From Environmental Parameters to Pareto Optimal Prediction: LLM-Based Large-Scale Dynamic Multi-Objective Optimization

Chenyang Li
College of Computer Science
Sichuan University
Chengdu, China
lcy.yang@foxmail.com

Gary G. Yen
College of Computer Science
Sichuan University
Chengdu, China
gyen@okstate.edu

Lei Liu
College of Computer Science
Sichuan University
Chengdu, China
liulei.cx@gmail.com

Zhenan He
College of Computer Science
Sichuan University
Chengdu, China
zhenan@scu.edu.cn

***Abstract*—Existing prediction-based dynamic multi-objective evolutionary algorithms (DMOEAs) face significant challenges in tracking dynamic Pareto optimal set (POS) for large-scale dynamic multi-objective optimization problems, including insufficient high-quality training data and the lack of prior knowledge about optimal solutions in the new environment. To address these issues, we propose a large language model (LLM)-based DMOEA. The proposed method jointly models environmental time-varying parameters and dynamic POS within a unified semantic space via LoRA-based fine-tuning. The fine-tuned LLM leverages parameter variation sequences from historical to new environments, together with historical POS sequences, as contextual information to accurately predict the change directions of POS in a new environment. In doing so, this facilitates the evolution progress in finding the POS in the new environment. Additionally, we design an LLM reasoning-guided evolutionary search that narrows the search region from the large-scale search space to a promising local region, generating an initial population with both convergence and diversity for the new environment. Comprehensive experimental results on widely used dynamic benchmark problems demonstrate that LLM-DMOEA significantly outperforms state-of-the-art algorithms, particularly in large-scale scenarios.**

***Keywords*—dynamic multi-objective optimization problems, large-scale multi-objective optimization problems, large language model, multi-objective evolutionary algorithms.**

I. Introduction

Dynamic Multi-Objective Optimization Problems (DMOPs) exist widely in the real world, where the objective function changes over time [1], [2], [3]. The major challenge is to maintain an efficient tracking of the dynamic Pareto Optimal Set (POS) and the corresponding Pareto Optimal Front (POF) as the environment continuously changes. This challenge becomes even more demanding in large-scale scenarios, which are often encountered in practical DMOP applications [4], [5]. Gas transportation in large-scale pipeline networks is a typical real-world example [6], [7]. Such systems involve hundreds or even thousands of decision variables, which represent node pressures, compressor power settings, and valve operations. Meanwhile, the environment is influenced by various time-dependent parameters that change continuously, including consumer demand, market fluctuation, temperature, and humidity. In such scenarios, optimization algorithms must rapidly identify the high-dimensional POS within a vast search space after each environmental change.

In the field of evolutionary computation, researchers have developed various dynamic multi-objective evolutionary algorithms (DMOEAs) [2]. Among these, prediction-based DMOEAs are the most representative ones, as they can learn change patterns of POS across environments or extract useful knowledge from historical environments to help populations rapidly converge to the optimal region in a new environment [4]. However, these prediction-based DMOEAs overlook the modeling complexity and prediction degradation introduced by the high-dimensional decision space in large-scale DMOPs [5]. Specifically, due to the strong coupling among high-dimensional decision vectors and the scarcity of high-quality training data, it is difficult to construct accurate prediction models for dynamic POS [4], [5], [8]. Although recurrent neural network [9], long short-term memory networks [10], and spatio-temporal tensor regression methods [11] have been employed to enhance prediction accuracy of dynamic POS, these models often fail to capture the complex dependencies among high-dimensional decision vectors when training data are inadequate [4], [12].

In addition, some existing methods assume that the POS distributions of historical and the new environment are independent and identically distributed [13], [14]. Consequently, prediction models trained solely on historical solutions exhibit significant performance degradation when encountering nonlinear environmental changes [15], [16]. Some studies have introduced autoencoders [17], artificial neural networks [18], or transfer learning mechanisms [15],

This work was supported by the Research Fund for International Senior Scientists from the National Natural Science Foundation of China under Grant W2531051, the National Natural Science Foundation of China under Grant 62422604, the National Key Research and Development Program of China under Grant 2024YFF0509600, the Sichuan University-Suining Municipal Strategic Cooperation Science and Technology Project under Grant 2024CDSN-05, and the Fundamental Research Funds for the Central Universities under Grant YJ202501.

Corresponding author: Gary G. Yen

[16], [19], [20] to directly learn mapping relationships between adjacent environments for cross-environment generalization. However, without effective conditional guidance and target-domain data, accurately inferring the change patterns of POS under nonlinear dynamics remains challenging, if not impossible.

Moreover, maintaining population diversity in dynamic and large-scale search spaces remains a major obstacle. Most prediction-based DMOEAs generate initial populations by combining these predicted solutions with uniform random sampling, which contributes little to diversity in a massive search space [4]. Recently, generative models such as diffusion models [21], [22], [23] and adversarial autoencoders [4], [5] have been introduced to learn the distribution of optimal regions in a new environment, enabling the generation of solutions with both promising convergence and diversity. However, these methods often rely on traditional prediction techniques, such as Markov predictor-based POS angle estimation [5] and centroid-based prediction [22], to pre-extract optimal information from the new environment for model training. Due to the high dimensionality, strong coupling, and non-liner dynamic of POS in large-scale DMOPs, such prediction techniques frequently suffer from limited accuracy, which may mislead the generative models and degrade the quality of generated solutions for the new environment.

The fundamental limitation of these methods is their reliance on the assumption that change patterns of dynamic POS can be captured via explicit numerical correlations extracted from historical data and generalized to a new environment. However, in complex large-scale DMOPs, this assumption becomes questionable, as the underlying dynamics of high-dimensional POS are characterized by strong nonlinearity and coupling among decision variables [4]. In large-scale DMOPs, limited high-quality training data are insufficient to support reliable learning of the change patterns of POS. This also hinders the accurate acquisition of optimal solution information in a new environment through simple prediction models, as employed by some existing methods [4], [15], [22], [24]. In practice, the change patterns of POS in the new environment may arise from the combined effects of historical dynamic patterns and the current environmental state of the objective function. Specifically, changes in the objective functions are typically driven by time-varying parameters within large-scale dynamic systems, such as temperature, humidity, and consumer demand in gas pipeline network optimization [6]. These parameters originate from diverse sources, including external environmental variations and unanticipated events. Changes in these parameters lead to adjustments in the objective function, which in turn reshape the fitness landscape and give rise to dynamic POSs [3]. Therefore, how to uncover the interactions among these multi-source time-varying parameters and their latent effects on change patterns of POSs across environments provides a key research direction for solving large-scale DMOPs.

Recent advances in large language models (LLMs) [25], [26], [27] offer a promising pathway toward achieving this breakthrough. LLMs possess powerful capabilities in semantic understanding, logical reasoning, and conditional generation, enabling them to capture complex contextual dependencies within high-dimensional semantic spaces [28], [29]. These capabilities make it possible to discover latent semantic associations between environmental time-varying parameters and POS changes across environments, providing a new perspective beyond traditional numerical modeling. Building on this idea, we propose an LLM-based large-scale dynamic multi-objective evolutionary optimization algorithm (LLM-DMOEA). In our work, we leverage the LLM to jointly model environmental time-varying parameters and dynamic POSs, enabling the prediction of POS in a new environment by simultaneously considering the variation trends of time-varying parameters from historical to new environments and the change patterns of historical POS.

Specifically, we formulate POS prediction in a new environment as a context-dependent token generation task, where cross-environment change sequences of historical POS and environmental time-varying parameter sequences serve as contextual information. To enable effective learning of this task, we fine-tune the pre-trained LLM (e.g., Llama [30], [31]) via the LoRA [32] to capture implicit semantic dependencies between environmental time-varying parameters and dynamic POS within a unified semantic space. In a new environment, the fine-tuned LLM leverages environmental time-varying parameters and historical POS as contextual information to predict a high-quality initial population. The main contributions of this paper are summarized as follows:

- We propose an LLM-DMOEA, which jointly models environmental time-varying parameters and dynamic POS within a unified semantic space via the LoRA-based fine-tuning. By jointly leveraging cross-environment change sequences of historical POS and environmental time-varying parameter sequences as contextual information, the proposed method enables high-probability prediction of promising search directions toward the true POS in the new environment, thereby accelerating the optimal convergence of multi-objective evolutionary algorithms (MOEAs).
- We introduce a directional compact representation for cross-environment sequences of high-dimensional POS matrices. Since directly employing high-dimensional matrices as inputs and outputs of the LLM is infeasible due to computational efficiency and context length constraints, we compress these matrices into cosine similarity representations which encode cross-environment directional changes. This representation provides the LLM with structured and dimensionally controllable modeling targets, enabling efficient and scalable POS prediction in large-scale DMOPs.
- We design a fine-tuned LLM reasoning–guided evolutionary search. Leveraging the evolutionary directions predicted by the LLM, this strategy enables the MOEA to conduct structured directional sampling in the high-dimensional dynamic decision space, thereby

generating a high-quality initial population for the new environment.

The remainder of this paper is organized as follows. Section II presents the proposed LLM-DMOEA in detail. Section III describes the experimental setup. Section IV provides a comprehensive analysis of the experimental results. Finally, Section V concludes the paper and outlines directions for future work.

II. The Proposed Algorithm

To leverage the implicit dependencies among multi-source environmental time-varying parameters on POS change patterns, we jointly model the cross-environment change sequences of time-varying parameters and the dynamic POS via the LLM. Specifically, let the multi-source time-varying parameter sequence be $\boldsymbol{P} = \{\boldsymbol{p_1}, \dots, \boldsymbol{p_i} \dots, \boldsymbol{p_{t+1}}\}$, where $\boldsymbol{p_i}$ denotes the time-varying parameter vector in the i-th environment. Accordingly, the cross-environment dynamic POS is represented as $\boldsymbol{S} = \{\boldsymbol{s_1}, \dots, \boldsymbol{s_i}, \dots, \boldsymbol{s_t}\}$, where $\boldsymbol{s_i}$ denotes the POS in the i-th environment. To predict the POS in the new $(t+1)$ environment, we simultaneously leverage the time-varying parameters $\boldsymbol{p_{t+1}}$ as conditional context, enabling the model to infer $\boldsymbol{s_{t+1}}$ based on the historical $\boldsymbol{S}$ and the $\boldsymbol{P}$. Since the task inherently requires the model to predict the sequential dependencies of cross-environment POS based on rich contextual information, it naturally aligns with the autoregressive token prediction paradigm of LLMs [25], [28]. This alignment enables the LLM to leverage its powerful sequence modeling capability and context-aware conditional generation ability to improve the prediction accuracy of dynamic POSs by incorporating environmental time-varying parameters.

Building upon this modeling approach, we propose the LLM-DMOEA, whose core architecture comprises three synergistically integrated modules: **1) Directional compact representation module**, which compresses cross-environment sequences of high-dimensional POS matrices into cosine similarity representations, providing the LLM with structured and dimensionally controllable modeling goals; **2) Instruction fine-tuning based LLM learning module**, which jointly models multi-source environmental time-varying parameters and dynamic POS within a unified semantic space via LoRA fine-tuning; **3) LLM reasoning-guided directional evolutionary search module**, which leverages the search directions toward the potential POS in the new environment inferred by the fine-tuned LLM to generate a high-quality initial population for the MOEA.

A. Directional representation of dynamic POSs for LLM

In large-scale DMOPs, cross-environment dynamic POSs are typically stored as high-dimensional real-valued matrices. Directly tokenizing such matrices leads to excessive token counts and lengthy contexts, hindering LLMs from focusing on key structural features of POS changes. Therefore, we design a directional representation module that extracts compact directional descriptors from cross-environment POS sequences, enabling the LLM to focus on the critical structural change patterns of POS.

Specifically, the cross-environment POS sequence represents $\boldsymbol{S} = \{\boldsymbol{s_1}, \dots, \boldsymbol{s_i}, \dots, \boldsymbol{s_t}\}$, where $\boldsymbol{s_i} \in \mathbb{R}^{K \times D}$ denotes the POS matrix in the i-th environment, with K being the number of Pareto optimal solutions and D being the dimension of decision space. To characterize the directional changes of POS across consecutive environments, we compute the cosine similarity between corresponding solutions in adjacent environments. For the k-th Pareto optimal solution pair across environments i and $i+1$, the cosine similarity is computed as:

$$c_i^{(k)} = \frac{\langle s_i^{(k)}, s_{i+1}^{(k)} \rangle}{||s_i^{(k)}|| \cdot ||s_{i+1}^{(k)}||} \tag{1}$$

where $s_i^{(k)}$ and $s_{i+1}^{(k)}$ denote the k-th Pareto optimal solution in environments i and $i+1$ respectively. The operator $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors, and $|| \cdot ||$ denotes the Euclidean norm. The cosine similarity $c_i^{(k)}$ lies within [-1, 1], encoding the directional change of the k-th Pareto optimal solution across adjacent environments. Accordingly, the cross-environment directional representation for the k-th Pareto optimal solution can be represented as:

$$\boldsymbol{C}^{(k)} = \{c_2^{(k)}, c_3^{(k)}, \dots, c_t^{(k)}\} \tag{2}$$

Furthermore, we obtain the complete directional representation for the cross-environment POS sequence:

$$\boldsymbol{C} = \{\boldsymbol{C}^{(1)}, \boldsymbol{C}^{(2)}, \dots, \boldsymbol{C}^{(K)}\} \tag{3}$$

where K representative Pareto optimal solutions are selected using the crowded comparison operator [33]. These directional sequences compactly encode the cross-environment directional changes of high-dimensional POS matrices.

B. Fine-Tuned LLM for POS Prediction

In the subsection, we further leverage the LLM to jointly model environmental time-varying parameters $\boldsymbol{P}$ and POS directional sequences $\boldsymbol{C}$ for dynamic POS prediction through supervised fine-tuning. Specifically, we introduce a sliding window of size w to partition the historical environments. The $\boldsymbol{P^t} = \{\boldsymbol{p_{t-w+1}}, \dots, \boldsymbol{p_t}\}$ denotes the cross-environment time-varying parameter sequence within a complete sliding window. The contextual input of LLM is formed by concatenating the $\boldsymbol{P^t}$, the new environment parameter $\boldsymbol{p_{t+1}}$, and the directional sequence of a cross-environment Pareto optimal solution $\boldsymbol{C}^{(k)} = \{c_2^{(k)}, c_3^{(k)}, \dots, c_t^{(k)}\}$ within the sliding window. The new environment's directional representation of the Pareto optimal solution $c_{t+1}^{(k)}$ serves as the supervised response. These input-response pairs constitute the fine-tuning dataset. Subsequently, the LoRA technique is employed for parameter-efficient fine-tuning of the pre-trained LLM [34].

During the fine-tuning process, a supervised training dataset is constructed from historical environments, defined as:

$$\mathcal{D} = \left\{c_{t+1}^{(k)} | \boldsymbol{P^t}, \boldsymbol{p_{t+1}}, \boldsymbol{C}^{(k)}\right\}, t \in [w, T-1] \tag{4}$$

Instruction: "Based on the historical movement directions of the 1st Pareto optimal solution and the trends of environmental time-varying parameters, predict the POS movement direction for the new environment."

Input: "Historical Pareto optimal solution movement directions: {0.89, 0.90, 0.92, 0.85, 0.90, 1.00, 1.00, 0.99, 1.00}. The sequence of parameter G: {0.16, 0.31, 0.45, 0.59, 0.71, 0.81, 0.89, 0.95, 0.99}. The sequence of parameter H: {1.37, 1.48, 1.59, 1.69, 1.78, 1.86, 1.92, 1.96, 1.99}. The new environment parameter G is 1.00. The new environment parameter H is 2.00."

Output: "1.00."

Fig. 1. An example of fine-tuning data construction on the DF1 test function [35].

where w denotes the window length, T is the total number of historical environments. The $k \in [1,2,\dots,K]$ denotes the index of the k-th selected Pareto optimal solution. Within each sliding window, the environmental time-varying parameter sequence $\{\boldsymbol{P^t}, \boldsymbol{p_{t+1}}\}$ together with the corresponding historical POS directional sequences $\boldsymbol{C}^{(k)}$ are used as contextual inputs while the $c_{t+1}^{(k)}$ serves as the supervised output. Accordingly, each selected Pareto optimal solution is formulated as a supervised training sample consisting of three components: instruction, input, and output.

- The instruction component provides the prediction task description, i.e., "Based on the historical movement directions of the k-th Pareto optimal solution and the trends of environmental time-varying parameters, predict the POS movement direction for the new environment."
- The input component provides the contextual information, i.e., "The environmental time-varying parameter sequence $\boldsymbol{P^t}$, the new environment parameter $\boldsymbol{p_{t+1}}$, and the historical Pareto optimal solution movement direction sequences $\boldsymbol{C}^{(k)}$".
- The output component corresponds to the supervised output of the new environment, i.e., "the movement direction of the k-th Pareto optimal solution in the new $(t+1)$ environment, recorded as $c_{t+1}^{(k)}$".

A simplified example of the instruction fine-tuning training sample construction is illustrated in Fig. 1. In the LoRA-based fine-tuning [32], the environmental time-varying parameter sequences $\boldsymbol{P^t}$, the new environment parameters $\boldsymbol{p_{t+1}}$ and the historical directional sequence of the k-th Pareto optimal solution $\boldsymbol{C}^{(k)}$ are all transformed into token representations, forming a unified input sequence for the pre-trained LLM.

C. LLM Reasoning–Guided Evolutionary Search

This subsection presents the LLM reasoning-guided evolutionary search for the new $(t+1)$ environment. Given the $\boldsymbol{P^t}, \boldsymbol{p_{t+1}}$ and $\boldsymbol{C}^{(k)}$, the fine-tuned LLM infers the potential change directions of the k-th Pareto optimal solution in the new environment. Leveraging both the predicted directions derived from the LLM's inference process, a guided sampling strategy is employed within the predicted promising local region in the large-scale search space. The resulting high-quality initial population provides structured search priors that significantly accelerate the convergence of the MOEA toward the Pareto optimal region in the new environment.

Algorithm 1 LLM-DMOEA

Input:
$F^t(\mathbf{x})$: Large-scale DMOP ($t = 1, 2, \dots, T$)
The fine-tuned LLM: $\mathbf{LLM}_{ft}$
Output:
$DPOS$: Dynamic POSs
$DPOF$: Dynamic POFs

$t = 1$ *# The first environment*
While $t < T$ **do**
  If *detects that the environment has changed* **do**
    If $t < w$ **do**
      $\boldsymbol{X}_0 \leftarrow Random\ Initialization\ Population$
    Else
      For $k \in [1,2,\dots,K]$ **do**
        Construct input $(\boldsymbol{P^t}, \boldsymbol{p_{t+1}}, \boldsymbol{C}^{(k)})$ (Sections II.A, II.B)
        $\hat{c}_{t+1}^{(k)} = \mathbf{LLM}_{ft}(\boldsymbol{P^t}, \boldsymbol{p_{t+1}}, \boldsymbol{C}^{(k)})$
        $d^{(k)} = \cos(\theta^{(k)}) \cdot e^{(k)} + \sin(\theta^{(k)}) \cdot v^{(k)}$
        $\mathbf{x}_{t+1}^{(k)} = \mathbf{x}_t^{(k)} + \alpha \cdot d^{(k)}$
        $\boldsymbol{X}_0 = \bigcup_{k=1}^{K} \mathbf{x}_{t+1}^{(k)}$
      End For
    End If
  End If
  $t = t + 1$
  $\boldsymbol{X}^t, \boldsymbol{Y}^t \leftarrow MOEA(F^t(\mathbf{x}), \boldsymbol{X}_0)$
  $DPOF \leftarrow DPOF \cup \boldsymbol{Y}^t$
  $DPOS \leftarrow DPOS \cup \boldsymbol{X}^t$
End While
Return $DPOS$, $DPOF$

Specifically, after detecting an environmental change [2], the fine-tuned LLM takes the contextual input ($\boldsymbol{P^t}$, $\boldsymbol{p_{t+1}}$, $\boldsymbol{C}^{(k)}$) and outputs the predicted directional representation of the k-th Pareto optimal solution for the new environment:

$$\hat{c}_{t+1}^{(k)} = \mathbf{LLM}_{ft}(\boldsymbol{P^t}, \boldsymbol{p_{t+1}}, \boldsymbol{C}^{(k)}) \tag{5}$$

where $\mathbf{LLM}_{ft}$ denotes the fine-tuned LLM, and $\hat{c}_{t+1}^{(k)} \in [-1,1]$ indicates the predicted cosine similarity of the k-th Pareto optimal solution from environment t to environment $t+1$.

Based on the predicted cosine similarity, we determine a local sampling region in large-scale search space. Specifically, the predicted cosine similarity $\hat{c}_{t+1}^{(k)}$ defines a conical region centered at the k-th Pareto optimal solution $\mathbf{x}_t^{(k)}$ in the t environment, within which all candidate solutions satisfy the predicted cosine similarity constraint. Accordingly, the directional vector $d^{(k)}$ can be uniformly sampled within this constrained local region via the following formulation:

$$d^{(k)} = \cos(\theta^{(k)}) \cdot e^{(k)} + \sin(\theta^{(k)}) \cdot v^{(k)} \tag{6}$$

where Equation (6) provides a geometrically interpretable mechanism for translating the LLM-predicted cosine similarity into a concrete search direction in the decision space. Specifically, the unit vector $e^{(k)}$ represents the normalized historical Pareto optimal solution $\mathbf{x}_t^{(k)}$, which serves as the principal axis of the local search region. The maximum deviation $\theta^{(k)}$ is computed from the predicted cosine similarity as $\arccos(\hat{c}_{t+1}^{(k)})$. The resulting direction $d^{(k)}$ is constructed as a convex combination of the axial component $e^{(k)}$ and an orthogonal component $v^{(k)}$, ensuring that $d^{(k)}$ lies on the surface of a hyper spherical cone centered at $e^{(k)}$ with half-angle $\theta^{(k)}$ [36]. To ensure diversity, $v^{(k)}$ can be randomly sampled from the orthogonal complement space of

$e^{(k)}$. Equation (6) enables sampling within the predicted conical region, thereby generating candidate solutions that are as close as possible to the true POS movement direction in the new environment.

Along the directional vector $d^{(k)}$, we perform uniform sampling to generate candidate solutions for the new environment. Specifically, multiple candidate solutions are generated with uniformly distributed step sizes:

$$\mathbf{x}_{t+1}^{(k)} = \mathbf{x}_t^{(k)} + \alpha \cdot d^{(k)}, \alpha \in [\alpha_{\min}, \alpha_{\max}] \quad (7)$$

where α is uniformly sampled within the predefined step size range $[\alpha_{\min}, \alpha_{\max}]$. The initial population $\boldsymbol{X}_0$ for the new environment is constructed by aggregating candidate solutions from all K representative directions:

$$\boldsymbol{X}_0 = \bigcup_{k=1}^{K} \mathbf{x}_{t+1}^{(k)} \quad (8)$$

As shown in Algorithm 1, the generated $\boldsymbol{X}_0$ serves as a high-quality initial population for the MOEA, accelerating convergence toward the optimal region in the new environment. This is because the proposed strategy effectively narrows the search region from the entire large-scale search space to a promising local region, significantly reducing the exploration cost. Furthermore, sampling within this localized promising region inherently provides solutions with both strong convergence and adequate diversity, as the region is constructed to encompass the predicted POS distribution in the new environment.

III. Experimental Setup

A. Test Functions and Performance Indicator

To evaluate the proposed LLM-DMOEA, we adopt the DF test suite [35], which is widely used benchmark for evaluating DMOEAs. The dimension of decision space is scaled to $D \in \{30, 300, 600\}$ to assess performance in small and large-scale scenarios. Following the recommended settings in [35], the severity and frequency of environmental changes are controlled by $n_t = 10$, and $\tau_t = 10$, respectively. The mean inverted generational distance (MIGD) [2] is employed as the performance indicator, which measures the average distance from the obtained POF to the true POF across all environmental changes. A smaller MIGD value indicates better performance. Due to page limitations, additional experiments with more dynamic settings and performance indicators will be further validated in future work.

B. Comparing Algorithms

To comprehensively evaluate the performance of the proposed LLM-DMOEA, four state-of-the-art DMOEAs are selected as baselines for comparison, covering three main categories of prediction-based methods [5]. Among them, CGLP-DMOEA [37] belong to time-series prediction method, while AE-DMOEA [17] directly learns the mapping relationship between POSs in adjacent environments from historical data. MMTL-DMOEA [38] is a transfer learning-based approach that leverages knowledge extracted from historical environments to accelerate convergence in a new environment. AAE-LSDMOEA [5] is a generative model-based method that employs adversarial autoencoders to learn the distribution of optimal regions and generate the high-quality initial population for large-scale DMOPs.

C. Parameter Settings

Since only AAE-LSDMOEA was specifically developed for LSDMOPs, we follow its original parameter settings, while for the other baselines we retain their standard recommended configurations under the same underlying large-scale multi-objective optimizer [39], thus ensuring a fair and reproducible comparison. The population size is set to $N = 100$, and the maximum number of function evaluations is set to $10^4 \times \tau_t$. Each algorithm is independently executed 10 times on each test function to ensure statistical reliability. The Wilcoxon rank-sum test with a significance level of 0.05 is applied to determine statistical significance. In the result tables, the symbols "+", "–", and "≈" indicate that LLM-DMOEA performs significantly better than, significantly worse than, or statistically similar to the compared algorithm, respectively.

The parameter settings for the fine-tuning and inference phases of LLM-DMOEA are described as follows. For the LLM reasoning–guided evolutionary search, the number of representative Pareto optimal solutions is set to $K = 20$. The sliding window size is set to $w = 10$, with a detailed sensitivity analysis to be presented in a future extended journal version. For the LoRA-based fine-tuning module, we adopt Llama 2.5-8B-insruct [30] as the base model. The LoRA rank is set to $r = 8$ with all linear layers as the target modules. The model is fine-tuned for 3 epochs using the Adam optimizer with a learning rate of 10^{-4}, a cosine learning rate scheduler, and a warmup ratio of 0.1 [40].

IV. Experimental Results and Analysis

TABLE I presents the mean and standard deviation of MIGD results for LLM-DMOEA and four comparison algorithms on DF1-DF14 test functions. The dimension of decision space is set to 30, 300, and 600 to evaluate algorithm performance in both small-scale, medium-scale, and large-scale scenarios. The best results for each test instance are highlighted in bold.

From the overall performance perspective, LLM-DMOEA demonstrates significant competitive advantages across 42 test instances. Specifically, LLM-DMOEA achieves significantly better/worse/similar results compared to MMTL-DMOEA, AAE-LSDMOEA, AE-DMOEA, and CGLP-DMOEA in 32/1/9, 26/2/14, 32/2/8, and 34/1/7 test instances, respectively. Notably, the advantage of LLM-DMOEA becomes more pronounced as the dimension of decision space increases. In the large-scale scenarios ($D = 600$), LLM-DMOEA significantly outperforms all comparison algorithms on the majority of test functions, demonstrating its effectiveness in addressing the challenges posed by large-scale DMOPs.

Compared to CGLP-DMOEA, AE-DMOEA, and MMTL-DMOEA, LLM-DMOEA exhibits significant performance advantages in medium/large-scale scenarios ($D = 300$ and $D = 600$). To understand this performance gap, we analyze the characteristics of each method. CGLP-DMOEA partitions the population into three subpopulations based on the correlation of individual moving directions across environments and employs a linear prediction model to track

TABLE I. MEAN AND STANDARD DEVIATION OF MIGD RESULTS FOR LLM-DMOEA AND OTHER COMPARATORS ON DF1-DF14.

	Dimension	LLM-DMOEA	MMTL-DMOEA [38]	AAE-LSDMOEA [5]	AE-DMOEA [17]	CGLP-DMOEA [37]
DF1	30	0.00638 ± 6.12E-05	0.00642 ± 6.57E-05 (≈)	0.00644 ± 6.51E-05 (≈)	**0.00636 ± 3.02E-05** (≈)	0.00644 ± 6.15E-05 (≈)
	300	**0.01171 ± 1.06E-03**	0.05472 ± 1.52E-02 (+)	0.01255 ± 3.80E-04 (+)	0.01521 ± 1.98E-03 (+)	0.02180 ± 1.12E-03 (+)
	600	**0.02168 ± 6.79E-03**	0.05304 ± 1.44E-02 (+)	0.04580 ± 5.46E-03 (+)	0.05075 ± 2.08E-02 (+)	0.07502 ± 2.64E-02 (+)
DF2	30	0.00655 ± 5.01E-05	**0.00652 ± 4.75E-05** (≈)	0.00657 ± 5.59E-05 (≈)	0.00655 ± 7.10E-05 (≈)	0.00657 ± 6.59E-05 (≈)
	300	**0.01991 ± 2.13E-03**	0.03479 ± 6.90E-03 (+)	0.02235 ± 3.16E-03 (≈)	0.02175 ± 7.44E-04 (≈)	0.02745 ± 1.28E-03 (+)
	600	0.05434 ± 9.79E-03	**0.04163 ± 6.05E-03** (−)	0.05811 ± 3.66E-03 (≈)	0.05561 ± 4.45E-03 (+)	0.07438 ± 4.21E-03 (+)
DF3	30	**0.19830 ± 4.67E-03**	0.20726 ± 4.32E-03 (+)	0.21609 ± 1.03E-02 (+)	0.22184 ± 7.88E-03 (+)	0.22663 ± 4.96E-03 (+)
	300	**0.39697 ± 7.51E-03**	0.41150 ± 5.93E-03 (+)	0.39889 ± 6.84E-03 (≈)	0.40842 ± 2.76E-03 (+)	0.41140 ± 5.62E-03 (+)
	600	0.48925 ± 1.61E-02	0.51013 ± 1.68E-02 (+)	**0.48702 ± 1.48E-02** (≈)	0.51203 ± 1.78E-02 (+)	0.51574 ± 2.30E-02 (+)
DF4	30	**0.04246 ± 1.76E-03**	0.04540 ± 2.24E-03 (+)	0.04499 ± 3.08E-03 (+)	0.04478 ± 1.70E-03 (+)	0.04440 ± 3.11E-03 (≈)
	300	**0.06559 ± 1.21E-02**	0.45009 ± 7.10E-02 (+)	0.08597 ± 9.59E-03 (+)	0.61406 ± 3.59E-02 (+)	0.42849 ± 3.72E-02 (+)
	600	**0.09223 ± 1.09E-02**	1.91921 ± 4.96E-01 (+)	0.77715 ± 1.31E-01 (+)	2.59049 ± 2.83E-01 (+)	2.26192 ± 1.57E-01 (+)
DF5	30	0.00649 ± 2.89E-05	0.00653 ± 5.25E-05 (≈)	0.00661 ± 5.33E-05 (+)	**0.00641 ± 3.49E-05** (−)	0.00658 ± 4.69E-05 (+)
	300	**0.00839 ± 3.53E-04**	0.03197 ± 6.03E-03 (+)	0.01240 ± 3.18E-04 (+)	0.02910 ± 5.16E-03 (+)	0.03176 ± 1.13E-03 (+)
	600	**0.01084 ± 1.44E-03**	0.14397 ± 3.02E-02 (+)	0.07909 ± 2.85E-03 (+)	0.09532 ± 1.15E-02 (+)	0.10997 ± 3.86E-03 (+)
DF6	30	**0.02299 ± 1.92E-02**	0.02647 ± 3.60E-03 (≈)	0.03213 ± 1.31E-02 (≈)	0.06787 ± 7.86E-02 (+)	0.10811 ± 8.62E-02 (+)
	300	**0.28949 ± 6.44E-01**	0.57418 ± 1.75E-01(+)	0.94434 ± 7.51E-01(+)	1.25291 ± 7.82E-01 (+)	2.51108 ± 5.41E-01 (+)
	600	**0.49074 ± 1.11E+00**	1.36092 ± 3.31E-01 (+)	1.28195 ± 6.30E-01 (+)	3.14438 ± 2.05E+00 (+)	8.26469 ± 7.32E+00 (+)
DF7	30	**0.23066 ± 5.46E-02**	0.38269 ± 3.68E-02 (+)	0.38766 ± 2.76E-02 (+)	0.48935 ± 1.97E-01 (+)	0.77248 ± 3.28E-02 (+)
	300	**0.36167 ± 1.82E-01**	1.01847 ± 6.04E-02 (+)	0.59168 ± 3.22E-01 (+)	0.58264 ± 7.61E-02 (+)	1.18842 ± 6.02E-02 (+)
	600	**0.52448 ± 1.13E-01**	0.93428 ± 1.08E-01 (+)	0.62579 ± 3.68E-01 (≈)	0.52768 ± 9.71E-02 (≈)	1.30448 ± 1.82E-01 (+)
DF8	30	**0.00811 ± 5.77E-04**	0.00840 ± 6.55E-04 (≈)	0.00841 ± 7.30E-04 (≈)	0.00865 ± 4.07E-04 (≈)	0.00812 ± 6.77E-04 (≈)
	300	**0.01376 ± 1.98E-03**	0.02673 ± 5.26E-03 (+)	0.01555 ± 1.22E-03 (+)	0.06122 ± 4.90E-03 (+)	0.04739 ± 5.69E-03 (+)
	600	**0.01631 ± 1.41E-03**	0.07630 ± 7.51E-03 (+)	0.03409 ± 4.28E-03 (+)	0.12530 ± 1.22E-02 (+)	0.11605 ± 1.51E-02 (+)
DF9	30	**0.03654 ± 1.28E-03**	0.07470 ± 3.77E-03 (+)	0.04569 ± 2.41E-03 (+)	0.06027 ± 3.64E-03 (+)	0.05166 ± 2.43E-03 (+)
	300	**0.23054 ± 1.29E-02**	0.92450 ± 1.03E-01 (+)	0.29596 ± 2.70E-02 (+)	0.66217 ± 6.41E-02 (+)	1.27793 ± 8.39E-02 (+)
	600	**0.33631 ± 2.30E-02**	2.80247 ± 3.00E-01 (+)	0.45965 ± 7.66E-02 (+)	1.45504 ± 1.69E-01 (+)	2.65908 ± 1.75E-01 (+)
DF10	30	**0.06477 ± 2.08E-03**	0.06552 ± 2.99E-03 (≈)	0.06549 ± 1.60E-03 (≈)	0.06748 ± 2.38E-03 (+)	0.06781 ± 2.76E-03 (+)
	300	**0.07098 ± 1.93E-03**	0.09276 ± 2.06E-03 (+)	0.07427 ± 2.49E-03 (+)	0.08997 ± 5.09E-03 (+)	0.09830 ± 3.71E-03 (+)
	600	**0.07599 ± 3.03E-03**	0.11157 ± 2.00E-03 (+)	0.08615 ± 3.21E-03 (+)	0.10259 ± 5.66E-03 (+)	0.11785 ± 2.51E-03 (+)
DF11	30	0.03732 ± 2.64E-04	0.03735 ± 2.25E-04 (≈)	**0.03715 ± 1.87E-04** (−)	0.03733 ± 3.29E-04 (≈)	0.03719 ± 2.86E-04 (≈)
	300	**0.04988 ± 1.65E-03**	0.07342 ± 6.66E-03 (+)	0.05804 ± 4.41E-03 (+)	0.05173 ± 3.16E-03 (+)	0.06573 ± 5.92E-03 (+)
	600	0.08318 ± 1.28E-02	0.09387 ± 1.82E-02 (+)	0.08973 ± 1.35E-02 (≈)	**0.06583 ± 5.28E-03** (−)	0.09960 ± 1.01E-02 (+)
DF12	30	0.07261 ± 1.65E-03	0.09390 ± 2.06E-03 (+)	**0.06883 ± 1.15E-03** (−)	0.08647 ± 5.76E-03 (+)	0.07070 ± 1.84E-03 (−)
	300	**0.11628 ± 2.26E-03**	0.21223 ± 1.15E-02 (+)	0.12028 ± 7.65E-03 (≈)	0.20768 ± 1.30E-02 (+)	0.19600 ± 6.29E-03 (+)
	600	**0.13193 ± 6.25E-03**	0.27157 ± 7.31E-03 (+)	0.15820 ± 1.41E-02 (+)	0.25913 ± 2.50E-02 (+)	0.26096 ± 1.28E-02 (+)
DF13	30	0.13226 ± 3.38E-03	0.13029 ± 4.34E-03 (≈)	0.13108 ± 4.45E-03 (≈)	**0.13008 ± 2.32E-03** (≈)	0.13028 ± 5.65E-03 (≈)
	300	**0.17858 ± 6.88E-03**	0.28749 ± 2.37E-02 (+)	0.18540 ± 4.55E-03 (+)	0.20901 ± 1.33E-02 (+)	0.28503 ± 1.30E-02 (+)
	600	**0.21071 ± 1.02E-02**	0.45993 ± 3.14E-02 (+)	0.27159 ± 2.00E-02 (+)	0.29929 ± 2.68E-02 (+)	0.45929 ± 6.31E-02 (+)
DF14	30	0.04732 ± 4.81E-04	**0.04730 ± 4.75E-04** (≈)	0.04764 ± 4.48E-04 (≈)	0.04741 ± 3.21E-04 (≈)	0.04751 ± 5.56E-04 (≈)
	300	**0.06078 ± 1.69E-03**	0.08900 ± 8.63E-03 (+)	0.06343 ± 1.67E-03 (+)	0.06811 ± 3.80E-03 (+)	0.09802 ± 5.85E-03 (+)
	600	**0.07475 ± 6.57E-03**	0.12744 ± 2.38E-02 (+)	0.12425 ± 4.06E-03 (+)	0.12410 ± 1.77E-02 (+)	0.23060 ± 1.12E-02 (+)
+/−/≈		/	32/1/9	26/2/14	32/2/8	34/1/7

the change trends of high-correlation individuals. In a different approach, AE-DMOEA utilizes an autoencoder to learn POS movement patterns from nondominated solutions in the previous environment. Additionally, MMTL-DMOEA predicts optimal individuals in a new environment through memory mechanisms and manifold transfer learning. However, these methods share a common limitation: they rely solely on historical POS information for prediction without effectively utilizing multi-source environmental time-varying parameters to guide the prediction process. In contrast, LLM-DMOEA incorporates these environmental time-varying parameter sequences as conditional context, enabling the LLM to jointly model parameter sequences and POS directional sequences, thereby capturing the implicit dependencies between them and providing more accurate prediction results. Furthermore, regarding diversity maintenance, LLM-DMOEA leverages LLM reasoning-guided search to identify a promising local region likely containing the POS within the large-scale search space for sampling, effectively enhancing population diversity while ensuring convergence.

AAE-LSDMOEA is an algorithm specifically developed for solving large-scale DMOPs. Compared to LLM-DMOEA, it performs significantly worse in 26 test instances, better in 2 instances, and statistically similar in 14 instances. AAE-LSDMOEA employs a Markov predictor to track the movement directions of the POS as auxiliary information for the adversarial autoencoder to learn the distribution of the POS in a new environment. Consequently, its performance is largely affected by the prediction bias of the Markov predictor [4]. In contrast, LLM-DMOEA leverages LoRA-based fine-tuning to learn the implicit dependencies between historical POS sequences and environmental time-varying parameter sequences in the parameter space, enabling accurate prediction of POS change directions in the new environment.

In summary, the experimental results validate the effectiveness of LLM-DMOEA on large-scale DMOPs. The proposed method effectively narrows the search region from the entire large-scale search space to a promising local region, significantly accelerating the convergence of the MOEA in a new environment. Due to page limitations, more comprehensive ablation studies will be provided in future work.

V. CONCLUSION

This paper proposes an LLM-DMOEA, which jointly models multi-source environmental time-varying parameters and dynamic POSs within a unified semantic space for large-scale DMOPs. Unlike traditional prediction-based DMOEAs that rely on explicit numerical modeling of historical POS change patterns, the proposed method employs LoRA-based fine-tuning to capture the implicit dependencies between time-varying parameters and POS dynamics, significantly improving the prediction accuracy of POS change directions in a new environment. Comprehensive experimental results on the DF test suite demonstrate that LLM-DMOEA significantly outperforms state-of-the-art algorithms, particularly in large-scale scenarios. Future work will further explore the performance boundaries and scalability of LLM-DMOEA, and investigate the applications of this framework to real-world large-scale dynamic optimization systems.

References

[1] D. Hu and J. He, 'A Dynamic Multi-objective Optimization approach for computing resource allocation in industrial model repository', *Swarm Evol. Comput.*, vol. 99, p. 102219, Dec. 2025.

[2] S. Jiang, J. Zou, S. Yang, and X. Yao, 'Evolutionary Dynamic Multi-objective Optimisation: A Survey', *ACM Comput. Surv.*, vol. 55, no. 4, pp. 1–47, Apr. 2023.

[3] M. Farighi, M. Soltani, M. Ziabasharhagh, P. Pourmoghadam, and W. Alfraidi, 'Dynamic multi-objective optimization of a solar-based combined cooling, heating, and power (CCHP) system based on the data-driven model', *Appl. Therm. Eng.*, vol. 283, p. 128897, Jan. 2026.

[4] C. Li, G. G. Yen, and Z. He, 'Generative Model-Driven Large-Scale Dynamic Multi-Objective Evolutionary Optimization', in *2025 IEEE Congress on Evolutionary Computation (CEC)*, Jun. 2025, pp. 1–8.

[5] C. Li, G. G. Yen, and Z. He, 'Adversarial AutoEncoder-Based Large-Scale Dynamic Multiobjective Evolutionary Algorithm', *IEEE Trans. Evol. Comput.*, vol. 29, no. 4, pp. 1112–1126, Aug. 2025.

[6] P. Domschke, O. Kolb, and J. Lang, 'Fast and reliable transient simulation and continuous optimization of large-scale gas networks', *Math. Methods Oper. Res.*, vol. 95, no. 3, pp. 475–501, Jun. 2022.

[7] A. Zlotnik, K. Sundar, A. M. Rudkevich, A. Beylin, and X. Li, 'Optimal Control for Scheduling and Pricing Intra-day Natural Gas Transport on Pipeline Networks', in *2019 IEEE 58th Conference on Decision and Control (CDC)*, Dec. 2019, pp. 4877–4884.

[8] J. Chang, Q. Fang, X. Qiao, and Q. Yao, 'On the Modeling and Prediction of High-Dimensional Functional Time Series', *J. Am. Stat. Assoc.*, vol. 120, no. 552, pp. 2181–2195, Oct. 2025.

[9] Y. Hu *et al.*, 'Dynamic Multiobjective Optimization Algorithm Guided by Recurrent Neural Network', *IEEE Trans. Evol. Comput.*, vol. 29, no. 5, pp. 1820–1833, Oct. 2025.

[10] G. Rang, B. Xu, W. Li, Z. Fan, and Y. Su, 'A Long Short-Term Memory Prediction-Based Dynamic Multi-Objective Evolutionary Optimization Algorithm', in *2023 IEEE Congress on Evolutionary Computation (CEC)*, Jul. 2023, pp. 1–8.

[11] X. Wang, Y. Zhao, L. Tang, and X. Yao, 'MOEA/D With Spatial–Temporal Topological Tensor Prediction for Evolutionary Dynamic Multiobjective Optimization', *IEEE Trans. Evol. Comput.*, vol. 29, no. 3, pp. 764–778, Jun. 2025.

[12] Y. Kim, S. Kwon, and H. Choi, 'Consistent model selection criteria on high dimensions', *J Mach Learn Res*, vol. 13, no. null, pp. 1037–1057, Apr. 2012.

[13] D. Xu, M. Jiang, W. Hu, S. Li, R. Pan, and G. G. Yen, 'An Online Prediction Approach Based on Incremental Support Vector Machine for Dynamic Multiobjective Optimization', *IEEE Trans. Evol. Comput.*, vol. 26, no. 4, pp. 690–703, Aug. 2022.

[14] W. HU, M. JIANG, X. Gao, K. C. TAN, and Y. Cheung, 'Solving Dynamic Multi-objective Optimization Problems Using Incremental Support Vector Machine', in *2019 IEEE Congress on Evolutionary Computation (CEC)*, Wellington, New Zealand: IEEE, Jun. 2019, pp. 2794–2799.

[15] M. Jiang, Z. Wang, H. Hong, and G. G. Yen, 'Knee Point-Based Imbalanced Transfer Learning for Dynamic Multiobjective Optimization', *IEEE Trans. Evol. Comput.*, vol. 25, no. 1, pp. 117–129, Feb. 2021.

[16] M. Jiang, Z. Wang, S. Guo, X. Gao, and K. C. Tan, 'Individual-based transfer learning for dynamic multiobjective optimization', *IEEE Trans. Cybern.*, vol. 51, no. 10, pp. 4968–4981, 2020.

[17] L. Feng, W. Zhou, W. Liu, Y.-S. Ong, and K. C. Tan, 'Solving Dynamic Multiobjective Problem via Autoencoding Evolutionary Search', *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 2649–2662, May 2022.

[18] X.-F. Liu, X.-X. Xu, Z.-H. Zhan, Y. Fang, and J. Zhang, 'Interaction-Based Prediction for Dynamic Multiobjective Optimization', *IEEE Trans. Evol. Comput.*, vol. 27, no. 6, pp. 1881–1895, Dec. 2023.

[19] Y. Hu *et al.*, 'A Learning Algorithm Based on Similarity Identification and Knowledge Transfer for Dynamic Multi-Objective Optimization', *IEEE Trans. Evol. Comput.*, pp. 1–1, 2025.

[20] J. Zou *et al.*, 'Knowledge Transfer With Mixture Model in Dynamic Multiobjective Optimization', *IEEE Trans. Evol. Comput.*, vol. 29, no. 5, pp. 1517–1530, Oct. 2025.

[21] C. Li, G. G. Yen, and Z. He, 'Diffusion Learning-guided Evolution for Large-scale Dynamic Multi-Objective Optimization', *IEEE Trans. Evol. Comput.*, no. Early Access, pp. 1–1, 2026.

[22] F. Wang, J. Xie, A. Zhou, and K. Tang, 'A New Prediction Strategy for Dynamic Multiobjective Optimization Using Diffusion Model', *IEEE Trans. Evol. Comput.*, vol. 29, no. 5, pp. 1575–1589, Oct. 2025.

[23] C. Li, G. G. Yen, and Z. He, 'Diffusion Meets Evolution: Constraint-Guided Learning for Large-Scale Dynamic Multi-Objective Evolutionary Optimization', in *2025 International Conference on Machine Intelligence and Nature-Inspired Computing (MIND)*, Oct. 2025, pp. 439–445.

[24] W. Song, Z. Liu, J. Yu, X. Sun, Y. Jin, and K. Wee Lai, 'Multisource and Hidden Source-Based Knowledge Transfer for Solving Dynamic Multiobjective Optimization Problems', *IEEE Trans. Evol. Comput.*, vol. 29, no. 5, pp. 1546–1560, Oct. 2025.

[25] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, and X. Wu, 'Unifying Large Language Models and Knowledge Graphs: A Roadmap', *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 7, pp. 3580–3599, Jul. 2024.

[26] B. Romera-Paredes *et al.*, 'Mathematical discoveries from program search with large language models', *Nature*, vol. 625, no. 7995, pp. 468–475, Jan. 2024.

[27] X. Wu, S.-H. Wu, J. Wu, L. Feng, and K. C. Tan, 'Evolutionary Computation in the Era of Large Language Model: Survey and Roadmap', *IEEE Trans. Evol. Comput.*, vol. 29, no. 2, pp. 534–554, Apr. 2025.

[28] X. Wang, M. Feng, J. Qiu, J. Gu, and J. Zhao, 'From News to Forecast: Integrating Event Analysis in LLM-Based Time Series Forecasting with Reflection', Oct. 30, 2024, *arXiv*: arXiv:2409.17515.

[29] S. Lee *et al.*, 'Reasoning Abilities of Large Language Models: In-Depth Analysis on the Abstraction and Reasoning Corpus', *ACM Trans. Intell. Syst. Technol.*, vol. 16, no. 6, pp. 1–52, Dec. 2025.

[30] H. Touvron *et al.*, 'Llama 2: Open Foundation and Fine-Tuned Chat Models', Jul. 19, 2023, *arXiv*: arXiv:2307.09288.

[31] A. Grattafiori *et al.*, 'The Llama 3 Herd of Models', Nov. 23, 2024, *arXiv*: arXiv:2407.21783. doi: 10.48550/arXiv.2407.21783.

[32] E. J. Hu *et al.*, 'LoRA: Low-Rank Adaptation of Large Language Models', Oct. 16, 2021, *arXiv*: arXiv:2106.09685.

[33] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, 'A fast and elitist multiobjective genetic algorithm: NSGA-II', *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

[34] S. Zhang *et al.*, 'Instruction Tuning for Large Language Models: A Survey', *ACM Comput. Surv.*, vol. 58, no. 7, pp. 1–36, May 2026.

[35] S. Jiang, S. Yang, X. Yao, K. C. Tan, M. Kaiser, and N. Krasnogor, 'Benchmark problems for CEC2018 competition on dynamic multiobjective optimisation', *Newctle. Univ.*, p. 18, 2018.

[36] D. C. Hall, 'Sampling random directions within an elliptical cone', *Comput. Phys. Commun.*, vol. 219, pp. 87–90, Oct. 2017.

[37] K. Yu *et al.*, 'A Correlation-Guided Layered Prediction Approach for Evolutionary Dynamic Multiobjective Optimization', *IEEE Trans. Evol. Comput.*, vol. 27, no. 5, pp. 1398–1412, Oct. 2023, doi: 10.1109/TEVC.2022.3193287.

[38] M. Jiang, Z. Wang, L. Qiu, S. Guo, X. Gao, and K. C. Tan, 'A Fast Dynamic Evolutionary Multiobjective Algorithm via Manifold Transfer Learning', *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3417–3428, Jul. 2021.

[39] S. Qin, C. Sun, Y. Jin, Y. Tan, and J. Fieldsend, 'Large-Scale Evolutionary Multiobjective Optimization Assisted by Directed Sampling', *IEEE Trans. Evol. Comput.*, vol. 25, no. 4, pp. 724–738, Aug. 2021.

[40] Y. Zheng *et al.*, 'LlamaFactory: Unified Efficient Fine-Tuning of 100+ Language Models', Jun. 27, 2024, *arXiv*: arXiv:2403.13372.