can become substantially harder; in particular, NP-hardness results are known in broad regimes [see, e.g., 15, 40, 58]. Intuitively, under these richer models the incremental value of adding a product can depend

strongly on what is already offered, so the revenue function may fail to exhibit simple diminishing returns structure that many classical greedy-type methods rely on. For specific choice models, there may be special structure that can be exploited to design efficient algorithms, but in general the problem is computationally challenging. This motivates algorithmic approaches that can effectively search the space of assortments even when $r(S)$ exhibits complex interactions, and that are applicable across different choice models without model-specific tailoring.

3 Our Method

In this section, we propose *Neural Assortment Optimization (`NAO`)*, a principled optimization framework designed to tackle the combinatorial hardness of the problem. Our method proceeds in a three-stage pipeline: *extend–particle search–round*.

We first extend the discrete assortment objective from $\{0,1\}^n$ to the continuous hypercube $\mathcal{X} = [0,1]^n$ via the Lovász extension ϕ, which preserves the discrete optimum value and admits an efficient sub gradient oracle (Section 3.1). To navigate the resulting non-convex landscape, we perform particle search by optimizing a pooled soft-min objective over a population of particle vectors using projected noisy subgradient descent (Sections 3.2 and 3.3). Finally, we employ a rounding and selection scheme to recover nearly optimal assortments from the continuous solutions (Section 3.4), and synthesize the complete procedure in Algorithm 1 (Section 3.5).

3.1 Continuous Extension

The assortment decision can be represented either as a subset $S \subseteq [n]$ or, equivalently, as its indicator vector $\mathbf{1}_S \in \{0,1\}^n$. Direct search over $\{0,1\}^n$ is combinatorial. Our approach relaxes the decision space to the hypercube $[0,1]^n$ and then maps the resulting continuous solution back to a feasible assortment. There are many ways to extend a discrete function to a continuous domain, but here we choose the Lovász extension [44], which is a piecewise-affine extension that preserves the original optimal value and has a nice landscape geometry that we can exploit for optimization.

Definition 1 (Lovász extension)**.** *Let $f : \{0,1\}^n \to \mathbb{R}$ and assume $f(\varnothing) = 0$ (equivalently, $f(\mathbf{0}) = 0$). For any $x \in [0,1]^n$, let π sort the coordinates of x in non-increasing order, $x_{\pi(1)} \geq \cdots \geq x_{\pi(n)}$, and define $x_{\pi(n+1)} := 0$. Define the chain of sets $S_i(x) := \{\pi(1), \ldots, \pi(i)\}$ with $S_0(x) := \varnothing$. The Lovász extension of f is the function $\phi : [0,1]^n \to \mathbb{R}$ given by*

$$\phi(x) := \sum_{i=1}^{n} \big(x_{\pi(i)} - x_{\pi(i+1)}\big)\, f\big(S_i(x)\big). \tag{4}$$

The definition can be read directly in assortment terms. A vector $x \in [0,1]^n$ assigns each product a *priority level* for inclusion in the offer set; larger x_j means product j is treated as higher priority. Sorting these priorities produces an ordering π, and the sets $S_i(x) = \{\pi(1), \ldots, \pi(i)\}$ are simply the assortments formed by taking the top i products under that ordering. Thus, although x is fractional, it induces a small, structured family of particle assortments, namely the chain $\{S_i(x)\}_{i=0}^{n}$.

A closely related and often more intuitive view is obtained by thresholding. For $t \in [0,1]$, define the level-set assortment

$$S_u(x) := \{i \in [n] : x_i \geq u\}.$$

As u decreases from 1 to 0, the offer set $S_u(x)$ expands by adding products in the order π. In particular, if $u \in (x_{\pi(i+1)}, x_{\pi(i)}]$ then $S_u(x) = S_i(x)$, and if $U \sim \text{Uniform}[0,1]$,

$$\mathbb{P}\big(S_U(x) = S_i(x)\big) = x_{\pi(i)} - x_{\pi(i+1)}.$$

Consequently, (4) can be written as

$$\phi(x) = \mathbb{E}_{U\sim\mathrm{Unif}[0,1]}\big[f(S_U(x))\big].$$

This representation highlights the connection to assortments: $\phi(x)$ is the average loss obtained by converting x into an offer set via a random threshold.

A key property for our method is that this relaxation is *tight* in optimal value, in the sense that minimizing ϕ over $[0,1]^n$ achieves the same minimum value as minimizing f over $\{0,1\}^n$. To see this, define

$$\alpha_0 := 1 - x_{\pi(1)}, \qquad \alpha_i := x_{\pi(i)} - x_{\pi(i+1)}, \quad i = 1,\dots,n.$$

Then $\alpha_i \geq 0$ and $\sum_{i=0}^{n} \alpha_i = 1$, and

$$\phi(x) = \sum_{i=0}^{n} \alpha_i\, f(S_i(x)).$$

The term $i = 0$ corresponds to $S_0(x) = \varnothing$. Therefore $\phi(x)$ is a weighted average of the losses of assortments on the chain induced by x, implying

$$\phi(x) \geq \min_{0\leq i\leq n} f(S_i(x)) \geq \min_{s\in\{0,1\}^n} f(s).$$

On the other hand, ϕ agrees with f on integral points: for any $S \subseteq [n]$, $\phi(\mathbf{1}_S) = f(S)$. These two facts imply the exactness of the relaxation, which we state as a formal proposition. For completeness, we include a short proof in Appendix C.1.1.

Proposition 1 (Tightness of Lovász extension)**.** *Let $f : \{0,1\}^n \to \mathbb{R}$ and let ϕ be its Lovász extension. Then*

$$\min_{s\in\{0,1\}^n} f(s) = \min_{x\in[0,1]^n} \phi(x).$$

The shape of ϕ depends on the substitution patterns encoded by the choice model. Under a diminishing-returns condition (submodularity), the Lovász extension is convex [44]. In many choice models, this condition fails, and ϕ is typically non-convex, motivating the particle search procedure developed next. This tightness property, however, does not depend on any structural assumptions on f (e.g., submodularity) and holds for any choice model, making it a powerful tool for our model-agnostic framework.

3.2 Particle-based Objective

Equipped with the continuous relaxation ϕ, we now describe how we search over $[0,1]^n$ for high-quality assortments.

The Lovász extension is piecewise affine: when the ordering of the coordinates of x is fixed (i.e., away from ties), ϕ is linear in x. At the same time, different orderings correspond to different affine pieces, so the overall landscape is generally non-convex. In this setting, a single projected subgradient trajectory can be sensitive to initialization and may settle in regions that are far from the global minimum. To improve exploration, we optimize a collection of M particle solutions in parallel. Let $\mathbf{x} = (x^m)_{m=1}^{M} \in \mathcal{X}^M$ denote the population, where $\mathcal{X} = [0,1]^n$ and each $x^m \in \mathcal{X}$ is a feasible priority vector over products, where we use boldface $\mathbf{x}$ to denote the entire particle system and regular font x^m to denote the m-th particle solution. Our goal is that at least one particle attains a small value of ϕ. This motivates the objective $\min_{\mathbf{x}\in\mathcal{X}^M} \min_{1\leq m\leq M} \phi(x^m)$. However, directly optimizing the inner minimum is inconvenient because it is non-differentiable. Instead, we replace the hard minimum across particles by a soft minimum. Specifically, we define the particle loss

$$L(\mathbf{x}) := \ell\big(\phi(x^1),\dots,\phi(x^M)\big), \tag{5}$$

where

$$\ell(u) := -\log\left(\frac{1}{M}\sum_{m=1}^{M} e^{-u_m}\right)$$

is the log-sum-exp function, which is a smooth approximation to the minimum that becomes tighter as M increases.

Equation (5) also admits a simple neural-network interpretation. Specifically, one can view the population $\{x^m\}_{m=1}^M$ as the parameters of a one-hidden-layer architecture: each hidden neuron takes the particle x^m and produces a scalar activation $-\phi(x^m)$, and the output layer aggregates these activations through log-sum-exp pooling. This perspective motivates the term *Neural Assortment Optimization* (`NAO`), an architectural view we explicitly adopt in our numerical implementation.

In the literature, the function $L(\mathbf{x})$ is also known as the *entropic risk* of the random variable $\phi(x^m)$ when m is uniformly sampled from $\{1,\dots,M\}$. It measures the expected loss of a randomly selected particle under an exponential weighting scheme that emphasizes better-performing particles. Importantly, this smooth objective maintains the global optimum value of our considered problem.

Proposition 2 (Exactness of the particle loss)**.** *The global minimum of L over $\mathcal{X}^M$ coincides with the global minimum of ϕ over $\mathcal{X}$:*

$$\min_{\mathbf{x}\in\mathcal{X}^M} L(\mathbf{x}) = \min_{x\in\mathcal{X}} \phi(x) = \min_{s\in\{0,1\}^n} f(s).$$

We remark that one can scale ϕ by a smoothing parameter $\beta > 0$ with the same exactness property, which allows for tuning the softness of the pooling and can improve optimization stability in practice; see Appendix A.2 for details.

Although ϕ is typically non-convex and non-smooth, it admits a simple first-order description. The unit hypercube can be partitioned into regions according to the ordering of the coordinates of x. On any region where this ordering is fixed, ϕ is affine, and therefore has a constant gradient on that region. This gradient can be written using marginal changes of the discrete objective $f(\cdot)$ along the chain induced by sorting x. Concretely, fix $x \in [0,1]^n$ and let π be any permutation that sorts x in non-increasing order (breaking ties arbitrarily). Recall the induced chain of assortments $S_i(x) := \{\pi(1),\dots,\pi(i)\}$ for $i = 1,\dots,n$. Define $g(x) \in \mathbb{R}^n$ by

$$g_{\pi(i)}(x) := f\big(S_i(x)\big) - f\big(S_{i-1}(x)\big), \qquad i = 1,\dots,n. \tag{6}$$

Thus $g_{\pi(i)}(x)$ is the incremental change in the objective when the i-th ranked product $\pi(i)$ is added to the offer set consisting of the higher-ranked products.

Proposition 3 (Subgradient of ϕ)**.** *For any $x \in [0,1]^n$ and any permutation π consistent with sorting x, the vector $g(x)$ defined in* (6) *belongs to the subdifferential of ϕ at x.*

Proposition 3 yields a direct computational routine. Given x, we sort its coordinates to obtain π, evaluate $f(S_i(x))$ along the chain, and form the marginal differences in (6). When x has a strict ordering (no ties), π is locally constant and ϕ is affine in a neighborhood of x, so $g(x)$ coincides with the classical gradient. When ties occur, ϕ is non-differentiable, but the same construction yields a valid generalized subgradient for any tie-breaking permutation π. Formal details are deferred to Appendix A.1.

We remark that when evaluations of $f(\cdot)$ are expensive (e.g., when expected revenue is estimated from data or simulation), we can reduce oracle calls using sampling. Specifically, sample $I \sim \text{Uniform}\{1,\dots,n\}$ and define the sparse estimator $\widehat{g}(x) \in \mathbb{R}^n$ by

$$\widehat{g}(x) := n\big(f(S_I(x)) - f(S_{I-1}(x))\big)e_{\pi(I)},$$

where e_i denotes the ith standard basis vector. Then $\mathbb{E}[\widehat{g}(x)] = g(x)$, so $\widehat{g}(x)$ can be used in stochastic subgradient updates. This reduces the number of function evaluations per iteration from $O(n)$ to $O(1)$ (although we do not exploit this advantage in our numerical experiment for fair comparison).

3.3 Particle Noisy Projected Gradient Descent

We minimize the particle objective $L(\mathbf{x})$ over the constrained domain $\mathcal{X} = [0,1]^n$. Starting from an initial population $\mathbf{x} = (x^m)_{m=1}^M \in \mathcal{X}^M$, at iteration k, a natural way is to apply the projected subgradient descent on $\mathbf{x}$:

$$x^m \leftarrow \Pi_{\mathcal{X}}\left(x^m - \eta M \nabla_{x^m} L(\mathbf{x})\right), \quad m = 1, \ldots, M, \tag{7}$$

where $\eta > 0$ is the stepsize and $\nabla L(\mathbf{x})$ denotes a subgradient of L at $\mathbf{x}$. The chain rule implies that the partial subgradient $\partial_{x^m} L(\mathbf{x})$ satisfies

$$w^m(\mathbf{x}) g^m \in \partial_{x^m} L(\mathbf{x}), \qquad \text{where } w^m(\mathbf{x}) = \frac{\exp(-\phi(x^m))}{\sum_{j=1}^M \exp(-\phi(x^j))}, \tag{8}$$

where g^m is any generalized subgradient of ϕ at x^m. The weight $w^m(\mathbf{x})$ acts as a soft selection rule within the particle population. Particles with smaller Lovász values $\phi(x^m)$ receive larger weights in (8), so their subgradients have greater influence on the joint update of the particle population, while the soft weighting keeps the remaining particles active for exploration. Because $w^m(\mathbf{x})$ depends on the entire population, the method is more than a collection of independent restarts: better-performing particles guide the coupled search throughout the run rather than only being selected at termination. We keep the scaling factor M explicit in (7) so that the overall step magnitude is comparable across different population sizes when the weights are not highly concentrated; equivalently, this factor can be absorbed into η. Because $\mathcal{X}$ is the unit hypercube, the Euclidean projection $\Pi_{\mathcal{X}}$ is simply coordinate-wise truncation: for every $y \in \mathbb{R}^n$, $i \in [n]$ and $m = 1, \ldots, M$,

$$\left[\Pi_{\mathcal{X}}(y)\right]_i = \min\{1, \max\{0, y_i\}\}.$$

In our setting ϕ is typically non-convex, and the basic update (7) can be sensitive to initialization and may stagnate near flat regions or along boundaries where the ordering of coordinates changes. We therefore use a noisy, regularized variant of (7). For each particle m we update

$$x^m \leftarrow \Pi_{\mathcal{X}}\left(x^m - \eta\left(M \nabla_{x^m} L(\mathbf{x}) + \lambda x^m\right) + \sqrt{2\eta\tau}\xi^m\right), \quad m = 1, \ldots, M, \tag{9}$$

where $\{\xi^m\}_{m=1}^M$ are independent standard Gaussian vectors; $\tau > 0$ controls the noise magnitude, a larger τ encouraging more exploration; and $\lambda > 0$ is a weight decay parameter that stabilizes the training dynamics and discourages overly aggressive movement toward near-binary corner solutions early in the run.

3.4 Rounding and Selection

The optimization phase produces a population of continuous solutions $\mathbf{x} = (x^m)_{m=1}^M \in \mathcal{X}^M$. The operational decision, however, is a discrete assortment $S \subseteq [n]$ (equivalently, an indicator vector in $\{0,1\}^n$). We therefore convert each continuous particle x^m into a feasible assortment via a deterministic rounding step that exploits the structure of the Lovász extension.

Each $x \in [0,1]^n$ induces a permutation π obtained by sorting its coordinates and a corresponding nested chain of assortments $\{S_i(x)\}_{i=0}^n$ (Definition 1). Since $\phi(x)$ evaluates $f(\cdot)$ only on this chain, it is natural to search for a discrete assortment by scanning the same chain [44]. We define the rounding operator $\mathrm{Round}(x)$ as follows:

$$\mathrm{Round}(x) \in \arg\min_{i=0,\ldots,n} f\big(S_i(x)\big), \tag{10}$$

where $\{S_i(x)\}_{i=0}^n$ is the chain induced from sorting the coordinates of x according to Definition 1.

This rounding rule comes with a simple guarantee relative to the continuous surrogate. By construction, $\phi(x)$ is a weighted average of the chain values $\{f(S_i(x))\}_{i=0}^n$ with nonnegative weights. A weighted average cannot be smaller than the minimum term being averaged, so the best assortment on the chain attains an objective value no larger than $\phi(x)$.

Lemma 1 (Chain rounding is no worse than ϕ)**.** *Let ϕ be the Lovász extension of f. For any $x \in [0,1]^n$,*

$$f\big(\text{Round}(x)\big) \le \phi(x). \tag{11}$$

Lemma 1 implies that any bound on the surrogate value $\phi(x)$ immediately transfers to the rounded assortment produced from x. The proof is deferred to Appendix C.1.2.

3.5 Algorithm

Algorithm 1 summarizes the complete Neural Assortment Optimization (`NAO`) procedure. At iteration k, the method maintains a population $\mathbf{x} = (x^m)_{m=1}^M \in \mathcal{X}^M$, where each $x^m \in \mathcal{X}$ is a priority vector over products. For each m, we first sort the n components of $x^m \in [0,1]^n$ to get a chain $\{S_i(x^m)\}_{i=0}^n$ from Definition 1; next, we evaluate f along this chain to obtain $\phi(x^m)$, a subgradient $g^m \in \partial\phi(x^m)$, and the rounded solution $\widehat{S}^m = \text{Round}(x^m)$. We then form the weights $w^m(\mathbf{x})$ via (8) and update each particle using the projected noisy subgradient (9), where the projection is simply the coordinate-wise truncation to $[0,1]$. We maintain an incumbent best assortment $\widehat{S}$ as the best rounded solution encountered so far across all particles and iterations. This best-so-far policy makes the method an anytime procedure: it produces a feasible assortment at every iteration and returns the all-time best particle at termination. Because the same chain evaluations are used for $\phi(x^m)$ and g^m, rounding adds negligible overhead in the implementation. Finally, there are various termination criteria; in practice, we monitor the improvement of the best rounded solution and stop when it fails to improve for a certain number of iterations.

These components are designed to work together. The Lovász extension provides a tractable continuous representation with an efficient chain-based subgradient oracle; the entropic-risk particle loss (5) couples the search across particles rather than treating them as independent restarts; and the particle structure allows chain evaluations and subgradient computations to be batched efficiently. Thus, the method relies on the combination of continuous representation, coupled multi-particle search, and parallel implementation.

ALGORITHM 1: Neural Assortment Optimization (`NAO`)

Input: $\mathbf{x} = (x^m)_{m=1}^M$, $x^m \sim \text{Uniform}([0,1]^n)$, $\widehat{S} \leftarrow \varnothing$, $\widehat{f} \leftarrow +\infty$
while *not converged* **do**
  for $m = 1$ **to** M **do**
    Sort $x^m \in [0,1]^n$ to obtain the chain $\{S_i(x^m)\}_{i=0}^n$ in Definition 1
    Compute $\phi(x^m)$ via (4) and $g^m \in \partial\phi(x^m)$ via (6)
    Compute weights $w^m(\mathbf{x})$ via (8)
    Update the particle x^m via the noisy gradient descent oracle (9)
    $S^m \leftarrow \text{Round}(x^m)$ via (10)
    if $f(S^m) < \widehat{f}$ **then**
      $\widehat{f} \leftarrow f(S^m)$, $\widehat{S} \leftarrow S^m$, $\widehat{x} \leftarrow x^m$
    end
  end
end
return $(\widehat{S}, \widehat{x})$

4 Handling Capacity Constraints

The development so far focuses on the unconstrained problem (2), where the Lovász extension ϕ provides a value-preserving continuous objective on $[0,1]^n$. In this section, we extend our method to the capacity-constrained assortment optimization problem (3), where the offered set is required to satisfy $|S| \le K$ for a

given number K. Our goal is to retain the advantages of the our approach—particle search on a continuous domain followed by fast rounding—while ensuring that the feasibility constraint is respected throughout the search process.

A direct use of the standard Lovász extension is not ideal in this setting because its chain representation evaluates $f(\cdot)$ on subsets of all sizes, including those consisting of more than K products. One could attempt to enforce feasibility only at the end by restricting rounding to $|S| \leq K$, but then the continuous objective being optimized would still be influenced by infeasible large sets, weakening the connection between descent in the surrogate and improvement in feasible assortments. This motivates a capacity-aware variant of the extension whose defining sets satisfy $|S| \leq K$ throughout.

A first attempt is to truncate the Lovász construction after the top-K positions of the sorted list. While this enforces feasibility in the sets that appear in the surrogate, it creates a practical problem for gradient-based search: once a product is ranked below position K, small changes in its coordinate have little or no effect on the truncated surrogate, and the algorithm receives essentially no feedback about that product. As a result, it becomes difficult for potentially attractive products that are currently below the cutoff to move upward in the ordering, especially when K is small. Indeed, empirically we find that this truncated surrogate works reasonably well when K is large, but its performance degrades significantly as K decreases.

We address this issue with a *rolling-window* construction. Instead of focusing exclusively on the top-K products, we evaluate $f(\cdot)$ on a sequence of feasible "windows" of size at most K that move along the sorted ranking. This way, the capacity constraint is enforced within the surrogate—every set evaluated by ϕ_K has size at most K—while each product still enters at least one window. As a result, products ranked below the current top-K continue to affect the surrogate loss through their window evaluations, so gradient-based updates can still reflect how changes in their position would change the objective.

Definition 2 (Rolling-window Lovász extension)**.** *Fix a capacity parameter $K \in \{1, \dots, n\}$ and a set function $f : \{0,1\}^n \to \mathbb{R}$ with $f(\varnothing) = 0$. For any $x \in [0,1]^n$, let π be a permutation sorting x in non-increasing order, $x_{\pi(1)} \geq \cdots \geq x_{\pi(n)}$, and define $x_{\pi(n+1)} := 0$. Define window sets $W_0(x) := \varnothing$ and, for $i = 1, \dots, n$,*

$$W_i(x) := \begin{cases} \{\pi(1), \dots, \pi(i)\}, & 1 \leq i \leq K, \\ \{\pi(i-K+1), \dots, \pi(i)\}, & i > K. \end{cases}$$

The rolling-window Lovász extension is the function $\phi_K : [0,1]^n \to \mathbb{R}$ defined by

$$\phi_K(x) := \sum_{i=1}^{n} \big(x_{\pi(i)} - x_{\pi(i+1)}\big) f\big(W_i(x)\big). \tag{12}$$

Our construction is closely related to the bounded-cardinality Lovász extension proposed by Karalias et al. [35]. Both constructions are based on the same family of capacity-feasible window sets $\{W_i(x)\}_{i=0}^n$. The difference lies in the choice of coefficients in the weighted sum that defines the extenstion. Karalias et al. [35] select weights for a primal-dual purpose (in particular, to satisfy a dual-feasibility condition in their analysis), whereas we retain the standard Lovász coefficients $(x_{\pi(i)} - x_{\pi(i+1)})$ in (12). This choice keeps the construction aligned with the unconstrained Lovász extension: it preserves the same threshold interpretation, which in turn gives an immediate tightness property and a rounding guarantee. Specifically, define the rounding operator under the capacity constraint:

$$\mathrm{Round}_K(x) \in \operatorname*{arg\,min}_{0 \leq i \leq n} f\big(W_i(x)\big).$$

By construction, $\mathrm{Round}_K(x)$ is feasible (its cardinality is no more than K) for all x, and the following analogue of Proposition 1 and Lemma 1 holds.

Proposition 4. *The rolling-window Lovász extension ϕ_K defined by* (12) *satisfies the following properties:*

$$\min_{x\in[0,1]^n} \phi_K(x) = \min_{S\subseteq[n],\,|S|\le K} f(S),$$

and for all $x \in [0,1]^n$, $\mathrm{Round}_K(x)$ is feasible and satisfies the bound

$$f\big(\mathrm{Round}_K(x)\big) \le \phi_K(x). \tag{13}$$

The rolling-window extension ϕ_K inherits the same geometry that makes the unconstrained Lovász extension convenient for gradient-based search. On any region where the sorting permutation π is fixed, the window sets $\{W_i(x)\}$ are fixed and ϕ_K is affine in x; hence ϕ_K is continuous and piecewise affine on $[0,1]^n$. Subgradients are computed by the same "sort-and-marginals" procedure as in (6), with rolling windows $W_i(x)$ in place of partial subsets $S_i(x)$. Fix $x \in [0,1]^n$ and a induced permutation $\pi(x)$. A valid generalized subgradient $g(x) \in \partial\phi_K(x)$ is given by

$$g_{\pi(k)}(x) = f\big(W_i(x)\big) - f\big(W_{i-1}(x)\big), \qquad i = 1,\dots,n.$$

With this oracle and the rounding rule $\mathrm{Round}_K(\cdot)$, our algorithm `NAO` extends to the capacity-constrained problem by direct substitution: replace ϕ by ϕ_K when evaluating particles and computing subgradients, and replace $\mathrm{Round}(\cdot)$ by $\mathrm{Round}_K(\cdot)$; all other steps remain unchanged. Empirically, we find that this choice performs well and efficiently, as will be shown in our numerical experiments.

5 Performance Guarantees

This section establishes performance guarantees for Algorithm 1. Let $r^\star := \max_{S\subseteq[n]} r(S)$ denote the optimal expected revenue. The algorithm returns a discrete assortment $\widehat{S}$, and our goal is to bound the expected optimality gap

$$r^\star - \mathbb{E}\big[r(\widehat{S})\big],$$

where the expectation is with respect to the algorithmic randomness, including initialization and injected Gaussian noise.

5.1 Interacting Particle Systems and their Mean-field Limit

Algorithm 1 can be viewed as an evolving dynamic system of M interacting particles. At each iteration, the update of particle m depends on the entire population through the exponential weights (8). This interaction is essential: the method is not M independent runs of a stochastic optimizer, but a coupled system whose collective behavior is designed to steer at least one particle toward low values of the surrogate loss.

A standard approach for analyzing such systems is to study the continuous-time diffusion limit as the step size $\eta \to 0$. This perspective serves as a rigorous analytical framework for non-convex dynamics [see, e.g., 9, 55, 60], facilitating the derivation of global performance guarantees. Let $\mathbf{x}_{(k)} = (x^m_{(k)})_{m=1}^M$ be the iterates generated by (9). Introduce the rescaled time $t = k\eta$ and define the piecewise-constant interpolation $X^m_t = x^m_{(k)}$ for $t \in [k\eta, (k+1)\eta)$. As $\eta \to 0$ (with a fixed time horizon), the interpolated process admits a continuous-time limit in an appropriate weak sense, described by the reflected interacting stochastic differential equation (SDE)

$$dX^m_t = b^m(\mathbf{X}_t)dt + \sqrt{2\tau}dB^m_t + dR^m_t, \quad m = 1,\dots,M, \tag{14}$$

where $\{B_t^m\}_{m=1}^M$ are independent Brownian motions and R_t^m is the reflection term induced by the projection onto $\mathcal{X}$. The drift $b^m(\cdot)$ corresponds to the deterministic part of (9) per unit time:

$$b^m(\mathbf{x}) = -\Big(M\, w^m(\mathbf{x})\, g(x^m) + \lambda x^m\Big), \qquad w^m(\mathbf{x}) = \frac{\exp(-\phi(x^m))}{\sum_{j=1}^M \exp(-\phi(x^j))},$$

with $g(x^m) \in \partial\phi(x^m)$ any generalized subgradient selection. This SDE viewpoint is widely used in the analysis of stochastic optimization, as continuous-time diffusions are amenable to tools from Markov processes and stability theory.

Directly analyzing the convergence of the coupled dynamics in (14) is difficult, due to both non-convexity and the dependence structure induced by interaction. However, when the particle size M is large (equivalently, for a wide neural network), the system admits a tractable *mean-field* limit that captures the collective behavior of the particle system. The key point is that particles are coupled only through the normalization in the weights $w^m(\mathbf{x})$. As $M \to \infty$, the empirical average $\frac{1}{M}\sum_{j=1}^M \exp(-\phi(X_t^j))$ concentrates around its population counterpart $\int_{\mathcal{X}} e^{-\phi(x)}\,\rho_t(dx)$, where ρ_t is the marginal law of a representative particle at time t. Consequently, a representative particle X_t evolves according to the mean-field (McKean-Vlasov) reflected SDE

$$dX_t = -\Big(\frac{e^{-\phi(X_t)}}{\int_{\mathcal{X}} e^{-\phi(x)}\rho_t(dx)}\, g(X_t) + \lambda X_t\Big)\, dt + \sqrt{2\tau}\, dB_t + dR_t, \tag{15}$$

and ρ_t satisfies the associated Fokker-Planck equation with reflecting boundary conditions.

A key advantage of working with ρ_t is that the mean-field evolution admits a variational interpretation: it can be viewed as a gradient-flow dynamics that decreases a free-energy functional over distributions. In particular, it solves the mean-field loss minimization problem

$$\rho^\tau := \arg\min_{\rho\in\mathcal{P}(\mathcal{X})} \; -\log \mathbb{E}_{X\sim\rho}\big[e^{-\phi(X)}\big] + \tau D_{\mathrm{KL}}(\rho\|\gamma), \tag{16}$$

where γ is the standard Gaussian distribution restricted to $\mathcal{X}$ and D_{KL} denotes the Kullback–Leibler divergence. Under mild conditions, ρ^τ is the unique minimizer, and ρ_t converges exponentially to ρ^τ as $t \to \infty$.

This "optimization over distributions" viewpoint underlies the global convergence behavior in the $M \to \infty$ limit. Although the finite-particle system is non-convex in $\mathcal{X}^M$, the mean-field dynamics can be interpreted as a descent flow on a distributional objective with a well-behaved landscape, aided by entropic smoothing from the injected noise. It also provides the natural reference dynamics against which the finite-M particle system can be compared, leading to explicit finite-particle approximation terms in our performance bounds. We refer to Appendix B.1 for further discussion and technical details.

5.2 Global Convergence

Algorithm 1 maintains a best-so-far assortment and returns the all-time-best rounded solution $\widehat{S}$ encountered across all particles and iterations. To analyze its performance, we upper bound the gap of $\widehat{S}$ by comparing it to the terminal solution at time T.

Let x_T^m denote the terminal value of particle m under the continuous-time proxy, and define its rounded assortment $S^m(T) := \mathrm{Round}(x_T^m)$. Let

$$\bar{S}(T) \in \arg\min_{m\in[M]} f(S^m(T)) = \arg\max_{m\in[M]} r(S^m(T))$$

be the best rounded assortment among the M terminal particles. Because $\widehat{S}$ is the best rounded solution observed over the entire run, it is no worse than any terminal particles; hence, for every realization,

$$f(\widehat{S}) \le f(\bar{S}(T)).$$

Next, let $\bar{x}_T \in \arg\min_{m\in[M]} \phi(x_T^m)$ be the terminal particle with the smallest surrogate loss. By Lemma 1, chain rounding is no worse than the Lovász surrogate. Applying this particle-wise yields

$$f(\bar{S}(T)) = \min_{m\in[M]} f(S^m(T)) = \min_{m\in[M]} f(\mathrm{Round}(x_T^m)) \le \min_{m\in[M]} \phi(x_T^m) = \phi(\bar{x}_T). \tag{17}$$

Moreover, the log-sum-exp soft minimum dominates the hard minimum: for any $u \in \mathbb{R}^M$, $\ell(u) \ge \min_{m\in[M]} u_m$. Applying this with $u_m = \phi(x_T^m)$ gives

$$L(\mathbf{x}_T) \ge \min_{m\in[M]} \phi(x_T^m) = \phi(\bar{x}_T).$$

Finally, recall that $f = -r$ and $\phi^\star = -r^\star$ (Proposition 1). Combining the inequalities above and taking expectations yields

$$\begin{aligned} r^\star - \mathbb{E}\big[r(\widehat{S})\big] &= r^\star + \mathbb{E}\big[f(\widehat{S})\big] = -\phi^\star + \mathbb{E}\big[f(\widehat{S})\big] \\ &\le -\phi^\star + \mathbb{E}\big[f(\bar{S}(T))\big] \le -\phi^\star + \mathbb{E}\big[\phi(\bar{x}_T)\big] \le -\phi^\star + \mathbb{E}\big[L(\mathbf{x}_T)\big]. \end{aligned} \tag{18}$$

Consequently, it suffices to control the surrogate suboptimality

$$\mathrm{Gap}_T := \mathbb{E}\left[L(\mathbf{x}_T)\right] - \phi^\star,$$

which upper bounds the expected revenue gap of the returned assortment $\widehat{S}$. We have the following explicit upper bound on Gap_T.

Theorem 1. *Let $\tau, \lambda > 0$ and assume the revenue of each item is bounded by $\bar{r}$. Then there exist constants $\tilde{\kappa} > 0$ such that, for sufficiently large M,*

$$\mathrm{Gap}_T \le \underbrace{e^{-\tilde{\kappa}T}\Delta_0}_{\textit{optimization convergence}} + \underbrace{O\left(\frac{1}{M}\right)}_{\textit{particle approximation}} + \underbrace{O\left(\tau \log\frac{1}{\tau} + \lambda\right)}_{\textit{regularization bias}}, \tag{19}$$

where Δ_0 is an initialization-dependent constant.

The three terms in (19) reflect distinct effects:

- The first term is a finite-time convergence error. In the mean-field limit, the law ρ_t of a representative particle evolves according to (15) and converges exponentially fast to the unique minimizer ρ^τ of the free-energy functional (16). This yields an exponential decay in T from the initial gap Δ_0, producing the term $e^{-\tilde{\kappa}T}\Delta_0$. The rate $\tilde{\kappa}$, dependent on (τ, λ), characterizes the "connectivity" of the non-convex optimization landscape: a larger $\tilde{\kappa}$ indicates better connectivity, leading to easier escape from poor stationary points and thus quicker convergence to the global optimum. In the worst-case, this constant can depend on the noise level τ unfavorably but is unavoidable without additional structural assumptions and standard for Langevin dynamics in non-convex landscapes [49, 55, 60]. Empirically, however, the convergence is substantially faster on the benchmark instances considered below.

- The second term is a finite-particle approximation error. Algorithm 1 implements the population interaction using M particles, replacing population expectations in the mean-field system by empirical averages. The resulting deviation from the mean-field reference is controlled via uniform-in-time propagation-of-chaos bounds [9], yielding an explicit $O(1/M)$ error term. This term formalizes the benefit of using a large number particles (equivalently, a large neural network): increasing M makes the implemented particle system track the mean-field dynamics more accurately.

- The third term is a regularization bias. Even when T and M are large, the limiting dynamics targets the minimizer of the regularized functional (16), which generally differs from the unregularized combinatorial optimum $\phi^\star = -r^\star$. The resulting bias scales as $O(\tau \log(1/\tau) + \lambda)$ and captures the cost of exploration and stabilization. This highlights a fundamental trade-off: larger hyperparameter values can improve convergence rate (potentially increasing $\tilde{\kappa}$), but they also increase bias relative to the exact combinatorial optimum.

This decomposition explains the observed performance by separating the effects of optimization time, particle population size, and regularization. In particular, the finite-particle approximation and regularization-bias terms are controlled by M, τ, and λ, and our numerical experiments indicate that these terms are small in the benchmark instances considered in Section 6.

Taken together, (18) and Theorem 1 yield a near-optimal revenue guarantee for the discrete assortment returned by Algorithm 1. Notably, the bound on the right-hand side of (19) contains no explicit rounding term. This is because the chain rounding selects an assortment on the Lovász chain whose loss is no larger than the surrogate value (Lemma 1). In addition, these guarantees extend directly to the capacity-constrained problem by substituting ϕ_K for ϕ, because the rolling-window extension ϕ_K introduced in Section 4 shares the same regularity properties (Lipschitz continuity and piecewise linearity) as the standard Lovász extension.

We provide the full explicit dependence of the bound on the dimensional parameters in Appendices B.5 and C.4.8. In particular, we derive a conservative upper bound on the convergence-rate constant, $\tilde{\kappa} = O(\frac{\lambda}{\tau}\exp(-1/\tau))$, which can be unfavorable for small τ. Nonetheless, as shown in [55], in the presence of non-convexity, such exponential dependence on τ cannot generally be avoided in general without additional structural assumptions; yet our numerical results in the next section suggest that $\tilde{\kappa}$ is reasonably large empirically. In Appendix B.6, we further show that the iteration complexity is $O(\exp(n/\epsilon)\log(1/\epsilon))$, recalling n is the number of products. This rate aligns with standard results for Langevin dynamics for non-convex optimization [e.g., 49, 55, 60] and is consistent with the combinatorial nature of the problem.

6 Numerical Experiments

We evaluate our method on the public benchmark dataset introduced by Guo et al. [28], which provides computationally challenging instances for assortment optimization under discrete choice models. The dataset covers two widely-studied choice models: the Mixed Multinomial Logit (MMNL) model and the Nested Logit (NL) model.

- For MMNL instances, the dataset varies the number of customer segments $C \in \{5, 10, 25\}$, with two revenue curve structures from the stylized patterns of Rogosinski et al. [56]: RS2 (convex curve), defined as $r_j = 1.0 - \frac{\log(1+j)}{\log(1+n+1)}$, and RS4 (concave curve), defined as $r_j = 1.0 - \sqrt{\frac{j-1}{n-1}}$, both scaled to $[0.2, 1.0]$. These two structures aim to produce the most computationally challenging instances among the patterns tested [28].

- For NL instances, the dataset considers $G \in \{5, 10, 20\}$ nests with n products per nest, examining both low ($v_0^g \sim U[0, 1]$) and high ($v_0^g \sim U[3, 4]$) within-nest no-purchase preferences, where v_0^g denotes the utility of the no-purchase option for each nest g.

Both experiments for MMNL and ML consider unconstrained and cardinality-constrained settings. In the constrained case, we evaluate performance across capacity ratios Cap $\in \{0.1, 0.3, 0.5\}$. The public dataset in Guo et al. [28] contains 100 instances for each configuration, defined by (n, C) and (n, C, Cap) for MMNL, and (n, G) and (n, G, Cap) for NL, with $n \in \{50, 100, 200\}$. Using the same data-generating process, we additionally construct large-scale instances with n up to 10,000 to assess the scalability; dataset details are provided in Section 6.3.

We compare our method against several prominent baselines from the literature: the Revenue-Ordered (RO) policy [5], which selects optimal assortments from revenue-ordered sets; the AlphaPhi heuristic [24], which approximates the choice model using parameterized linear programs; and the ADXOpt heuristic [25], a state-of-the-art local search method incorporating tabu-like mechanisms to escape local optima. We also include the neural network-based framework of Guo et al. [28], denoted NN for the baseline optimizer and NN_{pp} for its post-processed variant, which applies local-search refinement via single-product add and delete operations. Except for RO, which does not handle capacity constraints, all methods are evaluated under both unconstrained and constrained settings, following the setting in Guo et al. [28] for fair comparison (see Appendix D).

We study two perspectives: the optimality gap for convergence and the computational time for algorithmic efficiency. The optimality gap is defined as

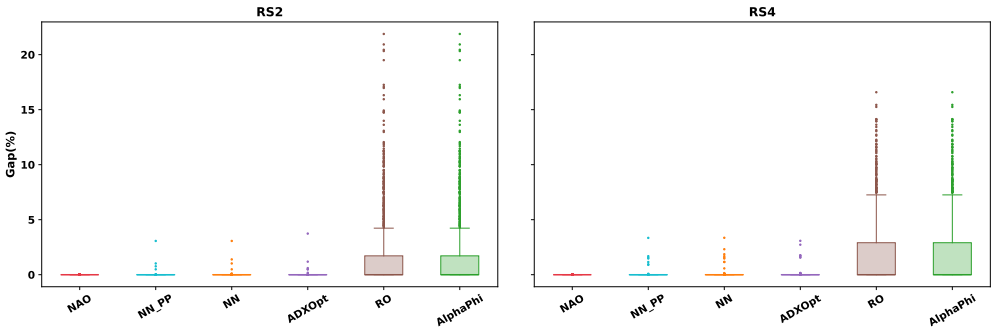
$$\text{Optimality Gap (\%)} = 100 \times \frac{r^\star - \widehat{r}}{r^\star},$$

where $\widehat{r}$ denote the revenue achieved by our framework or other state-of-the-art baselines. For MMNL instances, $r^\star$ denotes the optimal revenue, obtained by solving the conic integer formulation of Şen et al. [13] using Gurobi solver; when this formulation fails to yield the best solution within the time limit, $r^\star$ is taken as the maximum revenue achieved among all baselines. For NL instances, where exact solutions are computationally intractable at scale, $r^\star$ is replaced by a tight upper bound derived from the linear programming relaxation of Kunnumkal [37] (so the actual optimality gap can only be no more than the resulting ratio).

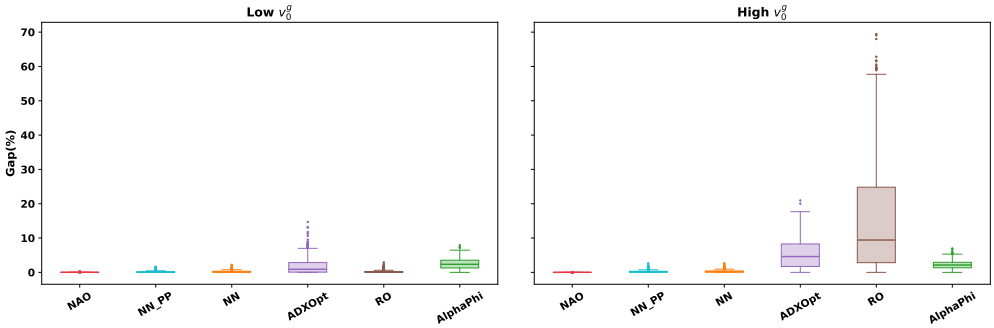
We run our method `NAO` with the following default setup unless otherwise specified. The number of particles is set to $M = 5000$, which we empirically found as a sufficient number for achieving near-optimality in all cases (although in many cases this choice is more than sufficient). The weight decay parameter λ and noise level τ are tuned for each configuration. Note that, unlike statistical learning problems, we do not have any generalization concerns for this optimization problem. A standard selection strategy is to choose the best-performing hyperparameter pairs in a grid. The stopping criterion is to terminate after 1,000 iterations or stop early if no improvement is observed over 400 cumulative iterations. For the unconstrained MMNL instances, a reduced budget is adopted (200 iterations and 50 patience), which is sufficient to ensure convergence. We report the average performance across all 100 instances (Section 6.1) and analyze the optimality gaps for the hardest instances as a stress test (Section 6.2). Moreover, the scalability of `NAO` is examined with larger problem sizes (Section 6.3). Finally, ablation studies on hyperparameters are provided in Appendix D.3.

6.1 Overall Performance for All Instances

Using the full public dataset in Guo et al. [28], in which each configuration contains 100 instances described above, we compare the overall performance of `NAO` with that of the benchmark methods in Figure 1. Our method `NAO` consistently outperforms the baselines in two key respects: (i) it achieves the smallest median optimality gap, which is close to zero, indicating near-optimal performance; and (ii) it exhibits the lightest tail behavior and the narrowest dispersion, and thus demonstrates strong robustness. The NN-based baselines perform the second best, followed by the remaining heuristics. Overall, these results demonstrate the consistent effectiveness of `NAO` across diverse choice behaviors, providing strong empirical support for our global convergence theory. For the constrained setting, the observations remain consistent, whose results are deferred in the Appendix (Figure 4 and Figure 7).



(a) MMNL



(b) NL

Figure 1: Boxplots of optimality gap over 100 instances under the unconstrained setting.

6.2 Stress Test for the Hardest Instances

Beyond evaluating average algorithmic performance over 100 instances in [28], we further stress-test the solution quality of `NAO` on a subset of hard instances formed by the union of the 5 hardest instances out of the 100 instances across all evaluated methods for each configuration. This selection ensures that each instance is difficult for at least one benchmark method. For unconstrained problems, Figure 2 and Figure 3 summarize the performance of all benchmarks across problem configurations using heatmaps, where the lighter color indicates better performance. The findings are as follows. (i) `NAO` shows a clear dominating quality of solutions for all scenarios except slightly worse for a single case of 200 items and 5 nests in MMNL (RS2). (ii) `NAO` exhibits a more evident advantage when the candidate set contains items with higher revenue margins (RS4) and when the probability of no purchase is higher (High v_0^g). These findings strengthen our conclusion of the superior empirical performance of `NAO`. For constrained problems, Figures 5 and 8 in the Appendix show consistent trends, with `NAO` winning in most instances across different capacity ratios.

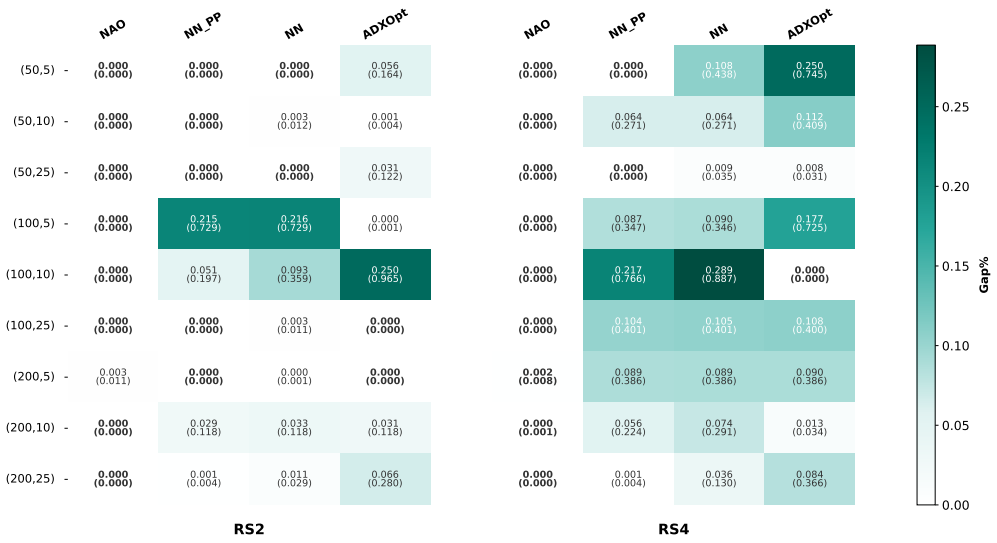


Figure 2: Average Gap and its Standard Deviation on the Hardest Instances of MMNL: Bolded values suggest the best-performed ones or those within a 0.0001% optimality gap. For clearer presentation, RO and AlphaPhi are omitted due to relatively huge gaps. Full results can be found in Table 5.

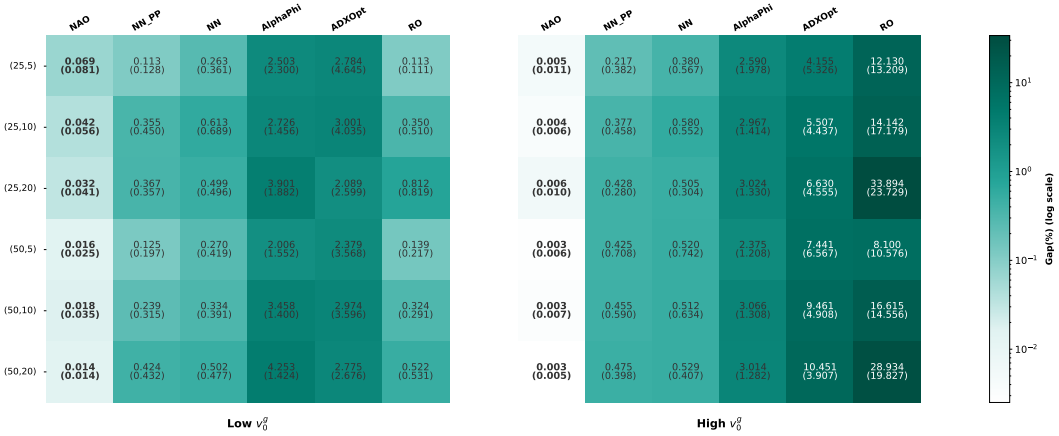


Figure 3: Average Gap and its Standard Deviation on the Hardest Instances of NL: Bolded values suggest the best-performed ones. The heatmap legend is log-scaled to enhance distinct color display.

6.3 Large-scale Experiment

To investigate the scalability and robustness of our method in large-scale settings, we generate another dataset using the same methodology in [28]. We focus on the MMNL case and evaluate across varying problem sizes by fixing the number of classes to $C = 10$ and varying $n \in \{100, 300, 1000, 3000, 10000\}$; both the unconstrained and highly capacitated scenarios (with 0.1 capacity ratio) are tested. For `NAO`, we set $M = 3000$ for efficient computation while maintaining strong performance (as supported by the ablation results in Section D.3.1). Additional details are in Appendix D.4 to ensure a fair comparison across all methods. For example, we fixed the tuning parameters in all methods, including `NAO`, for comparable computational time.

We report in Table 1 and 2 the results on the union of 5 hardest instances over all methods, illustrating how the runtime and (relative) optimality gap vary with problem size. Two key observations emerge. (i) `NAO` demonstrates competitive computational efficiency and scalability across all problem sizes n. As n increases

n	NAO	NN_PP	NN	ADXOpt	AlphaPhi	RO
100	0.46 (0.32)	8.30 (1.15)	8.16 (1.15)	0.02 (0.01)	0.36 (0.07)	**0.00 (0.00)**
300	0.39 (0.06)	9.59 (1.52)	8.74 (1.53)	0.33 (0.18)	1.39 (0.48)	**0.01 (0.00)**
1000	0.63 (0.12)	67.63 (7.01)	26.55 (3.25)	7.77 (3.92)	6.97 (3.96)	**0.11 (0.02)**
3000	1.99 (0.38)	120.52 (9.55)	46.75 (8.72)	89.80 (34.62)	8.82 (3.33)	**0.79 (0.01)**
10000	14.35 (4.22)	N/A	N/A	N/A	37.89 (11.27)	**8.61 (0.05)**
30000	**58.34 (14.15)**	N/A	N/A	N/A	170.78 (9.06)	100.61 (4.70)

(a) Average computation time (s) with sample standard deviation.

n	NAO	NN_PP	NN	ADXOpt	AlphaPhi	RO
100	**0.0001 (0.0000)**	**0.0001 (0.0000)**	0.0072 (0.0139)	**0.0001 (0.0000)**	9.3543 (1.5879)	9.3517 (1.5890)
300	**0.0001 (0.0000)**	0.0225 (0.0317)	0.2045 (0.2611)	**0.0001 (0.0000)**	6.3680 (3.2075)	6.3679 (3.2072)
1000	**0.0016 (0.0035)**	0.1161 (0.0959)	0.4209 (0.5228)	1.1771 (2.6320)	2.8142 (2.8306)	2.5328 (2.7720)
3000	**0.0098 (0.0218)**	6.2730 (7.9947)	6.6565 (7.9952)	3.5840 (5.2211)	1.0158 (2.2166)	1.0115 (2.2190)
10000	**0.0190 (0.0060)**	N/A	N/A	N/A	0.9088 (2.0029)	0.9024 (2.0019)
30000	**0.0286 (0.0244)**	N/A	N/A	N/A	0.8314 (1.8435)	0.8254 (1.8457)

(b) Average optimality gap (%) relative to the best-known solution, with sample standard deviation.

Table 1: Performance comparison under the large-scale unconstrained MMNL setting. Each entry reports the sample mean with the sample standard deviation in parentheses. N/A indicates that the method could not be executed within a reasonable time frame.

(where $n \geq 30000$ for the unconstrianed case and $n \geq 1000$ for the constrained case), NAO achieves the lowest runtime and remains stable across instances, whereas competing methods exhibit much steeper cost growth or become infeasible for large-scale problems. (ii) NAO delivers the best solution quality for all n, attaining the smallest optimality gaps with high stability, except slightly worse for $n = \{100, 300\}$ in Table 2b. Its performance distribution maintains a consistently light dispersion as problem size grows, while other methods show heavier or increasing dispersion.

Overall, NAO is the most scalable, efficient, and effective method among those evaluated.

7 Conclusion

In conclusion, this paper introduces NAO, a choice-model-agnostic framework for assortment optimization with end-to-end near-optimality guarantees and strong empirical performance. The framework is modular: the Lovász extension and entropic-risk objective can, in principle, be replaced by other continuous extensions or search objectives. We focus on this combination because it is scalable and theoretically tractable. A natural next step is to test the method under additional choice models beyond the benchmark families considered here. Beyond assortment optimization, the underlying technique may be useful for other set optimization problems that arise in economics and computer science.

n	NAO	NN_PP	NN	ADXOpt	AlphaPhi
100	0.63 (0.19)	22.09 (3.82)	21.30 (3.83)	**0.01 (0.00)**	0.04 (0.01)
300	0.76 (0.28)	24.75 (3.15)	21.02 (3.19)	**0.32 (0.14)**	1.12 (0.44)
1000	**1.41 (0.46)**	63.93 (5.93)	24.02 (3.45)	8.11 (3.32)	36.44 (15.31)
3000	**3.55 (0.50)**	88.88 (6.70)	18.60 (2.53)	73.93 (35.65)	1032.74 (572.33)
10000	**18.64 (6.68)**	N/A	N/A	N/A	N/A

(a) Average computation time (s) with sample standard deviation.

n	NAO	NN_PP	NN	ADXOpt	AlphaPhi
100	0.0326 (0.0513)	**0.0000 (0.0000)**	**0.0000 (0.0000)**	**0.0000 (0.0000)**	5.8367 (4.1150)
300	0.0005 (0.0010)	**0.0000 (0.0000)**	0.0018 (0.0039)	**0.0000 (0.0000)**	4.1169 (2.1568)
1000	**0.0027 (0.0059)**	0.1344 (0.0812)	0.4483 (0.5019)	1.1771 (2.6320)	1.8931 (2.1992)
3000	**0.2542 (0.5647)**	4.5145 (3.6873)	4.8376 (3.6229)	3.3355 (5.3701)	1.0130 (2.2182)
10000	**0.0000 (0.0000)**	N/A	N/A	N/A	N/A

(b) Average optimality gap (%) relative to the best-known solution, with sample standard deviation.

Table 2: Performance comparison under the large-scale constrained MMNL setting. Each entry reports the sample mean with the sample standard deviation in parentheses. N/A indicates that the method could not be executed within a reasonable time frame.

References

[1] Tarek Abdallah, Mohammad Amin Farzaneh, and Ashwin Venkataraman. Leveraging assortment similarities for data-driven demand prediction. *Available at SSRN 5133673*, 2025.

[2] Shipra Agrawal, Vashist Avadhanula, Vineet Goyal, and Assaf Zeevi. Mnl-bandit: A dynamic learning approach to assortment selection. *Operations Research*, 67(5):1453–1485, 2019.

[3] Lin An, Andrew A Li, Vaisnavi Nemala, and Gabriel Visotsky. Near-optimal real-time personalization with simple transformers. In *NeurIPS 2025 Workshop MLxOR: Mathematical Foundations and Operational Integration of Machine Learning for Uncertainty-Aware Decision-Making*, 2025.

[4] Ali Aouad, Retsef Levi, and Danny Segev. Approximation algorithms for dynamic assortment optimization models. *Mathematics of Operations Research*, 44(2):487–511, 2019.

[5] Gerardo Berbeglia and Gwenaël Joret. Assortment optimisation under a general discrete choice model: A tight analysis of revenue-ordered assortments. *Algorithmica*, 82(4):681–720, 2020.

[6] Jose Blanchet, Guillermo Gallego, and Vineet Goyal. A markov chain approximation to choice modeling. *Operations Research*, 64(4):886–905, 2016.

[7] Joris JM Bront, Isabel Mendez-Diaz, and Gustavo Vulcano. A column generation algorithm for choice-based network revenue management. *Operations Research*, 57(3):769–784, 2009.

[8] Sebastien Bubeck, Ronen Eldan, and Joseph Lehec. Finite-time analysis of projected langevin monte carlo. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[9] Fan Chen, Zhenjie Ren, and Songbo Wang. Uniform-in-time propagation of chaos for mean field Langevin dynamics. *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, 61(4):2357 – 2404, 2025.

[10] Ningyuan Chen, Saman Lagzi, and Joseph Milner. Using neural networks to guide data-driven operational decisions. *Available at SSRN 4217092*, 2022.

[11] Xi Chen, Yining Wang, and Yuan Zhou. Dynamic assortment optimization with changing contextual information. *Journal of machine learning research*, 21(216):1–44, 2020.

[12] Frank H. Clarke. *Optimization and Nonsmooth Analysis*, volume 5 of *Classics in Applied Mathematics*. SIAM, Philadelphia, PA, 1990.

[13] Alper Şen, Alper Atamtürk, and Philip Kaminsky. Technical note—a conic integer optimization approach to the constrained assortment problem under the mixed multinomial logit model. *Operations Research*, 66(4):994–1003, 2018.

[14] Bernard Dacorogna. *Direct methods in the calculus of variations*. Springer, 2008.

[15] James M Davis, Guillermo Gallego, and Huseyin Topaloglu. Assortment optimization under variations of the nested logit model. *Operations Research*, 62(2):250–273, 2014.

[16] James M Davis, Guillermo Gallego, and Huseyin Topaloglu. Assortment optimization under variants of the nested logit model. *Operations Research*, 62(2):250–273, 2014.

[17] Antoine Désir, Vineet Goyal, Srikanth Jagabathula, and Danny Segev. Assortment optimization under the mallows model. *Advances in Neural Information Processing Systems*, 29, 2016.

[18] Antoine Désir, Vineet Goyal, Danny Segev, and Chun Ye. Constrained assortment optimization under the markov chain–based choice model. *Management Science*, 66(2):698–721, 2020.

[19] Antoine Désir, Vineet Goyal, and Jiawei Zhang. Technical note—capacitated assortment optimization: Hardness and approximation. *Operations Research*, 70(2):893–904, 2022.

[20] Paul Dupuis and Richard S Ellis. *A Weak Convergence Approach to the Theory of Large Deviations*. John Wiley & Sons, 1997. See Lemma 1.4.3(c).

[21] Lawrence C. Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, R.I., second edition, 2010.

[22] Avner Friedman. *Partial Differential Equations of Parabolic Type*. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1964.

[23] Guillermo Gallego and Huseyin Topaloglu. Constrained assortment optimization for the nested logit model. *Management Science*, 60(10):2583–2601, 2014.

[24] Guillermo Gallego, Pin Gao, Shaoyu Wang, and Gerardo Berbeglia. Assortment optimization with downward feasibility: Efficient heuristics based on independent demands. *SSRN Electronic Journal*, October 2024.

[25] Guillermo Gallego, Srikanth Jagabathula, and Wentao Lu. Efficient local-search heuristics for online and offline assortment optimization. *SSRN Electronic Journal*, May 2024.

[26] Kumar Goutam, Vineet Goyal, and Henry Lam. Assortment optimization over dense universe is easy. *Available at SSRN 3649233*, 2020.

[27] Vineet Goyal, Retsef Levi, and Danny Segev. Near-optimal algorithms for the assortment planning problem under dynamic substitution and stochastic demand. *Operations Research*, 64(1):219–235, 2016.

[28] Qing Guo, Saman Lagzi, Chenhao Wang, Ningyuan Chen, Guillermo Gallego, Sumit Kunnumkal, Yao Wang, and Li Yu. Solving assortment optimization with first-order methods and neural networks: A computational framework and public benchmark. SSRN Electronic Journal, 2025.

[29] Taotao He, Yating Zhang, and Huan Zheng. Assortment optimization under history-dependent effects. *arXiv preprint arXiv:2408.10967*, 2024.

[30] Julia Heger and Robert Klein. Assortment optimization: A systematic literature review. *OR Spectrum*, 46(4):1099–1161, 2024.

[31] Richard Holley and Daniel W. Stroock. Logarithmic sobolev inequalities and stochastic ising models. *Journal of Statistical Physics*, 46(5-6):1159–1194, March 1987.

[32] Kaitong Hu, Zhenjie Ren, David Šiška, and Lukasz Szpruch. Mean-field Langevin dynamics and energy landscape of neural networks. *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, 57(4):2043 – 2065, 2021.

[33] Shinji Ito. Submodular function minimization with noisy evaluation oracle. *Advances in Neural Information Processing Systems*, 32, 2019.

[34] Srikanth Jagabathula. Assortment optimization under general choice. *Available at SSRN 2512831*, 2014.

[35] Nikolaos Karalias, Joshua Robinson, Andreas Loukas, and Stefanie Jegelka. Neural set function extensions: Learning with discrete functions in high dimensions. *Advances in Neural Information Processing Systems*, 35:15338–15352, 2022.

[36] Hassaan Khalid and Bradley Sturt. Assortment optimization and the sample average approximation. *arXiv preprint arXiv:2510.00850*, 2025.

[37] Sumit Kunnumkal. Technical note—new bounds for cardinality-constrained assortment optimization under the nested logit model. *Operations Research*, 71(4):1112–1119, 2023.

[38] Andrew Lamperski. Projected stochastic gradient langevin algorithms for constrained sampling and non-convex learning. In Mikhail Belkin and Samory Kpotufe (eds.), *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pp. 2891–2937. PMLR, 15–19 Aug 2021.

[39] Dominique Lépingle. Euler scheme for reflected stochastic differential equations. *Mathematics and Computers in Simulation*, 38(1):119–126, 1995.

[40] Guang Li, Paat Rusmevichientong, and Huseyin Topaloglu. The nested logit model: A concave maximization reformulation and an optimization approach. *Operations Research*, 63(5):1210–1230, 2015.

[41] Guokai Li, Pin Gao, Stefanus Jasin, and Zizhuo Wang. From small to large: A graph convolutional network approach for solving assortment optimization problems. *arXiv preprint arXiv:2507.10834*, 2025.

[42] Tao Li, Chenhao Wang, Yao Wang, Shaojie Tang, and Ningyuan Chen. Deep reinforcement learning for online assortment customization: A data-driven approach. *Available at SSRN 4870298*, 2024.

[43] Pierre-Louis Lions and Alain-Sol Sznitman. Stochastic differential equations with reflecting boundary conditions. *Communications on Pure and Applied Mathematics*, 37(4):511–537, 1984.

[44] L. Lovász. Submodular functions and convexity. In *Mathematical Programming: The State of the Art*, pp. 235–257. Springer, 1983.

[45] R Duncan Luce. *Individual choice behavior: A theoretical analysis*. John Wiley & Sons, 1959.

[46] Tarika Mane. Convergence of reflected langevin diffusion for constrained sampling. *arXiv preprint arXiv:2512.00386*, 2025.

[47] Daniel McFadden. Conditional logit analysis of qualitative choice behavior. *Frontiers in Econometrics*, pp. 105–142, 1973.

[48] Daniel McFadden and Kenneth Train. Mixed mnl models for discrete response. *Journal of Applied Econometrics*, 15(5):447–470, 2000.

[49] Atsushi Nitanda, Denny Wu, and Taiji Suzuki. Convex analysis of the mean field langevin dynamics. In *International Conference on Artificial Intelligence and Statistics*, pp. 9741–9757. PMLR, 2022.

[50] Zijing Ou, Tingyang Xu, Qinliang Su, Yingzhen Li, Peilin Zhao, and Yatao Bian. Learning neural set functions under the optimal subset oracle. *Advances in Neural Information Processing Systems*, 35: 35021–35034, 2022.

[51] Andrea Pascucci. *PDE and Martingale Methods in Option Pricing*, volume 2 of *Bocconi & Springer Series*. Springer-Verlag Italia, Milano, 2011. ISBN 978-88-470-1780-1.

[52] Yannik Peeters, Arnoud V. den Boer, and Michel Mandjes. Continuous assortment optimization with logit choice probabilities and incomplete information. *Operations Research*, 70(3):1613–1628, 2022. doi: 10.1287/opre.2021.2235. URL `https://doi.org/10.1287/opre.2021.2235`.

[53] Andrey Pilipenko. *An introduction to stochastic differential equations with reflection*. Universitätsverlag Potsdam, 2014.

[54] Meng Qi, Ho-Yin Mak, and Zuo-Jun Max Shen. Data-driven research in retail operations—a review. *Naval Research Logistics (NRL)*, 67(8):595–616, 2020.

[55] Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a global convergence guarantee. In *Conference on Learning Theory (COLT)*, pp. 1674–1703. PMLR, 2017.

[56] Stefan Rogosinski, Sven Müller, and Lorena Reyes-Rubiano. Distribution-specific approximation guarantees for the random-parameters logit assortment problem. *European Journal of Operational Research*, 2025. ISSN 0377-2217.

[57] Paat Rusmevichientong and Huseyin Topaloglu. Robust assortment optimization in revenue management under the multinomial logit choice model. *Operations research*, 60(4):865–882, 2012.

[58] Paat Rusmevichientong, Zuo-Jun Max Shen, and David B Shmoys. Assortment optimization under the multinomial logit model with random choice parameters. *Production and Operations Management*, 23 (11):2023–2039, 2014.

[59] Leszek S lomiński. Euler's approximations of solutions of SDEs with reflecting boundary. *Stochastic Processes and their Applications*, 94(2):317–337, August 2001.

[60] Taiji Suzuki, Denny Wu, and Atsushi Nitanda. Convergence of mean-field langevin dynamics: Time and space discretization, stochastic gradient, and variance reduction. In *Advances in Neural Information Processing Systems 36*, 2023.

[61] Kalyan Talluri and Garrett Van Ryzin. Revenue management under a general discrete choice model of consumer behavior. *Management Science*, 50(1):15–33, 2004.

[62] Hiroshi Tanaka. Stochastic differential equations with reflecting boundary condition in convex regions. *Hiroshima Mathematical Journal*, 9(1):163–177, 1979.

[63] K. Train. *Discrete Choice Methods with Simulation*. Cambridge University Press, 2 edition, 2009.

[64] Rajan Udwani. Submodular order functions and assortment optimization. In *International Conference on Machine Learning*, pp. 34584–34614. PMLR, 2023.

[65] Cédric Villani. *Optimal Transport: Old and New*, volume 338 of *Grundlehren der mathematischen Wissenschaften*. Springer, Berlin, Heidelberg, 2009.

[66] Feng-Yu Wang. Modified curvatures on manifolds with boundary and applications. *Potential Analysis*, 41(3):699–714, 2014.

[67] Hanzhao Wang, Zhongze Cai, Xiaocheng Li, and Kalyan Talluri. A neural network based choice model for assortment optimization. *arXiv preprint arXiv:2308.05617*, 2023.

[68] Klas Wijk, Ricardo Vinuesa, and Hossein Azizpour. SFESS: Score function estimators for k-subset sampling. In *The Thirteenth International Conference on Learning Representations*, 2025.

[69] HCWL Williams. On the formation of travel demand models and economic evaluation measures of user benefit. *Environment and planning A*, 9(3):285–344, 1977.

[70] Haixiang Zhang, Zeyu Zheng, and Javad Lavaei. Gradient-based algorithms for convex discrete optimization via simulation. *Operations research*, 71(5):1815–1834, 2023.

A Lovász extension and particle loss

This appendix provides the mathematical foundations related to the continuous relaxation used in our algorithm. We first detail the subgradient calculus of the Lovász extension, establishing the validity of the sorting-based gradient oracle. Subsequently, we analyze the properties of the particle loss, including its preservation of the global optimum.

A.1 Subgradient calculus for the Lovász extension

This subsection details the differentiability properties of ϕ and supports Proposition 3. We summarize the structural regularity of ϕ and record a standard characterization of its subdifferential.

Specifically, the following lemma formally establishes that ϕ is piecewise affine over ordering regions and globally Lipschitz continuous on the hypercube $[0,1]^n$.

Lemma 2 (Regularity of the Lovász extension)**.** *Let ϕ be the Lovász extension of f.*

1. *Piecewise affinity: For any permutation π, define the strict-order region*

$$C_\pi := \left\{x \in [0,1]^n : x_{\pi(1)} > x_{\pi(2)} > \cdots > x_{\pi(n)}\right\}.$$

   On each C_π, ϕ is affine.

2. *Lipschitz continuity: Let*

$$G := \max_{S \subseteq [n],\ j \notin S} |f(S \cup \{j\}) - f(S)|.$$

   Then for all $x, y \in [0,1]^n$,

$$|\phi(x) - \phi(y)| \le G\|x - y\|_1.$$

Proof. The proof is deferred to Appendix C.1.3. □

Lemma 2 implies that ϕ has well-defined classical gradients on each strict-order region C_π. However, when ties occur (i.e., $x_i = x_j$ for some $i \neq j$), the ordering changes across adjacent regions and ϕ becomes non-differentiable. Since standard gradients are undefined at these points, we utilize the *Clarke subdifferential* [12], a generalized gradient which aggregates limiting gradients from nearby differentiable points.

Definition 3 (Clarke Subdifferential)**.** *For a locally Lipschitz function $h : \mathbb{R}^n \to \mathbb{R}$, the Clarke subdifferential at x is*

$$\partial^\circ h(x) := \operatorname{conv}\left\{ \lim_{j\to\infty} \nabla h(x^j) \ :\ x^j \to x,\ h \text{ differentiable at } x^j \right\}. \tag{20}$$

At points where h is differentiable, $\partial^\circ h(x) = \{\nabla h(x)\}$.

Since the Lovász extension ϕ is locally Lipschitz and piecewise affine, $\partial^\circ\phi(x)$ provides a convenient notion of generalized gradient at points where coordinate ties occur. We next recall the gradient on strict-ordering regions and then take limits to obtain $\partial^\circ\phi(x)$.

Fix a permutation π and consider a strict-order region C_π on which the ordering is preserved. On C_π, the chain $(S_i(x))_{i=0}^n$ is fixed and ϕ is affine; hence its (classical) gradient is constant and equals the marginal vector g^π, whose components are given by the oracle in (6). In particular, for any $x \in C_\pi$ we have $\nabla\phi(x) = g^\pi$.

At a point x with ties, ϕ is typically non-differentiable. In this case, $\partial^\circ\phi(x)$ is obtained by taking the convex hull of the limiting gradients from neighboring strict-ordering regions, which leads to the following characterization.

Proposition 5 (Clarke subdifferential of the Lovász extension)**.** *For any $x \in [0,1]^n$, let $\Pi(x)$ denote the set of permutations consistent with sorting x in non-increasing order (ties allowed). Then*

$$\partial^\circ \phi(x) = \operatorname{conv}\{g^\pi(x) : \pi \in \Pi(x)\}.$$

In particular, for any tie-breaking permutation $\pi \in \Pi(x)$, the marginal vector $g^\pi(x)$ is a valid generalized subgradient at x.

Proof. The proof is deferred to Appendix C.1.4. □

This proposition is practically significant: it implies that to run a (sub)gradient-based update, we do not need numerical differentiation. In practice, we obtain a valid Clarke subgradient by sorting x to form the chain $(S_i(x))_{i=0}^n$ and evaluating the marginals $f(S_i(x)) - f(S_{i-1}(x))$ to construct $g^\pi(x)$. In particular, this avoids numerical differentiation and leads to the implementable oracle in Proposition 3.

A.2 General Particle Loss

Consider the generalized multi-particle loss with a scaling parameter $\beta > 0$:

$$L(\mathbf{x}) := -\frac{1}{\beta} \log\Big(\frac{1}{M} \sum_{m=1}^{M} \exp\big(-\beta\, \phi(X^m)\big)\Big). \tag{21}$$

The parameter β controls the accuracy of the soft-min approximation: larger β places more emphasis on the best particle. Our main text definition corresponds to the case $\beta = 1$.

The standard log-sum-exp inequality yields that

$$\min_{1\le m\le M} \phi(X^m) \;\le\; L(\mathbf{x}) \;\le\; \min_{1\le m\le M} \phi(X^m) + \frac{\log M}{\beta}. \tag{22}$$

The lower bound ensures that minimizing the surrogate L necessarily pushes down the minimum objective value among the particles. The upper bound shows that the approximation error is at most $\frac{\log M}{\beta}$. Notably, the particle loss preserves the exact global optimum of the original problem.

Proposition 6 (Optimality preservation of the particle loss)**.** *For any $M \ge 1$ and $\beta > 0$,*

$$\min_{\mathbf{x}\in\mathcal{X}^M} L(\mathbf{x}) \;=\; \min_{x\in\mathcal{X}} \phi(x) = \min_{S\subseteq[n]} f(S).$$

Proof. Let $\phi^\star := \min_{x\in\mathcal{X}} \phi(x)$. For any $\mathbf{x} \in \mathcal{X}^M$, we have $\min_m \phi(X^m) \ge \phi^\star$, and by (22), $L(\mathbf{x}) \ge \min_m \phi(X^m) \ge \phi^\star$, hence $\min_{\mathbf{x}} L(\mathbf{x}) \ge \phi^\star$. Conversely, take $x^\star \in \arg\min_{x\in\mathcal{X}} \phi(x)$ and set $x_1 = \cdots = X^m = x^\star$. Then $L(\mathbf{x}) = \phi(x^\star) = \phi^\star$, so $\min_{\mathbf{x}} L(\mathbf{x}) \le \phi^\star$. Combining the two inequalities proves the exactness. □

Accordingly, the soft-min objective L induces the normalized weights

$$w^m(\mathbf{x}) := \frac{\exp\big(-\beta\, \phi(X^m)\big)}{\sum_{j=1}^M \exp\big(-\beta\, \phi(x_j)\big)}, \qquad m = 1, \dots, M,$$

which satisfy $w^m(\mathbf{x}) \ge 0$ and $\sum_{m=1}^M w^m(\mathbf{x}) = 1$. These weights naturally emphasize particles with smaller objective values and appear in the gradient and stability analysis of the particle dynamics.

All results in this appendix are stated and proved for the general loss L in (21). The formulation used in the main text corresponds to the special case $\beta = 1$, or equivalently to rescaling the surrogate ϕ by a positive constant. Since such a rescaling does not affect the location of minimizers, all guarantees carry over directly.

Furthermore, our results extend straightforwardly to the cardinality-constrained setting. The constrained surrogate ϕ_K introduced in Section 4 satisfies the same regularity conditions—specifically, Lipschitz continuity and piecewise linearity—established in Lemma 2. Consequently, all convergence guarantees apply analogously when ϕ is replaced by ϕ_K.

B Detailed Summary of Performance Guarantees

This appendix provides technical details supporting the guarantees in Section 5. At a high level, Algorithm 1 can be viewed as a discretized implementation of an interacting Langevin-type dynamics with a feasibility mechanism on the hypercube $\mathcal{X} = [0,1]^n$. The injected Gaussian perturbations correspond to a Langevin/noisy-gradient viewpoint, the soft-min weights induce a mean-field interaction, and the projection step enforces the hypercube constraints.

Formally, Algorithm 1 constitutes a discrete-time approximation of *Reflected Mean-Field Langevin Dynamics* (MFLD) (e.g, [8, 38]). In the vanishing-step-size limit ($\eta \to 0$), the projected updates converge to a *reflected* diffusion process on the hypercube $\mathcal{X}$, i.e., a continuous-time dynamics equipped with a boundary reflection term that enforces feasibility. This reflected perspective is the main geometric distinction from standard MFLD analyses on $\mathbb{R}^n$ (e.g., 32, 49). The convergence properties of such projected/reflected Langevin dynamics on constrained domains justify projection/reflection as a principled way to handle constraints in sampling and optimization (e.g., 38, 46).

With this continuous-time mean-field perspective, we model the optimization process as a continuous interacting particle system evolving over the time interval $[0, T]$, where T represents the total optimization duration. Accordingly, our theoretical analysis focuses on bounding the *continuous-time expected optimality gap*:

$$r^\star - \mathbb{E}\left[r(\bar{S}(T))\right],$$

where $\bar{S}(T)$ is obtained by rounding each particle state $X^m(T)$ at time $T > 0$ and selecting the best rounded set (as in Sections 3.4 and 5.2). The expectation is taken over the randomness of the Brownian motion and initialization.

Notation. Throughout this appendix, we denote the single-particle domain by $\mathcal{X} = [0,1]^n$. To distinguish a single particle from the full M-particle configuration, we use boldface for objects in the product space. In particular, a point in $\mathcal{X}^M$ is written as $\mathbf{x} = (x^1, \ldots, x^M) \in \mathcal{X}^M = [0,1]^{n \times M}$, where the superscript $m \in \{1, \ldots, M\}$ indexes the m-th particle. For the dynamics, $X^m(t)$ denotes the state of particle m at time t, and $\mathbf{X}(t) = (X^1(t), \ldots, X^M(t))$ denotes the joint stochastic process on $\mathcal{X}^M$.

B.1 Continuous-Time Dynamics and Mean-Field Limit

We now formalize the limiting dynamics and identify the energy functionals that drive the optimization process. Our analysis proceeds by first characterizing the finite-M reflected diffusion and its associated *finite-particle functional*, and subsequently describing the limiting behavior governed by the *mean-field functional* as $M \to \infty$.

B.1.1 The Reflected Particle System.

By treating the step size η in the update rule (9) as an infinitesimal time increment dt, this recursion can be viewed as the Euler-Maruyama discretization of a continuous-time stochastic process. Specifically, in the limit $\eta \to 0$, the dynamics of the particle system $\mathbf{X}(t) = (X^1(t), \ldots, X^M(t))$ are governed by the following system of coupled *Reflected Stochastic Differential Equations (SDEs)* (or Skorokhod problem; see, e.g., 53) : for $m = 1, \ldots, M$,

$$dX^m(t) = -\underbrace{\left(M w^m(\mathbf{X}(t)) g(X^m(t)) + \lambda X^m(t)\right)}_{\text{Drift Force}} dt + \sqrt{2\tau} dB^m(t) + dR^m(t), \tag{23}$$

where $(B^m(t))_{m=1}^M$ are independent standard Brownian motions, $g(x) \in \partial^\circ \phi(x)$ is a subgradient, and $R^m(t)$ is the boundary reflection process (Skorokhod reflection), which enforces the constraint $X^m(t) \in \mathcal{X}$ by exerting

a reflection force along the inward normal vector whenever the particle hits the boundary $\partial\mathcal{X}$ [39, 59]. Here, $\tau > 0$ represents the noise level, and $\lambda > 0$ controls the weight decay strength. We use the notation $X^m(t)$ to denote the state of the m-th particle at time t, keeping the particle index m as a superscript.

The drift term drives the evolution: the weighted gradient pushes candidates toward high-revenue assortments, while the regularization term $\lambda X^m(t)$ centers the search to avoid overfitting.

By leveraging the convexity of the domain and the boundedness of the joint drift (for each fixed M), standard results for reflected diffusions guarantee the *existence of a weak solution* and its *uniqueness in law* [43, 62]. This ensures that the evolution of the probability distribution ρ_t is uniquely determined. We refer the reader to Appendix C.3 for a detailed verification.

B.1.2 The Global Objective Functional.

A key observation is that the drift force in (23) is not arbitrary; it corresponds to the subgradient of a unified potential function. Recall the empirical soft-min loss $L(\mathbf{X})$ defined in (5). A direct calculation shows that the weighted subgradient term satisfies that

$$Mw^m(\mathbf{X}(t))g(X_t^m) \in \partial_{X^m}\left(ML(\mathbf{X}(t))\right).$$

Consequently, the particle system $\mathbf{X}_t$ can be characterized as the (sub)gradient flow minimizing the finite-particle functional $\mathcal{F}^{(M)} : \mathcal{P}(\mathcal{X}^M) \to \mathbb{R}$, defined as:

$$\mathcal{F}^{(M)}(\rho^{(M)}) := \int_{\mathcal{X}^M} ML(\mathbf{x})\rho^{(M)}(d\mathbf{x}) + \tau D_{\mathrm{KL}}(\rho^{(M)} \| \gamma^{\otimes M}), \tag{24}$$

where $\rho^{(M)}$ denotes the joint distribution of the particles $\mathbf{X}$ on $\mathcal{X}^M$, $\gamma^{\otimes M}$ is the product measure of the reference prior γ on $\mathcal{X}$, and γ is the truncated Gaussian measure on $\mathcal{X}$:

$$\gamma(dx) = \frac{1}{Z_\gamma} \exp\left(-\frac{\|x\|_2^2}{2\sigma^2}\right) \mathbf{1}_{\mathcal{X}}(x)\, dx,$$

where the variance is set to $\sigma^2 := \tau/\lambda$.

This functional (24) reveals the dual implicit objectives driving the particle system. The first term, representing the expected soft-min loss, steers the entire collection of candidate assortments toward high-revenue regions of the landscape, effectively performing collective exploitation. Simultaneously, the KL divergence term acts as a functional regularizer; it prevents the system from premature collapse into a single strategy by penalizing overly concentrated distributions, thereby ensuring a diverse exploration of the assortment space and stabilizing the dynamics on the bounded domain $\mathcal{X}^M$.

B.1.3 Mean-Field Limit ($M \to \infty$).

To derive global guarantees, we consider the many-particle limit ($M \to \infty$). In this idealized regime, the system exhibits the *Propagation of Chaos* property, implying that particles become asymptotically independent and the marginal distribution of any single particle converges to a deterministic density ρ. Simultaneously, by the Law of Large Numbers, the empirical average inside the loss function converges to the population expectation, and then the finite-particle energy converges to the mean-field functional $\mathcal{F} : \mathcal{P}(\mathcal{X}) \to \mathbb{R}$, defined as:

$$\mathcal{F}(\rho) := \underbrace{-\frac{1}{\beta}\log \mathbb{E}_{X\sim\rho}\left[e^{-\beta\phi(X)}\right]}_{L(\rho)} + \tau D_{\mathrm{KL}}(\rho\|\gamma).$$

Let $\rho^\tau := \arg\min_{\rho\in\mathcal{P}(\mathcal{X})} \mathcal{F}(\rho)$ denote the unique minimizer of this functional. The well-posedness of this definition is guaranteed by the Direct Method in the Calculus of Variations [14]. Specifically, the existence of a minimizer follows from the weak lower semi-continuity of $\mathcal{F}$ on the compact domain $\mathcal{P}(\mathcal{X})$ [20], while uniqueness holds because $L(\rho)$ is convex in ρ and $D_{\mathrm{KL}}(\rho\|\gamma)$ is strictly convex, hence the functional $\mathcal{F}$ is strictly convex.

B.2 Optimality Gap Decomposition

We now derive the global convergence guarantee. Let $\rho_T^{(M)}$ denote the joint probability distribution of the particle system $\mathbf{X}(T)$ at time T. From the definition of the finite-particle functional (24), we observe that:

$$\frac{1}{M}\mathcal{F}^{(M)}(\rho_T^{(M)}) = \mathbb{E}_{\mathbf{X}\sim\rho_T^{(M)}}[L(\mathbf{X})] + \frac{\tau}{M}D_{\mathrm{KL}}(\rho_T^{(M)}\|\gamma^{\otimes M}) \geq \mathbb{E}_{\mathbf{X}\sim\rho_T^{(M)}}[L(\mathbf{X})].$$

Combining this with the inequality (17) (taking expectations) yields the crucial bridge that

$$-\mathbb{E}[r(\bar{S}(T))] \leq \mathbb{E}[L(\mathbf{X}(T))] \leq \frac{1}{M}\mathcal{F}^{(M)}(\rho_T^{(M)}).$$

With this connection established, we present our main decomposition theorem.

Theorem 2 (Performance Decomposition)**.** *Let $\rho_T^{(M)}$ be the distribution of the M-particle system at time T. The expected optimality gap satisfies that*

$$\begin{aligned} r^\star - \mathbb{E}[r(\bar{S}(T))] &\leq \mathbb{E}\left[\frac{1}{M}\mathcal{F}^{(M)}(\rho_T^{(M)}) - \phi^\star\right] \\ &= \underbrace{\mathbb{E}\left[\frac{1}{M}\mathcal{F}^{(M)}(\rho_T^{(M)}) - \mathcal{F}(\rho^\tau)\right]}_{\text{(I) Finite-Particle Optimization error}} + \underbrace{\left(\mathcal{F}(\rho^\tau) - \phi^\star\right)}_{\text{(II) Regularization Bias}}. \end{aligned}$$

This theorem provides a clear interpretation of the algorithm's performance drivers: Term (I) (optimization error) measures how far the M-particle distribution at time T is from the mean-field minimizer of the regularized objective. As we show later, this error decays exponentially, meaning the swarm quickly identifies the most promising regions of the assortment space. Term (II) (regularization bias) represents the bias introduced by the regularization parameters (τ, λ). This term vanishes as the regularization parameters $\tau, \lambda \to 0$, recovering the exact optimal revenue.

B.3 Finite-Particle Optimization Convergence

In this section, we establish the bound on the finite-particle optimization error:

$$\mathbb{E}\left[\frac{1}{M}\mathcal{F}^{(M)}(\rho_T^{(M)}) - \mathcal{F}(\rho^\tau)\right].$$

Here, the difference measures the performance gap (in terms of functional) between the distribution $\rho_T^{(M)}$ generated by our M-particle Reflected Langevin Dynamics after time T, against the optimal mean-field distribution ρ^τ that minimizes the regularized objective. As a reminder, the expectation is taken over the stochasticity of the Brownian motions injected into the dynamics. The goal of this subsection is to establish how this optimization error decays as the simulation time T and the number of particles M increase. We have the following result.

Proposition 7. *Under the setting of Theorem 1, for sufficiently large M, it holds that*

$$\mathbb{E}\left[\frac{1}{M}\mathcal{F}^{(M)}(\rho_T^{(M)}) - \mathcal{F}(\rho^\tau)\right] \le e^{-\tilde{\kappa}T}\left(\frac{1}{M}\mathcal{F}^{(M)}(\rho_0^{(M)}) - \mathcal{F}(\rho^\tau)\right) + \frac{C}{\tilde{\kappa}M},$$

where $\rho_0^{(M)}$ is the initial joint particle distribution.

In this bound, the main term $e^{-\tilde{\kappa}T}$ indicates a exponential decay in the optimization gap from the initial state as the simulation time T increases. The effective rate $\tilde{\kappa}$ is governed by the convexity and geometry of the regularized energy landscape on the compact hypercube: a stronger convexity of the regularized functional, leading to faster mixing of the reflected diffusion process and thus quicker convergence to the minimizer ρ^τ. Crucially, Appendix C.4.8 shows that for all sufficiently large M, the effective rate $\tilde{\kappa}$ can be chosen strictly positive and uniform in M. The proof combines a functional-inequality estimate at the mean-field limit (via a log-Sobolev argument) with a finite-M perturbation bound.

The remainder term in the bound, of the order $O(1/M)$, quantifies the finite-particle approximation error due to the finite number of particles. It captures the variance arising from approximating the mean-field interaction (which depends on the full distribution) with an empirical average over M particles.

Explicit statements and detailed derivations are given within Appendix C.4. Below, we provide a sketch of the proof.

Proof Sketch. Our proof strategy adapts the "Uniform-in-time Propagation of Chaos" framework established in Chen et al. [9]. We compare the time-evolution of the finite-particle system $\rho_t^{(M)}$ against the stationary mean-field limit ρ^τ.

We start by deriving the *Dissipation Identity* for Reflected Langevin Dynamics. Unlike the standard Euclidean setting in Chen et al. [9], our domain is bounded; however, we show that the Neumann boundary condition ($J \cdot \mathbf{n} = 0$) enforced by the reflection process ensures that boundary flux terms vanish, allowing us to relate the time derivative of the functional directly to the relative Fisher Information.

To control the convergence rate, following the *Leave-One-Out* technique in Chen et al. [9] which compares the system with and without particle i, we verify that the conditional distributions satisfy functional inequalities that imply geometric contraction. Finally, the error due to finite M arises from the difference between the true particle drift and the idealized mean-field drift. Using Wasserstein transport inequalities, we bound this drift error by $O(1/M)$. Integrating the resulting differential inequality yields the final bound.

B.4 Regularization Bias

In this section, we establish a bound on the regularization bias:

$$\mathcal{F}(\rho^\tau) - \phi^\star.$$

This quantity measures the difference between the minimum energy achieved by the regularized continuous relaxation and the true optimal revenue of the discrete assortment problem. It captures the bias induced by the KL-to-prior regularization $\tau\, D_{\mathrm{KL}}(\rho\|\gamma)$. Under our truncated Gaussian prior γ, this term can be viewed as combining an entropy component (scaled by τ) with a quadratic weight decay (scaled by λ). It prevents the optimizer ρ^τ from collapsing to a pure Dirac mass at the optimum. We have the following result.

Proposition 8 (Regularization Gap)**.** *Under the setting of Theorem 1, for sufficiently small τ, the regularization bias satisfies that*

$$\mathcal{F}(\rho^\tau) - \phi^\star \le O\left(\tau \log\left(\frac{1}{\tau}\right) + \lambda\right).$$

Proposition 8 shows that the regularization bias vanishes as $\tau, \lambda \to 0$, scaling principally as $O(\tau \log(1/\tau))$. We note the explicit linear dependence on the dimension n, which arises from the entropic cost of concentrating a probability measure in an n-dimensional space. Explicit statements and detailed derivations are given within Appendix C.5. A brief proof sketch is provided below.

Proof Sketch. We construct a specific trial measure that (i) concentrates near the true optimizer to minimize the potential energy, but (ii) maintains sufficient volume to control the entropic cost required by the regularizer. We then optimize the trade-off between these two terms to obtain the final bound.

Let x^* be a global minimizer of the potential ϕ. Since the exact Dirac measure δ_{x^*} has infinite entropy, we construct a "smoothed" candidate ρ_ϵ by distributing the mass uniformly on the intersection of the domain $\mathcal{X}$ and a small ball of radius ϵ centered at x^*. By the optimality of ρ^τ, we have that

$$\mathcal{F}(\rho^\tau) - \phi^\star \le \mathcal{F}(\rho_\epsilon) - \phi^\star = \underbrace{(L(\rho_\epsilon) - \phi^\star)}_{\text{Smoothing Error}} + \underbrace{\tau D_{\mathrm{KL}}(\rho_\epsilon \| \gamma)}_{\text{Entropic Cost}}.$$

Due to the Lipschitz continuity of ϕ, the first term scales linearly with the smoothing radius ϵ. The second term, capturing the entropic penalty of localization, scales as $-n\tau \log \epsilon$ due to the volume scaling of the ball in $\mathbb{R}^n$. Optimizing the smoothing strength ϵ balances these opposing forces (localization vs. entropy) and yields the final bound.

B.5 Provable Performance Guarantee

By consolidating the finite-particle convergence rate established in Proposition 7 and the regularization bias bound from Proposition 8, we arrive at our main theoretical result. This theorem provides a complete end-to-end performance guarantee for the assortment $\bar{S}(T)$ extracted from the particle system at time T, serving as the detailed restatement of Theorem 1.

Theorem 3. *Let $r^\star$ be the optimal expected revenue of the discrete assortment problem* (1)*, and let $r(\bar{S}(T))$ be the revenue of the assortment obtained from the continuous-time algorithm at time T. Then, the expected optimality gap satisfies that*

$$\mathbb{E}\left[r^\star - r(\bar{S}(T))\right] \le \underbrace{e^{-\tilde{\kappa}T}\left(\frac{1}{M}\mathcal{F}^{(M)}(\rho_0^{(M)}) - \mathcal{F}(\rho^\tau)\right)}_{\textit{optimization convergence}} + \underbrace{O\left(\frac{1}{\tilde{\kappa}M}\right)}_{\textit{particle approximation}} + \underbrace{O\left(n\tau \log\left(\frac{1}{\tau}\right) + \lambda\right)}_{\textit{regularization bias}}.$$

Theorem 3 clarifies how the computational resources (number of particles M, simulation time T) and the regularization hyperparameters (τ, λ) impact the quality of the learned assortment. The bound decomposes into three distinct components: optimization convergence, particle approximation, and regularization bias.

The optimization convergence term reflects how fast the M-particle dynamics approaches the minimizer of the regularized functional. The factor $e^{-\tilde{\kappa}T}$ captures exponential convergence in time toward the regularized optimum. This confirms that the system rapidly forgets its initialization with a rate $\tilde{\kappa}$ determined by the landscape geometry.

The particle approximation term, scaling with $O(1/M)$, represents the approximation error imposed by the finite number of particles. This captures the particle approximation error arising from approximating the mean-field interaction with M particles; similar to the finite-width error in neural networks, this error vanishes as the particle count $M \to \infty$.

The regularization bias quantifies the gap between the regularized objective and the original discrete problem. This bias is induced by the KL-to-prior regularization $\tau\, D_{\mathrm{KL}}(\rho\|\gamma)$, where τ controls the strength of the KL term and λ enters through the Gaussian prior γ (equivalently, the quadratic weight decay). Increasing

τ or λ typically improves mixing and can speed up convergence (through a larger $\tilde{\kappa}$), but it also increases the bias, moving the solution further away from the unregularized (exact combinatorial) optimum.

B.6 Complexity Analysis

We characterize the computational requirements of Algorithm 1 from two complementary perspectives: (i) how the convergence depends on fixed computational resources (particle size M and continuous time T) for given regularization, and (ii) how these quantities scale in the high-precision regime (target gap $\epsilon \to 0$). Throughout this analysis, the notation $a \asymp b$ denotes asymptotic equivalence, meaning there exist positive constants c_1, c_2 such that $c_1 b \le a \le c_2 b$.

The convergence rate of the particle system is governed by the effective spectral gap

$$\tilde{\kappa} = \kappa_0 - \frac{C_1}{M},$$

where κ_0 is the mean-field rate (depending on τ) and C_1/M captures the finite-particle perturbation; see Appendix C.4.8. In our bound, the interaction constant satisfies $C_1 = O(n^2)$. To ensure $\tilde{\kappa} > 0$ (so that the particle system tracks the mean-field limit with a uniform-in-time contraction), it suffices to take

$$M > M_{\text{critical}} := \frac{C_1}{\kappa_0}.$$

In particular, in regimes where the regularization keeps κ_0 bounded away from 0, this yields the baseline scaling $M = \Omega(n^2)$.

Given $\tilde{\kappa} > 0$, the transient optimization term in Theorem 1 decays as $e^{-\tilde{\kappa}T}\Delta_0$; thus reaching a tolerance level ϵ requires

$$T \;\ge\; \frac{1}{\tilde{\kappa}} \log\!\left(\frac{\Delta_0}{\epsilon}\right).$$

The steady-state floor is controlled by the particle approximation term, which in our derivation takes the form $C/(\tilde{\kappa}M)$ with $C = O(n^3)$. Hence, for fixed (τ, λ) with $\tilde{\kappa}$ bounded below, the particle complexity needed to maintain a prescribed steady-state accuracy scales polynomially in n (here, on the order of n^3).

To drive the overall expected gap below ϵ, we also need to control the regularization bias $O\,(n\tau \log(1/\tau) + \lambda)$. Since the entropic contribution scales linearly with n, a natural choice is

$$\tau \asymp \frac{\epsilon}{n}, \qquad \lambda \asymp \epsilon,$$

which makes the bias term $O(\epsilon)$. Under this scaling of (τ, λ), the mean-field spectral gap κ_0 (and hence $\tilde{\kappa}$) can become exponentially small in $1/\tau$ due to trapping in the local minima of the non-convex landscape, yielding the conservative scaling that

$$\tilde{\kappa} \;\asymp\; \frac{\lambda}{\tau} \exp\!\left(-\frac{O(1)}{\tau}\right) - \frac{C_1}{M} \;\asymp\; n \exp\!\left(-\frac{n}{\epsilon}\right),$$

where the second equivalence holds provided M is chosen large enough ($M > C_1/\kappa_0$) such that the mean-field term dominates the perturbation. Consequently, suppressing the transient term to $O(\epsilon)$ requires an exponentially large runtime that

$$T \;\gtrsim\; \frac{1}{\tilde{\kappa}} \log\!\left(\frac{1}{\epsilon}\right) \;\asymp\; \frac{1}{n} \exp\!\left(\frac{n}{\epsilon}\right) \log\!\left(\frac{1}{\epsilon}\right) \;\asymp\; \exp\!\left(\frac{n}{\epsilon}\right) \log\!\left(\frac{1}{\epsilon}\right),$$

which is consistent with the NP-hardness barrier in the hard-combinatorial limit.

C Detailed Proofs

C.1 Proofs for Section 3.1 and additional details in Appendix A

C.1.1 Exact Relaxation of Lovász Extension (Proof of Proposition 1)

We now present a proof of Proposition 1.

Proof. Fix any $x \in [0,1]^n$ and let π be a permutation such that $x_{\pi(1)} \geq \cdots \geq x_{\pi(n)}$ with $x_{\pi(n+1)} := 0$. Recall the induced chain $S_i(x) := \{\pi(1), \ldots, \pi(i)\}$ and set $S_0(x) := \varnothing$.

By Definition 1, we have that

$$\phi(x) = \sum_{i=1}^{n} \big(x_{\pi(i)} - x_{\pi(i+1)}\big)\, f\big(S_i(x)\big).$$

Recall the weights

$$\alpha_i = x_{\pi(i)} - x_{\pi(i+1)} \geq 0 \quad (i = 1, \ldots, n), \qquad \alpha_0 = 1 - x_{\pi(1)} \geq 0.$$

Then $\sum_{i=0}^{n} \alpha_i = 1$. Assuming $f(\varnothing) = 0$, we may add the null term $\alpha_0 f(\varnothing) = 0$ and rewrite

$$\phi(x) = \sum_{i=0}^{n} \alpha_i f\big(S_i(x)\big),$$

which is a convex combination of the values $\{f(S_i(x))\}_{i=0}^{n}$. Hence, we have that

$$\phi(x) \;\geq\; \min_{0 \leq i \leq n} f\big(S_i(x)\big) \;\geq\; \min_{S \subseteq [n]} f(S).$$

Note that the chain includes $S_0(x) = \varnothing$ (and we set $f(\varnothing) = 0$), but this does not imply that the global minimum is attained at $\varnothing$ nor that $\min_{S \subseteq [n]} f(S) \geq 0$; the second inequality holds simply because $\{S_i(x)\}_{i=0}^{n} \subseteq 2^{[n]}$. Since this holds for all $x \in [0,1]^n$, it follows that

$$\min_{x \in [0,1]^n} \phi(x) \;\geq\; \min_{S \subseteq [n]} f(S). \tag{25}$$

On the other hand, let $S^* \in \arg\min_{S \subseteq [n]} f(S)$ and take $x^* = \mathbf{1}_{S^*} \in [0,1]^n$. Sorting x^* puts all ones before zeros, so there exists an index $i^* := |S^*|$ such that $S_{i^*}(x^*) = S^*$, $x^*_{\pi(i^*)} = 1$, and $x^*_{\pi(i^*+1)} = 0$. Therefore, $\alpha_{i^*} = x^*_{\pi(i^*)} - x^*_{\pi(i^*+1)} = 1$ and $\alpha_i = 0$ for $i \neq i^*$ Note that the case $i^* = 0$ means all differences are zero and the only remaining weight is on $S_0 = \varnothing$. Then, the Lovász formula reduces to

$$\phi(x^*) = \phi(\mathbf{1}_{S^*}) = f\big(S_{i^*}(x^*)\big) = f(S^*).$$

Consequently, we have that

$$\min_{x \in [0,1]^n} \phi(x) \;\leq\; \phi(x^*) \;=\; f(S^*) \;=\; \min_{S \subseteq [n]} f(S). \tag{26}$$

Combining the two inequalities (25) and (26) proves the equivalence $\min_{x \in [0,1]^n} \phi(x) = \min_{S \subseteq [n]} f(S)$. □

C.1.2 Rounding Guarantee

We now present the proof of Lemma 1.

Proof. Fix any $x \in [0,1]^n$ and let π be a permutation such that $x_{\pi(1)} \geq \cdots \geq x_{\pi(n)}$, with $x_{\pi(n+1)} := 0$. Recall the chain $S_i(x) = \{\pi(1), \ldots, \pi(i)\}$ and $S_0(x) := \varnothing$.

Recall the coefficients

$$a_i = x_{\pi(i)} - x_{\pi(i+1)} \geq 0, \qquad i = 1, \ldots, n, \qquad \text{and} \qquad a_0 = 1 - x_{\pi(1)} \geq 0.$$

Then $\sum_{i=0}^{n} a_i = 1$. Moreover, by Definition 1,

$$\phi(x) = \sum_{i=1}^{n} a_i \, f\big(S_i(x)\big).$$

In our setting $f(\varnothing) = 0$, hence we may add the missing mass $a_0 f(\varnothing) = 0$ and rewrite

$$\phi(x) = a_0 f(\varnothing) + \sum_{i=1}^{n} a_i \, f\big(S_i(x)\big) \;\geq\; \min_{0 \leq i \leq n} f\big(S_i(x)\big),$$

because $\phi(x)$ is a convex combination of the values $\{f(S_i(x))\}_{i=0}^{n}$ with weights $\{a_i\}_{i=0}^{n}$.

By definition of the rounding operator, $\mathrm{Round}(x) \in \arg\min_{S \in \{S_i(x)\}_{i=0}^{n}} f(S)$, so $f(\mathrm{Round}(x)) = \min_{0 \leq k \leq n} f(S_i(x))$. Combining the two displays yields that

$$f\big(\,\mathrm{Round}(x)\big) \leq \phi(x),$$

which is exactly (11). □

C.1.3 Proof of Lemma 2

We prove each property separately.

We first prove the piecewise affinity. Fix an arbitrary permutation π and consider the strict-order region C_π. For any $x \in C_\pi$, the sorting order is unique. Recall the explicit formula for the Lovász extension:

$$\phi(x) = \sum_{i=1}^{n} x_{\pi(i)} \left(f(S_i) - f(S_{i-1}) \right),$$

where $S_i = \{\pi(1), \ldots, \pi(i)\}$ and $S_0 = \varnothing$. This expression is clearly linear in x. Thus, on the open region C_π, ϕ is affine (in fact, linear) with a constant gradient $\nabla\phi(x) = g^\pi$, where the components are given by

$$g^\pi_{\pi(i)} = f(S_i) - f(S_{i-1}) \quad \text{for } i = 1, \ldots, n.$$

Since the hypercube $[0,1]^n$ is the union of the closures of finitely many such regions, ϕ is piecewise affine.

Next, we prove the Lipschitz Continuity. We begin by bounding the norm of the gradient on each differentiable region. For any π and any $x \in C_\pi$, the ℓ_∞-norm of the gradient is

$$\|\nabla\phi(x)\|_\infty = \|g^\pi\|_\infty = \max_{1 \leq i \leq n} |f(S_i) - f(S_{i-1})|.$$

By the definition of $G := \max_{S, j \notin S} |f(S \cup \{j\}) - f(S)|$, we immediately have that

$$\|\nabla\phi(x)\|_\infty \leq G, \quad \forall x \in \bigcup_{\pi} C_\pi.$$

Now, consider any $x, y \in [0,1]^n$. Since $[0,1]^n$ is convex, the line segment $\gamma(t) = x + t(y - x)$ for $t \in [0,1]$ lies entirely within the domain. Although ϕ is not everywhere differentiable, it is continuous and piecewise affine, hence absolutely continuous along lines. By the fundamental theorem of calculus for absolutely continuous functions, we have that

$$\phi(y) - \phi(x) = \int_0^1 \langle g(\gamma(t)), y - x \rangle \, dt,$$

where $g(\gamma(t)) \in \partial\phi(\gamma(t))$ is any measurable selection from the subdifferential. Since the boundaries between regions C_π have measure zero, the choice of subgradient at these points does not affect the integral. Thus, applying Hölder's inequality ($\langle u, v \rangle \leq \|u\|_\infty \|v\|_1$) yields that

$$|\phi(y) - \phi(x)| \leq \int_0^1 \|g(\gamma(t))\|_\infty \|y - x\|_1 \, dt \leq \int_0^1 G\|y - x\|_1 \, dt = G\|y - x\|_1.$$

This concludes the proof. □

C.1.4 Proof of Proposition 5

By Lemma 2(2), the Lovász extension ϕ is Lipschitz continuous on $[0,1]^n$. Moreover, since ϕ is explicitly defined via the sorting formula, it extends naturally to a locally Lipschitz function on the entire space $\mathbb{R}^n$. Consequently, the Clarke subdifferential $\partial^\circ\phi(x)$ is well-defined for every $x \in [0,1]^n$.

Recall that ϕ is piecewise affine and hence differentiable on each open strict-order region

$$C_\pi^\circ := \{y \in \mathbb{R}^n : y_{\pi(1)} > \cdots > y_{\pi(n)}\},$$

on which the gradient is constant and equals the marginal vector g^π (see (6)).

We prove the equality by establishing inclusions in both directions.

Inclusion $\supseteq$. Fix any $\pi \in \Pi(x)$. We verify that $g^\pi \in \partial^\circ\phi(x)$. Construct a perturbation vector $d \in \mathbb{R}^n$ strictly decreasing according to π (i.e., $d_{\pi(1)} > \cdots > d_{\pi(n)}$), and define $y^{(\varepsilon)} := x + \varepsilon d$. Consider any pair of indices i, j such that π ranks i before j (implying $x_i \geq x_j$ and $d_i > d_j$). If strictly $x_i > x_j$, then continuity implies $y_i^{(\varepsilon)} > y_j^{(\varepsilon)}$ for sufficiently small ε. If $x_i = x_j$, the strict ordering of d ensures $y_i^{(\varepsilon)} - y_j^{(\varepsilon)} = \varepsilon(d_i - d_j) > 0$ for all $\varepsilon > 0$. Thus, for all sufficiently small $\varepsilon > 0$, $y^{(\varepsilon)} \in C_\pi^\circ$, which implies ϕ is differentiable at $y^{(\varepsilon)}$ with $\nabla\phi(y^{(\varepsilon)}) = g^\pi$. Letting $\varepsilon \downarrow 0$ yields $y^{(\varepsilon)} \to x$, and therefore $g^\pi \in \partial^\circ\phi(x)$. By the convexity of $\partial^\circ\phi(x)$, we conclude $\mathrm{conv}\{g^\pi : \pi \in \Pi(x)\} \subseteq \partial^\circ\phi(x)$.

Inclusion $\subseteq$. By definition (20), $\partial^\circ\phi(x)$ is the convex hull of all limiting gradients. It therefore suffices to show that every limiting gradient belongs to $\{g^\pi : \pi \in \Pi(x)\}$.

Let v be a limiting gradient, i.e., $v = \lim_{\ell\to\infty} \nabla\phi(y^\ell)$ for some sequence $y^\ell \to x$ at which ϕ is differentiable. For each ℓ, differentiability implies that y^ℓ lies in a strict-order region $C_{\pi^\ell}^\circ$, and hence $\nabla\phi(y^\ell) = g^{\pi^\ell}$. Since there are only finitely many permutations, we may extract a subsequence $\{y^{\ell_k}\}$ such that $\pi^{\ell_k} = \hat{\pi}$ is constant. Consequently, $v = \lim_{k\to\infty} \nabla\phi(y^{\ell_k}) = g^{\hat{\pi}}$.

It remains to show that $\hat{\pi} \in \Pi(x)$. Suppose for contradiction that $\hat{\pi} \notin \Pi(x)$. Then there exist indices a, b such that $x_a > x_b$, but $\hat{\pi}$ places b before a. However, since $y^j \in C_{\hat{\pi}}^\circ$, the strict ordering imposed by $\hat{\pi}$ implies $y_b^j > y_a^j$ for all j. Taking the limit as $j \to \infty$, we obtain $x_b \geq x_a$, which contradicts the assumption $x_a > x_b$. Therefore, $\hat{\pi}$ must be consistent with x (i.e., $\hat{\pi} \in \Pi(x)$). We conclude that any limiting gradient v belongs to $\{g^\pi : \pi \in \Pi(x)\}$. Taking the convex hull completes the proof. □

C.2 Proofs for Section 4

Proof of Proposition 4. Let $f_K^\star := \min\{f(S) : S \subseteq [n], |S| \le K\}$. Fix $x \in [0,1]^n$ and let π sort x in non-increasing order, with $x_{\pi(n+1)} := 0$. Define weights

$$w_0 := 1 - x_{\pi(1)}, \qquad w_k := x_{\pi(k)} - x_{\pi(k+1)}, \quad k = 1, \dots, n.$$

Then $w_k \ge 0$ and $\sum_{k=0}^n w_k = 1$. Since $f(\varnothing) = 0$ and $W_0(x) = \varnothing$, we can rewrite (12) as

$$\phi_K(x) = \sum_{k=0}^n w_k \, f\big(W_k(x)\big).$$

Because $|W_k(x)| \le K$ for all k, each $W_k(x)$ is feasible and hence $f(W_k(x)) \ge f_K^\star$. Therefore,

$$\phi_K(x) \ge \sum_{k=0}^n w_k \, f_K^\star = f_K^\star,$$

which implies $\min_{x\in[0,1]^n} \phi_K(x) \ge f_K^\star$.

For the reverse inequality, let $S^\star$ attain $f_K^\star$ and set $x^\star := \mathbf{1}_{S^\star}$. Let $s := |S^\star| \le K$. After sorting $x^\star$, we have $x^\star_{\pi(k)} = 1$ for $k \le s$ and $x^\star_{\pi(k)} = 0$ for $k > s$, so the only nonzero gap $x^\star_{\pi(k)} - x^\star_{\pi(k+1)}$ occurs at $k = s$ and equals 1. Since $s \le K$, the window definition yields $W_s(x^\star) = \{\pi(1), \dots, \pi(s)\} = S^\star$, and thus

$$\phi_K(x^\star) = f\big(W_s(x^\star)\big) = f(S^\star) = f_K^\star.$$

Hence $\min_{x\in[0,1]^n} \phi_K(x) \le f_K^\star$, proving exactness.

Finally, the same weighted-average representation gives the rounding bound. For any x, a weighted average is at least the minimum of its terms, so

$$\phi_K(x) = \sum_{k=0}^n w_k \, f\big(W_k(x)\big) \;\ge\; \min_{0\le k\le n} f\big(W_k(x)\big) \;=\; f\big(\,\mathrm{Round}_K(x)\big),$$

which is (13). Feasibility of $\mathrm{Round}_K(x)$ follows from $|W_k(x)| \le K$ for all k. □

C.3 Well-Posedness of the Reflected Particle System

We address the well-posedness of the coupled particle system defined in Eq. (23).

Lemma 3 (Existence and Uniqueness in Law)**.** *For any $\tau > 0$ and $M \ge 1$, the coupled system of reflected SDEs* (23) *admits a weak solution, and its probability law is unique.*

Proof. The proof proceeds by verifying the conditions for the Girsanov transformation on the product space, utilizing the results of Lions & Sznitman [43].

First, we establish the properties of the drift coefficient. The domain of the joint process is $\Omega = \mathcal{X}^M$. Since Ω is a compact convex polytope, the Skorokhod problem for the reflection term is well-posed on Ω. Fix a measurable selection $g(\cdot)$ such that $g(x) \in \partial^\circ \phi(x)$ for all $x \in \mathcal{X}$. (For instance, one may use a deterministic tie-breaking rule to define a permutation $\pi(x)$ and set $g(x) := g^{\pi(x)}(x)$ as in Proposition 5; this yields a Borel measurable map on $\mathcal{X}$.) Define the drift field $\mathbf{b} : \Omega \to \mathbb{R}^{nM}$ blockwise by

$$\mathbf{b}^m(\mathbf{x}) := -\Big(M w^m(\mathbf{x})\, g(X^m) + \lambda X^m\Big), \qquad m = 1, \dots, M.$$

The weights $w^m(\mathbf{x})$ are smooth in $\mathbf{x}$. Moreover, since ϕ is G-Lipschitz on the compact set $\mathcal{X}$, we have $\sup_{x\in\mathcal{X}} \sup_{v\in\partial^\circ\phi(x)} \|v\| \le G$, hence $\|g(x)\| \le G$. Together with $0 < w^m(\mathbf{x}) < 1$ and boundedness of $\mathcal{X}$, this yields a uniform bound $\|\mathbf{b}(\mathbf{x})\| \le C$ on Ω for some constant C.

Next, we construct the solution via a change of measure. Let $\mathbf{Y}_t$ be the reference process consisting of M independent reflected Brownian motions on Ω with diffusion coefficient $\sqrt{2\tau}$ and normal reflection. In its Skorokhod decomposition, $\mathbf{Y}_t$ admits

$$d\mathbf{Y}_t = \sqrt{2\tau}\, d\mathbf{B}_t + d\mathbf{R}_t,$$

where $\mathbf{B}_t$ is an $\mathbb{R}^{nM}$-valued Brownian motion and $\mathbf{R}_t$ is the reflection term of bounded variation.

Since the joint drift $\mathbf{b}(\mathbf{X}_t)$ is bounded, Novikov's condition (e.g., [51, Theorem 10.16]) holds, and the Girsanov exponential martingale

$$Z_t := \exp\left(\frac{1}{\sqrt{2\tau}}\int_0^t \mathbf{b}(\mathbf{Y}_s)\cdot d\mathbf{B}_s - \frac{1}{4\tau}\int_0^t \|\mathbf{b}(\mathbf{Y}_s)\|^2\, ds\right)$$

is a true martingale. By Lions & Sznitman [43, Theorem 3.2], define a new measure $\mathbb{Q}$ on $(\Omega, \mathcal{F}_t)$ by $\frac{d\mathbb{Q}}{d\mathbb{P}}\big|_{\mathcal{F}_t} = Z_t$. Then, under $\mathbb{Q}$, the reference process $\mathbf{Y}_t$ is a weak solution of (23). We denote this solution by $\mathbf{X}_t := \mathbf{Y}_t$ under $\mathbb{Q}$. Due to the bounded drift, the same theorem (equivalently, the one-to-one nature of the Girsanov transform) yields *uniqueness in law* of the coupled particle system $\mathbf{X}(t))$ in (23). □

C.4 Proof of Convergence Theorem 1

In this section, we provide the detailed proof for the exponential convergence of the finite-particle system (Proposition 7). Our proof strategy follows the Uniform-in-time Propagation of Chaos framework of Chen et al. [9, Theorem 2.6]. Below, we verify that our functional satisfies the structural assumptions required by their framework, and extend their analysis to our setting of reflected Langevin dynamics on a compact domain.

C.4.1 Correspondence to the Framework of Chen et al. [9]

To align our notation with the framework of Chen et al. [9], we introduce a reformulated objective functional. Recall our original Mean-Field functional defined via KL-divergence:

$$\mathcal{F}(\rho) = L(\rho) + \tau D_{\mathrm{KL}}(\rho\|\gamma).$$

Expanding the relative entropy term $D_{\mathrm{KL}}(\rho\|\gamma) = \int \rho\log\rho - \int \rho\log\gamma$, and noting that $\log\gamma(x) = -\frac{\lambda}{2\tau}\|x\|^2 - \log Z_\gamma$, we can rewrite the functional as

$$\mathcal{F}(\rho) = \underbrace{L(\rho) + \int_{\mathcal{X}} \frac{\lambda}{2}\|x\|^2\rho(dx)}_{L_\lambda(\rho)} + \tau \underbrace{\int_{\mathcal{X}} \rho(x)\log\rho(x)dx}_{\mathrm{Ent}(\rho)} + \tau\log Z_\gamma.$$

Here, $\tau\log Z_\gamma$ is a constant independent of ρ.

To facilitate the convergence analysis, we remove this additive constant and work with the equivalent functional

$$\widetilde{\mathcal{F}}(\rho) := L_\lambda(\rho) + \tau\mathrm{Ent}(\rho),$$

where

$$L_\lambda(\rho) := L(\rho) + \mathbb{E}_{x\sim\rho}\left[\frac{\lambda}{2}\|x\|^2\right] \tag{27}$$

collects the soft-min interaction term and the quadratic confinement term. This shift does not affect the optimization gap. Let $\widetilde{\mathcal{F}}^{(M)}$ be the corresponding shifted finite-particle functional. Since the constant term $\tau \log Z_\gamma$ cancels out in the difference:

$$\frac{1}{M}\mathcal{F}^{(M)}(\rho_t^{(M)}) - \mathcal{F}(\rho^\tau) = \frac{1}{M}\widetilde{\mathcal{F}}^{(M)}(\rho_t^{(M)}) - \widetilde{\mathcal{F}}(\rho^\tau),$$

establishing the convergence rate for the shifted functional $\widetilde{\mathcal{F}}$ is equivalent to proving it for our original objective $\mathcal{F}$.

Next, we verify that the effective potential $L_\lambda(\rho)$ defined in (27) satisfies the structural Assumptions (2.1)–(2.4) in Chen et al. [9]. All constants derived below are made explicit.

C.4.2 Verification of Assumptions

We now state the required assumptions as Lemmas and prove they hold for our Assortment Optimization problem on the compact hypercube $\mathcal{X}$.

The Lovász extension $\phi(x)$ is piecewise linear on $\mathcal{X}$. While it is not differentiable everywhere, it is globally Lipschitz continuous. In the following analysis, at any point x where ϕ is non-differentiable, we interpret the symbol $\nabla\phi(x)$ as an arbitrary element of its subdifferential $\partial^\circ\phi(x)$. Crucially, the Lipschitz property ensures that this generalized gradient is uniformly bounded: $\|\nabla\phi(x)\| \le G$ for all $x \in \mathcal{X}$.

The following lemma regarding convexity corresponds to the assumption (2.2) in 9.

Lemma 4 (Convexity)**.** *The functional $L_\lambda(\rho)$ is convex on $\mathcal{P}(\mathcal{X})$.*

Proof. Note that L_λ consists of the interaction term $L(\rho) = -\frac{1}{\beta}\log\int e^{-\beta\phi}d\rho$ and the quadratic potential $\int \frac{\lambda}{2}\|x\|^2 d\rho$. Since $-\log(\cdot)$ is convex and the integral operator is linear, $L(\rho)$ is convex. The quadratic term is linear in ρ. Thus, their sum is convex. □

The following lemma regarding Lipschitz Derivative corresponds to the assumption (2.2) in 9.

Lemma 5 (Lipschitz Derivative)**.** *The intrinsic derivative $\frac{\delta L_\lambda}{\delta\rho}$ exists, and its gradient is Lipschitz continuous with respect to the Wasserstein-1 distance W_1. Specifically, there exists $C_{\mathrm{Lip}} > 0$ such that for any $\rho, \nu \in \mathcal{P}(\mathcal{X})$ and any $x \in \mathcal{X}$:*

$$\left\|\nabla_x \frac{\delta L_\lambda}{\delta\rho}(\rho, x) - \nabla_x \frac{\delta L_\lambda}{\delta\rho}(\nu, x)\right\| \le C_{\mathrm{Lip}} W_1(\rho, \nu).$$

Proof. We first compute the explicit form of the derivative. The first variation is

$$\frac{\delta L_\lambda}{\delta\rho}(\rho, x) = -\frac{1}{\beta}\frac{e^{-\beta\phi(x)}}{Z_\rho} + \frac{\lambda}{2}\|x\|^2, \quad \text{where } Z_\rho = \int_{\mathcal{X}} e^{-\beta\phi(z)}\rho(dz).$$

Taking the gradient with respect to x yields that

$$\nabla_x \frac{\delta L_\lambda}{\delta\rho}(\rho, x) = \underbrace{\frac{e^{-\beta\phi(x)}g(x)}{Z_\rho}}_{\text{Term I}} + \underbrace{\lambda x}_{\text{Term II}},$$

where $g(x) \in \partial\phi(x)$ denotes a subgradient of the Lovász extension at x. Since ϕ is Lipschitz, $\|g(x)\|$ is bounded almost everywhere.

For Term II, this quadratic term λx is independent of the measure ρ. Thus, in the difference $\nabla_x \frac{\delta L_\lambda}{\delta\rho}(\rho) - \nabla_x \frac{\delta L_\lambda}{\delta\rho}(\nu)$, this term cancels out exactly.

For Term I, we must bound the difference:

$$\Delta_I = \left\| \frac{e^{-\beta\phi(x)}g(x)}{Z_\rho} - \frac{e^{-\beta\phi(x)}g(x)}{Z_\nu} \right\| = \underbrace{\|e^{-\beta\phi(x)}g(x)\|}_{I(a)} \underbrace{\left| \frac{1}{Z_\rho} - \frac{1}{Z_\nu} \right|}_{I(b)}.$$

For the term I(a), since $r(S) \in [0, \bar{r}]$ and we work with $f(S) = -r(S)$, the Lovász extension satisfies

$$\phi(x) \in [-\bar{r}, 0], \qquad \forall x \in \mathcal{X}. \tag{28}$$

Additionally, by the Lemma 2, since ϕ is $G-$Lipschitz, we have the bound $\|g(x)\|_1 \leq G$. Converting this to the Euclidean norm required for our analysis, we have that

$$\|g(x)\|_2 \leq \sqrt{n}\|g(x)\|_\infty \leq \sqrt{n}G.$$

Consequently, the numerator term is bounded by:

$$C_\phi := \sup_{x\in\mathcal{X}} \|e^{-\beta\phi(x)}g(x)\| \leq e^{-\beta(-\bar{r})}(\sqrt{n}G) = \sqrt{n}Ge^{\beta\bar{r}}.$$

For term I(b), we first observe that since $\phi(x) \leq 0$, the exponential term $e^{-\beta\phi(x)} \geq 1$ everywhere. Thus, for any probability measure ρ, the partition function is bounded from below that

$$Z_\rho = \int e^{-\beta\phi(z)}\rho(dz) \geq \min_{x\in\mathcal{X}} e^{-\beta\phi(x)} = 1.$$

Next, consider the difference $|Z_\rho - Z_\nu|$ that

$$|Z_\rho - Z_\nu| = \left| \int e^{-\beta\phi(z)}(\rho - \nu)(dz) \right|.$$

The integrand $h(z) = e^{-\beta\phi(z)}$ is Lipschitz continuous on $\mathcal{X}$. By the chain rule for generalized gradients, any subgradient $\zeta \in \partial h(z)$ satisfies $\|\zeta\| = \| - \beta e^{-\beta\phi(z)}g\|$ for some $g \in \partial\phi(z)$. Using the bounds $\phi(z) \geq -\bar{r}$ and $\|g\|_2 \leq \sqrt{n}G$, we have the uniform bound $\|\zeta\| \leq \beta e^{\beta\bar{r}}G$. Defining the constant $L_h := \beta\sqrt{n}Ge^{\beta\bar{r}}$, the Kantorovich-Rubinstein duality implies that

$$|Z_\rho - Z_\nu| \leq L_h W_1(\rho, \nu).$$

Now, applying the mean value theorem to the function $f(z) = 1/z$ and using the lower bound $Z \geq 1$, we obtain that

$$\left| \frac{1}{Z_\rho} - \frac{1}{Z_\nu} \right| = \frac{|Z_\rho - Z_\nu|}{Z_\rho Z_\nu} \leq \frac{L_h}{1 \cdot 1} W_1(\rho, \nu) = L_h W_1(\rho, \nu).$$

Combining these estimates, we define the aggregate Lipschitz constant explicitly as

$$C_{\text{Lip}} := C_\phi \cdot L_h \leq (\sqrt{n}Ge^{\beta\bar{r}}) \cdot (\sqrt{n}\beta Ge^{\beta\bar{r}}) = n\beta G^2 e^{2\beta\bar{r}}.$$

This explicit dependence ($C_{\text{Lip}} \propto n$) arises naturally from the Euclidean geometry of the diffusion process. We obtain $\Delta_I \leq C_{\text{Lip}} W_1(\rho, \nu)$, which completes the proof. □

The following lemma regarding Log-Sobolev Inequalities corresponds to the assumption (2.3) in 9.

Lemma 6 (Uniform LSI)**.** *Define the "conditional measure" ν_m for any configuration $\mathbf{x}^{-m}$ as:*

$$\nu_m(dx) \propto \exp\left(-\frac{1}{\tau}\frac{\delta L_\lambda}{\delta \rho}(\widehat{\mu}_{\mathbf{x}^{-m}}, x)\right) dx.$$

Then, ν_m satisfies the Log-Sobolev Inequality (LSI) with a constant $\alpha_{\text{unif}} > 0$ independent of M and $\mathbf{x}^{-m}$.

Proof. Substituting the expression for the intrinsic derivative, the density of ν_m is given by

$$\nu_m(dx) \propto \exp\left(-\frac{1}{\tau}\left[-\frac{1}{\beta Z_\mu}e^{-\beta\phi(x)} + \frac{\lambda}{2}\|x\|^2\right]\right) dx,$$

where $\mu = \widehat{\mu}_{\mathbf{x}^{-m}}$ is the empirical measure of the other particles. We prove the LSI via the Holley-Stroock Perturbation Principle([31]). We view ν_m as a perturbation of the reference measure γ.

Let $\gamma(dx) \propto \exp(-V_{\text{ref}}(x))dx$ with $V_{\text{ref}}(x) = \frac{\lambda}{2\tau}\|x\|^2$ restricted to the convex set $\mathcal{X}$. Since the potential V_{ref} is strongly convex with Hessian $\nabla^2 V_{\text{ref}} = \frac{\lambda}{\tau}I$ and the reflected (Neumann) diffusion is considered on a convex domain, the generalized Bakry–Émery criterion for manifolds with boundary implies that γ satisfies an LSI with constant $\alpha_{\text{ref}} = \lambda/\tau$ [see, e.g., 66].

We can write $\nu_m(dx) \propto e^{-\Psi(x)}\gamma(dx)$, where the perturbation potential is

$$\Psi(x) = -\frac{1}{\tau\beta Z_\mu}e^{-\beta\phi(x)}.$$

We analyze the oscillation of Ψ on the domain $\mathcal{X}$: $\text{Osc}(\Psi) = \sup\Psi - \inf\Psi$. First, recalling the range of ϕ from (28), we have $\phi(x) \in [-\bar{r}, 0]$ on $\mathcal{X}$. Consequently, for $Z_\mu := \int_{\mathcal{X}} e^{-\beta\phi(z)}\,\mu(dz)$ with $\mu = \widehat{\mu}_{\mathbf{x}^{-m}}$, we obtain that

$$1 = \inf_{z\in\mathcal{X}} e^{-\beta\phi(z)} \le Z_\mu \le \sup_{z\in\mathcal{X}} e^{-\beta\phi(z)} = e^{\beta\bar{r}}.$$

Since $e^{-\beta\phi(x)} \in [1, e^{\beta\bar{r}}]$, it follows that for $\Psi(x) = -\frac{1}{\tau\beta Z_\mu}e^{-\beta\phi(x)}$,

$$|\Psi(x)| \le \frac{1}{\tau\beta}\frac{\sup_x e^{-\beta\phi(x)}}{\inf_\mu Z_\mu} \le \frac{1}{\tau\beta}e^{\beta\bar{r}} =: C_\Psi, \qquad \forall x \in \mathcal{X},$$

and hence $\text{Osc}(\Psi) \le 2C_\Psi$. By the Holley–Stroock perturbation principle [31], we conclude that ν_m satisfies an LSI with

$$\alpha_{\text{unif}} \ge \alpha_{\text{ref}}\exp(-\text{Osc}(\Psi)) \ge \frac{\lambda}{\tau}e^{-2C_\Psi} > 0,$$

which is uniform in M and $\mathbf{x}^{-m}$ (although the bound can be conservative for small τ). □

Having verified the necessary regularity conditions, we now proceed to the detailed proof of the optimization gap convergence. The proof is structured into the following key steps: establishing the entropy sandwich bound, deriving the dissipation identity, and decomposing the Fisher information.

C.4.3 Entropy Sandwich

We now establish a crucial bound linking the convergence of the functional to the propagation of chaos. This result adapts Lemma 5.2 of Chen et al. [9] to our relative entropy formulation. It states that the distance between the finite-particle law and the tensorized mean-field invariant measure is bounded by the gap between their respective functionals.

Lemma 7 (Domination of Entropic Chaos)**.** *Let $\rho^{(M)} \in \mathcal{P}(\mathcal{X}^M)$ be any finite-particle distribution with finite entropy, and let ρ^τ be the unique minimizer of the mean-field functional $\mathcal{F}$. Then, it holds that*

$$\tau D_{\mathrm{KL}}(\rho^{(M)} \mid (\rho^\tau)^{\otimes M}) \leq \mathcal{F}^{(M)}(\rho^{(M)}) - M\mathcal{F}(\rho^\tau).$$

Proof. Recall the definitions of the functionals:

$$\mathcal{F}^{(M)}(\rho^{(M)}) = M\mathbb{E}_{\mathbf{X}\sim\rho^{(M)}}[L(\mu_{\mathbf{x}})] + \tau D_{\mathrm{KL}}(\rho^{(M)} \mid \gamma^{\otimes M}),$$
$$\mathcal{F}(\rho^\tau) = L(\rho^\tau) + \tau D_{\mathrm{KL}}(\rho^\tau \mid \gamma).$$

By the convexity of L (Lemma 4), for every empirical measure realization $\mu_{\mathbf{x}}$, we have that

$$L(\mu_{\mathbf{x}}) - L(\rho^\tau) \;\geq\; \int_{\mathcal{X}} \frac{\delta L}{\delta \rho}(\rho^\tau, x)\,(\mu_{\mathbf{x}} - \rho^\tau)(dx).$$

Applying this inequality to the random empirical measure $\mu_{\mathbf{x}} = \frac{1}{M}\sum_{m=1}^M \delta_{X^m}$ and taking the expectation with respect to the particle law $\rho^{(M)}$ yields that

$$\mathbb{E}[M\,L(\mu_{\mathbf{x}}) - M\,L(\rho^\tau)] \;\geq\; \mathbb{E}\left[M\int_{\mathcal{X}} \frac{\delta L}{\delta \rho}(\rho^\tau, x)\,(\mu_{\mathbf{x}} - \rho^\tau)(dx)\right]. \tag{29}$$

Next, we use the fixed-point condition for the mean-field invariant measure. Writing it in density form, we have that

$$\frac{d\rho^\tau}{d\gamma}(x) = \frac{1}{Z_*}\exp\left(-\frac{1}{\tau}\frac{\delta L}{\delta \rho}(\rho^\tau, x)\right),$$

where Z_* is the normalization constant for the distribution ρ^τ, then we obtain, for γ-a.e. x,

$$\frac{\delta L}{\delta \rho}(\rho^\tau, x) = -\tau \log \frac{d\rho^\tau}{d\gamma}(x) + C_*, \tag{30}$$

for the constant $C_* = -\tau \log Z_*$. Substituting (30) into the right-hand side of (29) and using the fact that $\int (\mu_{\mathbf{x}} - \rho^\tau)(dx) = 0$, the constant C_* cancels out. Then, we obtain that

$$\begin{aligned}\mathbb{E}\left[M\int_{\mathcal{X}} \frac{\delta L}{\delta \rho}(\rho^\tau, x)\,(\mu_{\mathbf{x}} - \rho^\tau)(dx)\right] &= -\tau\,\mathbb{E}\left[M\int_{\mathcal{X}} \log\frac{d\rho^\tau}{d\gamma}(x)\,(\mu_{\mathbf{x}} - \rho^\tau)(dx)\right] \\ &= -\tau\,\mathbb{E}\left[M\int_{\mathcal{X}} \log\frac{d\rho^\tau}{d\gamma}(x)\,\mu_{\mathbf{x}}(dx)\right] + \tau M\int_{\mathcal{X}} \log\frac{d\rho^\tau}{d\gamma}(x)\,\rho^\tau(dx)\end{aligned} \tag{31}$$

Moreover, by the definition of the empirical measure $\mu_{\mathbf{x}}$, the first term expands as that

$$\mathbb{E}\left[M\int_{\mathcal{X}} \log\frac{d\rho^\tau}{d\gamma}(x)\,\mu_{\mathbf{x}}(dx)\right] = \int_{\mathcal{X}^M} \log\frac{d(\rho^\tau)^{\otimes M}}{d\gamma^{\otimes M}}(\mathbf{x})\,\rho^{(M)}(d\mathbf{x}).$$

Therefore, combining the definition of $\mathcal{F}^{(M)}$, the convexity bound (29), and the expansion (31), we have that

$$\begin{aligned}\mathcal{F}^{(M)}(\rho^{(M)}) - M\mathcal{F}(\rho^\tau) &\geq -\tau\int_{\mathcal{X}^M} \log\frac{d(\rho^\tau)^{\otimes M}}{d\gamma^{\otimes M}}(\mathbf{x})\,\rho^{(M)}(d\mathbf{x}) + \tau D_{\mathrm{KL}}(\rho^{(M)}\|\gamma^{\otimes M}) \\ &= \tau\int_{\mathcal{X}^M} \rho^{(M)}(\mathbf{x})\left(\log\frac{\rho^{(M)}(\mathbf{x})}{\gamma^{\otimes M}(\mathbf{x})} - \log\frac{(\rho^\tau)^{\otimes M}(\mathbf{x})}{\gamma^{\otimes M}(\mathbf{x})}\right)d\mathbf{x} \\ &= \tau\,D_{\mathrm{KL}}\left(\rho^{(M)} \,\middle\|\, (\rho^\tau)^{\otimes M}\right).\end{aligned}$$

This completes the proof. □

C.4.4 Functional Dissipation under Reflection

We now derive the dissipation of the particle Functional under the reflected Langevin dynamics. The time evolution of the joint density $\rho_t^{(M)}$ is governed by the Fokker-Planck equation on the product space $\mathcal{X}^M$ with Neumann boundary conditions.

Lemma 8 (Reflected Dissipation Identity)**.** *Let $\rho_t^{(M)}$ be the solution to the reflected Fokker-Planck equation on $\mathcal{X}^M$. The time derivative of the finite-particle functional satisfies that*

$$\frac{d}{dt}\mathcal{F}^{(M)}(\rho_t^{(M)}) = -\tau^2 \mathcal{I}(\rho_t^{(M)} \mid \rho_\infty^{(M)}),$$

where $\mathcal{I}(\cdot\|\cdot)$ denotes the relative Fisher information.

Proof. The time derivative is given by

$$\frac{d}{dt}\mathcal{F}^{(M)}(\rho_t^{(M)}) = \int_{\mathcal{X}^M} \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}}\, \partial_t \rho_t^{(M)}\, d\mathbf{x}.$$

We invoke the continuity equation for the probability density, $\partial_t \rho_t^{(M)} + \nabla \cdot J_t = 0$, where J_t denotes the probability flux vector that

$$J_t := -\rho_t^{(M)} \nabla \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}}.$$

Substituting the continuity equation $\partial_t \rho_t^{(M)} = \nabla \cdot (\rho_t^{(M)} \nabla \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}})$ and applying integration by parts (Green's first identity) yield that

$$\frac{d}{dt}\mathcal{F}^{(M)} = -\int_{\mathcal{X}^M} \left\| \nabla \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}} \right\|^2 \rho_t^{(M)}\, d\mathbf{x} + \underbrace{\int_{\partial \mathcal{X}^M} \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}} \left(\rho_t^{(M)} \nabla \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}} \cdot \mathbf{n} \right) dS}_{\text{Boundary Term}}.$$

Unlike the full Euclidean space setting in Chen et al. [9] where boundary terms vanish at infinity, here the domain $\mathcal{X}^M$ is compact. However, the reflected dynamics enforce the *no-flux boundary condition*:

$$J_t \cdot \mathbf{n} = \left(\rho_t^{(M)} \nabla \frac{\delta \mathcal{F}^{(M)}}{\delta \rho^{(M)}} \right) \cdot \mathbf{n} = 0 \quad \text{on } \partial \mathcal{X}^M.$$

Consequently, the boundary integral vanishes identically. Noting that $\nabla \frac{\delta \mathcal{F}^{(M)}}{\delta \rho} = \tau \nabla \log(\rho_t^{(M)} / \rho_\infty^{(M)})$, the integral term becomes the relative Fisher information, proving the identity. □

C.4.5 Decomposition of Fisher Information

To utilize the Uniform LSI property (Lemma 6), we decompose the total Fisher information into conditional components.

Definition and Regularity of Conditional Measures. Before performing the decomposition, we explicitly define the conditional distributions. The joint density $\rho_t^{(M)}$ evolves according to the parabolic Fokker-Planck equation associated with the SDE (23). Since the diffusion coefficient $\tau > 0$ is non-degenerate and drift term in (23) is bounded on the compact domain$\mathcal{X}^M$ for each fixed M (as established by the Lipschitz property of ϕ and a measurable subgradient selection), this equation is strictly parabolic on the compact domain $\mathcal{X}^M$, subject to Neumann boundary conditions. By the classical Strong Maximum Principle for parabolic

operators (see, e.g., 21, 22), the solution $\rho_t^{(M)}(\mathbf{x})$ is strictly positive for all $t > 0$ and $\mathbf{x} \in \mathcal{X}^M$. Consequently, the marginal density of the particle configuration excluding i, defined as

$$\rho_t^{(M,-m)}(\mathbf{x}^{-m}) := \int_{\mathcal{X}} \rho_t^{(M)}(\mathbf{x})\, dx^m,$$

is also strictly positive and finite. This ensures that the conditional density of the i-th particle that

$$\rho_t^{(M),m|-m}(x_m \mid \mathbf{x}^{-m}) := \frac{\rho_t^{(M)}(\mathbf{x})}{\rho_t^{(M,-m)}(\mathbf{x}^{-m})}$$

is well-defined, smooth, and strictly positive everywhere. This allows us to legitimately operate on its logarithm.

C.4.6 Error Estimates for Finite-Particle Approximation

To facilitate the convergence analysis, we first bound the errors arising from the finite-particle approximation of the drift and potential.

Let $\delta_m(\mathbf{X}) := \nabla_{X^m} \frac{\delta L_\lambda}{\delta \rho}(\mu_{\mathbf{x}}, X^m) - \nabla_{X^m} \frac{\delta L_\lambda}{\delta \rho}(\widehat{\mu}_{\mathbf{x}^{-m}}, X^m)$. By the Lipschitz derivative property (Lemma 5) and the transport coupling bound from Chen et al. [9, Eq 5.11], we have $\|\delta_m(\mathbf{X})\| \le C_{\mathrm{Lip}} W_1(\mu_{\mathbf{x}}, \widehat{\mu}_{\mathbf{x}^{-m}})$. Following the argument in Chen et al. [9, Eq 5.12], summing over particles and taking expectations yields the upper bound for the drift error Δ_1 that

$$\Delta_1 := \sum_{m=1}^{M} \mathbb{E}[\|\delta_m(\mathbf{X})\|^2] \le \frac{6C_{\mathrm{Lip}}^2}{M} \left(\mathbb{E}[W_2^2(\mu_{\mathbf{x}}, \rho^\tau)] + \mathrm{Var}(\rho^\tau) \right). \tag{32}$$

The potential error Δ_2 accounts for the nonlinearity of the interaction $L(\rho)$. By applying Jensen's inequality and Lemma 5 as in Chen et al. [9, Eq 5.13], we define and bound the potential error Δ_2 as:

$$|\Delta_2| := \left| \mathbb{E}\left[ML(\mu_{\mathbf{x}}) - ML(\rho^\tau) - \sum_{m=1}^{M} \int_{\mathcal{X}} \frac{\delta L}{\delta \rho}(\widehat{\mu}_{\mathbf{x}^{-m}}, z)(\rho_t^{(M),m|-m}(dz) - \rho^\tau(dz)) \right] \right|, \tag{33}$$

which satisfies $|\Delta_2| \le C_{\mathrm{Lip}} \left(\frac{4}{M} \mathbb{E}[W_2^2(\mu_{\mathbf{x}}, \rho^\tau)] + 5\mathrm{Var}(\rho^\tau) \right)$.

To relate the sum of local conditional KL divergences to the global functional gap, we utilize the following variational identity (see 9, Lemma 5.4):

$$\tau \sum_{m=1}^{M} \mathbb{E}\left[D_{\mathrm{KL}}\left(\rho_t^{(M),m|-m} \parallel \nu_m \right) \right] = \mathcal{F}^{(M)}(\rho_t^{(M)}) - M\mathcal{F}(\rho^\tau) - \Delta_2. \tag{34}$$

C.4.7 Fisher Information Decomposition and Dissipation

We now relate the global dissipation to the previously defined error metrics. To do so, we decompose the joint Fisher information on the product space $\mathcal{X}^M$.

Using the inequality $\|a+b\|^2 \ge (1-\varepsilon)\|a\|^2 - C_\varepsilon \|b\|^2$ with $C_\varepsilon := \varepsilon^{-1} - 1$, we lower bound the Fisher

information by separating the conditional dissipation from the approximation error:

$$\mathcal{I}(\rho_t^{(M)} \mid \rho_\infty^{(M)}) = \sum_{m=1}^{M} \mathbb{E}_{\mathbf{X}} \left[\left\| \nabla_{X^m} \log \frac{\rho_t^{(M),m|-m}}{\nu_m} + \nabla_{X^m} \log \frac{\nu_m}{\rho_\infty^{(M)}} \right\|^2 \right]$$
$$\geq (1-\varepsilon) \underbrace{\sum_{m=1}^{M} \mathbb{E}\left[\mathcal{I}(\rho_t^{(M),m|-m} \parallel \nu_m) \right]}_{\text{Term A}} - C_\varepsilon \underbrace{\sum_{m=1}^{M} \mathbb{E}\left[\left\| \nabla_{X^m} \log \frac{\nu_m}{\rho_\infty^{(M)}} \right\|^2 \right]}_{\text{Term B}}.$$

We now express both terms in terms of our established error metrics Δ_1 and Δ_2. For Term B, noting that $\nabla \log(\nu_m/\rho_\infty^{(M)}) = \frac{1}{\tau}\delta_m(\mathbf{X})$, we have that

$$C_\varepsilon \sum_{m=1}^{M} \mathbb{E}\left[\left\| \frac{1}{\tau}\delta_m(\mathbf{X}) \right\|^2 \right] = \frac{C_\varepsilon}{\tau^2}\Delta_1.$$

For Term A, we first invoke the Uniform LSI (Lemma 6) to translate Fisher information into local KL divergences, and then utilize the variational relation of Δ_2 (34) to recover the global functional gap that

$$\text{Term A} \geq \sum_{m=1}^{M} \mathbb{E}\left[2\alpha_{\text{unif}} D_{\text{KL}}\left(\rho_t^{(M),m|-m} \parallel \nu_m \right) \right]$$
$$\geq \frac{2\alpha_{\text{unif}}}{\tau} \left(\mathcal{F}^{(M)}(\rho_t^{(M)}) - M\mathcal{F}(\rho^\tau) - |\Delta_2| \right).$$

Combining these results and multiplying both sides by τ^2, we obtain the final lower bound for the functional dissipation that

$$\tau^2 \mathcal{I}(\rho_t^{(M)} \mid \rho_\infty^{(M)}) \geq 2\alpha_{\text{unif}}(1-\varepsilon)\tau\Big((\mathcal{F}^{(M)} - M\mathcal{F}^*) - |\Delta_2| \Big) - C_\varepsilon \Delta_1. \tag{35}$$

This inequality characterizes the competition between the contractive force of the gradient flow and the errors introduced by the finite-particle approximation.

C.4.8 Final Differential Inequality and Convergence

We now combine the error estimates to derive the final convergence rate. Recall the dissipation identity from Lemma 8 that

$$\frac{d}{dt}\mathcal{F}^{(M)}(\rho_t^{(M)}) = -\tau^2 \mathcal{I}(\rho_t^{(M)} \mid \rho_\infty^{(M)}).$$

Substituting the lower bound for the Fisher information from (35) and dividing by M, we obtain the evolution of the normalized gap $\Delta_M(t) := \frac{1}{M}\mathcal{F}^{(M)}(\rho_t^{(M)}) - \mathcal{F}(\rho^\tau)$:

$$\frac{d}{dt}\Delta_M(t) \leq -\kappa_0 \left(\Delta_M(t) - \frac{|\Delta_2|}{M} \right) + \frac{C_\varepsilon}{M}\Delta_1, \tag{36}$$

where $\kappa_0 := 2\alpha_{\text{unif}}(1-\varepsilon)$.

As established in Lemma 6, the conditional measures satisfy LSI with constant α_{unif}. Consequently, the mean-field invariant measure ρ^τ also satisfies LSI with a constant $\alpha^* \geq \alpha_{\text{unif}}$ (see Remark 2.2 in Chen et al. [9]). This LSI property implies two key bounds [see, e.g., 65, Theorem 22.17]. First, by Talagrand's inequality (T_2), the transport cost is controlled by the functional gap that

$$\mathbb{E}[W_2^2(\mu_{\mathbf{x}}, \rho^\tau)] \leq \frac{2}{\alpha^* M}\left(\mathcal{F}^{(M)}(\rho_t^{(M)}) - M\mathcal{F}(\rho^\tau)\right).$$

Second, by the Poincaré inequality, the variance of the invariant measure is bounded by the dimension that

$$\mathrm{Var}(\rho^\tau) := \int_{\mathcal{X}} \|x - \mathbb{E}_{\rho^\tau}[x]\|^2 \rho^\tau(dx) \leq \frac{n\tau}{\alpha^*}.$$

Substituting the explicit bounds for Δ_1 from (32) and Δ_2 from (33) into (36), and grouping terms according to their dependence on $\Delta_M(t)$, we obtain that

$$\frac{d}{dt}\Delta_M(t) \leq -\left(\kappa_0 - \frac{C_1}{M}\right)\Delta_M(t) + \frac{C_2}{M},$$

where the constants C_1 and C_2 are defined as:

$$C_1 = \frac{4\kappa_0 C_{\mathrm{lip}}}{\alpha^*} + \frac{12 C_\varepsilon C_{\mathrm{lip}}^2}{\alpha^*},$$
$$C_2 = \left(5\kappa_0 C_{\mathrm{lip}} + 6C_\varepsilon C_{\mathrm{lip}}^2\right)\mathrm{Var}(\rho^\tau).$$

Notably, as $C_{\mathrm{lip}} = O(n)$ and both $\mathrm{Var}(\rho^\tau) = O(n)$, the constant $C_1 = O\left(n^2\right)$ and $C_2 = O\left(n^3\right)$ scale polynomially in n

For a sufficiently large number of particles M, the effective convergence rate $\tilde{\kappa} := \kappa_0 - C_1/M$ remains strictly positive. Applying Grönwall's lemma yields that

$$\Delta_M(t) \leq e^{-\tilde{\kappa}t}\Delta_M(0) + \frac{C_2}{\tilde{\kappa}M}.$$

This confirms that the optimization gap decays exponentially at rate $\tilde{\kappa}$ toward a steady-state error of order $O(n^2/M)$, which completes the proof of Proposition 7. □

C.5 Proof of Regularization Bias (Proposition 8)

In this section, we provide the detailed proof for the bound on the regularization bias. The argument is variational: we construct a localized trial measure around a minimizer of ϕ to upper bound the minimal functional.

Let $x^* \in \mathcal{X}$ be a global minimizer of $\phi(x)$, such that $\phi(x^*) = \phi^\star$. Recall that the mean-field functional is given by

$$\mathcal{F}(\rho) = L(\rho) + \tau D_{\mathrm{KL}}(\rho\|\gamma).$$

Since ρ^τ minimizes $\mathcal{F}$ over $\mathcal{P}(\mathcal{X})$, for any arbitrary trial measure $\rho_\epsilon \in \mathcal{P}(\mathcal{X})$, we have the upper bound that

$$\mathcal{F}(\rho^\tau) - \phi^\star \leq \mathcal{F}(\rho_\epsilon) - \phi^\star. \tag{37}$$

We construct ρ_ϵ to be the uniform distribution restricted to a local neighborhood of x^*. Let $\epsilon > 0$ be a smoothing radius. Define the region $B_\epsilon = \mathcal{X} \cap B_1(x^*, \epsilon)$, $B_1(x^*, \epsilon) = \{x \in \mathcal{X} : \|x - x^*\|_1 \leq \epsilon\}$. The trial density is that

$$\rho_\epsilon(x) = \frac{1}{\mathrm{Vol}(B_1)}\mathbf{1}_{B_1}(x).$$

In n dimensions, the volume of an ℓ_1-ball scales as $\mathrm{Vol}(B_1) \geq c_n \frac{\epsilon^n}{n!}$. Substituting this into the entropic cost, we obtain that

$$-\tau \log \mathrm{Vol}(B_1) \leq -n\tau \log \epsilon + \tau \log(n!) - \tau \log c_n. \tag{38}$$

Using Stirling's approximation, $\log(n!) \approx n \log n - n$, which is a lower-order term compared to the logarithmic singularity in τ.

We decompose the gap for the trial measure into two components as follows:

$$\mathcal{F}(\rho_\epsilon) - \phi^\star = \underbrace{\left(L(\rho_\epsilon) - \phi^\star\right)}_{\text{(I)}} + \underbrace{\tau D_{\mathrm{KL}}(\rho_\epsilon \| \gamma)}_{\text{(II)}}.$$

To bound the Term (I), by Jensen's inequality (concavity of the log function), we upper bound the Soft-min by the expectation:

$$L(\rho_\epsilon) \le \mathbb{E}_{X \sim \rho_\epsilon}[\phi(X)].$$

Since ϕ is G-Lipschitz continuous with respect to the ℓ_1-norm, for any $x \in B_\epsilon$, we have that

$$\phi(x) - \phi^\star = \phi(x) - \phi(x^*) \le G\|x - x^*\|_1 \le G\epsilon.$$

Averaging this over ρ_ϵ yields that

$$L(\rho_\epsilon) - \phi^\star \le \mathbb{E}_{\rho_\epsilon}[\phi(X) - \phi^\star] \le G\epsilon. \tag{39}$$

To bound the Term (II), recall that KL divergence term

$$\tau D_{\mathrm{KL}}(\rho_\epsilon \| \gamma) = \tau \int_{B_\epsilon} \rho_\epsilon(x) \log \frac{\rho_\epsilon(x)}{\gamma(x)} dx,$$

and $\gamma(x) = \frac{1}{Z_\gamma} e^{-V_{\mathrm{ref}}(x)/\tau} \mathbf{1}_{\mathcal{X}}(x)$, where $V_{\mathrm{ref}}(x) = \frac{\lambda}{2}\|x\|^2$. Substituting the density $\rho_\epsilon(x) = \frac{1}{\mathrm{Vol}(B_\epsilon)}$ gives

$$\begin{aligned}
\tau D_{\mathrm{KL}}(\rho_\epsilon \| \gamma) &= \tau \log\left(\frac{1}{\mathrm{Vol}(B_\epsilon)}\right) + \tau \log Z_\gamma + \int_{B_\epsilon} V_{\mathrm{ref}}(x) \rho_\epsilon(x) dx \\
&= -\tau \log \mathrm{Vol}(B_\epsilon) + \tau \log Z_\gamma + \mathbb{E}_{\rho_\epsilon}[V_{\mathrm{ref}}(X)].
\end{aligned}$$

For the first term, using the ℓ_1 volume bound (38) and Stirling's approximation ($\log n! \approx n \log n - n$), we have

$$\begin{aligned}
-\tau \log \mathrm{Vol}(B_\epsilon) &\le -\tau \log\left(c_n \frac{(2\epsilon)^n}{n!}\right) \\
&= -n\tau \log \epsilon + \tau \log(n!) - n\tau \log 2 - \tau \log c_n \\
&\le -n\tau \log \epsilon + n\tau \log n + O(n\tau).
\end{aligned}$$

For the second term, $\tau \log Z_\gamma$ is bounded for small τ and λ. For the last term, since $\mathcal{X}$ is bounded, let $D_{\mathcal{X}} = \sup_{x \in \mathcal{X}} \frac{1}{2}\|x\|^2$. Then $\mathbb{E}_{\rho_\epsilon}[V_{\mathrm{ref}}(X)] \le \lambda D_{\mathcal{X}}$.

Combining these, we obtain a bound dominated by the logarithmic singularity as $\epsilon \to 0$ that

$$\tau D_{\mathrm{KL}}(\rho_\epsilon \| \gamma) \le -n\tau \log \epsilon + n\tau \log n + C_{\mathrm{const}} + O(\lambda), \tag{40}$$

where C_{const} collects terms independent of ϵ, τ, λ.

Substituting (39) and (40) into (37) gives that

$$\mathcal{F}(\rho^\tau) - \phi^\star \le G\epsilon - n\tau \log \epsilon + n\tau \log n + C_{\mathrm{total}}.$$

Setting the derivative with respect to ϵ to zero yields $\epsilon^* = \frac{n\tau}{G}$. Plugging this back into the inequality yields that

$$\begin{aligned}
\text{Gap} &\leq G\left(\frac{n\tau}{G}\right) - n\tau \log\left(\frac{n\tau}{G}\right) + n\tau \log n + C_{\text{total}} \\
&= n\tau - n\tau(\log n + \log \tau - \log G) + n\tau \log n + C_{\text{total}} \\
&= n\tau \log\left(\frac{1}{\tau}\right) + n\tau(1 + \log G) + C_{\text{total}} \\
&= O\left(n\tau \log \frac{1}{\tau}\right) + O(\lambda).
\end{aligned}$$

Notably, the $n\tau \log n$ terms arising from the ℓ_1 volume and the optimal radius ϵ^* exactly cancel. This confirms the regularization bias is $O(n\tau \log \frac{1}{\tau})$, completing the proof. □

D Experiment Details

For the MMNL results (Section D.1) and NL results (Section D.2), NAO is run with $M = 5{,}000$ particles. Each problem configuration contains 100 randomly generated instances (seeds). To focus the comparison on the most challenging cases, we report performance on *hardest instances* constructed as follows: for each method, we select the 5 seeds on which it attains the largest optimality gap, then take the union of these seeds across all methods. The reported values are the mean and sample standard deviation of each method's optimality gap over this combined hard-instance set. This protocol, following Guo et al. [28], ensures that the evaluation stresses every method on its own weakest cases as well as those of its competitors.

Table 3 summarizes the hyperparameter settings for NAO. We perform a grid search over weight decay, temperature τ, inverse steepness β, and learning rate. Note that β is defined in (21). For the unconstrained MMNL setting, we use the Adam optimizer with 200 training steps and an early stopping patience of 50 iterations (triggered when the best revenue ceases to improve). For all other settings—constrained MMNL, and both unconstrained and constrained NL—we use the AdamW optimizer with 1,000 training steps and a patience of 400 iterations.

Table 3: `NAO` hyperparameter settings.

Parameter	**Value**
Grid Search Space	
Weight decay	$\{0,\ 0.1,\ 0.5\}$
Temperature (τ)	$\{0,\ 0.0005,\ 0.001\}$
Inverse steepness (β)	$\{5,\ 10,\ 20\}$
Learning rate	$\{0.01,\ 0.05,\ 0.1\}$
MMNL Unconstrained	
Optimizer	Adam
Training steps	200
Early stopping patience	50
MMNL Constrained / NL Unconstrained / NL Constrained	
Optimizer	AdamW
Training steps	1,000
Early stopping patience	400

We use Adam-type optimizers because they typically converge faster and more robustly in practice, consistent with standard neural-network benchmarks [28]. We adopt AdamW in particular since it decouples weight decay from gradient normalization, aligning with the regularization effect assumed in our analysis. Empirically, SGD reaches comparable final solution quality, but is consistently slower. On the theory side, obtaining general nonasymptotic convergence guarantees for Adam/AdamW in nonconvex settings remains an active area, and sharp global guarantees comparable to SGD-based analyses are not yet fully settled. We therefore interpret our theoretical results as formalizing the core mechanisms preserved by the implementation: (i) first-order projected updates on the relaxed domain, (ii) explicit exploration via noise injection, and (iii) weight decay regularization to stabilize particles and control the surrogate.

Table 4 summarizes the hyperparameter settings for all benchmark methods (NN, NN_{PP}, ADXOpt) used in both the MMNL and NL experiments. These settings follow those reported by Guo et al. [28].

All `NAO` experiments are conducted on a single NVIDIA RTX 5090 GPU (32 GB) with 25 vCPUs (Intel Xeon Platinum 8470Q).

Table 4: Hyperparameter settings for benchmark methods, following Guo et al. [28].

Parameter	**Value**
Network Parameters	
Number of models (M)	10
Max iterations ($T_{\max}$)	1500
Hidden layers (L) / STE steepness (K_{STE})	4 / 5
Propensity widths	$[256, 512, 512, 256]$ (n<400); $[512, 1024, 1024, 512]$ (400≤n<1500); $[2056, 4056, 4056, 2056]$ (n≥1500)
Threshold widths	$[128, 256, 256, 128]$ (n<400); $[256, 512, 512, 256]$ (400≤n<1500); $[1024, 2056, 2056, 1024]$ (n≥1500)
LR range (propensity / threshold)	$[10^{-6}, 10^{-4}]$ / $[10^{-5}, 10^{-3}]$
LR cycle	1000 iterations
Max gradient norm	1.0
Grouped softmax size / Dropout	50 / 0.2
Loss Function Parameters	
C_{floor} / λ_{base} / μ / τ_{tol} / C_{cap}	10 / 1 / 1 / 1 / 50
Early Stopping	
No-improvement / No-change / Warm-up	800 / 400 / 100 iterations
Post-Processing	
High-confidence assortments	1
Candidate assortments (K_{cand})	20
Heuristics	
ADXOpt max removals (b)	3
Gurobi time limit	600s

D.1 MMNL Results

D.1.1 Unconstrained

To identify the hardest instances in the MMNL model, we consider all benchmarked methods: NAO, NN, NN_{PP}, ADXOpt, RO, and AlphaPhi. For each parameter setting, we select the five instances on which each method attains the largest optimality gap, and take the union of these instances across all methods. The performance of every method is then evaluated on this combined set. When the conic formulation fails to solve within the time limit (600s), we use the maximum revenue found by any method as $r^\star$ for computing optimality gaps. The detailed average optimality gaps and sample standard deviations on these hardest instances are reported in Table 5 (RS2) and Table 6 (RS4).

Table 5: Average optimality gap (%) and standard deviation for each method on MMNL unconstrained RS2 instances.

(n, C)	NAO	NN_PP	NN	ADXOpt	RO	AlphaPhi
(50,5)	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.056 (0.164)	7.163 (7.325)	7.163 (7.325)
(50,10)	**0.000 (0.000)**	**0.000 (0.000)**	0.003 (0.012)	0.001 (0.004)	5.739 (7.357)	5.739 (7.357)
(50,25)	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.031 (0.122)	3.786 (4.869)	3.786 (4.869)
(100,5)	**0.000 (0.000)**	0.215 (0.729)	0.216 (0.729)	0.000 (0.001)	5.157 (6.943)	5.157 (6.943)
(100,10)	**0.000 (0.000)**	0.051 (0.197)	0.093 (0.359)	0.250 (0.965)	5.026 (7.145)	5.026 (7.145)
(100,25)	**0.000 (0.000)**	**0.000 (0.000)**	0.003 (0.011)	**0.000 (0.000)**	3.277 (4.279)	3.277 (4.279)
(200,5)	0.003 (0.011)	**0.000 (0.000)**	0.000 (0.001)	**0.000 (0.000)**	5.990 (5.881)	5.990 (5.881)
(200,10)	**0.000 (0.000)**	0.029 (0.118)	0.033 (0.118)	0.031 (0.118)	4.656 (6.052)	4.656 (6.052)
(200,25)	**0.000 (0.000)**	0.001 (0.004)	0.011 (0.029)	0.066 (0.280)	2.237 (3.015)	2.237 (3.015)

Table 6: Average optimality gap (%) and standard deviation for each method on MMNL unconstrained RS4 instances.

(n, C)	NAO	NN_PP	NN	ADXOpt	RO	AlphaPhi
(50,5)	**0.000 (0.000)**	**0.000 (0.000)**	0.108 (0.438)	0.250 (0.745)	4.579 (6.010)	4.579 (6.010)
(50,10)	**0.000 (0.000)**	0.064 (0.271)	0.064 (0.271)	0.112 (0.409)	4.962 (5.706)	4.962 (5.706)
(50,25)	**0.000 (0.000)**	**0.000 (0.000)**	0.009 (0.035)	0.008 (0.031)	3.794 (3.912)	3.794 (3.912)
(100,5)	**0.000 (0.000)**	0.087 (0.347)	0.090 (0.346)	0.177 (0.725)	4.644 (5.759)	4.644 (5.759)
(100,10)	**0.000 (0.000)**	0.217 (0.766)	0.289 (0.887)	**0.000 (0.000)**	3.840 (5.626)	3.840 (5.626)
(100,25)	**0.000 (0.000)**	0.104 (0.401)	0.105 (0.401)	0.108 (0.400)	3.606 (4.067)	3.606 (4.067)
(200,5)	**0.002 (0.008)**	0.089 (0.386)	0.089 (0.386)	0.090 (0.386)	4.244 (4.879)	4.244 (4.879)
(200,10)	**0.000 (0.001)**	0.056 (0.224)	0.074 (0.291)	0.013 (0.034)	4.396 (5.913)	4.396 (5.913)
(200,25)	**0.000 (0.000)**	0.001 (0.004)	0.036 (0.130)	0.084 (0.366)	2.757 (3.688)	2.757 (3.688)

D.1.2 Constrained

We next extend our analysis to the cardinality-constrained MMNL setting, where a capacity rate $\in \{0.1, 0.3, 0.5\}$ is imposed on each instance. Hardest instances are again identified following the same union-of-top-five protocol, applied independently to each parameter configuration (n, C, Cap). Since the RO method does not natively incorporate cardinality constraints, it is excluded from this constrained analysis.

Table 7: Average optimality gap (%) and standard deviation for each method on MMNL cardinality-constrained RS2 instances.

(n, C, Cap)	NAO	NN_PP	NN	ADXOpt	AlphaPhi
$(50, 5, 0.1)$	**0.000 (0.000)**	**0.000 (0.000)**	0.007 (0.034)	0.008 (0.038)	3.526 (5.815)
$(50, 5, 0.3)$	**0.000 (0.000)**	**0.000 (0.000)**	0.004 (0.014)	0.021 (0.068)	3.725 (6.018)
$(50, 5, 0.5)$	**0.000 (0.000)**	**0.000 (0.000)**	0.001 (0.004)	0.039 (0.138)	4.579 (5.972)
$(50, 10, 0.1)$	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	3.658 (5.258)
$(50, 10, 0.3)$	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.001 (0.004)	5.240 (5.294)
$(50, 10, 0.5)$	**0.000 (0.000)**	**0.000 (0.000)**	0.003 (0.011)	0.001 (0.004)	6.405 (5.341)
$(50, 25, 0.1)$	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.020 (0.087)	2.151 (3.790)
$(50, 25, 0.3)$	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.026 (0.112)	2.748 (3.740)
$(50, 25, 0.5)$	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.026 (0.112)	3.836 (4.115)
$(100, 5, 0.1)$	**0.000 (0.000)**	0.013 (0.049)	0.038 (0.104)	**0.000 (0.000)**	2.395 (4.659)
$(100, 5, 0.3)$	0.031 (0.103)	0.051 (0.218)	0.083 (0.248)	**0.000 (0.001)**	3.295 (4.889)
$(100, 5, 0.5)$	**0.000 (0.000)**	0.046 (0.218)	0.047 (0.218)	0.000 (0.001)	4.162 (5.272)
$(100, 10, 0.1)$	**0.000 (0.000)**	0.081 (0.373)	0.086 (0.373)	0.081 (0.373)	2.996 (4.499)
$(100, 10, 0.3)$	**0.000 (0.000)**	0.197 (0.857)	0.197 (0.858)	0.197 (0.857)	4.040 (4.797)
$(100, 10, 0.5)$	**0.000 (0.000)**	0.052 (0.208)	0.088 (0.353)	0.234 (0.934)	5.179 (5.490)
$(100, 25, 0.1)$	**0.000 (0.000)**	0.003 (0.014)	0.022 (0.075)	**0.000 (0.000)**	1.687 (3.034)
$(100, 25, 0.3)$	**0.000 (0.000)**	0.009 (0.040)	0.029 (0.105)	**0.000 (0.000)**	2.523 (3.315)
$(100, 25, 0.5)$	**0.000 (0.000)**	**0.000 (0.000)**	0.003 (0.010)	**0.000 (0.000)**	2.702 (4.151)
$(200, 5, 0.1)$	0.027 (0.057)	0.009 (0.021)	0.014 (0.031)	**0.000 (0.000)**	2.516 (4.059)
$(200, 5, 0.3)$	0.058 (0.255)	0.013 (0.041)	0.014 (0.044)	**0.000 (0.000)**	3.976 (4.326)
$(200, 5, 0.5)$	0.012 (0.034)	0.000 (0.001)	0.001 (0.002)	**0.000 (0.000)**	5.319 (5.015)
$(200, 10, 0.1)$	**0.001 (0.003)**	0.543 (1.592)	0.659 (1.954)	0.027 (0.109)	2.256 (3.951)
$(200, 10, 0.3)$	**0.000 (0.000)**	0.029 (0.118)	0.033 (0.118)	0.031 (0.118)	3.694 (4.355)
$(200, 10, 0.5)$	**0.000 (0.000)**	0.027 (0.115)	0.034 (0.115)	0.030 (0.115)	3.515 (5.396)
$(200, 25, 0.1)$	**0.000 (0.000)**	0.063 (0.272)	0.069 (0.271)	0.062 (0.272)	1.575 (2.558)
$(200, 25, 0.3)$	**0.000 (0.000)**	**0.000 (0.000)**	0.003 (0.008)	0.062 (0.272)	2.228 (2.980)
$(200, 25, 0.5)$	**0.000 (0.000)**	**0.000 (0.000)**	0.002 (0.007)	0.070 (0.288)	2.370 (3.126)

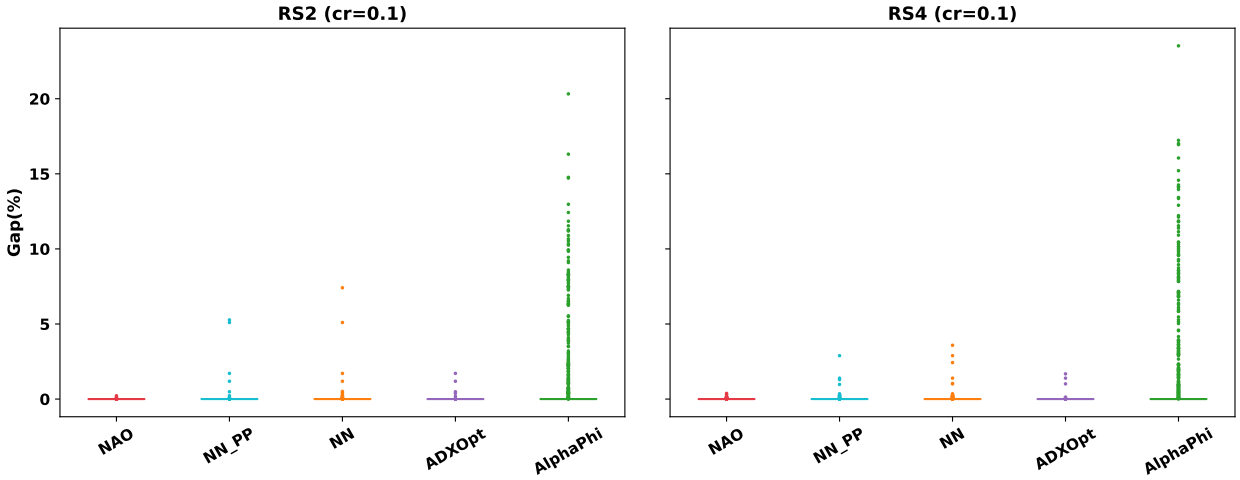
We begin with the overall performance. Figure 4 shows the distribution of optimality gaps across all 100 instances at each capacity rate. `NAO` consistently achieves gaps tightly concentrated near zero with negligible variance at all three constraint levels, demonstrating strong robustness to constraint tightness. AlphaPhi, by contrast, exhibits persistently large gaps ranging from 1.5% to over 6%, indicating its difficulty with cardinality constraints under the MMNL model. ADXOpt displays a notable pattern: its gap distribution is tighter under the most restrictive constraint (Cap Rate = 0.1), likely because the smaller feasible region channels its local search toward higher-quality solutions; as the constraint relaxes to Cap Rate = 0.5, however, its performance degrades with wider spread and more outliers, suggesting susceptibility to poor local optima in larger search spaces. NN and NN_{PP} generally perform well on average but produce occasional large outliers, particularly at larger problem sizes.

Turning to the hardest instances, Figure 5 presents heatmaps of the average optimality gap (log-scaled) for a more fine-grained comparison. `NAO` maintains near-zero gaps across virtually all parameter configurations under both the RS2 and RS4 revenue schemes. While NN_{PP} and ADXOpt remain competitive on many configurations, they degrade noticeably at larger scales—for instance, NN_{PP} reaches a gap of 0.543% at $(200, 10, 0.1)$ under RS2, whereas `NAO` stays at 0.001%. Similarly, ADXOpt matches `NAO` on several tightly constrained configurations but becomes less consistent as the capacity rate increases. The full

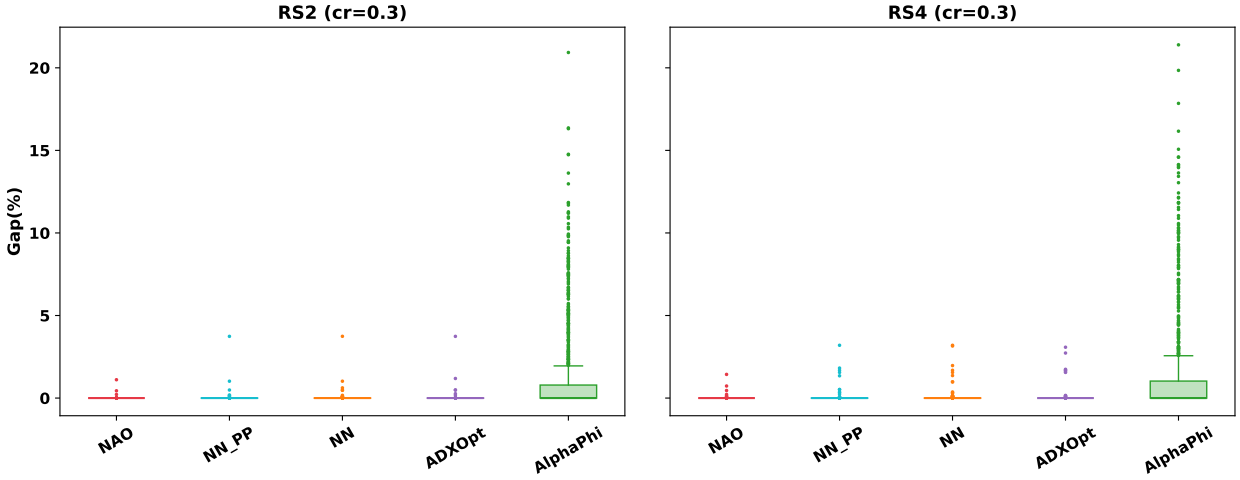
per-configuration means and sample standard deviations are reported in Table 7 (RS2) and Table 8 (RS4).

Table 8: Average optimality gap (%) and standard deviation for each method on MMNL cardinality-constrained RS4 instances.

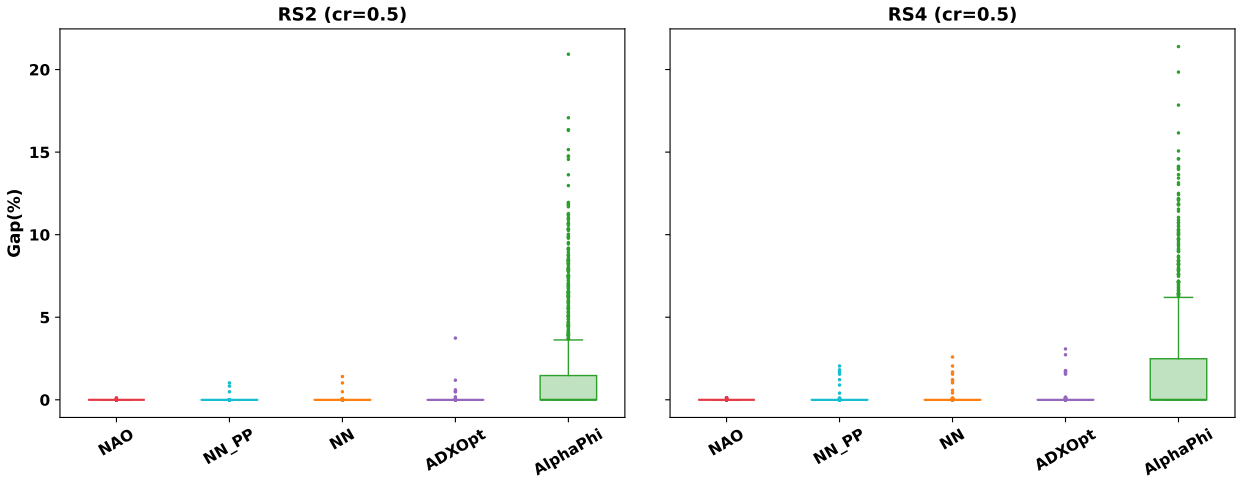
(n, C, Cap)	NAO	NN_PP	NN	ADXOpt	AlphaPhi
(50, 5, 0.1)	**0.000 (0.000)**	0.017 (0.077)	0.066 (0.236)	**0.000 (0.000)**	4.116 (7.098)
(50, 5, 0.3)	**0.000 (0.000)**	0.054 (0.270)	0.094 (0.392)	0.177 (0.629)	3.251 (6.243)
(50, 5, 0.5)	**0.005 (0.023)**	0.085 (0.418)	0.088 (0.418)	0.188 (0.650)	3.675 (6.260)
(50, 10, 0.1)	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.002 (0.007)	3.997 (6.196)
(50, 10, 0.3)	**0.000 (0.000)**	0.139 (0.667)	0.141 (0.666)	0.088 (0.363)	3.876 (6.215)
(50, 10, 0.5)	**0.000 (0.000)**	0.053 (0.254)	0.053 (0.254)	0.088 (0.363)	5.179 (6.136)
(50, 25, 0.1)	**0.000 (0.000)**	**0.000 (0.000)**	0.128 (0.557)	**0.000 (0.000)**	2.740 (4.631)
(50, 25, 0.3)	**0.000 (0.000)**	0.001 (0.003)	0.016 (0.045)	**0.000 (0.000)**	2.996 (4.732)
(50, 25, 0.5)	**0.000 (0.000)**	**0.000 (0.000)**	**0.000 (0.000)**	0.007 (0.031)	4.227 (4.516)
(100, 5, 0.1)	**0.000 (0.000)**	0.145 (0.589)	0.145 (0.589)	0.004 (0.022)	3.182 (5.508)
(100, 5, 0.3)	0.070 (0.183)	**0.024 (0.081)**	0.030 (0.084)	0.145 (0.656)	3.473 (5.239)
(100, 5, 0.5)	**0.001 (0.007)**	0.006 (0.020)	0.008 (0.023)	0.152 (0.671)	4.267 (5.377)
(100, 10, 0.1)	**0.000 (0.000)**	0.065 (0.289)	0.187 (0.800)	**0.000 (0.000)**	3.475 (5.714)
(100, 10, 0.3)	**0.000 (0.000)**	0.114 (0.429)	0.192 (0.742)	**0.000 (0.000)**	4.491 (5.656)
(100, 10, 0.5)	0.000 (0.002)	0.113 (0.411)	0.191 (0.610)	**0.000 (0.000)**	4.798 (5.697)
(100, 25, 0.1)	**0.000 (0.000)**	0.057 (0.236)	0.060 (0.246)	0.060 (0.246)	2.557 (4.233)
(100, 25, 0.3)	**0.000 (0.000)**	0.082 (0.356)	0.086 (0.356)	0.085 (0.356)	2.601 (3.945)
(100, 25, 0.5)	**0.000 (0.000)**	0.082 (0.356)	0.085 (0.356)	0.085 (0.356)	3.533 (3.981)
(200, 5, 0.1)	**0.063 (0.096)**	0.123 (0.320)	0.123 (0.320)	0.074 (0.320)	2.563 (4.573)
(200, 5, 0.3)	**0.082 (0.328)**	0.107 (0.389)	0.110 (0.388)	0.090 (0.386)	3.124 (4.517)
(200, 5, 0.5)	**0.006 (0.026)**	0.080 (0.367)	0.081 (0.367)	0.081 (0.367)	4.461 (4.581)
(200, 10, 0.1)	**0.010 (0.020)**	**0.010 (0.020)**	0.020 (0.039)	**0.010 (0.031)**	3.302 (4.740)
(200, 10, 0.3)	**0.003 (0.014)**	0.029 (0.122)	0.054 (0.231)	0.011 (0.032)	4.146 (5.166)
(200, 10, 0.5)	**0.000 (0.000)**	0.047 (0.205)	0.061 (0.267)	0.011 (0.032)	4.742 (5.392)
(200, 25, 0.1)	0.028 (0.076)	**0.006 (0.014)**	0.018 (0.042)	0.088 (0.385)	1.849 (3.380)
(200, 25, 0.3)	**0.000 (0.000)**	0.031 (0.122)	0.138 (0.382)	0.089 (0.376)	2.162 (3.361)
(200, 25, 0.5)	**0.007 (0.032)**	0.008 (0.026)	0.062 (0.160)	0.089 (0.376)	2.755 (3.810)



(a) Capacity rate = 0.1

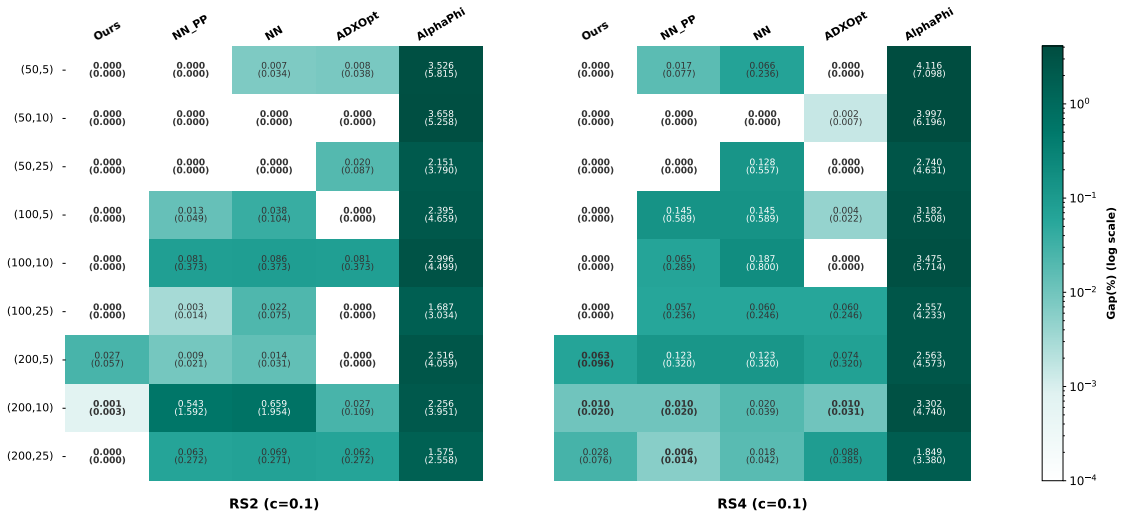


(b) Capacity rate = 0.3



(c) Capacity rate = 0.5

Figure 4: Distribution of optimality gaps (%) over 100 instances for the MMNL model under cardinality constraints with varying capacity rates.

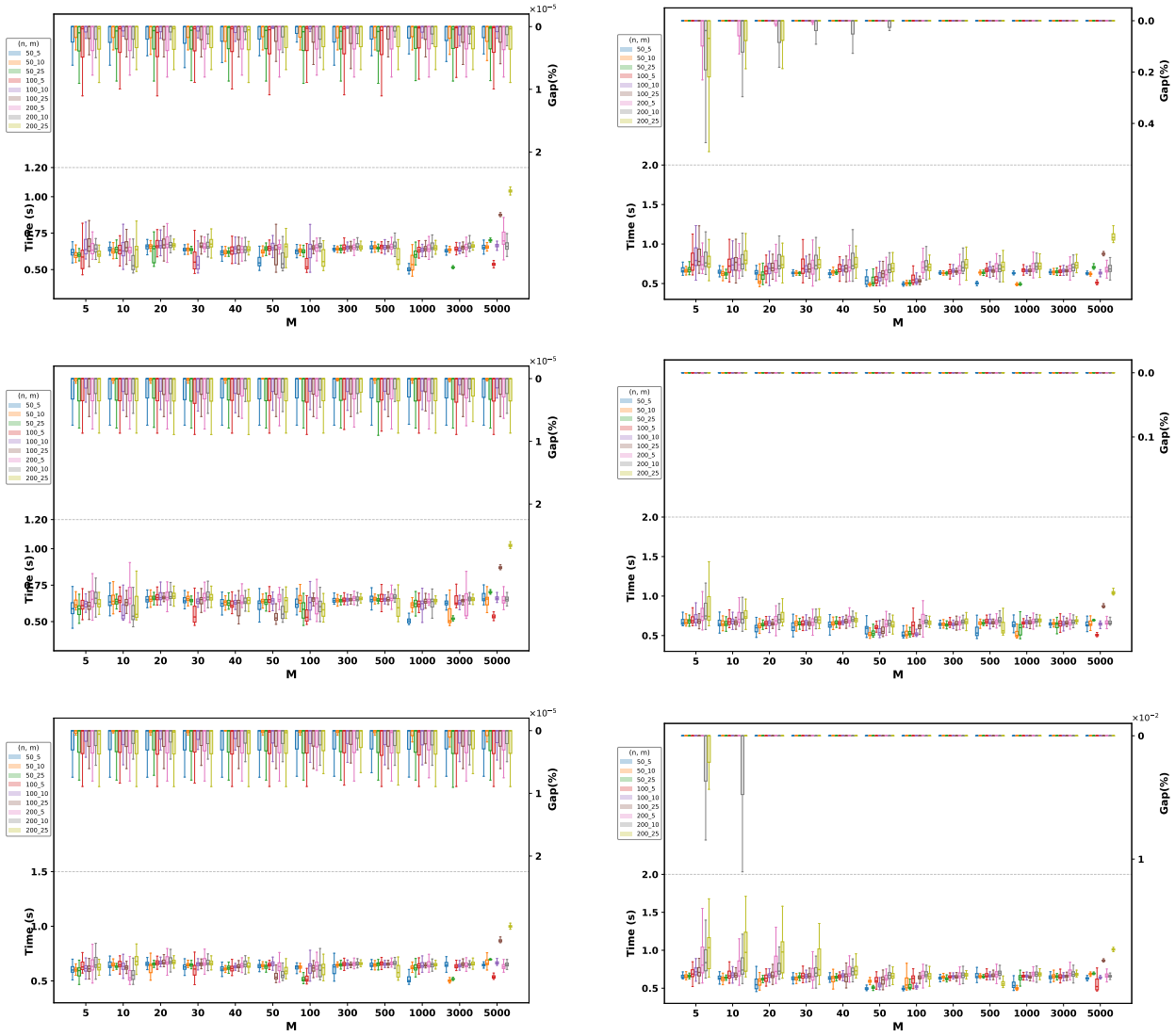


(a) Capacity rate = 0.1

(b) Capacity rate = 0.3

(c) Capacity rate = 0.5

Figure 5: Average optimality gap (%, log-scaled) on the hardest instances of MMNL under cardinality constraints. RO and AlphaPhi are omitted due to relatively large gaps. Full results can be found in Table 7 and Table 8.



RS2 RS4

Figure 6: Optimality gap & computational time vs. number of particles M for the MMNL model under cardinality constraints. Left column: RS2; right column: RS4. Rows from top to bottom: capacity rate = 0.1, 0.3, 0.5.

D.2 NL Results

For the NL model, `NAO` operates over a flattened decision space of $N = G \times n$ items, where each product j in nest g is mapped to a unique index $g \cdot n + j$. The solver maintains M particles $\mathbf{x} = (x_m)_{m=1}^{M} \in [0,1]^N$, each representing a continuous relaxation of the binary assortment decision.

Unconstrained NL In the unconstrained setting, each particle's entries are sorted in descending order to obtain a permutation π^{x_m}, inducing the chain of prefix sets $S_k(x_m) = \{\pi^{x_m}(1), \ldots, \pi^{x_m}(k)\}$. The chain revenue at each $S_k(x_m)$ requires evaluating the NL revenue formula, which depends on per-nest aggregate quantities. Specifically, for each nest g and prefix $S_k(x_m)$, the algorithm tracks the inclusive value $V_g(S_k) = v_0^g + \sum_{j \in S_k \cap V_g} v_{gj}$ and the revenue numerator $R_g(S_k) = \sum_{j \in S_k \cap V_g} r_{gj} \cdot v_{gj}$. Since items are added one at a time along the sorted order, both quantities can be updated incrementally: when the $(k+1)$-th item $\pi^{x_m}(k+1)$ belongs to nest g, only V_g and R_g change, while all other nests' aggregates remain the same.

This yields the full chain of N revenue values

$$r(S_k(x_m)) = \sum_{g=1}^{G} \frac{R_g(S_k)}{V_g(S_k)} \cdot \frac{V_g(S_k)^{\gamma_g}}{v_0 + \sum_{h=1}^{G} V_h(S_k)^{\gamma_h}}, \qquad k = 1, \ldots, N,$$

in $O(N \cdot G)$ time per particle.

Constrained NL with per-nest cardinality constraints. When each nest g is subject to an independent cardinality constraint $|S \cap V_g| \leq K_g$, the feasible assortments form a partition matroid over the product space. NAO exploits this structure by extending the rolling-window Lovász extension ϕ_K to operate independently within each nest, replacing the global window sets $W_k(x)$ with per-nest windows.

As items are added along the sorted order π^{x_m}, we maintain a running count $c_g(k)$ of how many items from nest g have appeared among $\{\pi^{x_m}(1), \ldots, \pi^{x_m}(k)\}$. When $c_g(k)$ exceeds K_g, the earliest item from nest g must be evicted to maintain feasibility—precisely the per-nest analogue of the global rolling window in Definition 2. Let $\mathrm{pos}_g(j)$ denote the sorted position at which the j-th item of nest g appears. The excess count at position k is $e_g(k) = \max(c_g(k) - K_g, 0)$, and the per-nest aggregates under the rolling window become

$$V_g(W_k) = v_0^g + \mathrm{cumsum}_g[k] - \mathrm{cumsum}_g[\mathrm{pos}_g(e_g(k))],$$
$$R_g(W_k) = \mathrm{rv\_cumsum}_g[k] - \mathrm{rv\_cumsum}_g[\mathrm{pos}_g(e_g(k))].$$

That is, only the most recent $\min(c_g(k), K_g)$ items from each nest contribute to the revenue at every position k.

Rounding. The rounding operator $\mathrm{Round}_K(\cdot)$ from Proposition 4 is applied independently within each nest. Items are visited in sorted order; for each nest g, only the most recent K_g items are retained. The position $k^* \in \arg\max_{0 \leq k \leq N} r(W_k(x_m))$ that maximizes the chain revenue determines the final assortment, which is then evaluated exactly under the NL revenue formula.

D.2.1 Unconstrained

Having described the per-nest rolling-window implementation of NAO for the NL model, we now evaluate its empirical performance. Since exact optimal solutions are generally intractable for the NL assortment problem, we follow Guo et al. [28] and measure performance against the tight LP relaxation upper bound proposed by Kunnumkal [37]. Specifically, for a heuristic solution with revenue R, the optimality gap is defined as $100 \times (UB - R)/UB$, where UB denotes the LP upper bound. In addition to the methods used in the MMNL experiments, we include the LP-based heuristic of Kunnumkal [37] as a benchmark, which is specifically designed for the NL model and provides strong performance in this setting.

Hardest instances are identified using the same union-of-top-five protocol as in the MMNL analysis.

Table 9: Average optimality gap (%) and standard deviation for each method on NL unconstrained instances with low $v_0^g \sim U[0, 1]$.

(n, G)	NAO	NN_PP	NN	AlphaPhi	ADXOpt	RO
(25,5)	**0.069 (0.081)**	0.113 (0.128)	0.263 (0.361)	2.503 (2.300)	2.784 (4.645)	0.113 (0.111)
(25,10)	**0.042 (0.056)**	0.355 (0.450)	0.613 (0.689)	2.726 (1.456)	3.001 (4.035)	0.350 (0.510)
(25,20)	**0.032 (0.041)**	0.367 (0.357)	0.499 (0.496)	3.901 (1.882)	2.089 (2.599)	0.812 (0.819)
(50,5)	**0.016 (0.025)**	0.125 (0.197)	0.270 (0.419)	2.006 (1.552)	2.379 (3.568)	0.139 (0.217)
(50,10)	**0.018 (0.035)**	0.239 (0.315)	0.334 (0.391)	3.458 (1.400)	2.974 (3.596)	0.324 (0.291)
(50,20)	**0.014 (0.014)**	0.424 (0.432)	0.502 (0.477)	4.253 (1.424)	2.775 (2.676)	0.522 (0.531)

Table 10: Average optimality gap (%) and standard deviation for each method on NL unconstrained instances with high $v_0^g \sim U[3,4]$.

(n, G)	NAO	NN_PP	NN	AlphaPhi	ADXOpt	RO
(25,5)	**0.005 (0.011)**	0.217 (0.382)	0.380 (0.567)	2.590 (1.978)	4.155 (5.326)	12.130 (13.209)
(25,10)	**0.004 (0.006)**	0.377 (0.458)	0.580 (0.552)	2.967 (1.414)	5.507 (4.437)	14.142 (17.179)
(25,20)	**0.006 (0.010)**	0.428 (0.280)	0.505 (0.304)	3.024 (1.330)	6.630 (4.555)	33.894 (23.729)
(50,5)	**0.003 (0.006)**	0.425 (0.708)	0.520 (0.742)	2.375 (1.208)	7.441 (6.567)	8.100 (10.576)
(50,10)	**0.003 (0.007)**	0.455 (0.590)	0.512 (0.634)	3.066 (1.308)	9.461 (4.908)	16.615 (14.556)
(50,20)	**0.003 (0.005)**	0.475 (0.398)	0.529 (0.407)	3.014 (1.282)	10.451 (3.907)	28.934 (19.827)

Tables 9 and 10 present the results on hardest instances under the low ($v_0^g \sim U[0,1]$) and high ($v_0^g \sim U[3,4]$) within-nest no-purchase utility settings, respectively.

In the low-v_0^g setting (Table 9), NAO achieves the smallest average gap relative to the LP upper bound across all six configurations, ranging from 0.014% to 0.069%. The RO heuristic proves surprisingly competitive in this regime, with gaps between 0.113% and 0.812%, while NN_{PP} follows closely behind NAO. ADXOpt and AlphaPhi both exhibit substantially larger gaps, frequently exceeding 2–4%.

The high-v_0^g setting (Table 10) reveals a much sharper separation among the methods. NAO maintains remarkably small gaps below 0.01% across all configurations. The RO heuristic, which performed reasonably in the low-v_0^g case, deteriorates dramatically—reaching gaps of 33.9% at $(50, 20)$—confirming that high within-nest no-purchase utilities fundamentally disrupt the revenue-ordering principle. ADXOpt also degrades substantially, with gaps growing monotonically from 4.2% at $(25, 5)$ to over 10% as the number of nests increases, suggesting that its local search becomes increasingly ineffective in this parameter regime. NN_{PP} remains the second-best method overall but still lags behind NAO by roughly two orders of magnitude. These results demonstrate that NAO's optimization framework is particularly effective under the NL structure, where specialized heuristics can fail under certain parameter regimes.

D.2.2 Constrained

We next impose per-nest cardinality constraints with capacity rates $\in \{0.1, 0.3, 0.5\}$ on each NL instance. Hardest instances are again identified using the union-of-top-five protocol, applied independently to each configuration $(n, G, \text{Cap Rate})$. The RO and LP-based methods are excluded from this constrained analysis as they do not natively handle per-nest cardinality constraints.

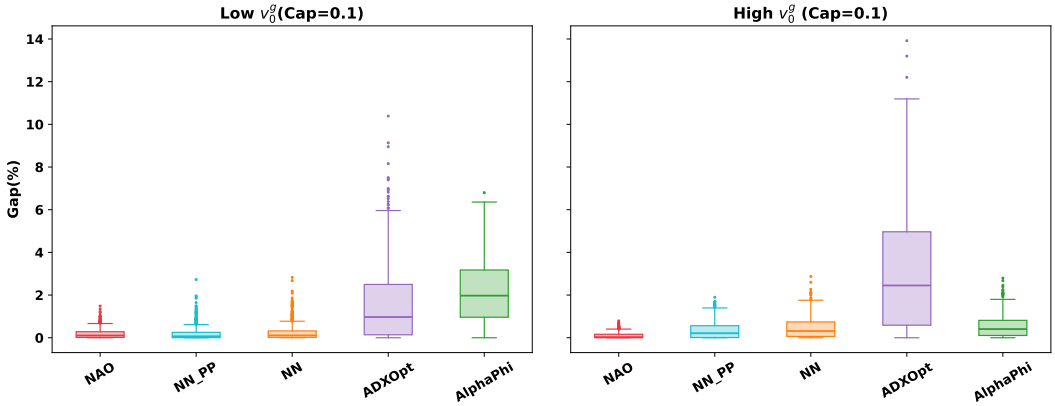
Table 11: Average optimality gap (%) and standard deviation for each method on NL cardinality-constrained instances with low $v_0^g \sim U[0, 1]$.

(n, G, Cap)	NAO	NN_PP	NN	ADXOpt	AlphaPhi
$(25, 5, 0.1)$	**0.097 (0.150)**	0.143 (0.241)	0.423 (0.582)	1.718 (2.633)	1.822 (1.564)
$(25, 5, 0.3)$	**0.133 (0.255)**	0.161 (0.226)	0.377 (0.495)	2.661 (4.442)	2.357 (2.247)
$(25, 5, 0.5)$	**0.069 (0.081)**	0.125 (0.147)	0.271 (0.370)	2.784 (4.645)	2.503 (2.300)
$(25, 10, 0.1)$	**0.212 (0.235)**	0.367 (0.535)	0.460 (0.618)	2.081 (2.373)	2.341 (1.068)
$(25, 10, 0.3)$	**0.073 (0.080)**	0.594 (0.783)	0.795 (0.935)	2.842 (4.008)	2.597 (1.506)
$(25, 10, 0.5)$	**0.045 (0.059)**	0.385 (0.489)	0.687 (0.785)	3.221 (4.175)	2.711 (1.510)
$(25, 20, 0.1)$	**0.270 (0.306)**	0.536 (0.599)	0.702 (0.760)	1.322 (1.602)	3.819 (1.358)
$(25, 20, 0.3)$	**0.047 (0.067)**	0.361 (0.374)	0.503 (0.483)	1.906 (2.499)	3.606 (1.957)
$(25, 20, 0.5)$	**0.031 (0.042)**	0.388 (0.365)	0.505 (0.480)	2.109 (2.653)	3.990 (1.868)
$(50, 5, 0.1)$	0.272 (0.330)	**0.258 (0.391)**	0.361 (0.545)	2.594 (3.468)	1.524 (1.378)
$(50, 5, 0.3)$	**0.028 (0.074)**	0.113 (0.192)	0.263 (0.449)	2.660 (3.596)	1.982 (1.545)
$(50, 5, 0.5)$	**0.016 (0.025)**	0.125 (0.197)	0.270 (0.419)	2.379 (3.568)	2.006 (1.552)
$(50, 10, 0.1)$	0.345 (0.288)	**0.299 (0.401)**	0.380 (0.437)	2.986 (2.605)	2.788 (1.282)
$(50, 10, 0.3)$	**0.027 (0.045)**	0.183 (0.285)	0.318 (0.430)	2.961 (3.499)	3.329 (1.464)
$(50, 10, 0.5)$	**0.023 (0.039)**	0.234 (0.311)	0.320 (0.395)	2.910 (3.531)	3.472 (1.371)
$(50, 20, 0.1)$	**0.569 (0.434)**	0.621 (0.668)	0.697 (0.675)	2.526 (2.052)	4.067 (1.221)
$(50, 20, 0.3)$	**0.038 (0.045)**	0.426 (0.436)	0.512 (0.488)	2.684 (2.779)	4.150 (1.400)
$(50, 20, 0.5)$	**0.022 (0.024)**	0.413 (0.429)	0.502 (0.466)	2.665 (2.682)	4.082 (1.487)

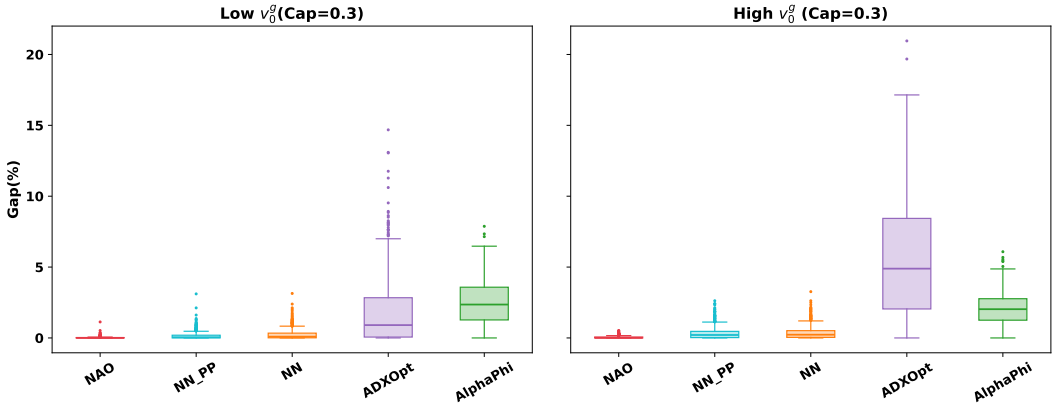
We begin with the overall performance. Figure 7 displays the distribution of optimality gaps (relative to the LP upper bound) across all 100 instances at each capacity rate. NAO consistently produces the tightest gap distributions, though the absolute gap levels are somewhat higher than in the MMNL constrained setting, reflecting both the greater inherent difficulty of the NL model and the conservatism of the LP upper bound. AlphaPhi shows relatively stable gaps in the 1–4% range, while ADXOpt exhibits substantially larger and more variable gaps, particularly in the high-v_0^g setting.

Turning to the hardest instances, Figure 8 presents heatmaps of the average optimality gap (log-scaled) for a finer-grained comparison. A clear distinction emerges between the two v_0^g regimes. In the low-v_0^g setting (Table 11), NAO achieves gaps mostly below 0.3%, with NN_{PP} as the closest competitor. Under tight constraints (Cap Rate = 0.1), NAO shows slightly elevated gaps at larger configurations—for example, 0.569% at $(50, 20, 0.1)$—but these remain far below those of ADXOpt (2.526%) and AlphaPhi (4.067%) on the same instances. Notably, at $(50, 5, 0.1)$ and $(50, 10, 0.1)$, NN_{PP} slightly outperforms NAO (0.258% vs. 0.272% and 0.299% vs. 0.345%, respectively), suggesting that tight per-nest constraints on larger problems can occasionally challenge NAO's continuous relaxation. As the constraint relaxes to Cap Rate = 0.3 and 0.5, however, NAO recovers its clear dominance, achieving gaps well below 0.1% in most configurations.

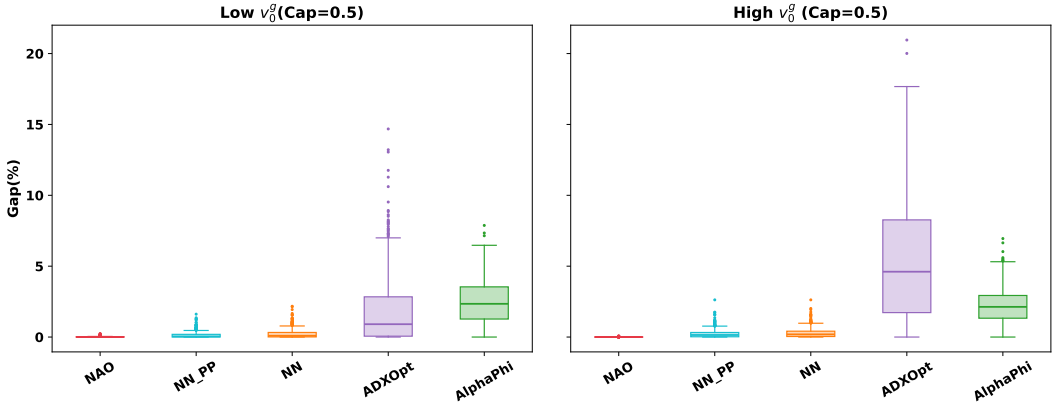
In the high-v_0^g setting (Table 12), the advantage of NAO becomes even more pronounced. NAO achieves gaps below 0.2% across nearly all configurations, while ADXOpt deteriorates sharply—frequently exceeding 5% and reaching 10.6% at $(50, 20, 0.3)$—indicating that its local search struggles with the combined difficulty of high nest-level no-purchase attractions and cardinality constraints. AlphaPhi performs noticeably better under tight constraints (Cap Rate = 0.1), with gaps around 0.6–1.0%, but degrades to 2–3% as the constraint relaxes. NN_{PP} maintains gaps in the 0.1–1.0% range, consistently ranking second behind NAO but trailing by a substantial margin. The full per-configuration means and sample standard deviations are reported in Table 11 (low v_0^g) and Table 12 (high v_0^g).



(a) Capacity rate = 0.1

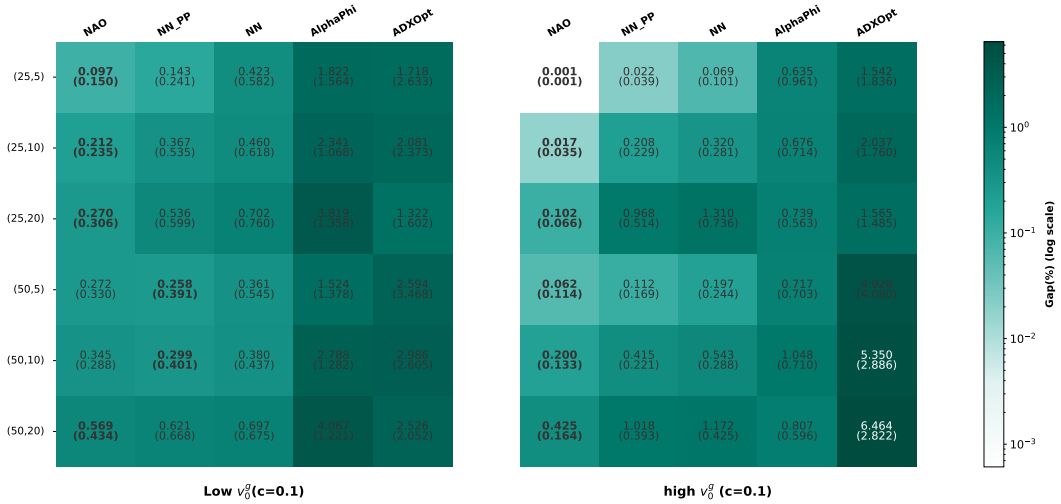


(b) Capacity rate = 0.3

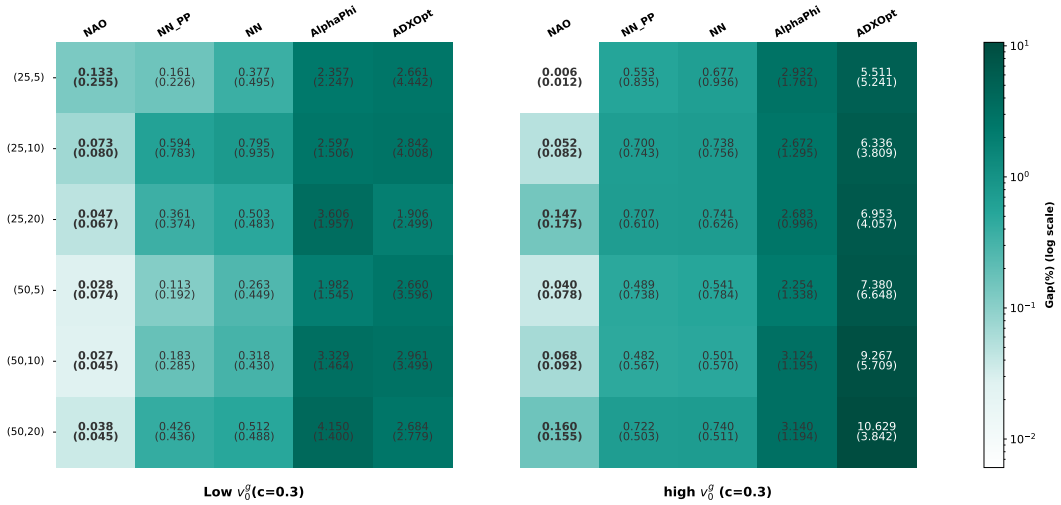


(c) Capacity rate = 0.5

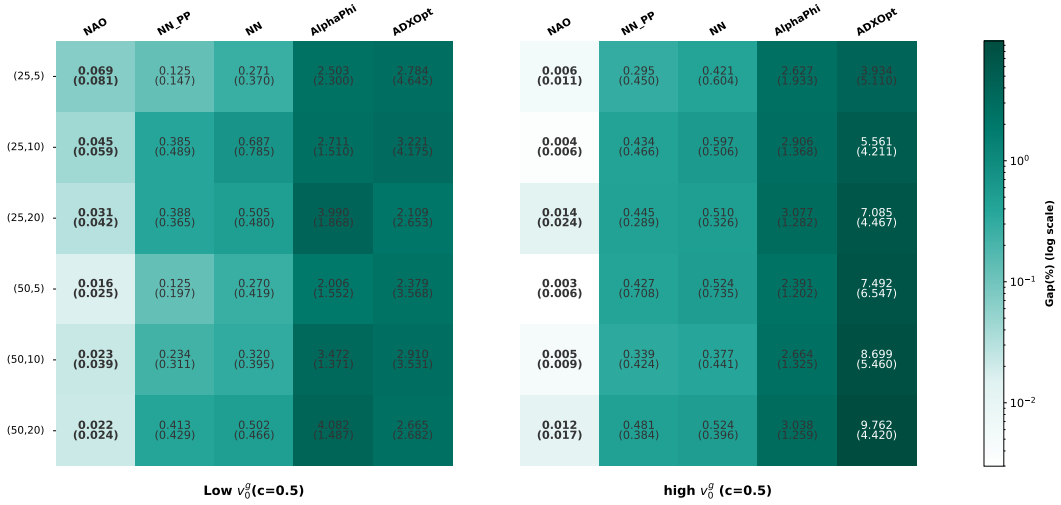
Figure 7: Distribution of optimality gaps (%) over 100 instances for the NL model under cardinality constraints with varying capacity rates.



(a) Capacity rate = 0.1

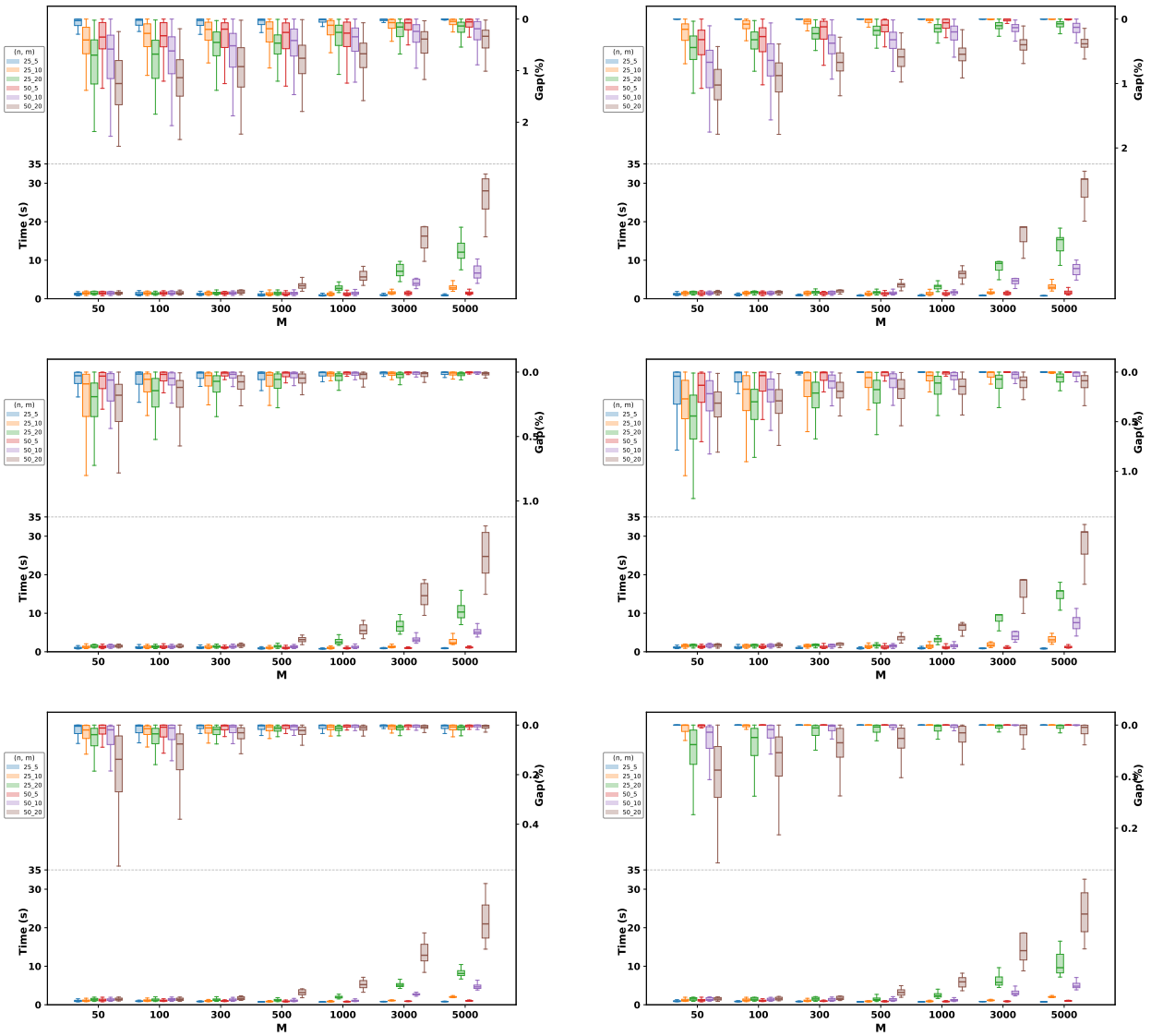


(b) Capacity rate = 0.3



(c) Capacity rate = 0.5

Figure 8: Average optimality gap (%, log-scaled) on hardest instances of NL under cardinality constraints. RO and AlphaPhi are omitted due to relatively large gaps. Full results can be found in Table 11 and Table 12.



Low $v_0^g \sim U[0,1]$ High $v_0^g \sim U[3,4]$

Figure 9: Optimality gap & computational time vs. number of particles M for the NL model under cardinality constraints. Left column: low $v_0^g \sim U[0,1]$; right column: high $v_0^g \sim U[3,4]$. Rows from top to bottom: capacity rate = 0.1, 0.3, 0.5.

Table 12: Average optimality gap (%) and standard deviation for each method on NL cardinality-constrained instances with high $v_0^g \sim U[3,4]$.

(n, G, Cap)	NAO	NN_PP	NN	ADXOpt	AlphaPhi
$(25, 5, 0.1)$	**0.001 (0.001)**	0.022 (0.039)	0.069 (0.101)	1.542 (1.836)	0.635 (0.961)
$(25, 5, 0.3)$	**0.006 (0.012)**	0.553 (0.835)	0.677 (0.936)	5.511 (5.241)	2.932 (1.761)
$(25, 5, 0.5)$	**0.006 (0.011)**	0.295 (0.450)	0.421 (0.604)	3.934 (5.110)	2.627 (1.933)
$(25, 10, 0.1)$	**0.017 (0.035)**	0.208 (0.229)	0.320 (0.281)	2.037 (1.760)	0.676 (0.714)
$(25, 10, 0.3)$	**0.052 (0.082)**	0.700 (0.743)	0.738 (0.756)	6.336 (3.809)	2.672 (1.295)
$(25, 10, 0.5)$	**0.004 (0.006)**	0.434 (0.466)	0.597 (0.506)	5.561 (4.211)	2.906 (1.368)
$(25, 20, 0.1)$	**0.102 (0.066)**	0.968 (0.514)	1.310 (0.736)	1.565 (1.485)	0.739 (0.563)
$(25, 20, 0.3)$	**0.147 (0.175)**	0.707 (0.610)	0.741 (0.626)	6.953 (4.057)	2.683 (0.996)
$(25, 20, 0.5)$	**0.014 (0.024)**	0.445 (0.289)	0.510 (0.326)	7.085 (4.467)	3.077 (1.282)
$(50, 5, 0.1)$	**0.062 (0.114)**	0.112 (0.169)	0.197 (0.244)	4.926 (4.080)	0.717 (0.703)
$(50, 5, 0.3)$	**0.040 (0.078)**	0.489 (0.738)	0.541 (0.784)	7.380 (6.648)	2.254 (1.338)
$(50, 5, 0.5)$	**0.003 (0.006)**	0.427 (0.708)	0.524 (0.735)	7.492 (6.547)	2.391 (1.202)
$(50, 10, 0.1)$	**0.200 (0.133)**	0.415 (0.221)	0.543 (0.288)	5.350 (2.886)	1.048 (0.710)
$(50, 10, 0.3)$	**0.068 (0.092)**	0.482 (0.567)	0.501 (0.570)	9.267 (5.709)	3.124 (1.195)
$(50, 10, 0.5)$	**0.005 (0.009)**	0.339 (0.424)	0.377 (0.441)	8.699 (5.460)	2.664 (1.325)
$(50, 20, 0.1)$	**0.425 (0.164)**	1.018 (0.393)	1.172 (0.425)	6.464 (2.822)	0.807 (0.596)
$(50, 20, 0.3)$	**0.160 (0.155)**	0.722 (0.503)	0.740 (0.511)	10.629 (3.842)	3.140 (1.194)
$(50, 20, 0.5)$	**0.012 (0.017)**	0.481 (0.384)	0.524 (0.396)	9.762 (4.420)	3.038 (1.259)

D.3 Ablation Studies

This section examines the impact of the hyperparameters–specifically M and (λ, τ)–in the NAO algorithm on the trade-offs between performance and efficiency.

D.3.1 Number of Particles

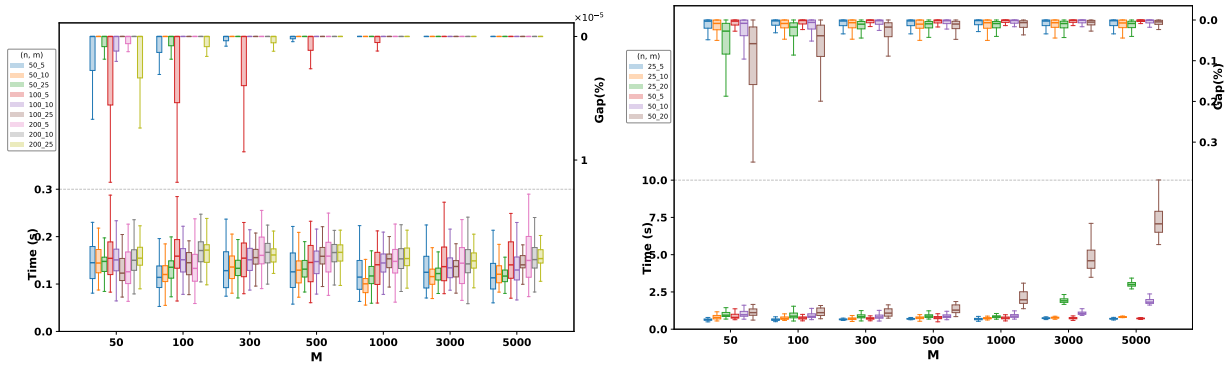
Using the full dataset, we evaluate the optimality gap and the computational time of NAO across $M = \{50, 100, 300, 500, 1000, 3000, 5000\}$. Figure 10 summarizes one representative setup each for MMNL and NL under the unconstrained setting, where M increases in a roughly exponential manner on the x-axis. The findings are summarized as follows.

- Performance: First of all, even with a small $M = 50$, the optimality gap remains extremely low, around $10^{-5}\%$ for MMNL and 0.3% for the other setting. Secondly, for both MMNL and NL, the optimality gap decreases quickly from $M = 50$ to $M = 500$ and becomes stable with sufficiently large $M = 3000, 5000$, consistent with the trend of $\frac{1}{M}$ approximation error in Theorem 1. This also justifies our decision to choose M not smaller than 3000, which suffices for the nearly optimal convergence. Note that the MMNL instances approach zero gap as they are easier problems that converge faster than NL instances.

- Time: For unconstrained MMNL instances, computational time does not vary significantly across different values of M. In contrast, for some harder NL configurations—such as larger problem instances $(25, 20)$, $(50, 10)$, and $(50, 20)$—we observe an approximately linear increase in runtime as the number of particles grows. This behavior is expected, since M particles must be evaluated and rounded at each iteration. Moreover, the patience-based stopping criterion further contributes to this trend: with more

particles, the likelihood of achieving an objective improvement in a given iteration increases, which in turn prolongs the time required to reach the stopping threshold.

Under the capacity-constrained setting, we report results in Figure 6 for MMNL and Figure 9 for NL as a function of the number of particles. For MMNL, we consider a wider range of particle counts, with the smallest value set to $M = 5$. Overall, the observations largely mirror those in the unconstrained setting. An interesting finding is that for MMNL (RS2) instances, even with as few as five particles, the optimality gap already shrinks to a sufficiently small level, and increasing the number of particles yields only marginal additional improvements.



(a) MMNL (RS2). (b) NL (low v_0^g).

Figure 10: Optimality Gap & Computational Time v.s. Number of Particles M under the unconstrained setting.

D.3.2 Weight decay and Noise

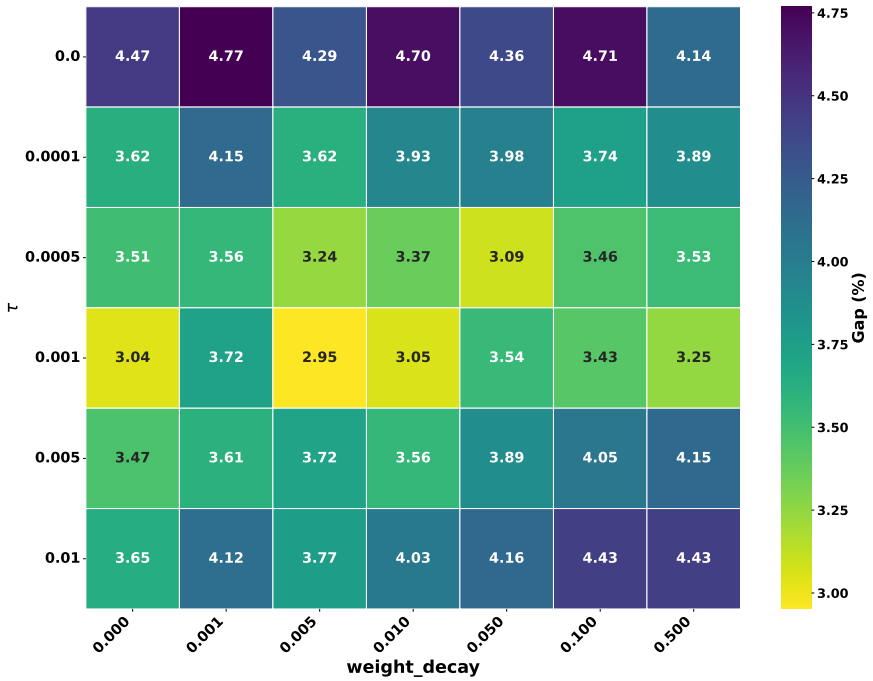


Figure 11: Grid of Optimality Gap w.r.t. Weight Decay and Noise pair (λ, τ) for union of five hardest constrained NL instances with low v_0^g, $n = 25$, $G = 5$, and 0.1 capacity rate.

We investigate the impact of two key tuning parameters—weight decay λ and noise level τ—on the performance of NAO by focusing on a hard constrained NL instance with a relatively large optimality gap. We fix $M = 10{,}000$ to minimize the effect of particle approximation error and evaluate all 42 parameter combinations with $\tau \in \{0, 0.0001, 0.0005, 0.001, 0.005, 0.01\}$ and $\lambda \in \{0, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5\}$. Our results indicate that (i) incorporating weight decay—and, in particular, adding noise—can substantially improve performance. For instance, the intermediate setting $\tau = 0.001$ and $\lambda = 0.005$ achieves the smallest optimality gap, whereas increasing weight decay in the absence of noise leads to degraded performance. Moreover, overly large parameter values can compromise optimality, as they introduce greater bias into the gradient estimates.

D.4 Large-Scale Experiments

We evaluate the scalability of all methods on large MMNL instances under the RS2 revenue scheme with $C = 10$ customer segments. For each problem size $n \in \{100, 300, 500, 1{,}000, 2{,}000, 3{,}000, 5{,}000, 10{,}000\}$, we generate 30 instances. All benchmark methods are executed using publicly available code from their respective authors, or, where unavailable, our own efficient implementations following the original papers. To ensure a fair comparison, the stopping criteria for competing methods are set generously and are not tuned in favor of NAO.

A key factor in NAO's scalability is its vectorized computation of the Lovász extension. Rather than evaluating $f(S_k)$ independently for each prefix set, NAO exploits the closed-form structure of the MMNL revenue to compute the entire chain of n revenue values simultaneously via cumulative sums on GPU. Specifically, sorting the particle entries costs $O(n \log n)$; the subsequent per-type numerators $\sum_{j \in S_k} r_j v_j^c$ and denominators $v_0^c + \sum_{j \in S_k} v_j^c$ are obtained as prefix cumulative sums in $O(n \cdot C)$, yielding all chain revenues $\{r(S_k)\}_{k=1}^n$ without any sequential loop. All M particles are processed as a single batched tensor operation, so the per-iteration cost is $O(M \cdot n \cdot C)$, fully utilizing GPU parallelism.

NAO uses $M = 3{,}000$ particles. To ensure fair comparison on the computational time, we also fix the tuning parameters of NAO with learning rate=0.01, $\lambda = 0.05$, $\tau = 0.0005$, $\beta = 5$, Optimizer=AdamW, as contrary to Table 3. For NN and NN_{PP}, we use `train_num` = 1,500 iterations with early stopping (`patience` = 800, `min_delta` = 10^{-5}, `nochange_epochs` = 400, `warmup_epochs` = 100). Due to hardware limitations, the number of independently trained models is reduced to 1 for $n = 5{,}000$ and set to 2 for all other problem sizes. For ADXOpt, the default per-instance time limit is 600 seconds; for $n \in \{3{,}000, 5{,}000\}$, this is reduced to 90 seconds with the maximum number of removals set to $b = 1$ to keep computation tractable. The conic integer formulation is excluded as a competing method in this comparison, as its solve time exceeds 600 seconds for $n \geq 1{,}000$. For $n = 10{,}000$, only NAO and AlphaPhi are reported, as NN, NN_{PP}, and ADXOpt could not be executed within a reasonable time frame given our hardware constraints.

Since exact optimal solutions are unavailable at these scales, we measure each method's optimality gap relative to the best-known solution found across all methods. For small problem sizes where the conic integer formulation terminates within its time limit, its solution is also included when determining the best-known value.

All experiments are conducted using the best available computational resources for each method. NAO, NN, and NN_{PP} ADXOpt are executed on a single NVIDIA RTX 5090 GPU (32GB) with 25 vCPUs (Intel Xeon Platinum 8470Q). AlphaPhi is executed on a local workstation with an Intel Core Ultra 7 155H CPU, as the Gurobi solver it relies on is only available on this machine. We acknowledge that the heterogeneous hardware makes wall-clock time comparisons across methods approximate rather than exact; however, we report runtimes transparently and note that the relative scaling behavior—how each method's runtime grows with n—remains informative regardless of the absolute hardware differences.

Besides the stated methods, we also tested XSET and ABD by [36], which solves the assortment problem under a different setting where the choice model is rank-based, and the choice probability is not fully

accessible.

To ensure faithful time estimation, we report the reproduction of their Experiment_1 on our device:

Device\Time(s)	N	M	$\widetilde{K}$	L	Budget	XSET	MIP	ABD
Our reproduction	50	5	25000	5	Inf	2.61	23.54	44.38
Khalid & Sturt [36]	50	5	25000	5	Inf	3.45	23.01	70.01

Table 13: Runtime reproduction of Experiment_1 from Khalid & Sturt [36].

Under the setup in the unconstrained large-scale experiment in Section 6.3, even with the maximum allowed runtime of 1800s and a small training size of $\widetilde{K} = 500$, all tests time out (N/A) except XSET at $n = 100$. NAO consistently outperforms in both runtime and quality.

2*n	Mean(Std) Time(seconds)			Mean(Std) Gap(%)		
	NAO	XSET	ABD	NAO	XSET	ABD
100	0.46 (0.32)	1135.78 (612.47)	N/A	0.0001 (0.0000)	2.2740 (1.1011)	N/A
300	0.39 (0.06)	N/A	N/A	0.0001 (0.0000)	N/A	N/A

Table 14: Comparison of NAO, XSET, and ABD. Time is reported in seconds, and gap is reported in percentage; sample standard deviations are shown in parentheses. N/A indicates that the method times out or could not be executed within the maximum allowed runtime.