Quantum-inspired Evolutionary Multitasking with Two-Level Transfer Learning Techniques

Eduardo N. Santiago Ramos
Departamento de Engenharia Elétrica
Pontificia Universidade Católica do Rio de Janeiro
Rio de Janeiro, Brazil
eng.producao.santiago@gmail.com

Alimed Celecia Ramos
Instituto Tecgraf/PUC-Rio
Pontificia Universidade Católica do Rio de Janeiro
Rio de Janeiro, Brazil
alimed@tecgraf.puc-rio.br

Marley Maria Bernardes Rebuzzi Vellasco
Departamento de Engenharia Elétrica
Pontificia Universidade Católica do Rio de Janeiro
Rio de Janeiro, Brazil
marley@ele.puc-rio.br

***Abstract*—This paper aims to explore alternative transfer learning techniques within the quantum-inspired evolutionary multitasking (Q-EMA) framework. We implement several novel transfer strategies, highlighting a two-level approach that performs a transfer between two tasks and later applies a mutation within the same task, adding randomness to the evolutionary process. The experimental results show that both the two-level approach and the mutation approach consistently achieve superior results compared to the baseline framework, with special attention to the gains obtained by the two-level approach on low similarity tasks. Moreover, both strategies converge more rapidly while maintaining competitive computational costs.**

***Index Terms*—Evolutionary Optimization, Evolutionary Multitasking, Transfer Learning, Quantum-inspired Evolutionary Algorithms**

I. Introduction

Evolutionary algorithms (EAs) are stochastic optimization methods inspired by biological evolution, emulating natural selection through a series of generations of candidate solutions [1]. This process aims to balance exploration and exploitation, iteratively selecting individuals better suited to solving a specific task. While this approach has been successful in solving a wide range of problems [2], evolutionary algorithms can also be applied to multiple tasks, leveraging similarities and diversity across search spaces to pursue optimal solutions. The evolutionary process of EAs, however, can be computationally expensive, as a large number of evaluations need to be done needlessly to reach convergence , thus slowly covering the search space in pursuit of the global optimum [1].

To address the limitation of solving tasks independently, *Evolutionary Multitasking* (EMT), also known as Multifactorial Optimization, has emerged as an approach that optimizes multiple related tasks simultaneously [2], [3]. By leveraging shared information and knowledge transfer across tasks, EMT enables faster convergence where genetic material from one task provides beneficial building blocks for solving related problems, exploiting latent commonalities in the search space.

Multitasking evolutionary algorithms have been successfully applied to photovoltaic parameter extraction, wherein single and double diode models [4] are optimized simultaneously [5]. The versatility of these approaches extends to complementary optimization problems, notably feature subset selection [6] [7] and algorithm configuration via hyperparameter tuning [8]. Transfer learning is a key component of EMT, characterized by the transfer of knowledge between tasks at the level of individuals or genes. This transfer expands the search space and allows the algorithm to discover optimal solutions, accelerating the convergence in each single objetive problem [3].

The use of EMTs, however, is limited by the quality of data transferred between tasks. If two tasks are too dissimilar, the transferred genetic material can disrupt the search process and lead to suboptimal solutions, leading to negative transfer, where the transferred population overall reduces the performance of the algorithm [2].

Quantum-Inspired Evolutionary Algorithms (QEAs) address the diversity and convergence limitations of classical EAs by incorporating principles from Quantum Computing [1]. While classical EAs encode the population as a fixed set of values, QEAs represent solutions using quantum-inspired representations (referred to as qubits) that define a probability distribution over possible solutions. These qubits are then collapsed into two states (0 or 1), defining binary solutions for the optimization problem. This approach allows QEAs to maintain a diverse set of individuals that, upon observation (or ”collapse”), can cover a broader range of the search space. These mechanisms help QEAs achieve more extensive search capabilities while also requiring smaller sample sizes to effectively reach convergence [9]. However, the model proposed in [1] exhibits sensitivity to parameter tuning, which when improperly tuned can lead to premature convergence, causing QEAs to become trapped in local optima [10].

QEAs binary representation is not suitable for numerical optimzation problems that would require real representation [11], this representation is introduced in [12], where the binary qubit framework is extended to continuous optimization problems by replacing discrete probability amplitudes with probability distribution functions over each variable's domain. Each variable is represented by a set of N rectangular pulses that collectively form a probability distribution from which candidate solutions are randomly sampled. The algorithm updates these distributions by repositioning pulse centers to the mean values of high-fitness individuals selected through roulette selection, while simultaneously contracting pulse widths exponentially according to $\sigma = (u-l)^{(1-t/T)^{\lambda}} - 1$, where u and l define the domain bounds, t represents the current generation, T denotes total generations, and λ controls the decay rate.

The intersection of evolutionary multitasking and quantum-inspired evolution gives rise to *Quantum-Inspired Evolutionary Multitasking*. [11] introduced the Quantum-inspired Evolutionary Multitasking Algorithm (Q-EMA), which combines quantum-inspired representations and operators with the multitasking framework. Q-EMA employs quantum subpopulations for each task, utilizing quantum-inspired representations to enhance exploration while facilitating information exchange between tasks through a knowledge transfer process. The Q-EMA also leverages the quantum inspired mechanisms to culminate in a convergence of the population in near-optimal regions of the search space faster than it's classical counterpart [1], The original Q-EMA demonstrated its effectiveness by outperforming single-objective approaches in most problems and achieving competitive performance against other state-of-the-art algorithms in multitasking benchmarks such as Griewank, Rastrigin, Ackley, Rosenbrock, Weierstrass, Sphere, and Schwefel [11].

Nevertheless, the quantum-inspired multitasking framework still faces notable constraints regarding *transfer learning*. Empirical investigations in [11] systematically assessed a variety of transfer intervals and rates, demonstrating that optimizing the timing and frequency of knowledge exchange is crucial for exploration. The study further revealed that evolving the transfer rate over time is instrumental in maximizing beneficial information exchange between complementary tasks, while simultaneously minimizing the risk of performance degradation due to negative transfer in incompatible task pairings. Despite these advances, the transfer process in Q-EMA has mostly focused on randomized arithmetic crossover between subpopulations of two different tasks, selecting the best individuals via elitism, and lacks more flexible or adaptive mechanisms for determining optimal transfer partners, timing, and quantity of knowledge exchanged. Importantly, the broader landscape of transfer learning presents additional possibilities, such as the transfer of suboptimal individuals, which may be ill-suited for their source tasks yet beneficial for others, and the intentional incorporation of greater randomness into the transfer process. Such strategies have previously been shown to enrich the search space and bolster the global search capability of evolutionary algorithms, suggesting untapped opportunities to further enhance multitasking optimization [3] [2].

The primary objective of this work is to enhance the transfer learning process within the Q-EMA framework by introducing and evaluating alternative transfer learning methodologies on the same benchmark tasks as in [11], potentially improving its performance and expanding its applicability to a broader range of multitasking optimization problems. Specifically, this article aims to:

- Investigate and implement novel transfer learning techniques that extend beyond the traditional randomized crossover approach used in the original Q-EMA model, including autoencoder-based latent space transfer, multi-armed bandit policy for adaptive transfer rate selection, and learning-based transfer rate optimization.
- Evaluate the effectiveness of a two-level transfer learning approach that combines inter-task knowledge transfer with intra-task refinement mechanisms in a similar vein to a mutation.

II. Methodology

The Q-EMA algorithm aims to address multiple optimization tasks simultaneously by leveraging the expanded search space of quantum-inspired representations and the knowledge transfer mechanisms of multitasking algorithms. This process consists of defining a quantum subpopulation for each task, which is collapsed into a classical population and then evaluated [11]. This subpopulation undergoes classical arithmetic crossover in its own task domain, and the best individuals are retained through elitism.

Afterwards, the subpopulation undergoes a knowledge transfer process at a set number of generations via the *transfer interval* parameter, which involves unifying search spaces and performing arithmetic crossover between pairs of individuals from different tasks, generating an offspring between the two tasks. The resulting individuals are then added to the current subpopulation, with elitism again used to retain the best individuals.

A. Knowledge Transfer

The knowledge transfer process involves exchanging information between tasks [3], enabling the algorithm to leverage knowledge from different tasks and expand the explored search space in directions that would otherwise remain unexplored in single-task optimization. This is done by generating an unified search space to serve as a common domain for comparison between tasks and can range from an interval between $[0, 1]$, where the bounds are defined by the tasks limitations [2] to a learned latent space via autoencoders [13]. In Q-EMA [11], the process begins with the collapse of quantum subpopulations into classical populations. These classical populations are then transformed into a unified common search space via a min-max normalization based on each tasks upper and lower bounds, with individuals appropriately padded to match the maximum dimensionality among all tasks.

Once a unified space is established, the classical pair of subpopulations from each task undergo an arithmetic crossover operation within this shared space, with the proportion of exchanged individuals controlled by the transfer rate parameter. This operation is performed for all pairs of tasks, and the resulting offspring are then transformed back into the original search space of the recipient task. Only individuals that improve upon the current subpopulation are retained, ensuring that knowledge transfer remains beneficial and reducing negative transfer effects.

To further mitigate negative transfer, Q-EMA incorporates an adaptive mechanism that dynamically adjusts the transfer rate based on observed improvements or deteriorations in subpopulation mean fitness after transfer. In the simplest approach, if knowledge transfer proves beneficial, the transfer rate increases by 0.25 to encourage further exchange, if it is detrimental, the rate decreases by 0.05 to limit negative impact. More sophisticated strategies maintain a sliding window of the last 5 transfer rates and compute the new rate as a weighted average of this history, where the weights can be uniform, exponentially decreasing with recency, or following a log-normal distribution to emphasize intermediate past values. Additionally, a predetermined exponential decay strategy reduces the transfer rate over generations regardless of fitness feedback. This feedback enables the algorithm to self-regulate the transfer process, using various strategies for updating the transfer rate [11].

Building upon the baseline Q-EMA knowledge transfer mechanism described in [11], this work introduces enhancements that target different stages of the transfer process. These modifications can be categorized into three complementary frameworks: (1) *unified search space enhancements*, which improve how task-specific solutions are represented in a common space for meaningful comparison and recombination, (2) *adaptive transfer rate mechanisms*, which dynamically adjust the intensity of knowledge exchange based on observed effectiveness, and (3) *transfer process extensions*, which introduce additional transfer operations beyond the standard inter-task crossover.

Unified Search Space Enhancements: **Autoencoder.** We employ an autoencoder based on the approach proposed in [14], where the latent space is used as a unified representation of individuals of both tasks. This approach is characterized by training an autoencoder on the combined subpopulations of all tasks, learning to encode task-specific solutions into a deep neural architecture that is then used to decode them back to their original domains. This process is done via iterating over a random combination of architectures and choosing the one with the smallest loss, as seen in Figure 1.

Transfers occur in this latent space, which is padded or filled with zeros to match the dimensionality of different tasks. The autoencoder is retrained at every transfer interval using the subpopulations of both tasks as input, ensuring that the latent space continually adapts to the current stage of evolution. After transfer, the latent representations are decoded back into their respective task-specific domains.

Adaptive Transfer Rate Mechanisms: **Multi-armed bandit policy.** To improve adaptation of the transfer rate, we incorporated a multi-armed bandit approach based on the methodology

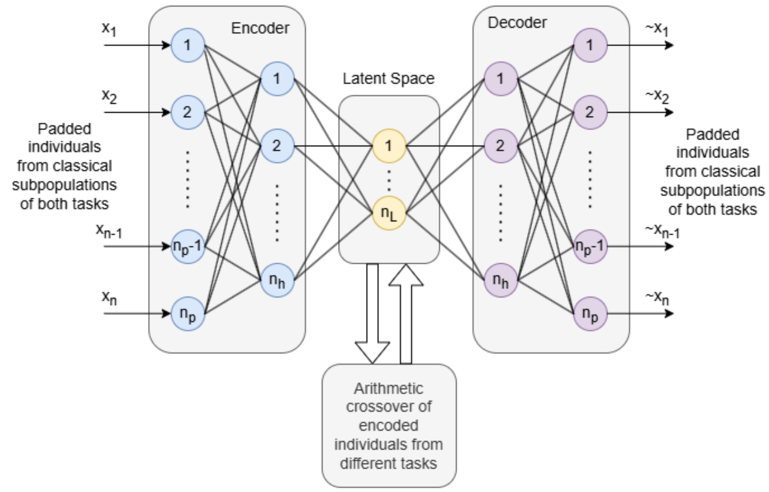

Fig. 1. Architecture of the autoencoder-based unified search space. Padded classical individuals (dimension $n_p = d_{\max}$) are encoded through a hidden layer ($n_h \in [32, 256]$ units) to a latent space ($n_l \in [5, 30]$ dimensions) where arithmetic crossover occurs. The decoder reconstructs the latent representations back to the padded dimension.

outlined in [15]. For each task, the transfer rate is chosen from a discrete set of candidate values: [0.05, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9], which were chosen to represent different variations of rates while not overwhelming the multi-armed bandit, given that the algorithm would need to iteratively reassess the confidence interval of all candidate values. After the first transfer, an initial rate is selected at random. The outcomes of subsequent transfers are tracked alongside confidence intervals for each rate. At every transfer, the rate with the highest upper confidence bound is selected. As a particular rate is tested repeatedly, its confidence interval shrinks, which encourages the exploration of other rates. Over time, the procedure converges through reinforcement towards an optimal transfer rate, thereby enhancing overall fitness.

Learning optimal transfer rate. Another approach that aims to achieve an optimal transfer rate uses incremental learning to update transfer parameters, as proposed in [16], Their approach consisted of training a Naive Bayes classifier to determine whether an individual from an original task would constitute a non-dominated solution for a target task, retraining at every generation to learn which individuals are better suited for transfer. In our approach a LightGBM Regressor model [17] is incrementally retrained on the historical pairings of transfer rate and achieved fitness in order to determine which rate is more adequate for the tasks. At every transfer interval, the combination between transfer rate and mean fitness is stored, after 5 transfers the model is trained with the transfer rate as regressor to determine what value of transfer rate will generate maximum fitness, at every transfer subsequent interval the model is retrained with more data.

Transfer Process Extensions:

Two-level transfer.: The two-level transfer approach [18] is characterized by a first level of actual transfer between tasks, where individuals best suited for another task are transferred to another one and evaluated, generating additional exploration opportunities in the search space. The second level constitutes an intra-task transfer mechanism, which involves transferring genes (dimensions) from different individuals within same task, increasing the randomness of the evolutionary process and potentially reaching unexplored regions of the search space.

Inter-task Transfer: The first level is composed of the already used transfer between two tasks, with the added difference that the individuals that are merged with the current subpopulation to be evaluated via elitism are the top-performing individuals of the donor task. This approach allows the transfer mechanism to introduce solutions from different regions of the search space, beyond those represented by the current best individuals of the recipient task. As a result, it can facilitate the exploration of a broader and more diverse set of candidate solutions.

Intra-task Refinement: The second stage involves searching within the same task by selecting chromosomes, referred to as "dimensions" in [18] and replacing the chromosomes of a recipient individual with values from the same dimension of other individuals.

In our work, we expanded this proposal with four approaches designed to maximize potential gains. This process involves gradually increasing the randomness in the evolutionary process, as suggested in [18] to improve the performance of multitasking algorithms. With these configurations, we are varying the levels of randomness introduced to the transfer process, starting with the default mode, a simplified version of the approach in [18], where only a single gene (dimension) is changed in an individual, followed by the sequential mode, which closely resembles the authors' approach, where multiple genes are changed in the same individual. The greedy mode tests all possible target dimensions and selects those where an improvement is achieved, while the random mode generates completely random individuals.

- **Classical mode:** The default proposal selected two individuals randomly from the same task and swapped the value randomly in a dimension of the donor and the dimension of the recipient, evaluating if a gain of fitness was achieved and keeping it in the new subpopulation. This method aims to generate small disturbances in individuals, and find instances of fitness improvement in regions close to the explored search space.
- **Sequential mode:** This iterates over each dimension of the individual's representation. For each dimension, the candidate gene is substituted with a donor's value. If fitness improves, the substitution is kept, otherwise, the original value is restored. This greedy process proceeds dimension by dimension, allowing cumulative improvement. This mode is most similar to the methodology in [18], and it's evaluation generates more disturbances in these individuals, reaching novel areas of the search space, while still constrained.
- **Greedy mode:** Here, for each target dimension, *all* possible donor substitutions are exhaustively tested, having as the donor individuals all the population besides the recipient. The substitution resulting in the largest fitness improvement is retained for that dimension. This approach is similar to the *default* method, but extensively testing new possibilities aims to achieve the best results within small disturbances.
- **Random mode:** A subset of dimensions is randomly selected and randomly altered. This approach is most similar to a random mutation, where the new gene is chosen from the bounds of the task, without being replaced by any value from a donor individual. This approach aims to reach novel areas of the search space with minimal constraints of the original individual, exploring completely random spaces to find possible improvements.

Combinations: In the experiments, the inter-task transfer and intra-task refinement methods described above were tested both separately and in combination. This allowed us to observe how each method, as well as their combinations, affected the quality of solutions and the improvement patterns throughout the search process.

All other benchmark definitions, task configurations, and default parameters remain identical to the original setup of tables I and II

TABLE I
PARAMETERS OF THE Q-EMA FOR BENCHMARK PROBLEMS (PART 1).

Parameter	CI+HS	CI+MS	CI+LS	PI+HS	PI+MS
Quantum Individuals	[10,10]	[10,10]	[10,10]	[10,10]	[10,10]
Observations	[2,2]	[2,2]	[2,2]	[2,2]	[2,2]
Generations	7500	7500	7500	7500	7500
Transfer Interval	30	50	10	30	50
Transfer Rate	1.0	1.0	1.0	1.0	1.0
Transfer Probability	0.5	0.5	0.5	0.5	0.8
Update Interval	[5,5]	[5,5]	[5,5]	[5,5]	[5,20]
Update Probability	[0.5,0.5]	[0.5,0.5]	[0.5,0.5]	[0.5,0.5]	[0.5,0.5]
Centre Update Intensity	[0.95,0.95]	[0.95,0.95]	[0.95,0.8]	[0.95,0.95]	[0.95,0.95]
Crossover Rate	[0.8,0.8]	[0.8,0.8]	[0.8,0.1]	[0.8,0.8]	[0.8,0.8]

TABLE II
PARAMETERS OF THE Q-EMA FOR BENCHMARK PROBLEMS (PART 2).

Parameter	PI+LS	NI+HS	NI+MS	NI+LS
Quantum Individuals	[10,10]	[10,10]	[10,10]	[10,10]
Observations	[2,2]	[2,2]	[2,2]	[2,2]
Generations	7500	7500	7500	7500
Transfer Interval	50	50	20	20
Transfer Rate	1.0	1.0	1.0	1.0
Transfer Probability	0.5	0.5	0.5	0.5
Update Interval	[10,10]	[20,5]	[5,5]	[5,5]
Update Probability	[0.5,0.5]	[0.5,0.5]	[0.5,0.5]	[0.5,0.5]
Centre Update Intensity	[0.95,0.95]	[0.95,0.95]	[0.95,0.95]	[0.95,0.8]
Crossover Rate	[0.8,0.8]	[0.8,0.8]	[0.8,0.8]	[0.8,0.1]

III. Results

The suite of benchmarks consists of task pairs, each combining two well-known continuous optimization problems (such as Griewank, Rastrigin, Ackley, Rosenbrock, Weierstrass, Sphere, and Schwefel) [19], with varied domains and similarity measures that are widely used in multitasking evolutionary algorithms. Each pair is designed to assess the algorithm's transfer capability across tasks of differing landscape similarity and separability. These combinations will be analyzed looking at the best fitness of 20 iterations of 7500 generations.

a) Fitness Analysis: The results seen in Table III show little meaningful and systematic improvement between the basic methods and the baseline observed. On the other hand, the results in Table IV show that using the two-level technique with default intra-task mutation parameters presents either improvement or comparable results in all algorithms, whereas other variants of two-level add too much randomness to the evolution, not achieving consistently good results. In Tables V and VI, where we observe just the mutation applied to the algorithm with the default transfer learning, the inter-task and inter-task greedy show promising results as well, with comparable or better fitness in all benchmarks.

TABLE III
BEST FITNESS VALUES ACHIEVED FOR EACH BASIC METHOD AND BENCHMARK TASK.

Experiment	Task	autoenc.	bandit	baseline	lightGBM	no transfer
CI+HS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	-6.55E-01
CI+LS	task1	-2.11E+01	-2.10E+01	-2.00E+01	-2.11E+01	-2.11E+01
	task2	-1.99E+03	-1.52E+03	-1.46E+03	-1.42E+03	-1.07E+03
CI+MS	task1	-2.29E-10	-1.73E-10	-1.50E-10	-2.24E-10	-1.87E-10
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	-3.76E+01
NI+HS	task1	-4.53E+01	-4.48E+01	-4.32E+01	-4.55E+01	-4.25E+01
	task2	-9.95E-01	0.00E+00	0.00E+00	0.00E+00	-6.07E+01
NI+LS	task1	-4.48E+01	-5.17E+01	-5.07E+01	-5.67E+01	-5.07E+01
	task2	-1.95E+03	-1.42E+03	-1.30E+03	-2.23E+03	-2.00E+03
NI+MS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	-2.19E+00	-1.75E+00	-2.50E+00	-1.45E+00	-6.69E+00
PI+HS	task1	**-2.89E+01**	-4.58E+01	-4.38E+01	-4.48E+01	-5.47E+01
	task2	-1.56E-19	-1.25E-19	-1.23E-19	-1.47E-19	-1.29E-19
PI+LS	task1	-1.91E-10	-1.22E-10	-1.60E-10	-1.43E-10	-1.89E-10
	task2	-2.13E-06	-3.21E-11	-1.59E-10	-2.03E-10	-7.60E-07
PI+MS	task1	-1.87E-10	-2.14E-10	-1.84E-10	-2.00E-10	-2.20E-10
	task2	-4.47E+01	-4.57E+01	-4.63E+01	-4.52E+01	**-4.34E+01**

b) Ranking Analysis: The ranks summarized in Figs. 2, 3, 4, and 5 show that the **two-level**, **simple intra-task**, and **intra-task greedy** strategies consistently attain the best performance overall. Of particular note, the two-level method secures the best (lowest) ranks, especially in cases of *low similarity* between tasks, reinforcing its robustness in more challenging transfer scenarios. This pattern remains evident when ranking methods by their best individual solutions across all tasks (Fig. 4), again confirming the superiority of these transfer approaches.

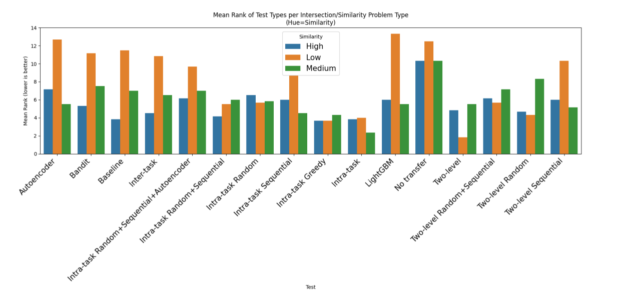

Fig. 2. Mean rank of each method by similarity level. Lower is better.

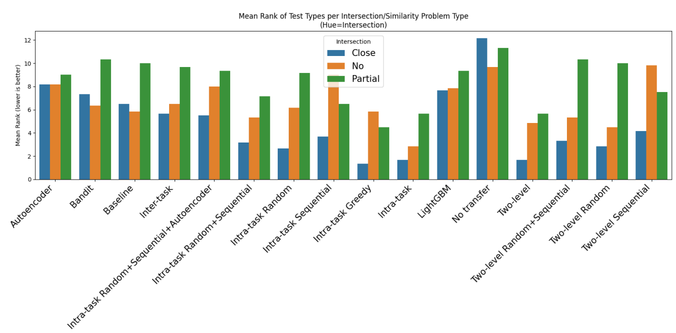

Fig. 3. Mean rank of each method by intersection type. Lower is better.

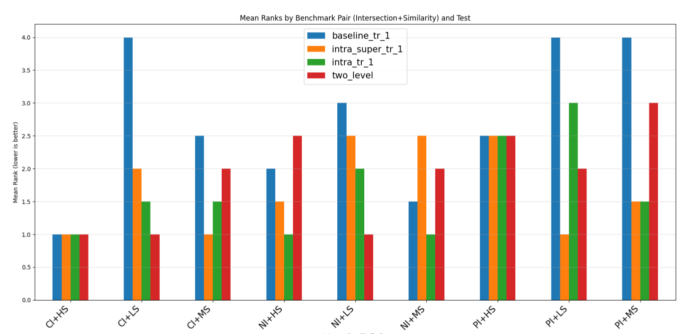

Fig. 4. Rank based on best solution for each method. Lower is better.

c) Evolution Across Benchmark Types: The fitness evolution plots for CI+M, NI+M, and PI+L benchmarks (Fig. 6) indicate that these leading methods two-level, simple intra-task, and intra-task greedy, all achieve very similar results over generations. This convergence in performance demon-

TABLE IV
BEST FITNESS VALUES FOR EACH TWO-LEVEL TRANSFER VARIANT AND BENCHMARK TASK.

Experiment	Task	Baseline	Two-level	Two-level Random+Sequential	Two-level Random	Two-level Sequential
CI+HS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
CI+LS	task1	-2.00E+01	**-1.71E-13**	-2.04E-10	-1.92E-10	-3.31E-06
	task2	-1.46E+03	**-6.36E-04**	**-6.36E-04**	**-6.36E-04**	-6.63E-04
CI+MS	task1	-1.50E-10	-2.72E-11	-1.34E-10	-1.27E-10	-2.01E-11
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
NI+HS	task1	-4.32E+01	-4.49E+01	-4.33E+01	-4.45E+01	-2.03E+01
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	-3.08E+01
NI+LS	task1	-5.07E+01	-4.58E+01	-5.07E+01	**-4.18E+01**	-1.13E+02
	task2	-1.30E+03	**-6.36E-04**	**-6.36E-04**	**-6.36E-04**	-1.18E+03
NI+MS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	-2.50E+00	-3.27E+00	-3.11E+00	-3.28E+00	-8.66E+00
PI+HS	task1	-4.38E+01	-5.27E+01	-5.27E+01	-4.88E+01	-9.65E+01
	task2	-1.23E-19	-3.72E-21	-1.80E-19	-1.23E-19	-2.77E-21
PI+LS	task1	-1.60E-10	-2.58E-12	-1.28E-10	-1.19E-10	-1.16E-10
	task2	-1.59E-10	-2.84E-14	-2.84E-14	-3.55E-14	-7.17E-11
PI+MS	task1	-1.84E-10	-3.29E-11	-2.04E-10	-2.31E-10	-2.38E-11
	task2	-4.63E+01	-4.55E+01	-4.57E+01	-4.58E+01	-4.53E+01

TABLE V
BEST FITNESS VALUES FOR INTRA-TASK AND INTER-TASK TRANSFER VARIANTS (PART 1).

Experiment	Task	Baseline	Inter-task	Intra-task Random+ Sequential+Autoencoder	Intra-task Random+Sequential	Intra-task Random
CI+HS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
CI+LS	task1	-2.00E+01	-2.00E+01	-4.88E+00	-1.85E-10	-1.52E-10
	task2	-1.46E+03	-1.48E+03	-6.37E-04	**-6.36E-04**	**-6.36E-04**
CI+MS	task1	-1.50E-10	-1.20E-10	-1.68E-10	-1.48E-10	-1.31E-10
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
NI+HS	task1	-4.32E+01	-4.48E+01	-4.54E+01	-4.31E+01	-4.47E+01
	task2	0.00E+00	0.00E+00	-9.95E-01	0.00E+00	0.00E+00
NI+LS	task1	-5.07E+01	-4.68E+01	-4.68E+01	-5.67E+01	-6.27E+01
	task2	-1.30E+03	-1.54E+03	**-6.36E-04**	**-6.36E-04**	**-6.36E-04**
NI+MS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	-2.50E+00	-3.09E+00	-3.30E+00	-1.93E+00	-2.18E+00
PI+HS	task1	-4.38E+01	-4.58E+01	-4.08E+01	-5.27E+01	-5.37E+01
	task2	-1.23E-19	-1.22E-19	-1.03E-19	-1.19E-19	-1.56E-19
PI+LS	task1	-1.60E-10	-1.42E-10	-2.14E-10	-1.09E-10	-1.11E-10
	task2	-1.59E-10	-1.33E-10	-1.07E-04	-1.42E-14	-2.84E-14
PI+MS	task1	-1.84E-10	-1.98E-10	-2.13E-10	-2.06E-10	-2.01E-10
	task2	-4.63E+01	-4.58E+01	-4.48E+01	-4.54E+01	-4.55E+01

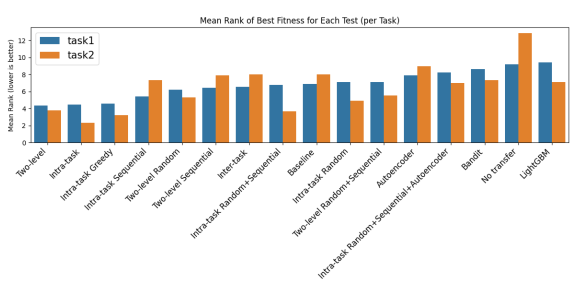

Fig. 5. Mean overall rank for each method. Lower is better.

strates that, regardless of the benchmark characteristics, these methods are able to reach comparable high-quality solutions.

d) Computational Cost Analysis: The computational cost analysis considers the number of evaluations performed per task. For the inter-task method, there are $1.5n$ evaluations per transfer because $n/2$ individuals are selected for transfer in addition to the original n. Table VII focuses on the computational cost for the most effective transfer learning strategies: **two-level**, **intra-task**, and **intra-greedy**. The two-level method consistently requires more fitness evaluations than intra alone, as it combines both inter and intra-task components in each transfer step. However, the number of evaluations for the intra-greedy method is strongly dependent on the problem dimensionality d: in our experiments, $d = 50$, so methods that scale linearly with d must evaluate each gene or perform substitutions for every dimension, significantly increasing their computational demand. This means that, while two-level incurs higher overall cost, above the default intra-

TABLE VI

BEST FITNESS VALUES FOR INTRA-TASK AND INTER-TASK TRANSFER VARIANTS (PART 2).

Experiment	Task	Baseline	Inter-task	Intra-task Sequential	Intra-task Greedy	Intra-task
CI+HS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
CI+LS	task1	-2.00E+01	-2.00E+01	-9.25E-09	-7.68E-13	-2.56E-13
	task2	-1.46E+03	-1.48E+03	-6.37E-04	**-6.36E-04**	**-6.36E-04**
CI+MS	task1	-1.50E-10	-1.20E-10	-1.22E-11	**-2.21E-13**	-2.50E-11
	task2	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
NI+HS	task1	-4.32E+01	-4.48E+01	**-1.37E+01**	-4.06E+01	-3.64E+01
	task2	0.00E+00	0.00E+00	-3.38E+01	0.00E+00	0.00E+00
NI+LS	task1	-5.07E+01	-4.68E+01	-1.04E+02	-1.04E+02	-5.27E+01
	task2	-1.30E+03	-1.54E+03	-9.49E+02	**-6.36E-04**	**-6.36E-04**
NI+MS	task1	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	task2	-2.50E+00	-3.09E+00	-6.22E+00	-6.14E+00	**-7.68E-01**
PI+HS	task1	-4.38E+01	-4.58E+01	-1.15E+02	-9.25E+01	-6.07E+01
	task2	-1.23E-19	-1.22E-19	-1.70E-21	**-2.56E-24**	-2.85E-21
PI+LS	task1	-1.60E-10	-1.42E-10	-3.45E-11	**-2.56E-13**	-4.58E-12
	task2	-1.59E-10	-1.33E-10	-1.78E-13	0.00E+00	-5.68E-14
PI+MS	task1	-1.84E-10	-1.98E-10	-1.62E-11	**-1.78E-13**	-2.31E-11
	task2	-4.63E+01	-4.58E+01	-4.53E+01	-4.54E+01	-4.49E+01

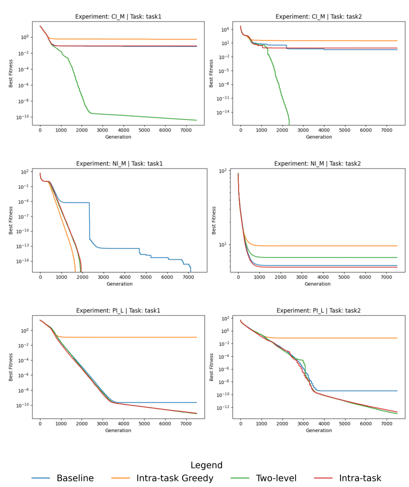

Fig. 6. Evolution of best fitness over generations for CI+M, NI+M, and PI+L benchmarks, respectively.

task transfer, intra-greedy can rapidly approach or even exceed this cost when d is large. Despite these costs, all three methods deliver similar solution quality, highlighting the trade-off between computational effort and performance, particularly for high-dimensional problems.

On Table VII, n denotes the population size, d is the problem dimension, and r is the transfer rate. For sequential greedy mode, the values reported represent worst-case scenarios; in practice, the number of evaluations is often lower due to early stopping.

We applied the non-parametric Friedman test [20] separately to the final best fitness ranks of the **baseline** method and **two-level**, **intra-task**, and **intra-greedy** to determine statistical relevance difference in ranks. On Table VIII we observe significantly different rankings in all experiments beside NI+HS, experiments with ”-” p-value have achieved convergence.

TABLE VII

NUMBER OF FITNESS EVALUATIONS PER METHOD.

Method	Evaluations per Transfer
Baseline	n
Inter-task	$1.5n$
Intra-task	$2rn = 2n$
Intra-task Random	$2rn = 2n$
Intra-task Sequential	$1 + rnd = nd$
Intra-task Greedy	$rn(1 + d) = n(1 + d)$
Intra-task Random+Sequential	$1 + rnd = nd$
Intra-task Random+Sequential+Autoencoder	$1 + rnd = nd$
Two-level	$1.5n + 2rn = 3.5n$
Two-level Sequential	$1.5n + (1 + rnd) = 1.5n + nd$
Two-level Random+Sequential	$1.5n + (1 + rnd) = 1.5n + nd$
Two-level Random	$1.5n + 2rn = 3.5n$

TABLE VIII

FRIEDMAN TEST P-VALUES FOR EACH BENCHMARK AND TASK. A VALUE OF “–” INDICATES THAT ALL METHODS CONVERGED TO IDENTICAL FITNESS VALUES, SO THE TEST STATISTIC IS UNDEFINED.

Experiment	Task	p-value	Significant
CI+HS	T_1	–	No
CI+HS	T_2	–	No
CI+MS	T_1	6.02×10^{-4}	Yes
CI+MS	T_2	1.06×10^{-3}	Yes
CI+LS	T_1	2.44×10^{-9}	Yes
CI+LS	T_2	5.88×10^{-13}	Yes
PI+HS	T_1	1.47×10^{-7}	Yes
PI+HS	T_2	4.99×10^{-10}	Yes
PI+MS	T_1	1.09×10^{-11}	Yes
PI+MS	T_2	7.82×10^{-1}	No
PI+LS	T_1	5.70×10^{-9}	Yes
PI+LS	T_2	3.08×10^{-9}	Yes
NI+HS	T_1	4.10×10^{-1}	No
NI+HS	T_2	4.38×10^{-3}	Yes
NI+MS	T_1	–	No
NI+MS	T_2	7.94×10^{-8}	Yes
NI+LS	T_1	2.61×10^{-8}	Yes
NI+LS	T_2	5.88×10^{-13}	Yes

IV. Conclusions

Overall, the transfer learning strategies evaluated prior to introducing the two-level inter and intra-task method did not achieve satisfactory results. The two-level transfer approach consistently yielded competitive results across the majority of tested scenarios, frequently outperforming or matching other leading solution methods. Its effectiveness stems from its ability to integrate both cross-task exploration and targeted intra-task refinement, which together maintain search balance and direction. Notably, the two-level method demonstrated strong convergence rates across a wide variety of tasks, allowing for the attainment of high-quality solutions more rapidly than many alternative strategies.

Importantly, the advantages of the two-level approach were most pronounced in cases where the similarity between tasks was low. In such settings, other transfer methods often struggled to leverage useful information, whereas the two-level strategy, by combining inter-task and intra-task mechanisms, was able to facilitate meaningful transfer and robust optimization. This highlights the method's adaptability and its particular strength in scenarios involving heterogeneous or weakly related tasks.

Notably, the strongest gains were often observed when using the **intra-greedy** approach for intra-task refinement, which systematically explores all possible donor substitutions for each gene. This exhaustive strategy leads to more substantial improvements than milder, less disruptive forms of transfer. Particularly in some tasks, combining the structured, comprehensive modifications of **intra-greedy** with a controlled degree of randomness in the search process resulted in superior outcomes, as it allowed both deep local optimization and extensive exploration. By contrast, purely inter-task transfer strategies such as **intra-task** lacked both the disruptive search depth of **intra-greedy** and its focused intra-task optimization, and thus performed less effectively. However, it is important to note that while moderate randomness, when guided by systematic search, can be beneficial, excessive or unstructured randomness tends to disrupt promising solutions and reduce overall search efficiency, thereby diminishing the advantages of more structured intra-task transfer.

In summary, the two-level transfer strategy stands out for its robustness and adaptability, especially when coupled with a small, controlled dose of randomness. Excessive or unstructured stochasticity, on the other hand, tends to blunt its effectiveness, highlighting the importance of carefully managing the balance between order and randomness in multifactorial search.

References

[1] K.-H. Han and J.-H. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 6, pp. 580–593, 2002.

[2] Y.-S. Ong and A. Gupta, "Evolutionary multitasking: A computer science view of cognitive multitasking," *Cognitive Computation*, vol. 8, no. 2, pp. 125–142, 2016.

[3] A. Gupta, Y.-S. Ong, and L. Feng, "Multifactorial evolution: Toward evolutionary multitasking," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 343–357, 2016.

[4] S. Li, Q. Gu, W. Gong, and B. Ning, "An enhanced adaptive differential evolution algorithm for parameter extraction of photovoltaic models," *Energy Conversion and Management*, vol. 205, p. 112443, 2020.

[5] S. Li, W. Gong, L. Wang, and Q. Gu, "Evolutionary multitasking via reinforcement learning," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 8, no. 1, pp. 762–775, 2023.

[6] Y. Feng, L. Feng, S. Liu, S. Kwong, and K. C. Tan, "Towards multi-objective high-dimensional feature selection via evolutionary multitasking," *Swarm and Evolutionary Computation*, vol. 89, p. 101618, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210650224001561

[7] J. Lin, Q. Chen, B. Xue, and M. Zhang, "Evolutionary multitasking for multiobjective feature selection in classification," *IEEE Transactions on Evolutionary Computation*, vol. 28, no. 6, pp. 1852–1866, 2024.

[8] S. Huang, J. Zhong, and W.-J. Yu, "Surrogate-assisted evolutionary framework with adaptive knowledge transfer for multi-task optimization," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 4, pp. 1930–1944, 2021.

[9] T. W. Lau, C. Y. Chung, K. P. Wong, T. S. Chung, and S. L. Ho, "Quantum-inspired evolutionary algorithm approach for unit commitment," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1503–1512, 2009.

[10] F. Lv, G. Yang, W. Yang, X. Zhang, and K. Li, "The convergence and termination criterion of quantum-inspired evolutionary neural networks," *Neurocomputing*, vol. 238, pp. 157–167, May 2017.

[11] A. R. Celecia, "Quantum-inspired evolutionary multitasking," Ph.D. thesis, Pontificia Universidade Católica do Rio de Janeiro (PUC-Rio), Rio de Janeiro, Brazil, 2022.

[12] A. V. Abs da Cruz, C. R. Hall Barbosa, M. A. C. Pacheco, and M. Vellasco, "Quantum-inspired evolutionary algorithms and its application to numerical optimization problems," in *Neural Information Processing*, N. R. Pal, N. Kasabov, R. K. Mudi, S. Pal, and S. K. Parui, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 212–217.

[13] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] S. Liu, Q. Lin, L. Feng, K.-C. Wong, and K. C. Tan, "Evolutionary multitasking for large-scale multiobjective optimization," *IEEE Transactions on Evolutionary Computation*, vol. 27, no. 4, pp. 863–877, 2023.

[15] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[16] L. Feng, K. Li, Q. Zhang, and K. C. Tan, "Multiobjective multitasking optimization based on incremental learning," in *Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2019, pp. 447–454, inspiration for the learning-based transfer rate adaptation.

[17] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 3149–3157.

[18] X. Ma, Q. Chen, Y. Yu, Y. Sun, L. Ma, and Z. Zhu, "A two-level transfer learning algorithm for evolutionary multitasking," *Frontiers in Neuroscience*, vol. 13, p. 1408, 2020.

[19] B. Da, Y.-S. Ong, L. Feng, A. K. Qin, A. Gupta, Z. Zhu, C.-K. Ting, K. Tang, and X. Yao, "Evolutionary multitasking for single-objective continuous optimization: Benchmark problems, performance metric, and baseline results," 2017. [Online]. Available: https://arxiv.org/abs/1706.03470

[20] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.