Multi-Objective Optimization via Wasserstein-Fisher-Rao Gradient Flow

Yinuo Ren Stanford University
Tesi Xiao Amazon
Tanmay Gangwani Amazon
Anshuka Rangi Amazon
Holakou Rahmanian Amazon
Lexing Ying Stanford University
Subhajit Sanyal Amazon

Abstract

Multi-objective optimization (MOO) aims to optimize multiple, possibly conflicting objectives with widespread applications. We introduce a novel interacting particle method for MOO inspired by molecular dynamics simulations. Our approach combines overdamped Langevin and birth-death dynamics, incorporating a "dominance potential" to steer particles toward global Pareto optimality. In contrast to previous methods, our method is able to relocate dominated particles, making it particularly adept at managing Pareto fronts of complicated geometries. Our method is also theoretically grounded as a Wasserstein-Fisher-Rao gradient flow with convergence guarantees. Extensive experiments confirm that our approach outperforms state-of-the-art methods on challenging synthetic and real-world datasets.

1 INTRODUCTION

Multi-objective optimization (MOO) addresses optimization scenarios where multiple objectives are simultaneously and systematically optimized. Given that these objectives may inherently conflict, MOO seeks to identify a diversified set of solutions on the *Pareto front*, where no solution can enhance one objective without deteriorating at least one other. Determining the Pareto front is intricate due to its typically non-closed-form expression and potentially complicated geometries. Real-world applications of MOO span various domains, including control systems (Gambier and Badreddin, 2007), energy saving (Cui et al., 2017), and economics and finance (Tapia and Coello, 2007).

Over the past few decades, MOO has been extensively explored in literature. Notable traditional methods include the evolutionary algorithms (Tamaki et al., 1996; Deb et al., 2002a; Konak et al., 2006; Reyes-Sierra et al., 2006; Zhang and Li, 2007; Zhou et al., 2011), and Bayesian optimization (Laumanns and Ocenasek, 2002; Belakaria et al., 2020; Konakovic Lukovic et al., 2020; Tu et al., 2022). These methods, while effective, can be computationally expensive and less adaptable to high-dimensional problems. Recently, gradient-based MOO methods have been developed for various machine learning tasks (Sener and Koltun, 2018). Many such methods rely on *preference vectors* (Lin et al., 2019; Mahapatra and Rajan, 2020; Liu and Vicente, 2021). However, their efficacy often hinges on the vector selection, making them potentially heuristic for unknown Pareto fronts. Another line of research has explored *hypernetwork* methods (Navon et al., 2020; Lin et al., 2020; Ruchte and Grabocka, 2021b; Chen and Kwok, 2022; Hoang et al., 2023), which may fail to capture the discontinuity of the Pareto front and thus only work for simple Pareto front geometries. We refer the readers to Appendix A for a more comprehensive literature review.

Contributions. In this paper, we propose a novel interacting particle method for MOO inspired by molecular dynamics simulations in computational physics[1]. Particle diversity is maintained by integrating a repulsive two-body interatomic potential. A standout feature of our method is the introduction of the *dominance potential*, which assigns higher potential to particles being dominated. Together with the incorporation of the stochastic birth-death process,

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume TBD.

[1]Code is accessible at `https://github.com/yinuoren/particlewfr`

we facilitate the direct relocation or, more intuitively, the *teleportation* of dominated particles to the Pareto front. This not only ensures the global Pareto optimality of the solutions but also significantly bolsters performance.

- We propose the *Particle-WFR* method, evolving a population of randomly-initialized *particles* by a combination of overdamped Langevin and stochastic birth-death dynamics towards the Pareto front.
- Theoretical grounding of our method is achieved by formulating it as a Wasserstein-Fisher-Rao gradient flow in the space of probability measures, providing provable convergence guarantees.
- Extensive experiments are conducted on both synthetic and real-world datasets, and the results demonstrate the superiority of our method over the state-of-the-art methods in addressing challenging tasks with complicated Pareto fronts.

2 PRELIMINARIES

This section defines the multi-objective optimization problem and introduces fundamental concepts and notations. A brief overview of gradient flows in the space of probability measures is also provided.

2.1 Pareto Optimality

In MOO, we consider the following problem of minimizing m objective functions simultaneously:

$$\min_{\mathbf{x}\in\mathcal{D}} \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \cdots, f_m(\mathbf{x})), \tag{1}$$

where $\mathcal{D} \subseteq \mathbb{R}^d$ is the feasible region, and $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [m] = \{1, \ldots, m\}$ are objective functions. Unlike single-objective optimization problems, we need the following notion of *Pareto optimality* to determine the superiority of a solution:

Definition 1 (Pareto optimality)**.** *A solution* $\mathbf{x}^*$ *is said to be* Pareto optimal *if there does not exist another solution* $\mathbf{x}'$ *such that* $f_i(\mathbf{x}') \leq f_i(\mathbf{x}^*)$ *for all* $i \in [m]$ *and* $f_j(\mathbf{x}') < f_j(\mathbf{x}^*)$ *for at least one* $j \in [m]$*. A solution* $\mathbf{x}^*$ *is said to be* locally Pareto optimal *if there exists a neighborhood* $\mathcal{N}(\mathbf{x}^*)$ *of* $\mathbf{x}^*$ *such that* $\mathbf{x}^*$ *is Pareto optimal in* $\mathcal{N}(\mathbf{x}^*)$*.*

In real applications, we are often interested in the set of Pareto optimal solutions, called the *Pareto front*, denoted by $\mathcal{P}$. Our goal is to find a set of solutions on the Pareto front, which should favorably be diversified, explicitly showcasing the characteristics of the problem for the decision maker to make the final choice (Tamaki et al., 1996). However, the Pareto front may exhibit complicated geometries that are disconnected or highly non-convex (Kulkarni et al., 2022), making it very challenging to compute. Singularities may arise even with only two quadratic objective functions (Sheftel et al., 2013).

2.2 Wasserstein-Fisher-Rao Gradient Flow

In many sampling problems, one would like to design an evolution of probability measures ρ_t that converges to a target distribution ρ^* as $t \to \infty$. One of the most intuitive and powerful tools for this purpose is the gradient flow. Specifically, a gradient flow represents a continuous-time dynamical system guiding ρ_t towards the target distribution ρ^*, recognized as the minimizer of a certain energy functional $\mathcal{E}[\rho]$. Generally, the introduction of different Riemannian metrics yields various gradient flows, each defining unique geodesics within the space of probability measures.

Consider two probability measures ρ_0 and ρ_1, the *Wasserstein* distance is defined as

$$d_{\mathsf{W}}(\rho_0, \rho_1) = \inf_{\pi\in\Pi(\rho_0,\rho_1)} \int_{\mathbb{R}^d\times\mathbb{R}^d} \|\mathbf{x}-\mathbf{y}\|^2 \pi(\mathrm{d}\mathbf{x}, \mathrm{d}\mathbf{y}), \tag{2}$$

where $\Pi(\rho_0, \rho_1)$ denotes the set of all joint probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ with marginals ρ_0 and ρ_1. The Benamou-Brenier theorem (Benamou and Brenier, 2000) provides an insightful geodesic interpretation for the Wasserstein metric:

$$\begin{aligned} &d_{\mathsf{W}}(\rho_0, \rho_1) \\ &= \inf\left\{ \int_0^1 \int \|\mathbf{v}_t\|^2 \mathrm{d}\rho_t \mathrm{d}t \,\middle|\, \partial_t\rho_t = -\nabla\cdot(\rho_t\mathbf{v}_t) \right\}, \end{aligned}$$

and the corresponding *Wasserstein gradient flow* of the energy $\mathcal{E}[\rho]$ is

$$\partial_t\rho_t = \nabla\cdot(\rho_t\nabla\delta_\rho\mathcal{E}[\rho_t]), \tag{3}$$

where $\delta_\rho\mathcal{E}[\rho]$ denotes the Fréchet derivative of $\mathcal{E}[\rho]$ w.r.t. ρ. One notable relevant result is that when $\mathcal{E}[\rho]$ is selected as the Kullback-Leibler divergence between ρ and ρ^*, the resulting Wasserstein gradient flow is the overdamped Langevin dynamics (Jordan et al., 1998).

In parallel, the *Fisher-Rao* metric is another important metric in the space of probability measures, resonating with concepts including the Fisher information and the Hellinger distance familiar to statisticians. It also has a geodesic interpretation:

$$d_{\mathsf{FR}}(\rho_0, \rho_1) = \inf\left\{ \int_0^1 \int \widetilde{\beta}_t^2 \mathrm{d}\rho_t \mathrm{d}t \,\middle|\, \partial_t\rho_t = \rho_t\widetilde{\beta}_t \right\},$$

where $\widetilde{\cdot}$ is a shorthand notation for $\cdot - \mathbb{E}_{\rho_t}[\cdot]$. The corresponding *Fisher-Rao gradient flow* is given by

$$\partial_t\rho_t = -\rho_t\widetilde{\delta_\rho\mathcal{E}[\rho_t]}. \tag{4}$$

Intuitively, as the Wasserstein gradient flow derives from the optimal transport problem (2), it redistributes probability densities by transporting them along paths directed by the Kantorovich potential $\delta_\rho \mathcal{E}[\rho]$. On the other hand, the Fisher-Rao gradient flow teleports mass, *i.e.* locally reshape probability densities, according to the deviation of $\delta_\rho \mathcal{E}[\rho]$ from its expectation.

Recent advances (Liero et al., 2016, 2018; Chizat et al., 2018) suggest the following *Wasserstein-Fisher-Rao* (WFR) distance, also known as the *spherical Hellinger-Kantorovich* distance:

$$d_{\mathsf{WFR}}(\rho_0, \rho_1) = \inf \left\{ \int_0^1 \int \|\mathbf{v}_t\|^2 + \widetilde{\beta}_t^2 \mathrm{d}\rho_t \mathrm{d}t \middle| \partial_t \rho_t = -\nabla \cdot (\rho_t \mathbf{v}_t) + \rho_t \widetilde{\beta}_t \right\},$$

as a natural combination of the Wasserstein and Fisher-Rao distances for the study of unbalanced optimal transport, *e.g.* accelerating Langevin sampling (Lu et al., 2019) and learning Gaussian mixture (Yan et al., 2023). It leads to the following WFR gradient flow of $\mathcal{E}[\rho]$:

$$\partial_t \rho_t = \nabla \cdot (\rho_t \nabla \delta_\rho \mathcal{E}[\rho_t]) - \rho_t \widetilde{\delta_\rho \mathcal{E}[\rho_t]}, \tag{5}$$

which offers promising implications, which we delve deeper into in the next section.

3 ALGORITHM

In this section, we present our proposed method for MOO. We first design the functional $\mathcal{E}[\rho]$ on which we employ the *Wasserstein-Fisher-Rao* (WFR) gradient flow and thereby encourage the diversity and global Pareto optimality. We then offer the theoretical formulation of our method as the WFR gradient flow. Finally, we discuss the interacting particle implementation of the WFR gradient flow, which is realized through the alternate application of the overdamped Langevin and birth-death dynamics.

3.1 Designing the Functional $\mathcal{E}[\rho]$

Consider the initial setup where the probability measure ρ_0 is arbitrarily initialized within the feasible region $\mathcal{D}$. As we implement the WFR gradient flow of $\mathcal{E}[\rho]$ to evolve ρ_t towards the target distribution ρ^*, we would like to design the functional $\mathcal{E}[\rho]$ such that its minimizer ρ^* has the following properties:

1. **Global Pareto optimality:** ρ^* should in close proximity to the Pareto front. Particularly, ρ^* should only cover *global* Pareto optimal solutions and exclude those only *local* Pareto optimal.
2. **Diversity:** To ensure a comprehensive representation, ρ^* should be diversified, spanning the entirety of the Pareto front. It should not be concentrated only on a subset of the Pareto front.

For each property above, we propose a corresponding term in the functional $\mathcal{E}[\rho]$, as explained below:

Objective Function Term. A straightforward strategy to force the minimizer ρ^* closer to the Pareto front is the weighted sum method, *i.e.* minimizing a linear combination of the objective functions $F(\mathbf{x}) = \sum_{i=1}^m \alpha_i f_i(\mathbf{x})$ where α_i, $i \in [m]$ are predetermined parameters. However, this method is susceptible to the varied scales and ranges of the objective functions, and the concavity of the Pareto front.

Addressing this challenge, the Multi-Gradient Descent Algorithm (MGDA) (Désidéri, 2012) relaxes the parameters α_i to be space-dependent determined by solving the following minimal norm optimization problem for each $\mathbf{x}$:

$$\min_{\boldsymbol{\alpha} \in \Delta_m} \left\| \sum_{i=1}^m \alpha_i(\mathbf{x}) \nabla f_i(\mathbf{x}) \right\|, \tag{6}$$

where Δ_m denotes the m-dimensional probability simplex. We will denote the optimal linear combination in (6) at each $\mathbf{x}$ as $\mathbf{g}^\dagger(\mathbf{x})$.

An argument by Lagrangian duality allows us to derive that $\mathbf{g}^\dagger(\mathbf{x})$ conforms to:

$$-\|\mathbf{g}^\dagger(\mathbf{x})\|^2 = \min_{\|\mathbf{g}\| \le 1} \min_{i \in [m]} -\mathbf{g}^\top \nabla f_i(\mathbf{x}).$$

Intuitively, $-\mathbf{g}^\dagger$ is the direction where objective functions see the most common decrease, quantified by $\|\mathbf{g}^\dagger\|^2$. As shown in Désidéri (2012, Theorem 2.2), a small magnitude of $\|\mathbf{g}^\dagger\|$ indicates misalignment among the objective function, *i.e.* $\mathbf{x}$ is close to local Pareto optimality.

Thus, we design the objective function term $\mathcal{F}_1[\rho]$ as

$$\mathcal{F}_1[\rho] = \int_{\mathcal{D}} \|\mathbf{g}^\dagger(\mathbf{x})\|^2 \rho(\mathrm{d}\mathbf{x}), \text{ with } \delta_\rho \mathcal{F}_1(\cdot) = \|\mathbf{g}^\dagger(\cdot)\|^2,$$

pushing ρ_t towards the Pareto front.

Dominance Potential Term. To ensure the global Pareto optimality of the minimizer ρ^*, we also design a *dominance potential* term $\mathcal{F}_2[\rho]$ of the following form:

$$\mathcal{F}_2[\rho] = \int_{\mathcal{D}} \left(\int_{\mathcal{P}} D(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y})) \mu_{\mathcal{P}}(\mathrm{d}\mathbf{y}) \right) \rho(\mathrm{d}\mathbf{x}), \tag{7}$$

where $\mu_{\mathcal{P}}$ is a predetermined measure on the Pareto front, and the kernel $D(\cdot, \cdot)$ satisfies

$$D(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y})) = \prod_{i=1}^m \max\{0, f_i(\mathbf{x}) - f_i(\mathbf{y})\}. \tag{8}$$

One should notice this potential term is still linear in ρ with the Fréchet derivative given by

$$\delta_\rho \mathcal{F}_2(\cdot) = \int_{\mathcal{P}} D(\mathbf{f}(\cdot), \mathbf{f}(\mathbf{y}))\mu_{\mathcal{P}}(\mathrm{d}\mathbf{y}). \tag{9}$$

As shown in Figure 1a, $D(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))$ is asymmetric and is non-zero if and only if $\mathbf{x}$ is dominated by $\mathbf{y}$, and $\int_{\mathcal{P}} D(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))\mu_{\mathcal{P}}(\mathrm{d}\mathbf{y})$ indicates how much $\mathbf{x}$ is dominated by the Pareto front $\mathcal{P}$. Ideally, the target distribution ρ^* is expected to have a small value of $\mathcal{F}_2[\rho^*]$. This term is crucial for implementing the birth-death process, which eliminates dominated particles, ensures global Pareto optimality, and accelerates the convergence of our method.

Entropy Term. We add a negative entropy term $-\mathcal{H}[\rho]$ to the functional $\mathcal{E}[\rho]$ to encourage the diversity of the minimizer ρ^*:

$$-\mathcal{H}[\rho] = \int_{\mathcal{D}} \rho(\mathbf{x}) \log \rho(\mathbf{x}) \mathrm{d}\mathbf{x},$$

and its Fréchet derivative is given by $\delta_\rho(-\mathcal{H}[\rho]) = \log \rho(\mathbf{x}) + 1$. As we will show later, this term effectively injects stochasticity into the evolution of ρ_t, thus also encouraging the exploration of the entire Pareto front.

Repulsive Potential Term. To further encourage the diversity of the minimizer ρ^* explicitly, we also design two-body *repulsive potential* term $\mathcal{G}[\rho]$ of the following form:

$$\mathcal{G}[\rho] = \frac{1}{2} \int_{\mathcal{D}\times\mathcal{D}} \rho(\mathrm{d}\mathbf{x}) R(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y})) \rho(\mathrm{d}\mathbf{y}),$$

with the Fréchet derivative being

$$\delta_\rho \mathcal{G}[\rho] = \int_{\mathcal{D}} R(\mathbf{f}(\cdot), \mathbf{f}(\mathbf{y}))\rho(\mathrm{d}\mathbf{y}).$$

Specifically, the repulsive kernel $\mathbb{R}(\cdot, \cdot)$ can adopt various forms depending on the requirements, including the Gaussian potential $R(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2/\sigma^2)$ or the Coulomb potential $R(\mathbf{x}, \mathbf{y}) = 1/\|\mathbf{x} - \mathbf{y}\|$. As illustrated in Figure 1b, this term pushes particles in the mass ρ away from each other, contributing to enhancing the dispersion of the minimizer ρ^*.

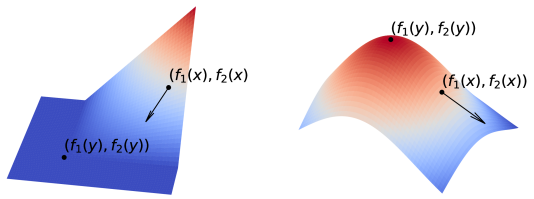
Combining the above terms, we define the functional $\mathcal{E}[\rho]$ as

$$\mathcal{E}[\rho] = \mathcal{F}[\rho] + \beta\mathcal{G}[\rho] - \gamma\mathcal{H}[\rho], \tag{10}$$

where $\mathcal{F}[\rho] = \alpha_1\mathcal{F}_1[\rho] + \alpha_2\mathcal{F}_2[\rho]$, with α_1, α_2, β, and γ being hyperparameters.

3.2 Theoretical Analysis

As we are implementing the WFR gradient flow of $\mathcal{E}[\rho]$, we have the following overall convergence guarantee:



(a) $D(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))$ (b) $R(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))$

Figure 1: Illustration of structural potential terms. This visualization explains the role of the dominance potential $\mathcal{F}_2$ and the repulsive potential $\mathcal{G}$ in a setting with two objective functions ($m = 2$). (a) Suppose $\mathbf{y}$ is from $\mu_{\mathcal{P}}(\cdot)$ with corresponding objective function values $\mathbf{f}(\mathbf{y}) = (f_1(\mathbf{y}), f_2(\mathbf{y}))$. When a point $\mathbf{x}$ is introduced, the dominance kernel $D(\cdot, \mathbf{f}(\mathbf{y}))$ acts to shift $\mathbf{x}$ out of the region dominated by $\mathbf{y}$. (b) For two samples $\mathbf{x}$ and $\mathbf{y}$ the repulsive kernel $R(\cdot, \mathbf{f}(\mathbf{y}))$ repels the objective function values of $\mathbf{x}$ away from those of $\mathbf{y}$.

Theorem 1. *Let ρ_t follow the WFR gradient flow of $\mathcal{E}[\rho]$ (5), then the following decay of the functional value $\mathcal{E}[\rho_t]$ holds:*

$$\partial_t \mathcal{E}[\rho_t] = -\int_{\mathcal{D}} \rho_t \left\|\nabla \delta_\rho \mathcal{E}[\rho_t]\right\|^2 + \rho_t \widetilde{\delta_\rho \mathcal{E}[\rho_t]}^2 \, \mathrm{d}\mathbf{x} \leq 0. \tag{11}$$

Furthermore, if $\beta > 0$ or $\gamma > 0$, the density ρ_t converges to the unique *minimizer ρ^* of $\mathcal{E}[\rho]$, as $t \to \infty$.*

For the special case where the diversity of the mass is only fostered by the entropy $\mathcal{H}[\rho]$, we have the following exponential convergence guarantee:

Theorem 2. *Let ρ_t follow the WFR gradient flow of $\mathcal{E}[\rho]$ in (5) with $\beta = 0$. The unique minimizer ρ^* has the following explicit Gibbs-type expression:*

$$\rho^* \propto \exp\left(-\frac{\alpha_1\|\mathbf{g}^\dagger\|^2 + \alpha_2 \int_{\mathcal{P}} D(\mathbf{f}(\cdot), \mathbf{f}(\mathbf{y}))\mu_{\mathcal{P}}(\mathrm{d}\mathbf{y})}{\gamma}\right). \tag{12}$$

Assume the initialization satisfies $\inf_{\mathbf{x}\in\mathcal{D}} \rho_0(\mathbf{x})/\rho^*(\mathbf{x}) \geq e^{-M}$, then the following exponential convergence holds:

$$\mathrm{KL}(\rho_t|\rho^*) \leq Me^{-\gamma t} + e^{-\gamma t + Me^{-\gamma t}} \mathrm{KL}(\rho_0|\rho^*). \tag{13}$$

Proofs are deferred to Appendix B.

Remark 1. *One should notice that the exponential convergence discussed above does not require the strong*

*convexity of the functional $\mathcal{E}[\rho]$, which is often necessary for Wasserstein gradient flows (*e.g. *via the log-Sobolev inequality (Villani, 2021)). The exclusion of $\mathcal{G}[\rho]$ in Theorem 2 is due to technicalities behind the nonlinearity of $\mathcal{E}[\rho]$ and the absence of a closed-form expression of the minimizer ρ^*, which we leave for future work.*

3.3 Interacting Particle Method

Since probability densities ρ_t as infinite-dimensional objects are generally difficult to keep track of or optimize, we propose to approximate ρ_t by a set of N *interacting particles* $\{\mathbf{x}_k\}_{k=1}^n$ as

$$\rho_t \approx \frac{1}{N}\sum_{k=1}^{n}\delta(\mathbf{x}-\mathbf{x}_k),$$

in which $\delta(\cdot)$ denotes the Dirac delta function and each particle $\mathbf{x}_k$ evolves with the time t. This method has been widely used in computational fluid dynamics (Koshizuka et al., 2018). We also employ a straightforward time discretization scheme with a small time step τ and use the notation $\mathbf{x}_k^{(\ell)} = \mathbf{x}_k(\ell\tau)$, for $k \in [m]$ and $\ell \geq 0$.

As studied by Gallouët and Monsaingeon (2017), WFR gradient flow can be approximated by the splitting scheme, *i.e.*, alternatively updating with (a) the gradient flow of the Wasserstein metric, and (b) that of the Fisher-Rao metric. Next, we will show that these two updates can be implemented with the overdamped Langevin and birth-death dynamics, respectively.

Overdamped Langevin Dynamics. Plugging (10) into the Wasserstein gradient flow (3), we obtain the following governing equation of ρ_t:

$$\partial_t\rho_t = \nabla\cdot\left(\rho_t\nabla\left(\delta_\rho\mathcal{F}+\delta_\rho\mathcal{G}[\rho_t]\right)\right)+\gamma\Delta\rho_t,$$

which is exactly the Fokker-Planck equation of the following stochastic differential equation (SDE):

$$\mathrm{d}\mathbf{x}_t = -\nabla\left(\delta_\rho\mathcal{F}+\delta_\rho\mathcal{G}[\rho_t]\right)\mathrm{d}t+\sqrt{2\gamma}\mathrm{d}\mathbf{w}_t,$$

where $\mathbf{w}_t$ is the standard Brownian motion. Therefore, the update in this step can be easily approximated by the *overdamped Langevin dynamics* of the particles for each $k \in [N]$:

$$\mathbf{x}_k^{(\ell+1/2)} = \mathbf{x}_k^{(\ell)} - \frac{\tau}{2}\nabla\left(\delta_\rho\mathcal{F}+\delta_\rho\mathcal{G}[\rho_t]\right)+\sqrt{\gamma\tau}\varepsilon_k^{(\ell)}, \quad (14)$$

where $\varepsilon_k^{(\ell)}$ are sampled independently from a standard Gaussian. As mentioned before, the stochasticity is introduced by the entropy term $\mathcal{H}[\rho]$.

Birth-death Dynamics. To simulate the Fisher-Rao gradient flow (4), we draw inspiration from the birth-death process in the queueing theory and the Gillespie algorithm for biochemical system simulation (Gillespie, 2007). Specifically, we reformulate (4) as

$$\partial_t\log\rho_t = -\widetilde{\delta_\rho\mathcal{E}[\rho_t]} := -\Lambda_t, \quad (15)$$

where Λ_t is the *instantaneous birth-death rate* function. Using the forward Euler scheme, (15) can be approximated by $\rho_{t+\tau/2} \approx \rho_t\exp(-\Lambda_t\tau/2)$, *i.e.* ρ_t should increase by $\exp(-\Lambda_t\tau/2)-1$ if $\Lambda_t<0$, and if $\Lambda_t>0$, it should decrease by $1-\exp(-\Lambda_t\tau/2)$.

In the context of our interacting particle method, we compute the instantaneous birth-death rate function $\Lambda_{(\ell+1/2)\tau}$ for each particle $\mathbf{x}_k^{(\ell+1/2)}$. This is done by approximating the expectation as

$$\begin{aligned}\Lambda_{(\ell+1/2)\tau}(\mathbf{x}_k^{(\ell+1/2)}) &= \delta_\rho\mathcal{E}[\rho_t](\mathbf{x}_k^{(\ell+1/2)}) - \mathbb{E}_{\rho_t}\left[\delta_\rho\mathcal{E}[\rho_t]\right]\\ &\approx \delta_\rho\mathcal{E}[\rho_t](\mathbf{x}_k^{(\ell+1/2)}) - \frac{1}{N}\sum_{k'=1}^{N}\delta_\rho\mathcal{E}[\rho_t](\mathbf{x}_{k'}^{(\ell+1/2)}).\end{aligned} \quad (16)$$

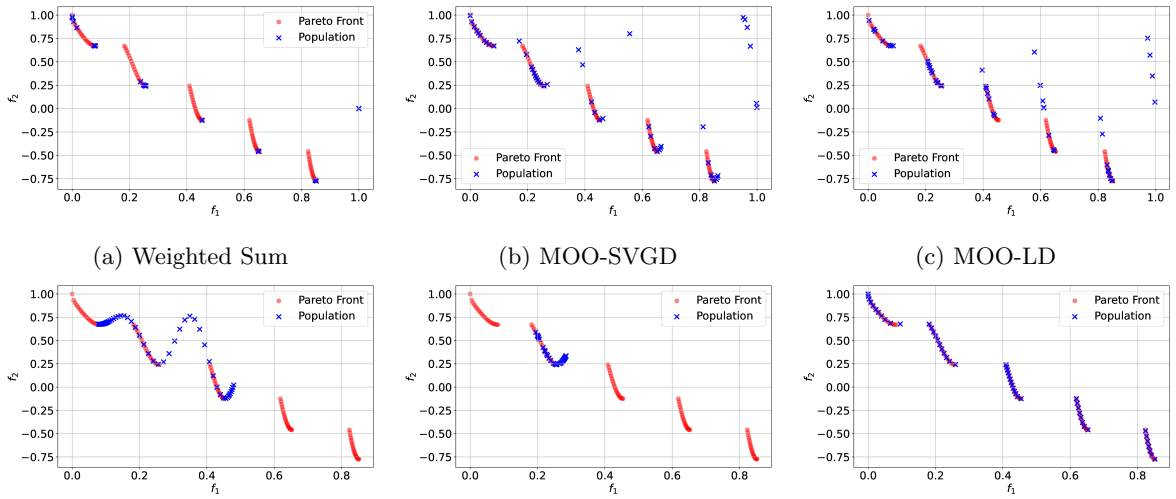
Then for each particle $\mathbf{x}_k^{(\ell+1/2)}$, depending on the sign of $\Lambda_{(\ell+1/2)\tau}$, we either duplicate it with probability $\exp(-\Lambda_{(\ell+1/2)\tau}(\mathbf{x}_k^{(\ell+1/2)})\tau/2)-1$ or remove it with probability $1-\exp(-\Lambda_{(\ell+1/2)\tau}(\mathbf{x}_k^{(\ell+1/2)})\tau/2)$. To compensate for the change in the total mass, we also randomly remove (or duplicate) a particle from the entire population whenever a duplication (or removal) occurs. As demonstrated in Proposition 5.1 of Lu et al. (2019), the above birth-death implementation converges to the original Fisher-Rao gradient flow (4) when $\tau\to 0$ and the number of particles $N\to\infty$.

In conclusion, we summarize our proposed interacting particle method in Algorithm 1 and refer to *Particle-WFR* in below. Due to space limit, we refer to Appendix C for additional implementation details.

Remark 2. *While most existing methods adopt the MGDA gradient $\mathbf{g}^\dagger(\mathbf{x})$ for updates, $\|\mathbf{g}^\dagger(\mathbf{x})\|=0$ only indicates local Pareto optimality, not a global guarantee. In our method, local Pareto optimal points are eliminated from the population by the birth-death dynamics acting on the dominance potential $\mathcal{F}_2[\rho]$ and replaced by other particles with lower dominance potential, thus ensuring global Pareto optimality.*

4 EXPERIMENT RESULTS

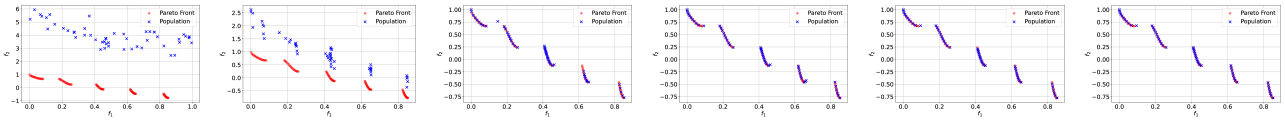
In this section, we conduct experiments on both synthetic and real-world datasets to evaluate the performance of our method. We compare our method with several recent state-of-the-art gradient-based methods, including PHN-LS and PHN-EPO (Navon et al.,



(a) Weighted Sum (b) MOO-SVGD (c) MOO-LD

(d) COSMOS (e) GMOOAR-HV (f) **Particle-WFR**

Figure 2: Performance comparison of different methods on the ZDT3 problem. The Pareto front is shown in red, and the solutions found by different methods are shown in blue. Our method (Particle-WFT) perfectly captures the complicated geometry of the Pareto front.



(a) Epoch 0 (b) Epoch 1000 (c) Epoch 2000 (d) Epoch 3000 (e) Epoch 4000 (f) Epoch 5000

Figure 3: Evolution of the particle population by Particle-WFR on the ZDT3 problem. The Pareto front is shown in red, and the current population is shown in blue.

2020), COSMOS (Ruchte and Grabocka, 2021b), MOO-SVGD and MOO-LD[2] (Liu et al., 2021), and GMOOAR-HV and GMOOAR-U (Chen and Kwok, 2022).

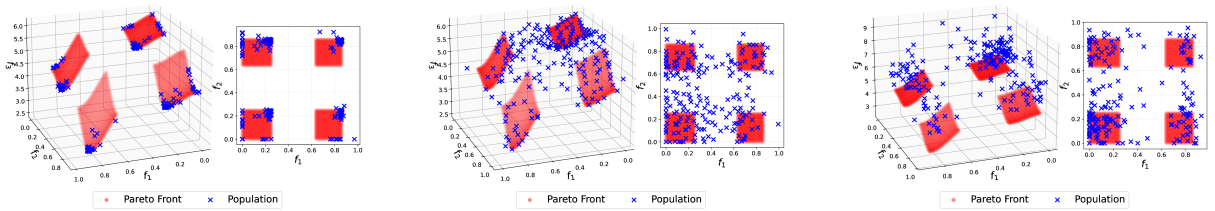
4.1 ZDT3 Problem

We first consider the ZDT3 problem (Zitzler et al., 2000), which has also been studied by Custódio et al. (2011); Liu et al. (2021). The ZDT3 problem is a 30-dimensional two-objective optimization problem ($d = 30$, $m = 2$), with the closed-form formula in Appendix D.1. Unlike ZDT1 and ZDT2 problems with continuous smooth Pareto fronts that can be easily handled Using the naive weighted sum method (Boyd and Vandenberghe, 2004) or preference vectors-based penalty methods (Lin et al., 2019), the ZDT3 problem presents a non-convex consisting of five disconnected segments. This complicated geometry makes it particularly challenging for gradient-based MOO methods. Due to the page limit, the experiments and discussions on the ZDT1 and ZDT2 problems are provided in Appendix D.1.1 and D.1.2, respectively.

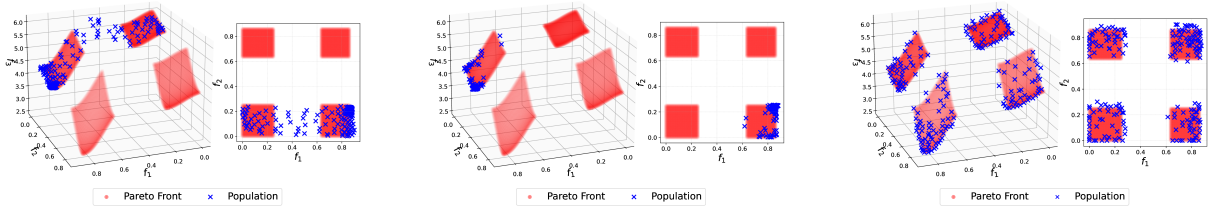
Figure 2 compares the performance of our method with five methods: the naive weighted sum method, COSMOS, MOO-SVGD, and GMOOAR-HV[3]. For a fair comparison, we use the same number of particles (or equivalently, uniformly distributed preference vectors or test rays) $N = 50$ and run all methods over 5000 iterations (or equivalently, epochs). As observed, the weighted sum method only identifies the convex hull of the Pareto front, aligning with the theoretical analysis of Boyd and Vandenberghe (2004). Due to the inherent continuity of neural networks, the solutions obtained by COSMOS delineate only a small portion of the image manifold of $\mathbf{f}(\mathbf{x})$ and consequently interpolate different Pareto front segments. The optimization of preference vectors in GMOOAR-HV also fails in this case, resulting in limited coverage of the Pareto

[2]Our implementation of MOO-SVGD and MOO-LD made several necessary modifications to its open-source version. Additionally, we utilized a different number of epochs, leading to results that are slightly distinct from those presented in Liu et al. (2021).

[3]We are only comparing with GMOOAR-HV in the ZDT3 and DTLZ7 problem because the result of GMOOAR-U is similar for these two examples.



(a) Weighted Sum (b) MOO-SVGD (c) MOO-LD



(d) COSMOS (e) GMOOAR-HV (f) **Particle-WFR**

Figure 4: Performance comparison of different methods on the DTLZ7 problem. The Pareto front is shown in red, and the solutions found by different methods are shown in blue. Alongside the 3D visualization, a bird's-eye perspective is also provided for each problem. Our method (Particle-WFR) achieves the best coverage of the Pareto front across all methods.

front. Both MOO-SVGD and MOO-LD achieve good coverage of the Pareto front but fall short of discerning and eliminating local Pareto optimal solutions. In contrast, our method outperforms all others, achieving a comprehensive and accurate coverage of the Pareto front.

We also showcase the evolution of the particle population by our method Particle-WFR in Figure 3. The particles are initially dispersed within the feasible region $\mathcal{D}$. Driven by the objective function potential term $\mathcal{F}_1[\rho]$, the particles are gradually attracted to the Pareto front, during which the birth-death dynamics systematically purge any particles that fall into local Pareto optimal regions. Roughly after epoch 2000, most of the particles are concentrated on the Pareto front, and the subsequent epochs focus on fine-tuning until a balance between the repulsive potential term $\mathcal{G}[\rho]$ and the dominance potential term $\mathcal{F}_2[\rho]$ is reached, ensuring optimal coverage of the Pareto front.

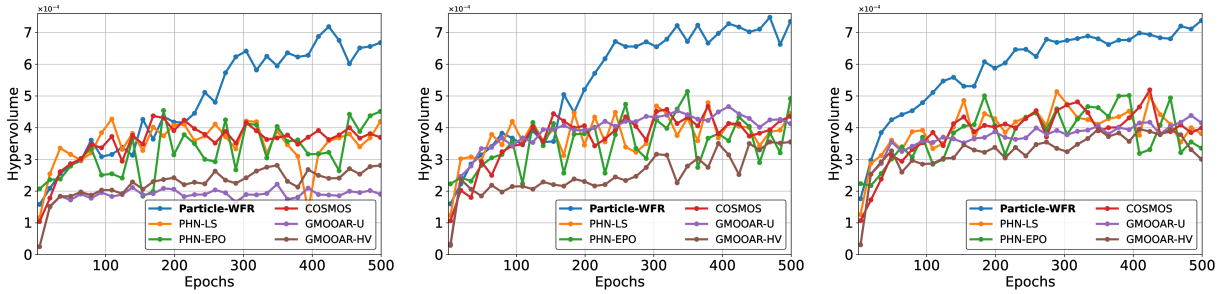
4.2 DTLZ7 Problem

Moving to three-objective optimization, we consider the DTLZ7 problem (Deb et al., 2002b), which is another 30-dimensional optimization problem with the closed-form description in Appendix D.2. As highlighted by Li et al. (2013), the DTLZ7 problem features a mixed, disconnected, and multimodal Pareto front, making it one of the most challenging problems within the DTLZ test suite when compared with the three-objective MaF1 and DTLZ2 problems used in Liu et al. (2021); Chen and Kwok (2022).

The solutions obtained by the naive weighted sum method, MOO-SVGD, MOO-LD, COSMOS, GMOOAR-HV, and Particle-WFR, are shown in Figure 4. We fix the number of particles $N = 200$ and the number of iterations as 3000. Similar phenomena as in the ZDT3 problem are observed: the solutions obtained by the weighted sum method cluster around the corners of the Pareto front. In contrast, the solutions obtained by COSMOS and GMOOAR-HV only span a limited portion of the Pareto front. Although MOO-SVGD can explore the entire Pareto front, a majority of the solutions end up in the local Pareto optimal regions. Relying on the noise for exploration, MOO-LD fails to balance the diversity and the convergence of the particle population, leading to a sub-optimal coverage of the Pareto front. In contrast, our method, Particle-WFR, achieves the best coverage of the Pareto front across all methods.

4.3 MSLR-WEB10K Dataset

In this example, We test MOO methods on the learning-to-rank (LTR) task (Dai et al., 2011; Hu and Li, 2018; Carmel et al., 2020; Mahapatra et al.,



(a) $N = 8$ (b) $N = 12$ (c) $N = 16$

Figure 5: Performance comparison of different methods on the MSLR-WEB30K dataset. Our method achieves the best HV value on test NDCG@10 and performance improves as particle count N increases from 8 to 16.

2023a,b). In LTR tasks (Liu et al., 2009), we deal with a collection of *query groups* $\Psi = \{\Psi^{(p)}\}_{p=1}^{|\Psi|}$, where each query group $\Psi^{(p)}$ consists of $n^{(p)}$ items. These items are characterized by a feature vector $\mathbf{x}_j^{(p)} \in \mathbb{R}^{d_f}$, generated from upstream tasks, and an associated relevance label $y_j^{(p)}$. The goal is to derive an ordering $\pi^{(p)}$ for the items in each query group $\Psi^{(p)}$ given the feature vectors $\mathbf{x}_j^{(p)}$, optimizing the utility $u(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})$ of the ordered list. The Normalized Discounted Cumulative Gain (NDCG) (Wang et al., 2013) is a widely adopted ranking metric.

Following the current LTR techniques, we employ a neural network f_θ, with θ denoting the parameters, that accepts the feature vector as input and produces a score, based on which we sort the items in each query group and obtain the ordering $\pi^{(p)}$. The neural network is trained using the empirical loss of the following form:

$$\mathcal{L}(\theta; \Psi) = \frac{1}{|\Psi|} \sum_{p=1}^{|\Psi|} \ell\left(\{f_\theta(\mathbf{x}_j^{(p)})\}_{j=1}^{n^{(p)}}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}\right), \quad (17)$$

where $\ell(\cdot, \cdot)$ is the query group-wise loss function. As the differentiable surrogate for the non-differentiable NDCG metric, we adopt the *Cross-Entropy loss* (Cao et al., 2007) for ℓ, which is one of the most robust choices as supported by Qin et al. (2021). Further details on the settings, metrics, and losses can be found in Appendix D.3.1.

In real-world scenarios, an item might have multiple labels of interest, denoted as $y_j^{(p),i}$. Each of these labels signifies the relevance of the corresponding item concerning the i-th ranking objective for $i \in [m]$. This gives rise to an MOO problem w.r.t. the neural network parameters θ and the m loss functions, $\mathcal{L}_i(\theta; \Psi)$ as the objective functions, obtained by substituting $\{y_j^{(p)}\}_{j=1}^{n^{(p)}}$ in (17) with the respective label $\{y_j^{(p),i}\}_{j=1}^{n^{(p)}}$.

Remark 3. *Unlike various studies from Sener and Koltun (2018) that integrate multi-task learning into their experimental designs, we use the multi-objective LTR task for benchmarking, as suggested by Ruchte and Grabocka (2021a).*

We conduct experiments on the Microsoft Learning-to-Rank Web Search (MSLR-WEB10K) dataset (Qin and Liu, 2013). The MSLR-WEB10K dataset consists of 10,000 query groups ($|\Psi| = 10^4$) and each item is associated with 136 features and a relevance label. Following the practice of Mahapatra et al. (2023a), we treat the first 131 features as the input ($d_f = 131$) and combine the last 5 features, *viz.* Query-URL Click Count, URL Dwell Time, Quality Score 1, Quality Score 2, with the relevance label, as six different ranking objectives ($m = 6$). Our Particle-WFR method is implemented with Distributed Data Parallel (DDP) in PyTorch (Paszke et al., 2019) with extensive scalability, and further details are provided in Appendix D.3.2.

In Figure 5, we present the learning curves of the hypervolume (HV)[4] values of the testing NDCG@10 across several benchmark methods: PHN-LS, PHN-EPO, COSMOS, GMOOAR-HV, and GMOOAR-U on the MSLR-WEB10K dataset over 500 epochs of training in the neural network setting[5]. As listed in Table 1, hypernetwork-based methods struggle to cover the Pareto front. This might be partially attributed to their dependence on the structure of the loss space and their intrinsic sensitivity, resulting in less desirable generalization. Our Particle-WFR method outperforms the other methods, achieving the highest HV value in three instances where the number of particles $N = 8, 12, 16$. Notably, the HV value of our method

[4]Hypervolume is a quality indicator for assessing the solutions in MOO (*cf.* Appendix D.3.3).

[5]We do not include a comparison with MOO-SVGD and MOO-LD, as their performance in neural network contexts has been found limited (Chen and Kwok, 2022) and our method consistently outperforms these other methods on synthetic datasets.

Algorithm 1: Particle-WFR Method for MOO

Data: Objective functions f_i, $i \in [m]$, feasible region $\mathcal{D}$, time step τ, number of particles N, number of iterations T

Result: Evolved population $\{\mathbf{x}_k^{(T)}\}_{k=1}^N$

Randomly initialize $\{\mathbf{x}_k^{(0)}\}_{k=1}^N \subset \mathcal{D}$;
for $\ell = 1$ **to** T **do**
  `// Overdamped Langevin dynamics`
  for $k = 1$ **to** N **do**
    Update $\mathbf{x}_k^{(\ell+1/2)}$ from $\mathbf{x}_k^{(\ell)}$ by (14);
  end
  `// Birth-death dynamics`
  for $k = 1$ **to** N **do**
    $\mathbf{x}_k^{(\ell+1)} \leftarrow \mathbf{x}_k^{(\ell+1/2)}$;
    Compute $\lambda = \Lambda_{(\ell+1/2)\tau}(\mathbf{x}_k^{(\ell+1/2)})$ by (16);
    $k' \sim \mathrm{Unif}([N])$, $\eta \sim \mathrm{Unif}([0,1])$;
    if $\eta < |1 - \exp(-\lambda\tau/2)|$ **then**
      if $\lambda < 0$ **then**
        $\mathbf{x}_{k'}^{(\ell+1)} \leftarrow \mathbf{x}_k^{(\ell+1/2)}$;
      else
        $\mathbf{x}_k^{(\ell+1)} \leftarrow \mathbf{x}_{k'}^{(\ell+1/2)}$;
      end
    end
  end
end

has surpassed all other methods after only 100 epochs. This demonstrates the effectiveness and efficacy of our method in solving the MOO problem within the neural network-driven LTR context.

5 CONCLUSION

This paper proposes a novel interacting particle method based on the Wasserstein-Fisher-Rao gradient flow for solving the MOO problem. Our method enjoys interpretable and intuitive physical meanings with provable convergence guarantees. We implement the Wasserstein-Fisher-Rao gradient flow by the splitting scheme, where the Wasserstein gradient flow is approximated by the overdamped Langevin dynamics and the Fisher-Rao gradient flow by the birth-death dynamics. We compare our proposed method with several recent state-of-the-art methods on challenging datasets. The results show that our method is favorable when dealing with complicated Pareto fronts.

Acknowledgements

This work was conducted during Yinuo Ren's internship as an Applied Scientist at Amazon Search. We thank the Amazon Search team for their support and feedback. We also thank the anonymous reviewers for their insightful comments and suggestions.

References

Belakaria, S., Deshwal, A., Jayakodi, N. K., and Doppa, J. R. (2020). Uncertainty-aware search framework for multi-objective bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10044–10052.

Benamou, J.-D. and Brenier, Y. (2000). A computational fluid mechanics solution to the monge-kantorovich mass transfer problem. *Numerische Mathematik*, 84(3):375–393.

Boyd, S. P. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. (2007). Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Carmel, D., Haramaty, E., Lazerson, A., and Lewin-Eytan, L. (2020). Multi-objective ranking optimization for product search using stochastic label aggregation. In *Proceedings of The Web Conference 2020*, pages 373–383.

Chen, W. and Kwok, J. (2022). Multi-objective deep learning with adaptive reference vectors. *Advances in Neural Information Processing Systems*, 35:32723–32735.

Chizat, L., Peyré, G., Schmitzer, B., and Vialard, F.-X. (2018). An interpolating distance between optimal transport and fisher–rao metrics. *Foundations of Computational Mathematics*, 18:1–44.

Cui, Y., Geng, Z., Zhu, Q., and Han, Y. (2017). Multi-objective optimization methods and application in energy saving. *Energy*, 125:681–704.

Custódio, A. L., Madeira, J. A., Vaz, A. I. F., and Vicente, L. N. (2011). Direct multisearch for multiobjective optimization. *SIAM Journal on Optimization*, 21(3):1109–1140.

Dai, N., Shokouhi, M., and Davison, B. D. (2011). Multi-objective optimization in learning to rank. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pages 1241–1242.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002a). A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197.

Deb, K., Thiele, L., Laumanns, M., and Zitzler, E. (2002b). Scalable multi-objective optimization test

problems. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, volume 1, pages 825–830. IEEE.

Désidéri, J.-A. (2012). Multiple-gradient descent algorithm (mgda) for multiobjective optimization. *Comptes Rendus Mathematique*, 350(5-6):313–318.

Fonseca, C. M. and Fleming, P. J. (1995). An overview of evolutionary algorithms in multiobjective optimization. *Evolutionary computation*, 3(1):1–16.

Gallouët, T. O. and Monsaingeon, L. (2017). A jko splitting scheme for kantorovich–fisher–rao gradient flows. *SIAM Journal on Mathematical Analysis*, 49(2):1100–1130.

Gambier, A. and Badreddin, E. (2007). Multi-objective optimal control: An overview. In *2007 IEEE international conference on control applications*, pages 170–175. IEEE.

Gillespie, D. T. (2007). Stochastic simulation of chemical kinetics. *Annu. Rev. Phys. Chem.*, 58:35–55.

Hoang, L. P., Le, D. D., Tuan, T. A., and Thang, T. N. (2023). Improving pareto front learning via multi-sample hypernetworks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7875–7883.

Hu, J. and Li, P. (2018). Collaborative multi-objective ranking. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1363–1372.

Jordan, R., Kinderlehrer, D., and Otto, F. (1998). The variational formulation of the fokker–planck equation. *SIAM journal on mathematical analysis*, 29(1):1–17.

Konak, A., Coit, D. W., and Smith, A. E. (2006). Multi-objective optimization using genetic algorithms: A tutorial. *Reliability engineering & system safety*, 91(9):992–1007.

Konakovic Lukovic, M., Tian, Y., and Matusik, W. (2020). Diversity-guided multi-objective bayesian optimization with batch evaluations. *Advances in Neural Information Processing Systems*, 33:17708–17720.

Koshizuka, S., Shibata, K., Kondo, M., and Matsunaga, T. (2018). *Moving particle semi-implicit method: a meshfree particle method for fluid dynamics*. Academic Press.

Kulkarni, A., Kohns, M., Bortz, M., Küfer, K.-H., and Hasse, H. (2022). Regularities of pareto sets in low-dimensional practical multi-criteria optimisation problems: analysis, explanation, and exploitation. *Optimization and Engineering*, pages 1–22.

Laumanns, M. and Ocenasek, J. (2002). Bayesian optimization algorithms for multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 298–307. Springer.

Li, M., Yang, S., and Liu, X. (2013). Shift-based density estimation for pareto-based algorithms in many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 18(3):348–365.

Liero, M., Mielke, A., and Savaré, G. (2016). Optimal transport in competition with reaction: The hellinger–kantorovich distance and geodesic curves. *SIAM Journal on Mathematical Analysis*, 48(4):2869–2911.

Liero, M., Mielke, A., and Savaré, G. (2018). Optimal entropy-transport problems and a new hellinger–kantorovich distance between positive measures. *Inventiones mathematicae*, 211(3):969–1117.

Lin, X., Yang, Z., Zhang, Q., and Kwong, S. (2020). Controllable pareto multi-task learning. *arXiv preprint arXiv:2010.06313*.

Lin, X., Zhen, H.-L., Li, Z., Zhang, Q.-F., and Kwong, S. (2019). Pareto multi-task learning. *Advances in neural information processing systems*, 32.

Liu, Q. (2017). Stein variational gradient descent as gradient flow. *Advances in neural information processing systems*, 30.

Liu, Q. and Wang, D. (2016). Stein variational gradient descent: A general purpose bayesian inference algorithm. *Advances in neural information processing systems*, 29.

Liu, S. and Vicente, L. N. (2021). The stochastic multi-gradient algorithm for multi-objective optimization and its application to supervised machine learning. *Annals of Operations Research*, pages 1–30.

Liu, T.-Y. et al. (2009). Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, 3(3):225–331.

Liu, X., Tong, X., and Liu, Q. (2021). Profiling pareto front with multi-objective stein variational gradient descent. *Advances in Neural Information Processing Systems*, 34:14721–14733.

Lu, Y., Lu, J., and Nolen, J. (2019). Accelerating langevin sampling with birth-death. *arXiv preprint arXiv:1905.09863*.

Lu, Y., Slepčev, D., and Wang, L. (2023). Birth–death dynamics for sampling: global convergence, approximations and their asymptotics. *Nonlinearity*, 36(11):5731.

Mahapatra, D., Dong, C., Chen, Y., and Momma, M. (2023a). Multi-label learning to rank through multi-objective optimization. In *Proceedings of the 29th*

ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pages 4605–4616.

Mahapatra, D., Dong, C., and Momma, M. (2023b). Querywise fair learning to rank through multi-objective optimization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, page 1653–1664.

Mahapatra, D. and Rajan, V. (2020). Multi-task learning with user preferences: Gradient descent with controlled ascent in pareto optimization. In *International Conference on Machine Learning*, pages 6597–6607. PMLR.

Navon, A., Shamsian, A., Chechik, G., and Fetaya, E. (2020). Learning the pareto front with hypernetworks. *arXiv preprint arXiv:2010.04104*.

Parsopoulos, K. E. and Vrahatis, M. N. (2002). Particle swarm optimization method in multiobjective problems. In *Proceedings of the 2002 ACM symposium on Applied computing*, pages 603–607.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Qin, T. and Liu, T.-Y. (2013). Introducing letor 4.0 datasets. *arXiv preprint arXiv:1306.2597*.

Qin, Z., Yan, L., Zhuang, H., Tay, Y., Pasumarthi, R. K., Wang, X., Bendersky, M., and Najork, M. (2021). Are neural rankers still outperformed by gradient boosted decision trees? In *International Conference on Learning Representations (ICLR)*.

Reyes-Sierra, M., Coello, C. C., et al. (2006). Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *International journal of computational intelligence research*, 2(3):287–308.

Ruchte, M. and Grabocka, J. (2021a). Multi-task problems are not multi-objective. *arXiv preprint arXiv:2110.07301*.

Ruchte, M. and Grabocka, J. (2021b). Scalable pareto front approximation for deep multi-objective learning. In *2021 IEEE international conference on data mining (ICDM)*, pages 1306–1311. IEEE.

Santambrogio, F. (2017). {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7:87–154.

Sener, O. and Koltun, V. (2018). Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31.

Sheftel, H., Shoval, O., Mayo, A., and Alon, U. (2013). The geometry of the p areto front in biological phenotype space. *Ecology and evolution*, 3(6):1471–1483.

Tamaki, H., Kita, H., and Kobayashi, S. (1996). Multi-objective optimization by genetic algorithms: A review. In *Proceedings of IEEE international conference on evolutionary computation*, pages 517–522. IEEE.

Tapia, M. G. C. and Coello, C. A. C. (2007). Applications of multi-objective evolutionary algorithms in economics and finance: A survey. In *2007 IEEE congress on evolutionary computation*, pages 532–539. IEEE.

Tu, B., Gandy, A., Kantas, N., and Shafei, B. (2022). Joint entropy search for multi-objective bayesian optimization. *Advances in Neural Information Processing Systems*, 35:9922–9938.

Villani, C. (2021). *Topics in optimal transportation*, volume 58. American Mathematical Soc.

Wang, Y., Wang, L., Li, Y., He, D., and Liu, T.-Y. (2013). A theoretical analysis of ndcg type ranking measures. In *Conference on learning theory*, pages 25–54. PMLR.

Wright, S. J. (1997). *Primal-dual interior-point methods*. SIAM.

Yan, Y., Wang, K., and Rigollet, P. (2023). Learning gaussian mixtures using the wasserstein-fisher-rao gradient flow. *arXiv preprint arXiv:2301.01766*.

Zhang, Q. and Li, H. (2007). Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731.

Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P. N., and Zhang, Q. (2011). Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm and evolutionary computation*, 1(1):32–49.

Zitzler, E., Deb, K., and Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2):173–195.

Zitzler, E. and Künzli, S. (2004). Indicator-based selection in multiobjective search. In *International conference on parallel problem solving from nature*, pages 832–842. Springer.

A RELATED WORKS

Gradient-Free MOO Methods. In the previous decades, research in multi-objective optimization has been focused on the evolutionary algorithms and particle swarm methods (Tamaki et al., 1996; Deb et al., 2002a; Parsopoulos and Vrahatis, 2002; Konak et al., 2006; Reyes-Sierra et al., 2006; Zhang and Li, 2007). These methods are based on the idea of maintaining a population of solutions, which are iteratively improved by applying genetic operators such as mutation and crossover. Distributed computing techniques have also been explored in this context (Zhou et al., 2011). However, because these methods assume that gradient information is unavailable, they are often computationally expensive and do not scale well to high-dimensional problems. Another line of research in MOO is the Bayesian optimization (Laumanns and Ocenasek, 2002; Belakaria et al., 2020; Konakovic Lukovic et al., 2020; Tu et al., 2022). These methods are effective for small-scale black-box optimization problems but also suffer from the curse of dimensionality in machine learning tasks.

Gradient-Based Methods. In recent years, gradient-based methods have been proposed to solve the MOO problem. These methods are primarily designed and believed to effectively sacle up for high0dimensional machine learning applications. To profile the Pareto front, one of the most straightforward gradient-based approaches is to parametrize the Pareto front by preference vectors, *i.e.* the vector formed by the values of the objective functions on the Pareto front. The prototype of all preference vector-based methods is the *weighted sum method*, aka *Linear Scalarization (LS)* that uses a linear combination of weights and objective functions to find optimal solutions. However, the weighted sum method cannot handle the concavity of the Pareto front (Boyd and Vandenberghe, 2004). To address this issue, several methods are proposed to find local Pareto optimal points catering to predetermined preference vectors, including PF-SMG (Liu and Vicente, 2021), PMTL (Lin et al., 2019), and EPO (Mahapatra and Rajan, 2020). These methods often rely on selecting preference vectors, which may be difficult or even impossible for complicated Pareto fronts, resulting in sub-optimal solutions.

Recently, hypernetworks (Lin et al., 2020; Ruchte and Grabocka, 2021b; Chen and Kwok, 2022; Hoang et al., 2023) have been proposed to learn the Pareto front directly from data originated from PHN-LS and PHN-EPO (Navon et al., 2020), which generalizes the weighted sum method and EPO directly. Hypernetwork methods aim to design neural networks that accept preference vectors as inputs and directly generate solutions on the Pareto front. However, as demonstrated in our experiments, these methods are generally not very robust when dealing with complicated Pareto fronts and challenging tasks.

Gradient Flows and Interacting Particle Method. Gradient flows have been studied widely as one of the most important techniques in the literature of optimal transport and sampling (Villani, 2021). Originated from (Jordan et al., 1998), Wasserstein gradient flow is one of the most renowned gradient flows (*cf.* Santambrogio (2017)). Stein Variational Gradient flow (SVGD) (Liu and Wang, 2016; Liu, 2017) can be viewed as the gradient flow with respect to a kernelized Wasserstein metric. MOO-SVGD (Liu et al., 2021) is a recent method that uses SVGD to solve the MOO problem. Wasserstein-Fisher-Rao metric and the corresponding gradient flow (Chizat et al., 2018; Liero et al., 2018, 2016) are recently proposed to study unbalanced optimal transport problems and have been applied to sampling by interacting particle methods (Lu et al., 2019; Yan et al., 2023).

Multi-Objective Learning-to-Rank. MOO finds extensive applications in Learning-to-Rank (LTR) because it naturally involves multiple, potentially conflicting ranking metrics, such as precision and recall, or multiple relevance labels, such as product quality and purchase likelihood in e-commerce product search. Unlike supervised learning problems, where each sample has a clearly defined target as a single categorical label or numerical value, LTR tasks aim to identify an optimal permutation within a large and discrete search space for each query-group. This optimization is usually conducted to maximize a linear additive ranking metric, such as Normalized Discount Cumulative Gain (NDCG) (Wang et al., 2013). As a parametrized ranking model always operates as a scoring function, generating numerical scores to rank documents within a query, all ranking metrics are non-differentiated with respect to the predicted scores. To reframe LTR as a supervised learning problem, various differentiable loss functions are introduced as alternatives to optimize ranking metrics; see Qin et al. (2021) and references therein.

In the context of Multi-Objective LTR, existing work can be categorized into two main approaches: label aggregation (Dai et al., 2011; Carmel et al., 2020) and loss aggregation (Hu and Li, 2018; Mahapatra et al., 2023a,b). In the former, labels assigned to the same document are combined into a single label, which is then

used as input for a ranking loss function. In the latter, the aim is to optimize aggregated ranking loss functions, each corresponding to a different objective. In our experiments, we adopt later approach.

B MISSING PROOFS

In this section, we provide the missing proofs in the theoretical analysis part (Section 3.2) of the main text.

Proof of Theorem 1. The decay of the functional $\mathcal{E}[\rho]$ (11) is due to the following calculation by plugging in (5):

$$\begin{aligned}\partial_t \mathcal{E}[\rho_t] &= \int_{\mathcal{D}} \delta_\rho \mathcal{E}[\rho_t] \partial_t \rho_t \mathrm{d}\mathbf{x} \\ &= \int_{\mathcal{D}} \delta_\rho \mathcal{E}[\rho_t] \nabla \cdot (\rho_t \nabla \delta_\rho \mathcal{E}[\rho_t]) - \rho_t \delta_\rho \mathcal{E}[\rho_t] \widetilde{\delta_\rho \mathcal{E}[\rho_t]} \mathrm{d}\mathbf{x} \\ &= -\int_{\mathcal{D}} \rho_t \left\| \nabla \delta_\rho \mathcal{E}[\rho_t] \right\|^2 + \rho_t \widetilde{\delta_\rho \mathcal{E}[\rho_t]}^2 \mathrm{d}\mathbf{x},\end{aligned}$$

where the last equality is due to the integration-by-parts and the fact that

$$\int_{\mathcal{D}} \rho_t \widetilde{\delta_\rho \mathcal{E}[\rho_t]} \mathrm{d}\mathbf{x} = \int_{\mathcal{D}} \rho_t \left(\delta_\rho \mathcal{E}[\rho_t] - \mathbb{E}_{\rho_t} \left[\delta_\rho \mathcal{E}[\rho_t] \right] \right) \mathrm{d}\mathbf{x} = 0.$$

Since ρ^* is the minimizer of $\mathcal{E}[\rho]$, we have the following optimality condition:

$$\nabla \delta_\rho \mathcal{E}[\rho^*] = 0, \ \textit{a.e.}, \text{ and } \widetilde{\delta_\rho \mathcal{E}[\rho^*]} = 0, \ \textit{a.e.},$$

which implies

$$\delta_\rho \mathcal{E}[\rho^*] = \alpha_1 \delta_\rho \mathcal{F}_1 + \alpha_2 \delta_\rho \mathcal{F}_2 + \beta \delta_\rho \mathcal{G}[\rho^*] - \gamma \delta_\rho \mathcal{H}[\rho^*] = C^*, \ \textit{a.e.}, \tag{18}$$

where $C^* = \mathbb{E}_{\rho^*} \left[\delta_\rho \mathcal{E}[\rho^*] \right]$.

Now suppose ρ' is another minimizer of $\mathcal{E}[\rho]$, a similar argument yield

$$\delta_\rho \mathcal{E}[\rho'] = \alpha_1 \delta_\rho \mathcal{F}_1 + \alpha_2 \delta_\rho \mathcal{F}_2 + \beta \delta_\rho \mathcal{G}[\rho'] - \gamma \delta_\rho \mathcal{H}[\rho'] \equiv C', \tag{19}$$

where $C' = \mathbb{E}_{\rho'} \left[\delta_\rho \mathcal{E}[\rho'] \right]$.

Subtracting the above two equations (18) and (19), we obtain

$$\begin{aligned}C' - C^* =& \beta \delta_\rho \mathcal{G}[\rho^*] - \beta \delta_\rho \mathcal{G}[\rho'] - \gamma \delta_\rho \mathcal{H}[\rho^*] + \gamma \delta_\rho \mathcal{H}[\rho'] \\ =& \beta \int_{\mathcal{D}} R(\mathbf{f}(\cdot), \mathbf{f}(\mathbf{y}))(\rho^* - \rho')(\mathrm{d}\mathbf{y}) + \gamma \log \rho^* - \gamma \log \rho'.\end{aligned}$$

Multiply both sides with $\rho^* - \rho'$ and integrate over $\mathcal{D}$, we have

$$\begin{aligned}0 =& \int_D (C' - C^*)(\rho^* - \rho')(\mathrm{d}\mathbf{x}) \\ =& \beta \int_{\mathcal{D}} (\rho^* - \rho')(\mathrm{d}\mathbf{x}) R(\mathbf{f}(\mathbf{x}), \mathbf{f}(\mathbf{y}))(\rho^* - \rho')(\mathrm{d}\mathbf{y}) + \gamma \int_D \log \frac{\rho^*}{\rho'}(\mathbf{y})(\rho^* - \rho')(\mathrm{d}\mathbf{y}),\end{aligned}$$

implying that $\rho^* = \rho'$ and therefore the minimizer ρ^* is unique. □

To prove Theorem 2, we need the following lemma:

Lemma 3 (Lu et al. (2023, Theorem 2.4 and Remark 2.6))**.** *Let ρ_0 and ρ^* be two probability measures absolutely continuous with respect to the Lebesgue measure and have the density functions $\rho_0(\mathbf{x})$ and $\rho^*(\mathbf{x})$, respectively. Suppose that the initial condition ρ_0 satisfies*

$$\inf_{\mathbf{x} \in \mathcal{D}} \frac{\rho_0(\mathbf{x})}{\rho^*(\mathbf{x})} \geq e^{-M}$$

for some constant M, we have for all $t > 0$, the Wasserstion-Fisher-Rao gradient flow ρ_t (5) with respect to the KL divergence $\mathrm{KL}(\rho_t|\rho^*)$ between ρ_t and ρ^* satisfies

$$\mathrm{KL}(\rho_t|\rho^*) \leq Me^{-t} + e^{-t+Me^{-t}}\mathrm{KL}(\rho_0|\rho^*).$$

Then, we are ready to prove Theorem 2.

Proof of Theorem 2. The constant C may change from line to line in this proof.

As we are considering the scenario where the repulsive potential term $\mathcal{G}[\rho]$ is turned off, *i.e.* $\beta = 0$, the functional $\mathcal{E}[\rho]$ can be simplified as

$$\mathcal{E}[\rho] = \alpha_1\mathcal{F}_1[\rho] + \alpha_2\mathcal{F}_2[\rho] - \gamma\mathcal{H}[\rho],$$

and the corresponding minimizer of the energy potential $\mathcal{E}[\rho]$ satisfies

$$\delta_\rho\mathcal{E}[\rho^*] = \alpha_1\delta_\rho\mathcal{F}_1 + \alpha_2\delta_\rho\mathcal{F}_2 - \gamma\delta_\rho\mathcal{H}[\rho^*] = C, \tag{20}$$

that is

$$\log\rho^* = -\frac{\alpha_1\delta_\rho\mathcal{F}_1 + \alpha_2\delta_\rho\mathcal{F}_2}{\gamma} + C,$$

and thus the minimizer ρ^* satisfies the Gibbs-type distribution as in (12):

$$\rho^* \propto \exp\left(-\frac{\alpha_1\delta_\rho\mathcal{F}_1 + \alpha_2\delta_\rho\mathcal{F}_2}{\gamma}\right) = \exp\left(-\frac{\alpha_1\|\mathbf{g}^\dagger\|^2 + \alpha_2\int_{\mathcal{P}} D(\mathbf{f}(\cdot),\mathbf{f}(\mathbf{y}))\mu_{\mathcal{P}}(\mathrm{d}\mathbf{y})}{\gamma}\right),$$

which is also the unique minimizer of $\mathcal{E}[\rho]$, as the result of Theorem 1.

Notice that the Fréchet derivative of the energy functional $\mathcal{E}[\rho]$ can be rewritten as:

$$\begin{aligned}\delta_\rho\mathcal{E}[\rho] &= \gamma\left(\frac{\alpha_1\delta_\rho\mathcal{F}_1 + \alpha_2\delta_\rho\mathcal{F}_2}{\gamma} - \delta_\rho\mathcal{H}[\rho]\right)\\ &= \gamma\left(-\log\rho^* + \log\rho\right) + C\\ &= \gamma\log\frac{\rho}{\rho^*} + C = \gamma\delta_\rho\mathrm{KL}(\rho|\rho^*) + C,\end{aligned} \tag{21}$$

we reparametrize the time with $\tau = t/\gamma$, and thus rewrite the Wasserstein-Fisher-Rao gradient flow (5) as

$$\begin{aligned}\partial_t\rho_\tau = \frac{1}{\gamma}\partial_\tau\rho_\tau &= \frac{1}{\gamma}\left[\nabla\cdot(\rho_\tau\nabla\delta_\rho\mathcal{E}[\rho_\tau]) - \rho_\tau\left(\delta_\rho\mathcal{E}[\rho_\tau] - \int_{\mathcal{D}}\rho_\tau\delta_\rho\mathcal{E}[\rho_\tau]\mathrm{d}\mathbf{x}\right)\right]\\ &= \nabla\cdot\left(\rho_\tau\nabla\log\frac{\rho_\tau}{\rho^*}\right) - \rho_\tau\left(\log\frac{\rho_\tau}{\rho^*} - \int_{\mathcal{D}}\rho_\tau\log\frac{\rho_\tau}{\rho^*}\mathrm{d}\mathbf{x}\right),\end{aligned}$$

that is the Wasserstein-Fisher-Rao gradient flow of the KL divergence between ρ_τ and ρ^*.

Together with the following assumption on the initial distribution in the theorem statement:

$$\inf_{\mathbf{x}\in\mathcal{D}}\frac{\rho_0(\mathbf{x})}{\rho^*(\mathbf{x})} \geq e^{-M},$$

we have by Lemma 3 that

$$\mathrm{KL}(\rho_\tau|\rho^*) \leq Me^{-t} + e^{-t+Me^{-t}}\mathrm{KL}(\rho_0|\rho^*),$$

and thus (13) holds. □

C ALGORITHM DETAILS

In this section, we provide additional implementation details of our method (Algorithm 1), including several techniques and heuristics on the hyperparameter selection and optimization strategy.

Multiple-stage optimization. In general, the hyperparameters α_i, $i = 1, 2$, β and γ in the expression (10) should be chosen in a way that the corresponding terms are balanced, and thus the minimizer ρ^* satisfies the desired properties, *i.e.* diversity and global Pareto optimality.

Since the main aim of multi-objective optimization is to profile the Pareto front instead of aggregating all potentials together as in (10), an alternative understanding of the problem is the following constraint optimization problem:

$$\min_{\mathrm{supp}\rho\subset\mathcal{D}} \quad \mathcal{G}[\rho] \quad \text{s.t.} \quad \mathcal{F}[\rho] \le C, \tag{22}$$

where the tolerance C controls how close the population ρ is to the Pareto front. This type of constraint optimization problem is often solved by the interior point method or the primal-dual method (Wright, 1997). However, unfortunately, the Euclidean geometry exploited by these methods is generally unavailable in the space of probability measures.

Another popular method of solving problems of this kind is the *penalty method*. The idea is to relax the original problem by adding a penalty term that penalizes the violation of the constraint. The convergence of this method is only asymptotic, *i.e.* the penalty parameter should be gradually increased to infinity.

Therefore, by viewing our method as a penalty relaxation to the above constraint optimization problem, one of the most important heuristics for the hyperparameter selection and tuning in our method is to split the optimization into multiple stages:

- In the first stage, we only impose a relatively small penalty on the constraint (with small dominance potential coefficient α_2, but large repulsive potential coefficient β and diffusion coefficient γ in (10)), encouraging the population to diversify and explore;
- As we gradually increase the penalty, the objective functions (constraints) will be balanced with the structural potentials being optimized, and thus, the population will be pushed towards the Pareto front while preserving diversity.
- In the final stage, we impose a larger penalty (with a large dominance potential coefficient α_2, but small repulsive potential coefficient β and diffusion coefficient γ in (10)), eliminating dominated particles and thus obtaining a population that is close to the Pareto front.

Numerical approximation of $\nabla\|\mathbf{g}^\dagger\|^2$. Denote $\boldsymbol{\alpha}^\dagger(\mathbf{x}) = (\alpha_1^\dagger(\mathbf{x}), \cdots, \alpha_m^\dagger(\mathbf{x}))$ as the optimal solution of (6) at point $\mathbf{x}$. Then we have $\mathbf{g}^\dagger(\mathbf{x}) = \sum_{i=1}^m \alpha_i^\dagger(\mathbf{x})\nabla f_i(\mathbf{x}) := (\nabla\mathbf{f})\boldsymbol{\alpha}^\dagger$. Moreover, the optimality of $\boldsymbol{\alpha}^\dagger$ yields the following relation:

$$\nabla\boldsymbol{\alpha}^\dagger(\nabla\mathbf{f})^\top\mathbf{g}^\dagger = \mathbf{0}. \tag{23}$$

Then $\nabla\delta_\rho\mathcal{F}_1[\rho]$ can be computed as

$$\begin{aligned}\nabla\delta_\rho\mathcal{F}_1[\rho] &= \nabla\|\mathbf{g}^\dagger\|^2 = 2(\nabla\mathbf{g}^\dagger)\mathbf{g}^\dagger \\ &= 2(\nabla^2\mathbf{f} : \boldsymbol{\alpha}^\dagger + \nabla\boldsymbol{\alpha}^\dagger(\nabla\mathbf{f})^\top)\mathbf{g}^\dagger \\ &= 2(\nabla^2\mathbf{f} : \boldsymbol{\alpha}^\dagger)\mathbf{g}^\dagger,\end{aligned} \tag{24}$$

where $\nabla^2\mathbf{f} : \boldsymbol{\alpha}^\dagger = \sum_{i=1}^m \alpha_i^\dagger\nabla^2 f_i$.

However, the computation of $\nabla^2\mathbf{f}$ is very expensive, as it involves the second-order derivatives of the objective functions. Therefore, we treat $\nabla^2\mathbf{f} : \boldsymbol{\alpha}^\dagger$ as a preconditioner and approximate it by the identity matrix.

Numerical approximation of $\delta_\rho\mathcal{F}_2$. In (7), we use a predetermined measure $\mu_\mathcal{P}$ on the Pareto front, which is not known in advance. However, as mentioned earlier, we only turn on the dominance potential term $\mathcal{F}_2[\rho]$ in the final stage of our method, and thus, we can use the empirical measure of the population ρ_t as a proxy of $\mu_\mathcal{P}$.

Furthermore, one can apply the following relaxation to the dominance kernel $D(\cdot,\cdot)$ presented in (8):

$$D(\mathbf{f}(\mathbf{x}, \mathbf{f}(\mathbf{y}))) = \prod_{i=1}^m \left(\max\{0, f_i(\mathbf{x}) - f_i(\mathbf{y})\} + c\mathbf{1}\{f_i(\mathbf{x}) - f_i(\mathbf{y}) \ge 0\}\right),$$

where $c > 0$ is a small constant and $\mathbf{1}\{\cdot\}$ is the indicator function. This relaxation deals with the rare scenario that there exists a point $\mathbf{x}$ that is dominated by $\mathbf{y}$ in some objectives but has the exactly same values in some other objectives.

Numerical approximation of $\delta_\rho \mathcal{H}[\rho_t]$. The computation of the instantaneous birth-death rate Λ_t (16) in the implementation of the birth-death dynamics involves the computation of the Fréchet derivative of the entropy term $\mathcal{H}[\rho_t]$. However, as we are approximating ρ_t by a set of particles $\{\mathbf{x}_k\}_{k=1}^N$, the Fréchet derivative $\delta_\rho \mathcal{H}[\rho_t]$ cannot be directly computed. Instead, we approximate it by the following *kernel density estimation* technique, as also adopted by Lu et al. (2019):

$$\rho_t(\mathbf{x}) \approx \frac{1}{N} \sum_{k=1}^{N} K\left(\frac{\mathbf{x} - \mathbf{x}_k}{h}\right), \tag{25}$$

where $K(\cdot)$ is a smooth kernel function, such as the Gaussian kernel $K(\mathbf{x}) = \exp(-\|\mathbf{x}\|^2)$, where h is the bandwidth parameter. Then the Fréchet derivative of $\delta_\rho \mathcal{H}[\rho_t]$ can be approximated by

$$\delta_\rho \mathcal{H}[\rho_t](\mathbf{x}) \approx -\log \frac{1}{N} \sum_{k=1}^{N} K(\mathbf{x} - \mathbf{x}_k). \tag{26}$$

Another possible technique is to use the Gaussian mixture model (GMM) to approximate the probability density ρ_t (Yan et al., 2023), which would result in a more straightforward computation of the Fréchet derivative $\delta_\rho \mathcal{H}[\rho_t]$. However, finding the particle weights deviates from our method's goal, and we choose to use the kernel density estimation for simplicity.

In practice, as we are imposing the repulsive potential term explicitly, the contribution of the entropy term is relatively small in the stochastic birth-death dynamics.

D EXPERIMENT DETAILS

In this section, we provide additional details of the experiments conducted in Section 4. We present the closed-form formulas of the ZDT3 problem in Appendix D.1, and additional experiments on the ZDT1 and ZDT2 problems are shown in D.1.1 and D.1.2, respectively. The closed-form formula of the DTLZ7 problem is provided in Appendix D.2. We also provide the experiment details of the learning-to-rank task and the MSLR-WEB10K dataset in Appendix D.3.

D.1 ZDT3 Problem

The closed-form formula of the ZDT3 problem is as follows:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1, \\ f_2(\mathbf{x}) &= g(\mathbf{x}) h(f_1(\mathbf{x}), g(\mathbf{x})), \end{aligned} \tag{27}$$

where $\mathbf{x} = (x_1, \cdots, x_{30})$,

$$g(\mathbf{x}) = 1 + \frac{9}{29} \sum_{i=2}^{30} x_i. \tag{28}$$

and

$$h(f_1, g) = 1 - \sqrt{\frac{f_1(\mathbf{x})}{g(\mathbf{x})}} - \frac{f_1(\mathbf{x})}{g(\mathbf{x})} \sin(10\pi f_1(\mathbf{x})). \tag{29}$$

The feasible region is $\mathcal{D} = [0, 1]^{30}$.

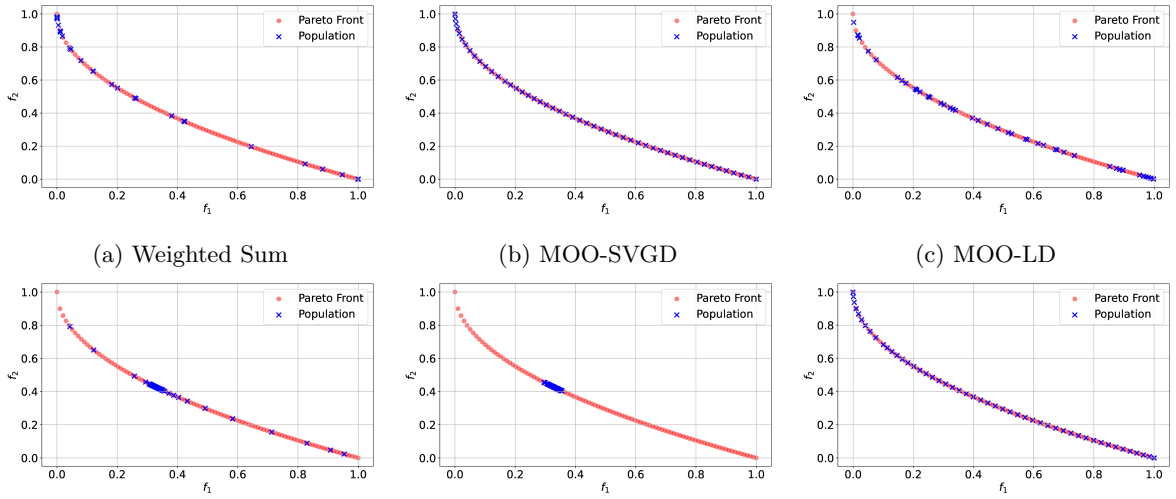
D.1.1 Additional Experiments on the ZDT1 Problem

The ZDT1 problem is another 30-dimensional two-objective optimization problem of the same form as in (27) but with

$$h(f_1, g) = 1 - \sqrt{\frac{f_1(\mathbf{x})}{g(\mathbf{x})}}.$$

The feasible region is $\mathcal{D} = [0, 1]^{30}$. The Pareto front of this problem is convex, continuous, and smooth, making it a relatively easy problem for MOO methods. This problem is also considered in Liu et al. (2021).

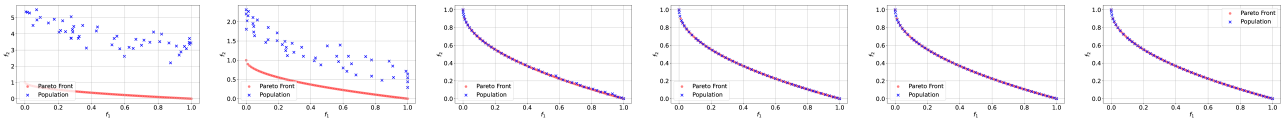
As shown in Figure 6, most of the methods can cover the whole Pareto front, but the weighted sum method, MOO-LD, and COSMOS are not able to cover the Pareto front uniformly. Our Particle-WFR method and MOO-SVGD give the best and most uniform coverage of the Pareto front. The evolution of the particle population by Particle-WFR is shown in Figure 7.



(a) Weighted Sum (b) MOO-SVGD (c) MOO-LD

(d) COSMOS (e) GMOOAR-HV (f) **Particle-WFR**

Figure 6: Performance comparison of different methods on the ZDT1 problem. The Pareto front is shown in red, and the solutions found by different methods are shown in blue.



(a) Epoch 0 (b) Epoch 1000 (c) Epoch 2000 (d) Epoch 3000 (e) Epoch 4000 (f) Epoch 5000

Figure 7: Evolution of the particle population by Particle-WFR on the ZDT1 problem. The Pareto front is shown in red, and the current population is shown in blue.

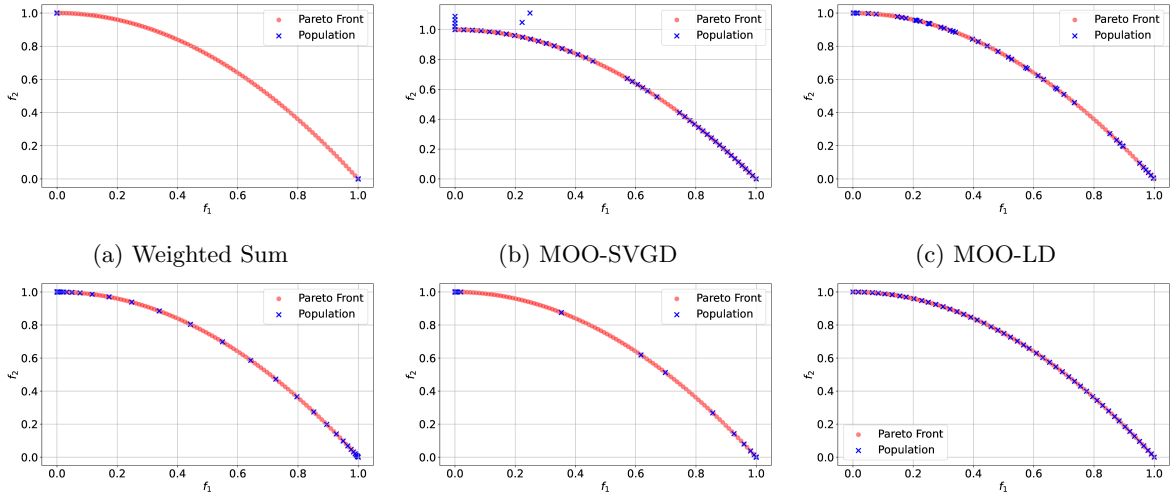
D.1.2 Additional Experiments on the ZDT2 Problem

The ZDT2 problem (Zitzler et al., 2000) is also a 30-dimensional two-objective optimization problem of the same form as in (27) but with

$$h(f_1, g) = 1 - \left(\frac{f_1(\mathbf{x})}{g(\mathbf{x})}\right)^2.$$

where the feasible region is $\mathcal{D} = [0, 1]^{30}$. Unlike the ZDT1 problem, the Pareto front of the ZDT2 problem is concave. Similar examples of concave Pareto fronts have been used in the literature, including the Fonseca problem (Fonseca and Fleming, 1995; Sener and Koltun, 2018; Lin et al., 2019; Mahapatra and Rajan, 2020) and the DTLZ2 problem (Chen and Kwok, 2022).

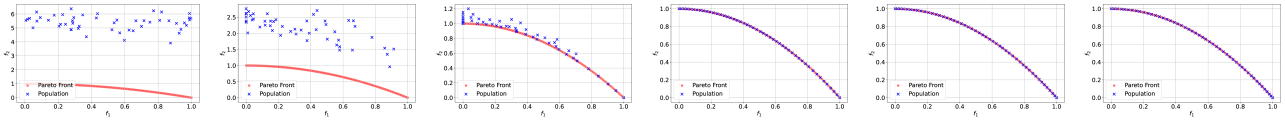
As shown in Figure 8, our Particle-WFR method can still cover the whole Pareto front uniformly. The weighted sum method fails in this case, and all solutions are concentrated on the two ends of the Pareto front. MOO-LD, COSMOS, and GMOOAR-HV are able to cover the Pareto front but not uniformly. MOO-SVGD performs well at most regions of the Pareto front, but two gaps of solutions are observed on the Pareto front, and several sub-optimal points exist near the upper-left end $(0, 1)$ of the Pareto front. The evolution of the particle population by Particle-WFR is shown in Figure 9.



(a) Weighted Sum (b) MOO-SVGD (c) MOO-LD

(d) COSMOS (e) GMOOAR-HV (f) **Particle-WFR**

Figure 8: Performance comparison of different methods on the ZDT2 problem. The Pareto front is shown in red, and the solutions found by different methods are shown in blue.



(a) Epoch 0 (b) Epoch 1000 (c) Epoch 2000 (d) Epoch 3000 (e) Epoch 4000 (f) Epoch 5000

Figure 9: Evolution of the particle population by Particle-WFR on the ZDT2 problem. The Pareto front is shown in red, and the current population is shown in blue.

D.2 DTLZ7 Problem

The DTLZ7 problem has the following form of $\mathbf{f}(\mathbf{x})$:

$$\begin{aligned} f_1(\mathbf{x}) &= x_1, \\ f_2(\mathbf{x}) &= x_2, \\ f_3(\mathbf{x}) &= (1 + g(\mathbf{x}))h(f_1(\mathbf{x}), f_2(\mathbf{x}), g(\mathbf{x})), \end{aligned} \tag{30}$$

where $\mathbf{x} = (x_1, \cdots, x_{30})$,

$$g(\mathbf{x}) = 1 + \frac{9}{29}\sum_{i=3}^{30} x_i$$

and

$$h(f_1(\mathbf{x}), f_2(\mathbf{x}), g(\mathbf{x})) = 3 - \sum_{i=1}^{2} \frac{f_i(\mathbf{x})}{1 + g(\mathbf{x})}(1 + \sin(3\pi f_i(\mathbf{x}))).$$

The feasible region is also $\mathcal{D} = [0, 1]^{30}$.

D.3 MSLR-WEB10K Dataset

In this section, we provide additional details of the learning-to-rank task and the MSLR-WEB10K dataset.

D.3.1 Learning-to-Rank Task

For readers' convenience, we repeat the settings of the learning-to-rank task as provided in Section 4.3 in the main text as follows.

Suppose we have a collection of *query groups* $\Psi = \{\Psi^{(p)}\}_{p=1}^{|\Psi|}$, where each query group $\Psi^{(p)}$ consists of $n^{(p)}$ documents. These items are characterized by a feature vector $\mathbf{x}_j^{(p)} \in \mathbb{R}^{d_f}$, generated from upstream tasks, and an associated relevance label $y_j^{(p)}$. In our case, the relevance labels are assumed to be positive. The goal is to derive an ordering $\pi^{(p)}$, *i.e.*

$$\pi^{(p)} : \{1, \cdots, n^{(p)}\} \to \{1, \cdots, n^{(p)}\}$$
$$j \mapsto \pi_j^{(p)},$$

for the items in each query group $\Psi^{(p)}$ given the feature vectors $\{\mathbf{x}_j^{(p)}\}_{j=1}^{n^{(p)}}$, that optimizes the utility $u(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})$ of the ordered list. We will denote the set of all possible orderings of the items in $\Psi^{(p)}$ as $\Pi^{(p)}$.

Utility functions (or ranking metrics) are operators that measure the quality of the ordering $\pi^{(p)}$ with respect to the relevance labels $\{y_j^{(p)}\}_{j=1}^{n^{(p)}}$. Intuitively, the utility function should be large if the items with higher relevance labels are ranked higher in the ordering $\pi^{(p)}$. One of the most widely adopted measures for the utility function $u(\cdot;\cdot)$ above is the Normalized Discount Cumulative Gain (NDCG) (Wang et al., 2013), which is defined as

$$\mathrm{NDCG}(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}) = \frac{\mathrm{DCG}(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})}{\mathrm{DCG}(\pi^{(p),*}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})}, \tag{31}$$

where $\mathrm{DCG}(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})$ is the discounted cumulative gain of the ordering $\pi^{(p)}$, defined as

$$\mathrm{DCG}(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}) = \sum_{j=1}^{n^{(p)}} \frac{2^{y_{\pi_j^{(p)}}^{(p)}} - 1}{\log_2(1+j)},$$

and $\pi^{(p),*}$ is the optimal ordering of the items in $\Psi^{(p)}$, *i.e.*

$$\pi^{(p),*} = \operatorname*{argmax}_{\pi^{(p)} \in \Pi^{(p)}} \mathrm{DCG}(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}).$$

In practical settings, one may also be interested in the truncated versions of the NDCG, denoted as NDCG@k, where only the top k items in the ordering $\pi^{(p)}$ are considered, *i.e.*

$$\mathrm{NDCG@}k(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}) = \frac{\mathrm{DCG@}k(\pi^{(p)}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})}{\mathrm{DCG@}k(\pi^{(p),*}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}})}, \tag{32}$$

where $\mathrm{DCG@}k(\cdot;\cdot)$ is defined by replacing the summation in $\mathrm{DCG}(\cdot;\cdot)$ with the summation over the top k items in the ordering $\pi^{(p)}$. For datasets with more than one query group, the utility function $u(\cdot;\cdot)$ is defined as the average of the NDCG@k over all the query groups.

Most of the current LTR methods employ a neural network f_θ, with θ denoting the parameters, to produce a score for each item, based on which the ordering $\pi^{(p)}$ is obtained. In particular, the neural network f_θ accepts the feature vector $\mathbf{x}_j^{(p)}$ of the j-th item in the query group $\Psi^{(p)}$ as input and produces a score $f_\theta(\mathbf{x}_j^{(p)})$ for the item. Then, the ordering $\pi^{(p)}$ is obtained by sorting the items in $\Psi^{(p)}$ according to the scores produced by f_θ.

The neural network is trained using the empirical loss of the following form:

$$\mathcal{L}(\theta; \Psi) = \frac{1}{|\Psi|} \sum_{p=1}^{|\Psi|} \ell\left(\{f_\theta(\mathbf{x}_j^{(p)})\}_{j=1}^{n^{(p)}}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}\right),$$

where $\ell(\cdot,\cdot)$ is the query group-wise loss function. One should notice that the loss function $\ell(\cdot;\cdot)$ has a different nature than the utility function $u(\cdot;\cdot)$, as the latter takes in an ordering $\pi^{(p)}$, while the former takes in a set of scores $\{f_\theta(\mathbf{x}_j^{(p)})\}_{j=1}^{n^{(p)}}$. Consequently, the loss function $\ell(\cdot;\cdot)$ is differentiable, while the utility function $u(\cdot;\cdot)$ is not.

In order to bridge this gap caused by the non-differentiability of the utility functions, such as the NDCG. Many works use differentiable surrogates for the NDCG metric as the loss function for training, including the *Cross-Entropy (CE) loss* (Cao et al., 2007) for ℓ, defined as:

$$\ell_{\mathrm{CE}}\left(\{f_\theta(\mathbf{x}_j^{(p)})\}_{j=1}^{n^{(p)}}; \{y_j^{(p)}\}_{j=1}^{n^{(p)}}\right) = -\sum_{j=1}^{n^{(p)}} y_j^{(p)} \log \frac{\exp(f_\theta(\mathbf{x}_j^{(p)}))}{\sum_{j=1}^{n^{(p)}} \exp(f_\theta(\mathbf{x}_j^{(p)}))}, \tag{33}$$

where we employ the softmax function to the scores produced by the neural network f_θ to obtain a probability distribution over the items in $\Psi^{(p)}$.

As analyzed and empirically verified in Qin et al. (2021), this choice of the loss function, referred to as the softmax loss, is one of the simplest and most robust choices for differentiable surrogates in the LTR task. We direct the readers to Qin et al. (2021) for more details on other popular surrogates and the discussions therein.

D.3.2 Implementation Details of MSLR-WEB10K Dataset

The Microsoft Learning-to-Rank Web Search (MSLR-WEB10K) dataset (Qin and Liu, 2013) is one of the most widely used benchmark datasets for the LTR task. The MSLR-WEB10K dataset consists of 10,000 query groups ($|\Psi| = 10^4$), each representing a query issued by a user. Each query group contains a list of items, each of which is a URL retrieved by the search engine in response to the query. Each item is characterized by a feature vector $\mathbf{x}_j^{(p)} \in \mathbb{R}^{136}$, extracted from the webpage, and a relevance label $y_j^{(p)} \in \{0, 1, 2, 3, 4\}$, indicating the relevance of the item to the query. Following the practice of (Mahapatra et al., 2023a), we treat the first 131 features as the input ($d_f = 131$) and combine the last 5 features, *viz.* Query-URL Click Count, URL Dwell Time, Quality Score 1, Quality Score 2, with the relevance label, as six different ranking objectives ($m = 6$).

We adopt a simple Multi-Layer Perception (MLP) of architecture $[131, 32, 1]$ as the neural network f_θ for the LTR task. We train the neural network with the Adam optimizer with a learning rate of 10^{-3} and a batch size of 512. The training loss is chosen to be the CE loss ℓ_{CE} as defined in (33) and (17). NDCG@10 (32) is used as the test metric. The training process is terminated after 500 epochs.

As the first practice of applying the interacting particle method to a multi-objective LTR task, our Particle-WFR method features the acceleration of Distributed Data Parallel (DDP) in PyTorch (Paszke et al., 2019). The DDP is a distributed training strategy that allows the training process to be distributed across multiple GPUs. In our case, we use 4 GPUs to train the neural network f_θ in parallel, and we expect further scalability by using more GPUs in real applications. Our code will be made publicly available upon publication.

D.3.3 Hypervolume Indicator

The hypervolume (HV) indicator (Zitzler and Künzli, 2004) is a widely used metric for evaluating the performance of MOO methods. The hypervolume of a set of points $\hat{\mathcal{P}}$ that approximate the Pareto front $\mathcal{P}$ is defined as the volume of the dominated region of $\hat{\mathcal{P}}$ with respect to a reference point $\boldsymbol{r}$, *i.e.*

$$\mathrm{HV}(\hat{\mathcal{P}}) = \int_{\mathbb{R}^m} \mathbf{1}\{\mathbf{x} \preceq \boldsymbol{r} \mid \exists \mathbf{y} \in \hat{\mathcal{P}} \text{ s.t. } \mathbf{y} \preceq \mathbf{x}\} \mathrm{d}\mathbf{x}, \tag{34}$$

where $\preceq$ denotes the Pareto dominance relation as in Definition 1. Hypervolume not only measures the optimality of the solutions found by the MOO methods but also measures the diversity of the solutions. A larger hypervolume indicates that the solutions are more diverse and closer to the Pareto front. Furthermore, the Pareto front $\mathcal{P}$ itself achieves the highest possible hypervolume.

Figure 5 and Table 1 presents the performance comparison of different methods on the MSLR-WEB10K dataset. The hypervolume values therein are computed with respect to the NDCG@10 metric evaluated on the test set, apart from the training loss. In the case of NDCG@10, we have to modify the definition of the hypervolume indicator to account for the fact that the NDCG@10 metric is being maximized, which can be resolved by adding a minus sign to all the values involved and choosing the reference point $\boldsymbol{r}$ to be the origin. The values are of the order of 10^{-4} because of the high-dimensionality ($m = 6$) of the MOO problem, noticing that the value of NDCG is between 0 and 1.

Method	HV of Test NDCG@10		
	$N = 8$	$N = 12$	$N = 16$
PHN-LS (Navon et al., 2020)	3.51±0.63 (4.58)	4.01±0.41 (4.79)	4.17±0.40 (5.04)
PHN-EPO (Navon et al., 2020)	3.60±0.63 (4.51)	3.65±0.65 (4.91)	4.15±0.67 (5.23)
COSMOS (Ruchte and Grabocka, 2021b)	3.81±0.33 (4.37)	4.00±0.37 (5.02)	4.19±0.32 (5.19)
GMOOAR-HV (Chen and Kwok, 2022)	2.57±0.17 (2.86)	3.25±0.29 (3.69)	3.65±0.32 (4.05)
GMOOAR-U (Chen and Kwok, 2022)	1.90±**0.09** (2.11)	4.26±0.25 (4.66)	4.07±**0.16** (4.42)
Particle-WFR (Ours)	**6.48**±0.38 (**7.27**)	**7.07**±**0.23** (**7.48**)	**6.95**±0.26 (**7.60**)

Table 1: Performance comparison of different methods on the MSLR-WEB10K dataset. The hypervolume (HV) is presented in the form of mean ± std (max) over the last 30 epochs, with the unit being 10^{-4}.