Cognizant multitasking in multiobjective multifactorial evolution : MO-MFEA-II

Bali, Kavitesh Kumar; Gupta, Abhishek; Ong, Yew-Soon; Tan, Puay Siew

2020

Bali, K. K., Gupta, A., Ong, Y. & Tan, P. S. (2020). Cognizant multitasking in multiobjective multifactorial evolution : MO-MFEA-II. IEEE Transactions On Cybernetics, 51(4), 1784-1796.
<https://dx.doi.org/10.1109/TCYB.2020.2981733>

<https://hdl.handle.net/10356/147964>

<https://doi.org/10.1109/TCYB.2020.2981733>

© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at:
<https://doi.org/10.1109/TCYB.2020.2981733>.

Downloaded on 26 Apr 2023 22:05:48 SGT

Cognizant Multitasking in Multi-Objective Multifactorial Evolution: MO-MFEA-II

Kavitesh Kumar Bali, Abhishek Gupta, Yew-Soon Ong, and Puay Siew Tan

Abstract—Humans have the ability to identify recurring patterns in diverse situations encountered over a lifetime, constantly understanding relationships between tasks and efficiently solving them through knowledge reuse. The capacity of artificial intelligence systems to mimic such cognitive behaviors for effective problem solving is deemed invaluable, particularly when tackling real world problems where speed and accuracy are critical. Recently, the notion of *evolutionary multitasking* has been explored as a means of solving multiple optimization tasks simultaneously using a single population of evolving individuals. In the presence of similarities (or even partial overlaps) between high quality solutions of related optimization problems, the resultant scope for inter-task genetic transfer often leads to significant performance speedup - as the cost of re-exploring overlapping regions of the search space is reduced. While multitasking solvers have led to recent success stories, a known shortcoming of existing methods is their inability to adapt the extent of transfer in a principled manner. Thus, in the absence of any prior knowledge about the relationships between optimization functions, a threat of predominantly *negative (harmful) transfer* prevails. With this in mind, the current paper presents a realization of a *cognizant evolutionary multitasking engine* within the domain of multi-objective optimization. Our proposed algorithm learns inter-task relationships based on overlaps in the probabilistic search distributions derived from data generated during the course of multitasking - and accordingly adapts the extent of genetic transfers *online*. The efficacy of the method is substantiated on multi-objective benchmark problems as well as a practical case study of knowledge transfers from low fidelity optimization tasks to substantially reduce the cost of high fidelity optimization.

Index Terms—Evolutionary Multitasking, multifactorial optimization, probabilistic modeling, online similarity learning, multi-objective optimization.

I. INTRODUCTION

The human brain is a self-learning system. Among various other things, it has the general capability of comprehending its surroundings - being “*aware of what is going on*” and “*figuring out*” what to do [1, 2]. This intrinsic feature enables humans to draw inferences from what one experiences,

This work was supported in part by the A*STAR Cyber-Physical Production Systems research project under IAF-PP Grant A19C1a0018, and in part by the Singapore Institute of Manufacturing Technology-Nanyang Technological University (SIMTech-NTU) Joint Laboratory and Collaborative Research Programme on Complex Systems. (*Corresponding author: Kavitesh Bali.*)

Kavitesh Kumar Bali is a doctoral candidate at the School of Computer Science and Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798 (e-mail: bali0001@e.ntu.edu.sg).

Abhishek Gupta and Puay Siew Tan are with the Singapore Institute of Manufacturing Technology (SIMTech), Agency for Science, Technology and Research (A*STAR), Singapore 138634 (e-mail: ABHISHEK_GUPTA, pstan@simtech.a-star.edu.sg).

Yew-Soon Ong is with the Data Science and Artificial Intelligence Research Centre, School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: asysong@ntu.edu.sg) and also with the Agency for Science, Technology and Research (A*STAR), Singapore 138632.

recognize recurring patterns (relationships) between diverse tasks, and share knowledge across related problems [3]. It is indeed this ability to leverage on what we already know to tackle related tasks that plays a significant role in boosting our efficiency of problem-solving and decision-making [4].

The ability to learn from experience is considered invaluable for machines as well. Given modern technologies with large-scale data storage and seamless communication facilities, the development of machines that mimic human-like behaviors of autonomously extracting, processing and transferring learned knowledge across related problems is surely an attractive proposition. In this regard, the concept of *multitasking* has drawn much interest in the field of computational intelligence [5–9]. Not surprisingly, present day artificial intelligence systems are emerging as powerful tools for multitasking, capable of simultaneously handling myriad tasks with greater speed and accuracy.

Evolutionary multitasking, a relatively new paradigm in the field of evolutionary computation, has recently been explored as a population-based search methodology for solving multiple optimization problems concurrently [7]. Notably, in the presence of underlying relationships between constituent problems, evolutionary multitasking leads to the possibility of fruitful knowledge transfers, thereby facilitating improved convergence characteristics. Many success stories have thus surfaced in recent years, encompassing the domains of discrete, continuous, single- and multi-objective optimization, including various applications in machine learning [10–22]. The practicality of evolutionary multitasking has also been demonstrated across a wide array of real-world applications spanning manufacturing process design [14], robot controller design [23], vehicle routing [4, 7], cloud computing [24] and software engineering [25], to name a few.

While practitioners have many algorithmic options to choose from, it is noted that the applicability and success of present day multitasking solvers depends strongly on task relatedness. For those cases where optimization problems share little in common, blindly transferring knowledge (in the form of encoded solutions) across them can lead to predominantly negative (harmful) effects [14, 26, 27]. Susceptibility to negative inter-task interactions has in fact been shown to impede overall convergence behavior [23]. Existing evolutionary multitasking algorithms such as the *multi-objective multifactorial evolutionary algorithm* (MO-MFEA) [14], generally lack the cognitive capability of *situational awareness* [2], and thus, are unable to decipher and adapt to the degree of similarity between distinct tasks on the fly.

In light of the above, a matter of increasing importance has

been the design of algorithms that are capable of processing incoming streams of data from multiple optimization tasks, and identifying relationships between them. Accordingly, in this paper, we present a *cognizant* evolutionary multitasking approach, labeled as MO-MFEA-II, which is adept at *on-line* inter-task similarity learning and adaptive transfer. The MO-MFEA-II incorporates probabilistic modeling of the data generated online during the multitasking search to unveil recurring solution patterns (measured by the similarity in search distributions) across different tasks. The proposed *data-driven* approach showcases *awareness* as to *when* and *how much* knowledge is to be transferred - much like the conscious human mind that decides which tasks to focus on and when to suppress irrelevant information [28]. We note that the foundations of MO-MFEA-II encompass ideas described in a recently introduced *multifactorial evolutionary algorithm with online learning capability* (MFEA-II) [23]. While MFEA-II focusses purely on single-objective optimization, the MO-MFEA-II proposed herein analyzes how similar concepts can apply to the domain of multi-objective optimization as well. Thus, to the best of our knowledge, this paper is among the first to utilize probabilistic modeling to capture inter-task relationships between multi-objective optimization tasks in the context of evolutionary multitasking.

The efficacy of MO-MFEA-II is showcased first on a series of synthetic test functions. Further, we highlight a natural and practically useful application of the approach to the domain of multi-fidelity optimization; i.e., leveraging on low fidelity function approximations to accelerate the optimization performance of related high fidelity tasks via multitasking.

The organization of the remainder of this paper is as follows. Section II presents the backgrounds of multi-objective optimization and associated multi-objective evolutionary algorithms. In addition, we provide a brief review of related works in the literature and summarize the existing MO-MFEA framework. The MO-MFEA-II with online transfer adaptation capability is presented in Section III. Experimental results and analyses on a wide range of multi-objective benchmarks and a practical case study are provided in Sections IV and V, respectively. Section VI concludes the paper with a discussion on future works.

II. BACKGROUND

In this section, we first present preliminaries of multi-objective optimization and associated single-task evolutionary algorithms. Next, a brief review of the recent advances in evolutionary multitasking, particularly in the domain of multi-objective optimization is presented. The outlined contributions of this paper are compared and contrasted against those available in the literature. Finally, the existing MO-MFEA is discussed and its potential limitations highlighted.

A. Multi-objective Optimization

The goal of solving multi-objective optimization problems (MOOPs) is to find multiple trade-off solutions that offer an optimal compromise between conflicting objectives of interest.

In what follows, we provide a general formulation for a MOOP.

Given a decision space $\mathbf{X} \subset \mathbb{R}^D$, a multi-objective minimization problem can be defined (without loss of generality) as follows:

$$\text{minimize } F(x) = (f_1(x), f_2(x), \dots, f_M(x)), \quad (1)$$

where $f_1(x), f_2(x), \dots, f_M(x)$ are M different minimization objectives, $F(x)$ is the overall objective vector, and $x \in \mathbf{X}$ is a decision vector. For the sake of brevity, additional constraint functions have been suppressed in Eq. (1). Following the concept of *Pareto dominance* [14], a solution x_1 is said to *dominate* another solution x_2 (i.e. $x_1 \prec x_2$) iff $\forall i \in \{1, 2, \dots, M\}: f_i(x_1) \leq f_i(x_2)$, and $\exists j \in \{1, 2, \dots, M\}: f_j(x_1) < f_j(x_2)$. Accordingly, a solution x^* is considered to be Pareto optimal if x^* is not dominated by any other solution in $\mathbf{X}$. The objective function values corresponding to all the Pareto optimal solutions collectively form the Pareto front [29].

B. Multi-objective Evolutionary Algorithms (MOEAs)

Evolutionary algorithms (EAs) are considered a popular choice for solving MOOPs. In particular, the implicit parallelism offered by a population enables simultaneous sampling, evaluation and processing of multiple regions of the search space, leading to a reasonable representation of the entire set of Pareto optimal solutions to be found in only a single run of the solver [9, 26]. In the literature, some examples of Multi-objective Evolutionary Algorithms (MOEAs) that are commonly used today include NSGA-II [30], NSGA-III [31, 32], SPEA2 [33] and [14], MOEA/D [34–36], to name a few. While a plethora of MOEAs have been proposed over the years, the focus has mostly been to efficiently solve only one MOOP at a time. On the other hand, attempts to tackle multiple related MOOPs simultaneously (in the spirit of evolutionary multitasking) have been relatively scarce.

C. Evolutionary Multitasking in Multi-objective Optimization

Since the conceptualization of evolutionary multitasking, associated algorithmic developments have been seen in the domain of multi-objective optimization as well [14, 37]. In particular, a handful of methods equipped with adaptive knowledge transfer capabilities have also come to the fore. For example, Feng *et. al.* [8] proposed to learn optimal linear mappings between different continuous multi-objective tasks using a denoising autoencoder. In this method, the learned mappings serve as a bridge between tasks, providing a transformation of search spaces (and solutions) such that adaptive knowledge transfers can be conducted.

Further, in [38], a credit assignment-based technique was employed to adapt knowledge transfers between tasks in a manner that promotes exploration, thereby preventing solutions from getting trapped in local optima. More recently, Lin *et. al.* [21] have utilized incremental Naive Bayes classifiers to select valuable solutions to be transferred during the multi-tasking search. In their proposed method, the authors also use a randomized mapping that enhances the exploration capacity

of transferred solutions, with the aim of improving overall convergence behavior.

While there has been growing interest in designing adaptive multitask solvers, many existing methods are designed based on prior assumptions of task relatedness, such that solutions from a source may be directly transferable to a target task. Some other methods have also been proposed that learn a mapping between tasks with the goal of inducing high (ordinal) correlation between their respective objective functions [27]. The learned mappings are then applied to transferred solutions, increasing the likelihood of them being beneficial to the target task. Nevertheless, it is noted that existing methods typically do not provide any explicit measures to guard against the threat of negative transfer; for example, when the learned mapping is inaccurate. *In this regard, the key novelty of our proposed algorithm is that it takes a data-driven approach to explicitly guard against negative transfers.* Our method is based on the simple idea that by learning the extent to which search distributions of different tasks overlap, the amount of knowledge to be transferred between them can be adapted.

D. Overview of the Existing MO-MFEA

The MO-MFEA [14] is a recently proposed method that allows efficient resolution of multiple MOOPs simultaneously by promoting *omnidirectional* knowledge transfers [10]. By this we mean that all tasks can benefit from each other via mutual knowledge sharing. For this to be possible, a *unified search space* encoding solutions to all constituent optimization tasks is first defined. At least in the case of box-constrained continuous optimization, such unification is achieved in a straightforward manner by a simple linear map between the task-specific solution space and the unified space [39] - which is also continuous and is defined in the range $[0, 1]^D$.

In the MO-MFEA, the primary mode of knowledge transmission is through implicit genetic transfers during *inter-task crossovers* between parent solutions belonging to different tasks. Essentially, the extent of genetic transfer is mandated by a user-defined and fixed scalar transfer parameter defined as the *random mating probability (rmp)*. For further details about the MO-MFEA, the reader is referred to [14].

1) *Tuning the rmp parameter:* It is noteworthy that the *rmp* parameter in MO-MFEA is *manually* specified based on the intuition of a decision maker. That is, if the optimization tasks are known to have a high degree of inter-task similarity, the *rmp* is usually set to a value close to 1. On the other hand, a low *rmp* value (closer to 0) is prescribed between optimization tasks that are known to bear low inter-task similarity. It is indeed patently clear that such an *offline rmp assignment* scheme is heavily dependent on the existence of prior knowledge in the mind of the practitioner about the relationships between different optimization problems. In practical settings, access to domain specific knowledge about different optimization problems is often limited, particularly in general black-box optimization. Given the possible lack of prior knowledge about inter-task relationships, the appropriate selection of *rmp* can become a significant hurdle for a practitioner. In fact, it has also been shown that inappropriate (blind) prescription of *rmp*

values (via trial and error) risks the possibility of harmful genetic transfers, thereby leading to significant performance slowdowns [23].

III. COGNIZANT MULTITASKING MO-MFEA-II

This section provides details of the proposed MO-MFEA-II, which offers a data-driven and cognitive enhancement of its predecessor MO-MFEA. Specifically, we would want the algorithm to be able to automatically decide *when* and *how much* knowledge to transfer such that the threat of negative transfers is minimized. To this end, the MO-MFEA-II learns the *rmp* parameter *online* (in place of an *offline rmp* assignment) and thus adapts the extent of genetic transfers between diverse tasks in a multitask setting. In order to develop an online *rmp* estimation technique with knowledge transfer adaptation capabilities, two key ideas have been incorporated in the existing MO-MFEA. These include:

- Utilizing a *RMP* matrix in place of a scalar *rmp* to effectively multitask across more than two tasks with possibly diverse inter-task relationships. For example, given three different optimization tasks, the degree of similarity between the first and the second task maybe different from that between the first and the third. Therefore, using the same *rmp* value for all task pairs is considered to be too restrictive.
- Defining the MO-MFEA framework from a *probabilistic modeling perspective*, so as to allow capturing of inter-task similarities based on the quantification of overlaps in search distributions. The MO-MFEA-II is equipped to model the data generated online during the course of a multitasking search to unveil recurring solution patterns across different tasks.

A. RMP matrix

Given K optimization tasks in a multitask setting, the *RMP* takes the form of a symmetric $K \times K$ matrix of pairwise *rmp* values represented as,

$$RMP = \begin{bmatrix} rmp_{1,1} & rmp_{1,2} & \cdot & \cdot \\ rmp_{2,1} & rmp_{2,2} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}, \quad (2)$$

where $rmp_{k,j} = rmp_{j,k}$ reflects the extent of transfer between the k^{th} and j^{th} task-pair. Note that $rmp_{k,k} = 1, \forall k$. Unlike the scalar *rmp* of the present day MO-MFEA, the *RMP* matrix offers the distinct advantage of adapting the extent of knowledge transmissions between diverse task-pairs with possibly non-uniform inter-task similarities.

B. Latent Search Distributions in the MO-MFEA

Here, we briefly describe how knowledge transfers are actualized in the original MO-MFEA, through the *implicit mixture of search distributions drawn from different tasks*.

For $k \in \{1, 2, \dots, K\}$, consider the parent subpopulation associated with the k^{th} task T_k to be denoted as $P^k(t)'$ at iteration t . Further, let the subpopulations associated with each of the K tasks originate from the *true* (but unknown) underlying

probability distributions $p^1(x, t), p^2(x, t), \dots, p^K(x, t)$ respectively, i.e., $P^k(t)' \sim p^k(x, t)$ for all $k \in \{1, 2, \dots, K\}$.

As a consequence of the *inter-task* interactions in the MO-MFEA (through inter-task crossovers), the offspring population generated for the k^{th} task at the t^{th} iteration can be shown to be drawn from the following mixture distribution, under the simplifying assumption of *parent-centric* evolutionary operations (see supplemental material of [23]).

$$p_c^k(x, t) = \left[1 - \frac{0.5}{K} \cdot \sum_{k \neq j} rmp_{k,j} \right] \cdot p^k(x, t) + \frac{0.5}{K} \sum_{j \neq k} rmp_{k,j} \cdot p^j(x, t). \quad (3)$$

The finite mixture $p_c^k(x, t)$ is a linear combination of all K task distributions, with the degree of mixing given by $rmp_{k,j}$'s. Note that $0 \leq rmp_{k,j} \leq 1$. Clearly, the most trivial way to prevent any harmful (negative) inter-task interactions is to simply set the $rmp_{k,j}$ values to 0, leading to the cancellation of all knowledge transfers. In this case, Eq. (3) implies that $p_c^k(x, t) = p^k(x, t)$.

C. Preliminaries on Probabilistic Model-based Search

For an ideal multitasking engine, the goal is to promote fruitful transfers whenever possible (i.e., by learning positive pairwise rmp values) while also minimizing negative inter-task interactions. The proposed idea is thus to leverage on the data generated during the course of a multitask optimization run, and to learn an *RMP* matrix that leads to an effective mixture distribution in Eq. (3). For instance, if the search distributions of the respective tasks are far from each other in a unified search space, such that no relationship can be inferred between them, we would want the corresponding pairwise rmp values to be automatically very low (near zero). On the other hand, if the search distributions are overlapping, higher rmp values should ideally be prescribed. Such a learning strategy is expected to benefit related tasks while not hurting the performance of unrelated tasks.

In order to do this learning, we draw from existing theories in the field of *probabilistic model-based search algorithms*. These algorithms guide the optimization search by iteratively building probabilistic models of parent solutions, and subsequently generating new offspring by sampling these models. The typical steps can be summarized as follows.

- **Step 1:** Start with a uniform distribution model $q(x, 0)$ over the search space
- **Step 2:** At iteration t , sample N offspring solutions from the current model $q(x, t-1)$
- **Step 3:** Based on the fitness of sampled solutions, *select* $n < N$ parent solutions
- **Step 4:** Build a probabilistic model $q(x, t)$ of the parent solutions
- **Step 5:** Increment iteration count and go to Step 2, unless a termination condition is satisfied.

A well known result which supports this class of algorithms is summarized in Theorem 1. Briefly, the result highlights the benefit of learning probabilistic models $q(x, t)$ that capture the true underlying parent distribution $p(x, t)$ accurately, in order to attain asymptotic convergence guarantees. It is this theorem that also serves as the basis of the mathematical program

proposed in the paper to learn the *RMP* matrix online, as shall be detailed in Section (III-E).

Theorem 1. *Consider a continuous optimization task T with objective function f and global optimum f^* . The prior distribution $q(x, 0)$ from which initial solutions are sampled is assumed to be positive and continuous everywhere in the search space $\mathbf{X}$, with $N \rightarrow \infty$. With this, at every iteration “ t ” of a probabilistic model-based EA, if the learned model $q(x, t)$ – from which offspring are sampled – is identical to the true parent distribution $p(x, t)$, then the search asymptotically converges to a distribution $p(x, \infty)$ such that $\int_{\mathbf{X}} f(x) \cdot p(x, \infty) \cdot dx = f^*$.*

The proof of the above can be found in [40]. We note that this result originally pertains to the case of single-objective optimization, and has recently been used in [23] for learning optimized mixture models in the context of multitasking with single-objective optimization tasks. In contrast, in this paper, we investigate how incorporation of the same result can lead to performance gains in the MO-MFEA as well. Details of the resultant MO-MFEA-II are presented next.

D. Probabilistic Mixture Modeling in MO-MFEA-II

At any iteration t of MO-MFEA-II, we start by building probabilistic models $q^k(x, t)$ for parent subpopulations $P^k(t)' \in \{P^1(t)', P^2(t)', \dots, P^K(t)'\}$. The model $q^k(x, t)$ is seen as a limited data approximation of the *true* parent distribution $p^k(x, t)$. Given the learned models above, Eq. (3) can equivalently be written as,

$$q_c^k(x, t) = \left[1 - \frac{0.5}{K} \cdot \sum_{k \neq j} rmp_{k,j} \right] \cdot q^k(x, t) + \frac{0.5}{K} \sum_{j \neq k} rmp_{k,j} \cdot q^j(x, t), \quad (4)$$

where $q_c^k(x, t)$ is a mixture model approximating the *true* $p_c^k(x, t)$ in Eq. (3).

From Theorem 1, we know that in this probabilistic interpretation, asymptotic convergence is guaranteed in the limit $N \rightarrow \infty$ if the offspring distribution is equal to the parent distribution. This result suggests that learning the *RMP* matrix such that mixture model $q_c^k(x, t)$ in Eq. (4) accurately replicates $p^k(x, t)$, for all $k \in \{1, 2, \dots, K\}$, shall implicitly decrease the tendency of negative inter-task interactions.

E. Learning the RMP Matrix Online in MO-MFEA-II

To this end, given the probabilistic models $q^k(x, t)$ for all tasks $k \in \{1, 2, \dots, K\}$, the following mathematical program is formulated to learn the *RMP* matrix,

$$\max_{RMP} \sum_{k=1}^K \sum_{i=1}^n \log q_c^k(x_{ik}, t), \quad (5)$$

where x_{ik} is the i^{th} sample (individual) in parent subpopulation $P^k(t)'$. The rationale behind this formulation can be seen from the following theorem.

Theorem 2. *Maximizing the likelihood function in Eq. (5), in the limit $n \rightarrow \infty$, is equivalent to minimizing the Kullback-Leibler divergence $KL(p^k(x, t) || q_c^k(x, t))$ averaged across all tasks $\{T_1, T_2, \dots, T_k, \dots, T_K\}$.*

The proof of the above can be found in [23], where the Kullback-Leibler divergence is used as a standard measure of the discrepancy between probability distributions. With regards to the computational tractability of optimizing the RMP matrix, we note that the mathematical program in Eq. (5) is convex upward. Therefore, in practice, the RMP matrix can be learned at little computational overhead (as long as K is not large) using classical convex optimization solvers.

It is also worth highlighting the importance of the type of constituent probabilistic models $q^k(x, t)$ to be chosen for approximating the parent distributions $p^k(x, t)$. If we choose to build a complex (highly expressive) model on a typically small subpopulation dataset $P^k(t)'$, then this may cause model *overfitting* - which in turn can be shown to result in the cancellation of inter-task knowledge transfers in Eq. (4) as the trivial solution of $rmpr_{k,j} = 0 \forall k \neq j$ optimizes the program in Eq. (5). To alleviate this overfitting issue, we propose to use simple (less expressive) and fast probabilistic models, e.g., univariate marginal distributions ignoring variable dependencies, to approximate the parent distributions; thus, allowing all other models in the multitask setting to be mobilized to fill in the distribution gaps and actualize knowledge transfers. An additional step that actively prevents the cancellation of inter-task transfers is to inject some random noise into the available subpopulation datasets. Specifically, a simple modification that is found to work well in practice is to add a small amount of uniformly sampled noise to the dataset $P^k(t)' \forall k$, resulting in a corrupted version $P^k(t)''$. Thereafter, building the model $q^k(x, t)$ on the corrupted dataset $P^k(t)''$ naturally prevents overfitting to the original dataset $P^k(t)'$ [41].

F. Summary of the MO-MFEA-II

Given all the enhancements above, we can now summarize the MO-MFEA-II algorithm.

1) *Preliminary definitions*: Consider the case of solving K multi-objective optimization tasks $\{T_1, T_2, \dots, T_K\}$ concurrently using a single population P of evolving individuals, encoded in a unified space (denoted as $\mathbf{Y}$). Let the subpopulation associated with the k^{th} task be denoted as P^k . An approach for comparing the fitness of candidate solutions in such a multitask setting is formulated by defining the following properties for each individual in P .

- **Definition 1 (Skill Factor)**: The *skill factor* τ_i corresponds to the one task (out of the available K tasks) the i^{th} individual is associated with in the multitasking environment.
- **Definition 2 (Scalar Fitness)**: The *scalar fitness* of the i^{th} individual is defined as $\varphi_i = 1/r_{\tau_i}^i$, where $r_{\tau_i}^i$ corresponds to the rank of the i^{th} individual on task τ_i . Note that $\max\{\varphi_i\} = 1$.

2) *Algorithm summary*: Algorithm 1 provides the pseudocode for MO-MFEA-II. The algorithm is built upon the *Non-dominated Sorting Genetic Algorithm* (NSGA-II) [30], which utilizes *lexicographic ranking* based on each candidate solution's *non-dominated front* (NF) and *crowding distance* (CD) to guide the population as a whole through complex multi-objective search spaces. For the sake of brevity, further details of the NSGA-II have not been reproduced in this paper.

Algorithm 1: Pseudocode of the MO-MFEA-II

```

1 Randomly sample  $N \cdot K$  individuals in  $\mathbf{Y}$  to form initial population  $P(0)$ 
2 for every individual  $p_i$  in  $P(0)$  do
3   Assign skill factor  $\tau_i = \text{mod}(i, K) + 1$ , for the case of  $K$  tasks
4   Evaluate  $p_i$  for task  $\tau_i$ 
5 Compute scalar fitness  $\varphi_i \forall p_i$  via lexicographic ranking of  $NF_i, CD_i$ 
6 Set  $t = 1$ 
7 while stopping conditions are not satisfied do
8   Configure offspring population  $P_c(t) = \emptyset$ 
9    $P(t)' = \text{Tournament Selection of } n \cdot K \text{ parent solutions from } P(t)$ 
10  Learn  $RMP(t)$  as per Eq. (5)
11  while offspring generated for each task  $< N$  do
12    Sample two individuals  $x_i$  and  $x_j$  at random from  $P(t)'$ 
13    if  $\tau_i == \tau_j$  then
14       $[x_a, x_b] \leftarrow \text{Intra-task crossover} + \text{mutate}(x_i, x_j)$ 
15      Assign offspring  $x_a$  and  $x_b$  skill factor  $\tau_i$ 
16    else if  $\text{rand} \leq \text{rmpr}_{\tau_i, \tau_j}$  then
17       $[x_a, x_b] \leftarrow \text{Inter-task crossover} + \text{mutate}(x_i, x_j)$ 
18      Each offspring is randomly assigned skill factor  $\tau_i$  or  $\tau_j$ 
19    else
20      Randomly select  $x'_i$  with skill factor  $\tau_i$ 
21       $[x_a] \leftarrow \text{Intra-task crossover} + \text{mutate}(x_i, x'_i)$ 
22      Assign offspring  $x_a$  skill factor  $\tau_i$ 
23      Randomly select  $x'_j$  with skill factor  $\tau_j$ 
24       $[x_b] \leftarrow \text{Intra-task crossover} + \text{mutate}(x_j, x'_j)$ 
25      Assign offspring  $x_b$  skill factor  $\tau_j$ 
26    Evaluate  $[x_a, x_b]$  for their assigned skill factors only
27     $P_c(t) = P_c(t) \cup [x_a, x_b]$ 
28   $R(t) = P_c(t) \cup P(t)$ 
29  Update scalar fitness of all individuals in  $R(t)$ ;
30  Select  $N \cdot K$  fittest members from  $R(t)$  to form  $P(t+1)$ 
31   $t = t + 1$ 

```

The MO-MFEA-II evolves a single population of $N \cdot K$ individuals encoded in a unified space $\mathbf{Y}$. A uniform resource allocation scheme is adopted whereby each task $T_k \in \{T_1, T_2, \dots, T_K\}$ is equally assigned N individuals to comprise its subpopulation P^k . Lines 7 – 31 in Algorithm 1 form the main evolutionary multitasking loop, describing the steps associated with offspring generation, task-specific fitness evaluation and environmental selection.

For offspring creation, a pool of parent candidates $P(t)'$ is first selected through binary tournament selection (line 9). The selection is based on the scalar fitness of each of the individuals such that $P^1(t)' \cup P^2(t)' \cup \dots, P^K(t)' = P(t)'$. The selected parent subpopulations are then used to build probabilistic models and learn the RMP matrix according to Eq. (5). Notice that the RMP matrix is currently learned and adapted online at every iteration t (line 10).

Apart from the conventional *intra-task crossover* between parent solutions belonging to the same task (lines 13 – 15), *inter-task crossover* (lines 16 – 18) also takes place between parent solutions belonging to different tasks (i.e., parents having different skill factors). In particular, the *extent* of implicit genetic transfer in the MO-MFEA-II is governed by the pairwise $rmpr$ values in the learned RMP matrix - which mandates the probability of inter-task crossover between two solutions x_i and x_j that may have different skill factors. Clearly, if the learned RMP is the identity matrix, then no knowledge transfer will be actualized.

3) *Selective evaluations in MO-MFEA-II*: Each of the offspring individuals is evaluated with respect to only one task (its assigned skill factor) amongst all other tasks in the multitasking environment (line 26). This salient feature combats the issue of otherwise exhaustively evaluating every individual for

every task, which would often be computationally demanding. Post evaluation, the fittest $N \cdot K$ individuals survive for the next iteration of the evolutionary search (line 30).

The aforementioned steps (lines 7 – 31 in Algorithm 1) repeat until some stopping condition is met.

IV. EXPERIMENTAL STUDY

In this section, experimental results are presented that show-case the efficacy of the proposed MO-MFEA-II in the domain of multi-objective optimization. The performance of MO-MFEA-II is compared against its predecessor, the MO-MFEA and its baseline algorithm, i.e., NSGA-II performing a single task at a time. For the case of synthetic benchmarks tackling two MOOPs at a time (Sections IV-D1 & IV-D2), the *evolutionary multitasking via explicit autoencoding* (EMEA) [8] algorithm is considered as an additional state-of-the-art multitasking baseline for comparison. Essentially, the EMEA leverages on the search biases from two different MOEAs (i.e., NSGA-II & SPEA2) acting upon two distinct tasks, respectively. During knowledge transfers, candidate solutions are first transformed via a linear mapping before being transferred from one task to the other.

A. Experimental Configuration

The experimental setup is outlined in what follows. To ensure consistency, the population size N (per task) is kept the same for all algorithms. For all the test problems under consideration, the following general settings are applied:

- 1) Continuous unified search space $[0, 1]^D$;
- 2) Evolutionary operators used for MO-MFEA-II, MO-MFEA, and NSGA-II are:
 - a) SBX crossover with probability $(p_c) = 1$ and distribution index $\eta_c = 10$;
 - b) Polynomial mutation with probability $(p_m) = 1/D$ and distribution index $\eta_m = 10$;
- 3) Probability model in MO-MFEA-II: Normal distribution;
- 4) Population size (N):
 - a) For synthetic benchmarks: 50;
 - b) For practical study: 20;
- 5) Maximum number of generations: 250;
- 6) *RMP* is learned online for MO-MFEA-II. For MO-MFEA, *rmp* values set as either 0.9, or 0.6, or 0.3;
- 7) Transfer interval in the EMEA: 10 generations.

Note that no uniform crossover like variable swap is applied between offspring to ensure parent-centric crossover operations with minimum chromosome disruptions [42], thus allowing us to clearly observe the effects of inter-task implicit genetic transfers.

B. Specification of Benchmark MOOPs

We highlight that the following benchmarks are created based on prior knowledge of the relationships between tasks in multitask settings. The relationship is measured in terms of the overlap between their optimal solutions and the rank

(ordinal) correlations between their function landscapes [26]. Given such prior knowledge, evolutionary multitasking is naturally expected to benefit from the scope for genetic transfers between tasks that share a degree of inter-task similarity. On the other hand, it may be less effective to conduct knowledge exchange across tasks that have little in common, as this could potentially lead to negative transfers and impede the optimization process. In the latter scenario, an ideal multitasking algorithm should be able to automatically reduce inter-task interactions, thereby performing no worse than its baseline single-tasking algorithm.

In this particular study, we consider a variety of MOOP benchmarks proposed for evolutionary multitask optimization [14, 26]. These problems are characterized by different properties, including the type of the Pareto front, multimodality and separability.

1) *Problem 1*: In this multi-objective problem, $g(\mathbf{x})$ takes the form of the Sphere function,

$$\begin{aligned} \min f_1(\mathbf{x}) &= g(\mathbf{x}) \cos\left(\frac{\pi x_1}{2}\right), \\ \min f_2(\mathbf{x}) &= g(\mathbf{x}) \sin\left(\frac{\pi x_1}{2}\right), \\ g(\mathbf{x}) &= 1 + \sum_{i=2}^D x_i^2, \\ x_1 &\in [0, 1], x_i \in [-100, 100], i = 2, 3, \dots, D. \end{aligned} \quad (6)$$

2) *Problem 2*: This particular MOOP is defined as follows,

$$\begin{aligned} \min f_1(\mathbf{x}) &= x_1; \\ \min f_2(\mathbf{x}) &= g(\mathbf{x}) \left(1 - \left(\frac{x_1}{g(\mathbf{x})}\right)^2\right), \\ g(\mathbf{x}) &= 1 + \frac{9}{n-1} \sum_{i=2}^D |x_i|, \\ x_1 &\in [0, 1], x_i \in [-100, 100], i = 2, 3, \dots, D. \end{aligned} \quad (7)$$

3) *Problem 3*: Here, $g(\mathbf{x})$ takes the form of the Griewank function [43],

$$\begin{aligned} \min f_1(\mathbf{x}) &= g(\mathbf{x}) \cos\left(\frac{\pi x_1}{2}\right) \cos\left(\frac{\pi x_2}{2}\right), \\ \min f_2(\mathbf{x}) &= g(\mathbf{x}) \cos\left(\frac{\pi x_1}{2}\right) \sin\left(\frac{\pi x_2}{2}\right), \\ \min f_3(\mathbf{x}) &= g(\mathbf{x}) \sin\left(\frac{\pi x_1}{2}\right), \\ g(\mathbf{x}) &= 2 + \frac{1}{4000} \sum_{i=3}^D (z_i)^2 - \prod_{i=3}^D \cos\left(\frac{z_i}{\sqrt{i-2}}\right) \\ (z_3, z_4, \dots, z_D)^T &= (x_3, x_4, \dots, x_D)^T - \mathbf{s}, \\ x_1 &\in [0, 1], x_2 \in [0, 1], x_i \in [-50, 50], i = 3, 4, \dots, D, \end{aligned} \quad (8)$$

where $\mathbf{s} = (20, 20, \dots, 20)$ represents a shift vector.

4) *Problem 4*: Here, $g(\mathbf{x})$ takes the form of the Ackley function [44],

$$\begin{aligned} \min f_1(\mathbf{x}) &= \frac{1}{2}(x_1 + x_2), \\ \min f_2(\mathbf{x}) &= g(\mathbf{x})\left(1 - \left(\frac{x_1 + x_2}{2 \cdot g(\mathbf{x})}\right)^2\right), \\ g(\mathbf{x}) &= -20 \exp\left(-0.2 \sqrt{\frac{1}{D-2} \sum_{i=3}^D x_i^2}\right) \\ &\quad - \exp\left(\frac{1}{D-2} \sum_{i=3}^D \cos(2\pi x_i)\right) + 21 + e \\ x_1 &\in [0, 1], x_2 \in [0, 1], x_i \in [-100, 100], i = 3, 4, \dots, D. \end{aligned} \quad (9)$$

5) *Problem 5*: In this case, $g(\mathbf{x})$ takes the form of the Rastrigin function [45, 46],

$$\begin{aligned} \min f_1(\mathbf{x}) &= x_1, \\ \min f_2(\mathbf{x}) &= g(\mathbf{x})\left(1 - \sqrt{\frac{x_1}{g(\mathbf{x})}}\right), \\ g(\mathbf{x}) &= 1 + 10(D-1) + \sum_{i=1}^{D-1} z_i^2 - 10 \cos(4\pi z_i) \quad (10) \\ (z_2, z_3, \dots, z_D)^T &= M \cdot (x_2, x_3, \dots, x_D)^T, \\ x_1 &\in [0, 1], x_i \in [-5, 5], i = 2, 3, \dots, D, \end{aligned}$$

where M is a randomly generated $(D-1) \times (D-1)$ rotation matrix. The properties of these MOOPs are summarized in Table I. In all cases, the dimensionality of the search space is set to $D = 10$. The Inverted Generational Distance (IGD) measure [47] is used for performance comparison between all the algorithms under consideration.

TABLE I
SUMMARY OF TEST PROBLEMS

	Pareto Front	Properties
Problem 1	$f_1^2 + f_2^2 = 1$ $f_1 \geq 0, f_2 \geq 0$	concave, unimodal, separable
Problem 2	$f_2 = 1 - f_1^2$ $0 \leq f_1 \leq 1$	concave, unimodal, separable
Problem 3	$\sum_{i=1}^3 f_i^2 = 1$ $f_i \geq 0, i = 1, 2, 3$	concave, multimodal, nonseparable
Problem 4	$f_2 = 1 - f_1^2$ $0 \leq f_1 \leq 1$	concave, multimodal, nonseparable
Problem 5	$f_2 = 1 - \sqrt{f_1}$ $0 \leq f_1 \leq 1$	convex, multimodal, nonseparable

C. Measuring Complementarity Between Constitutive Tasks

In [26], two quantities have been proposed to measure the complementarity between different multi-objective optimization tasks in a multitask setting. Firstly, the degree of overlap of the global optima of the $g(\mathbf{x})$ functions of constitutive tasks is considered. A scenario in which there is complete overlap between the respective global optima in the unified space is defined as *complete intersection*, and is denoted as CI. On the other hand, the global optima of task-pairs

may also *not intersect*, which we denote as NI. Secondly, a Spearman's rank correlation (R_s) measure is utilized to capture the similarity between the overall trends of the fitness landscape of tasks. The task-pairs for which the Spearman rank correlation $R_s < 0.2$ are characterized as having low inter-task similarity (LS); the task-pairs in the range $0.2 \leq R_s < 0.8$ are defined as having medium similarity (MS); and $R_s \geq 0.8$ implies high inter-task similarity (HS).

D. Experimental Results

1) *Complete Intersection and High Similarity: CI-HS*: We begin by considering a composite MOOP set with complete intersection and high similarity. In this setup, task T_1 is represented by *Problem: 1* (Eq. (6)) while task T_2 is represented by *Problem: 2* (Eq. (7)). The inter-task similarity is measured to be $R_s = 0.97$. The convergence trends (IGD metric averaged over 30 independent runs) for the first 100 generations are shown in Figure 1.

(a) Convergence trends of T_1 in CI-HS (b) Convergence trends of T_2 in CI-HS

(c) learned inter-task rmp 's of MO-MFEA-II

Fig. 1. **Top**: Convergence trends (averaged over 30 independent runs) achieved by all the algorithms in the CI-HS composite MOOP benchmark. **Below**: Pairwise rmp 's learned between tasks T_1 and T_2 over successive generations. The shaded regions span one half standard deviation on either side of the mean.

Since the composite MOOPs share high degrees of inter-task similarities, the effect of knowledge transfers in all the multitasking algorithms is observed to be beneficial. The results indicate that both MO-MFEA-II and MO-MFEA outperform the single tasking NSGA-II. Figures 1(a) and 1(b) show that MO-MFEA variants with high specified rmp values (i.e., 0.9 and 0.6, inducing high frequency of genetic transfers) showcase better convergence trends than MO-MFEA with a

lower prescribed rmp value of 0.3. As such, note that the MO-MFEA requires careful rmp tuning in order to achieve the desired performance. In contrast, the MO-MFEA-II alleviates the hurdle of manual parameter tuning. By learning the pairwise rmp values online and adapting knowledge transfers on the fly, the MO-MFEA-II has recorded improved convergence characteristics for both tasks T_1 and T_2 . Figure 1(c) reveals that the learning module of MO-MFEA-II is able to deduce rmp values that agree well with the *a priori* known inter-task similarity; high rmp values are learned throughout the optimization process implying high frequency of transfers.

We further compare the MO-MFEA-II with the results achieved by the EMEA algorithm for the CI-HS case. Recall that the EMEA employs two different solvers to tackle the two different tasks; T_1 is optimized by NSGA-II while T_2 is optimized by SPEA2. As observed from Figure 1, the EMEA has shown similar performance behavior as the MO-MFEA-II and other MO-MFEA variants. It leverages on the scope of knowledge transfers to improve its baseline algorithms for both the tasks.

2) *No Intersection and Low Similarity: NI-LS*: Next, we consider the extreme case of tackling two unrelated MOOPs with no intersection of the global optima and having low inter-task similarity. In this setup, task T_1 is represented by *Problem: 3* (Eq. (8)) while task T_2 is represented by *Problem: 4* (Eq. (9)). The inter-task similarity is measured to be $R_s = 0.08$. Experimental results are reported in Figure 2.

(a) Convergence trends of T_1 in NI-LS (b) Convergence trends of T_2 in NI-LS

(c) learned inter-task rmp 's of MO-MFEA-II

Fig. 2. **Top:** Convergence trends of all the algorithms in the NI-LS composite MOOP benchmark. **Bottom:** Pairwise learned rmp 's.

The lack of inter-task similarity in this scenario suggests that there exists little scope for crossover-based inter-task knowledge transfers. However, consider the convergence trends of

the MO-MFEA variants with higher frequency of transfers (i.e $rmp = 0.6$ and $rmp = 0.9$). While these configurations seem to have improved the performance of task T_2 in comparison to NSGA-II (Figure 2(b)), we observe that their performance deteriorates significantly on task T_1 (Figure 2(a)). This may be a counter-effect of the *omnidirectional* transfer mechanism, which, if not controlled, could lead to undesired negative transfers for one of the tasks. This conjecture is also supported by the results of EMEA on T_1 , which is found to be hampered by negative transfers as well. As such, the MO-MFEA variants are seen to improve one task at the expense of worsening the other, but, with a decreased rmp value (0.3), such counter-effects are minimized. On the other hand, MO-MFEA-II has recorded superior convergence trends for both tasks. As shown by the learned rmp curve in Figure 2(c) the frequency of transfer is low, thus making the algorithm less prone to the impact of negative transfers.

3) *Study with more than two tasks*: Studies focusing on many-tasking have been carried out in the domain single-objective optimization [13, 48]. Along similar lines, we extend our current experimental study to more than two multi-objective tasks. In particular, we attempt to analyze the effects of omnidirectional transfers in the context of pooling several tasks (up to five MOOPs) in the same multitasking environment. For this set of experiments, all the five problems defined in section IV-B are considered. Tasks $T_1, T_2, \dots, T_5$ are represented by *Problems: 1, 2, ..., 5*, respectively. The inter-task similarities are given in Table II.

TABLE II
INTER-TASK SIMILARITIES R_s

	T_1	T_2	T_3	T_4	T_5
T_1	1	0.970	0.298	0.778	0.402
T_2	-	1	0.299	0.753	0.391
T_3	-	-	1	0.08	0.130
T_4	-	-	-	1	0.817
T_5	-	-	-	-	1

Experimental results are reported in Table III and Figure 3. According to the results, MO-MFEA-II records the overall best performance. More importantly, no harmful effects of negative transfer is seen for Mo-MFEA-II in any of the five tasks. Clearly, such superior performance is attributed to the fact that MO-MFEA-II incorporates a *RMP* matrix, capable of capturing diverse inter-task relationships between various task pairs and adapting the extent of transfers accordingly.

In contrast, the MO-MFEA variants that employ a user-specified scalar rmp are found to suffer. Since the rmp is fixed, the extent of transfer remains the same (uniform) across all task pairs (irrespective of the existence of non-uniform inter-task synergies reported in Table II). In such a restrictive approach, the threats of negative transfers are observed to be predominant. For instance, while analyzing the convergence trends in Figure 3, it can be observed that prescribing a relatively high rmp (e.g., 0.6) may bolster performance in a few tasks while hampering others (in most cases, the conventional MO-MFEA performs worse than single-tasking NSGA-II).

TABLE III
AVERAGE IGD VALUES ACHIEVED BY NSGA-II, MO-MFEA-II AND MO-MFEA (WITH A RANGE OF rmp VALUES) FOR ALL $K = 5$ TASKS. BOLD FACE ENTRIES ARE THE BEST (BY WILCOXON TEST WITH SIGNIFICANCE LEVEL 0.05).

	Tasks				
	T_1	T_2	T_3	T_4	T_5
NSGA-II	$7.75e-02$ ($3.31e-02$)	$3.46e-01$ ($1.45e-01$)	$1.90e+01$ ($4.88e+00$)	$3.97e-01$ ($5.65e-02$)	$1.70e+00$ ($8.21e-01$)
MO-MFEA-II	$1.98e-02$ ($5.70e-03$)	$2.27e-01$ ($6.94e-02$)	$1.67e+01$ ($7.40e+00$)	$3.74e-01$ ($5.14e-02$)	$1.58e+00$ ($7.88e-01$)
MO-MFEA	($rmp = 0.90$)	$1.24e-01$ ($5.92e-02$)	$8.11e-01$ ($2.47e-01$)	$2.04e+01$ ($5.14e-02$)	$8.86e-01$ ($1.98e-01$)
	($rmp = 0.60$)	$5.24e-02$ ($2.31e-02$)	$5.07e-01$ ($1.30e-01$)	$1.90e+01$ ($7.40e+00$)	$5.60e-01$ ($5.68e-02$)
	($rmp = 0.30$)	$3.57e-02$ ($1.39e-02$)	$3.91e-01$ ($1.06e-01$)	$1.88e+01$ ($4.77e+00$)	$4.38e-01$ ($5.44e-02$)

(a) Convergence trends of T_1

(b) Convergence trends of T_2

(c) Convergence trends of T_3

(d) Convergence trends of T_4

(e) Convergence trends of T_5

Fig. 3. Convergence trends (averaged over 30 independent runs) achieved by all the algorithms on the 5 tasks.

Furthermore, Tables IV and V show numerical results for pooling $K = 4$ and $K = 3$ tasks (MOOPs) together, respectively. Performance trends similar to Table III are observed.

V. EVOLUTIONARY MULTITASKING FOR HIGH FIDELITY OPTIMIZATION: A PRACTICAL STUDY

One of the key beneficiaries of our proposed MO-MFEA-II framework is expected to be in the domain of high fidelity optimization settings. The fact that real world high fidelity numerical simulations can take anywhere between several minutes to several hours for a single run, the use of traditional single-task optimization approaches still remain too costly. On the other hand, knowledge exploitation from cheaper optimization problems to cut down the computational costs of (related)

expensive problems has shown to be promising [6, 49]. In this section, we demonstrate the efficacy of MO-MFEA-II in leveraging knowledge transfers from low fidelity tasks to speed up the performance of higher fidelity tasks.

In particular, we consider multi-objective optimization of the unmanned aerial vehicle (UAV) path planning problem. Before we proceed to the details of our experimental settings, we present a brief overview of the UAV path planning procedure. UAVs have emerged as promising tools for autonomous inspection of large geographical areas and performing missions in hazardous environments, such as military surveillance, nuclear power plant operations and outer-space exploration [50, 51]. Autonomous navigation by UAVs through unknown environments is a challenging task that requires safe and efficient path planning strategies, such that operational risks (in the

TABLE IV
AVERAGE IGD VALUES FOR ALL $K = 4$ TASKS

		Tasks			
		T_1	T_2	T_3	T_4
NSGA-II		$7.75e-02$ ($3.31e-02$)	$3.46e-01$ ($1.45e-01$)	$1.90e+01$ ($4.88e+00$)	$3.97e-01$ ($5.65e-02$)
MO-MFEA-II		$2.44e-02$ ($1.12e-02$)	$2.83e-01$ ($9.23e-02$)	$1.83e+01$ ($6.34e+00$)	$3.92e-01$ ($7.46e-02$)
MO-MFEA	$(rmp = 0.90)$	$1.21e-02$ ($1.12e-02$)	$7.46e-01$ ($1.77e-01$)	$2.04e+01$ ($1.30e+01$)	$9.00e-01$ ($8.57e-02$)
	$(rmp = 0.60)$	$4.38e-02$ ($1.42e-02$)	$4.75e-01$ ($1.03e-01$)	$1.87e+01$ ($5.24e+00$)	$5.30e-01$ ($1.27e-01$)
	$(rmp = 0.30)$	$3.43e-02$ ($8.30e-03$)	$3.42e-01$ ($7.47e-02$)	$2.03e+01$ ($2.19e-02$)	$4.44e-01$ ($6.97e-02$)

TABLE V
AVERAGE IGD VALUES FOR ALL $K = 3$ TASKS

		Tasks		
		T_1	T_2	T_3
NSGA-II		$7.75e-02$ ($3.31e-02$)	$3.46e-01$ ($1.45e-01$)	$1.90e+01$ ($4.88e+00$)
MO-MFEA-II		$2.27e-02$ ($6.20e-03$)	$2.75e-01$ ($7.35e-02$)	$1.31e+01$ ($9.45e+00$)
MO-MFEA	$(rmp = 0.90)$	$4.19e-02$ ($1.27e-02$)	$4.26e-01$ ($7.50e-02$)	$1.88e+01$ ($4.87e+00$)
	$(rmp = 0.60)$	$3.85e-02$ ($1.77e-02$)	$4.02e-01$ ($1.03e-01$)	$1.86e+01$ ($5.67e+00$)
	$(rmp = 0.30)$	$2.74e-02$ ($8.80e-03$)	$3.04e-01$ ($9.67e-02$)	$2.04e+01$ ($6.56e-02$)

presence of hazards including obstacles or environmental factors) are minimized. The risks associated with UAVs may vary from collisions to loss of control. Collisions are usually caused by static or dynamic obstacles while loss of control can result from system failure, navigating beyond signal range (e.g. radio, WiFi or GSM networks) or adverse environmental conditions (e.g. bad weather or bad GPS reception) [52]. In our experimental study, we consider a realistic scenario with multiple UAVs operating in a specified region in the southwest of Singapore. The operational environment is characterized by different types of hazards and environmental factors (e.g. signal strength and weather conditions). For our simulations, a data-driven multitask Gaussian process (GP) model of the environment is created. The probabilistic predictions from such a GP model are then used to define a *path-integral* risk metric, which gives the probability of an unsafe outcome along a UAV's traversed path. Background details of safety risk assessment in UAVs, GP modeling of the environment and the mathematical definitions of the path-integral risk metric are not reproduced herein for the sake of brevity, but can be found in [52].

A. Multi-objective Risk Averse UAV Path Planning

The problem of minimizing operational risk for multiple UAVs in uncertain environments can be defined in a multi-objective optimization framework, addressing the trade-off between the path-integral risk measure and path-efficiency (travel distance). The considered MOOP can be represented as follows,

$$\text{minimize } (\text{Risk}, \text{Distance}), \quad (11)$$

where the first objective corresponds to minimizing the probability of an unsafe event occurring for at least one UAV, while the second objective accounts for the total distance travelled by all the UAVs from their respective origins to destinations.

With regard to the risk measure, the path-integral over a continuous path for a particular UAV is evaluated through a discretization of its trajectory. We note that the runtime of evaluating the risk metric along a finite discretized path is highly dependent on the discretization step size. To elaborate, a smaller step size implies numerical integrations of high fidelity, making the optimization process computationally expensive. Such high fidelity simulations, albeit more accurate, can take up several minutes or even hours. On the other hand, lower fidelity simulations with larger discretization step sizes can be much faster, although they do not offer the same level of accuracy in the risk estimations.

In this demonstration, we consider five UAVs operating in a region of about $10 \times 7 \text{ km}^2$ in the southwest of Singapore. We introduce two environmental factors that translate into hazards: 1) signal strength (4G signal) and 2) the weather conditions (defined by different wind speeds). Further, the region includes geofences which UAVs cannot enter, and there exist a couple of other constraints such as maintaining a minimum separation distance between UAVs and flying within specified altitude boundaries. The trajectories of UAVs are defined using cubic spline interpolation [53]. For full experimental details, the reader is referred to [52].

Table VI shows two different experimental configurations used to demonstrate the efficacy of MO-MFEA-II for high fidelity multi-criteria optimization of UAV trajectories.

TABLE VI

PRACTICAL STUDY EXPERIMENTAL CONFIGURATIONS. DIFFERENT TASK INSTANCES ARE GENERATED BY DEFINING DIFFERENT DISCRETIZATION STEP SIZES (d_{step})

	Tasks	
	Main task	Helper task(s)
Configuration 1	$T_1 : d_{step} = 500m$	$T_2 : d_{step} = 1000m$
Configuration 2	$T_1 : d_{step} = 500m$	$T_2 : d_{step} = 750m$ $T_3 : d_{step} = 1000m$

As shown, different instances of tasks (representing different fidelity levels) are generated using different discretization step sizes (d_{step}) associated with the path-integral risk measure. In both configurations, the high fidelity task T_1 represents the main task of interest, while the low fidelity tasks T_2 and T_3 are considered as helper tasks. Since all the specified tasks essentially solve the same MOOP, but with different fidelity levels, there inherently exists underlying complementarities that can be exploited via multitasking.

While the baseline NSGA-II solves the main tasks in isolation, the MO-MFEA-II pools together the main and helper task(s) for simultaneous optimization. The hypervolume metric [54] is used as an indicator to measure the quality of solutions obtained by the two different algorithms.

Figure 4(a) shows the convergence trends of T_1 mapped against the total run time for experimental Configuration 1. As is seen, the MO-MFEA-II leads to a significant boost in the optimization search during the initial stages of the main task. Specifically, to achieve a hypervolume measure of 0.415 for the high fidelity task T_1 , MO-MFEA-II takes 33% less time compared to NSGA-II. The corresponding learned *rpm* values are depicted in Figure 4(b). As shown, T_1 experiences high frequency of transfers from its lower fidelity helper task T_2 , thereby boosting the overall convergence trend.

Next, in experimental Configuration 2, we demonstrate the effects of introducing an additional low fidelity helper task T_3 . As shown in Figure 5(a), the overall convergence trend of the high fidelity task T_1 (blue curve) is further boosted relative to the former experimental setup with only one helper task (shown by the red curve in Figure 5(a)). The performance enhancement is attributed to the learned *rpm* curves given by Figure 5(b). Therein, T_1 is observed to be experiencing knowledge transfer from both helper tasks T_2 and T_3 , with the learned rate of transfer from T_2 being much more frequent; this agrees with our intuition from Table VI that T_1 is more similar to T_2 than it is to T_3 . Notably, from Figure 5(b), it is also seen that T_3 helps T_2 (see $T_2 - T_3$ *rpm* curve), which in turn helps boost the optimization performance of the high fidelity task T_1 .

VI. CONCLUSION

In this paper, we presented a cognizant multitasking multi-objective multifactorial evolution algorithm, namely, MO-MFEA-II. The salient feature of MO-MFEA-II is its *self*

awareness, constantly modeling diverse relationships in the optimization environment and adapting its own mechanisms accordingly. We demonstrate the efficacy of MO-MFEA-II experimentally on various benchmark MOOPs. The experimental results show that the proposed method is capable of learning inter-task relationships between different MOOPs, demonstrating the algorithm's ability to decipher *when* and *how much* knowledge to transfer across different tasks. As opposed to the restrictive scalar transfer parameter in the present day MO-MFEA, the flexibility of MO-MFEA-II to capture diverse inter-task relationships is enhanced through a *RMP* matrix representation.

In addition to the above, a practically useful application is found to be in the domain of multi-fidelity optimization. The simulation results highlight the ability of MO-MFEA-II to exploit relevant knowledge from low fidelity (cheaper) tasks to quickly optimize high fidelity tasks, thereby leading to significant cost saving benefits. Further, the empirical evidence in this paper shows the potential benefits of incorporating multiple low fidelity tasks for performance enhancement of the high fidelity task.

Although the MO-MFEA-II is adept at capturing inter-task relationships across several tasks at once, we however note that the associated learning procedure becomes computationally demanding as the number of tasks grows increasingly large. In the literature, there have however been some recent works to efficiently handle *many-task* optimization [13, 18]. While the direct scalability of MO-MFEA-II to many-tasking can be achieved through simplifications such as reduced frequency of *RMP* learning, we shall take up a more rigorous study of this issue in our future work.

REFERENCES

- [1] L. S. Gottfredson, "Mainstream science on intelligence: An editorial with 52 signatories, history, and bibliography," 1997.
- [2] C. D. Wickens, "Situation awareness: Review of mica endsley's 1995 articles on situation awareness theory and measurement," *Human factors*, vol. 50, no. 3, pp. 397–403, 2008.
- [3] K. Tirri and P. Nokelainen, *Measuring multiple intelligences and moral sensitivities in education*. Springer Science & Business Media, 2012, vol. 5.
- [4] 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.
- [5] R. Caruana, "Multitask learning," in *Learning to learn*. Springer, 1998, pp. 95–133.
- [6] K. Swersky, J. Snoek, and R. P. Adams, "Multi-task bayesian optimization," in *Advances in neural information processing systems*, 2013, pp. 2004–2012.
- [7] A. Gupta, Y.-S. Ong, and L. Feng, "Multifactorial evolution: towards evolutionary multitasking," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 343–357, 2016.
- [8] L. Feng, L. Zhou, J. Zhong, A. Gupta, Y.-S. Ong, K.-C. Tan, and A. Qin, "Evolutionary multitasking via explicit autoencoding," *IEEE transactions on cybernetics*, no. 99, pp. 1–14, 2018.
- [9] A. Gupta and Y.-S. Ong, "Back to the roots: Multi-x evolutionary computation," *Cognitive Computation*, pp. 1–17, 2019.
- [10] A. Gupta, Y.-S. Ong, and L. Feng, "Insights on transfer optimization: Because experience is the best teacher," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1, pp. 51–64, 2017.
- [11] X. Zheng, A. Qin, M. Gong, and D. Zhou, "Self-regulated evolutionary multi-task optimization," *IEEE Transactions on Evolutionary Computation*, 2019.
- [12] B. Zhang, A. K. Qin, and T. Sellis, "Evolutionary feature subspaces generation for ensemble classification," in *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 2018, pp. 577–584.

Fig. 4. **Left:** Convergence trends (averaged over 10 independent runs) achieved by NSGA-II and MO-MFEA-II for experimental Configuration 1. **Right:** Pairwise rmp 's learned between tasks T_1 and the helper task T_2 over total run time. The shaded regions span one half standard deviation on either side of the mean.

Fig. 5. **Left:** Convergence trends achieved by NSGA-II and MO-MFEA-II for experimental Configuration 2. The MO-MFEA-II employs two helper tasks, respectively. **Right:** Pairwise rmp 's learned between tasks T_1 and the helper tasks T_2 and T_3 . The shaded regions span one half standard deviation on either side of the mean.

- [13] R.-T. Liaw and C.-K. Ting, "Evolutionary manytasking optimization based on symbiosis in biocoenosis," in *Thirty-Third AAAI Conference on Artificial Intelligence*, 2019.
- [14] A. Gupta, Y.-S. Ong, L. Feng, and K. C. Tan, "Multiobjective multifactorial optimization in evolutionary multitasking," *IEEE Transactions on cybernetics*, 2017.
- [15] Y.-W. Wen and C.-K. Ting, "Learning ensemble of decision trees through multifactorial genetic programming," in *Evolutionary Computation (CEC), 2016 IEEE Congress on*. IEEE, 2016, pp. 5293–5300.
- [16] R. Chandra, A. Gupta, Y.-S. Ong, and C.-K. Goh, "Evolutionary multi-task learning for modular knowledge representation in neural networks," *Neural Processing Letters*, pp. 1–17, 2017.
- [17] E. O. Scott and K. A. De Jong, "Automating knowledge transfer with multi-task optimization," in *2019 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2019, pp. 2252–2259.
- [18] Y. Chen, J. Zhong, L. Feng, and J. Zhang, "An adaptive archive-based evolutionary framework for many-task optimization," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2019.
- [19] G. Li, Q. Lin, and W. Gao, "Multifactorial optimization via explicit multipopulation evolutionary framework," *Information Sciences*, vol. 512, pp. 1555–1570, 2020.
- [20] S. Huang, J. Zhong, and W. Yu, "Surrogate-assisted evolutionary framework with adaptive knowledge transfer for multi-task optimization," *IEEE Transactions on Emerging Topics in Computing*, 2019.
- [21] J. Lin, H. Liu, B. Xue, M. Zhang, and F. Gu, "Multi-objective multi-tasking optimization based on incremental learning," *IEEE Transactions on Evolutionary Computation*, pp. 1–1, 2019.
- [22] J. Mo, Z. Fan, W. Li, Y. Fang, Y. You, and X. Cai, "Multi-factorial evolutionary algorithm based on m2m decomposition," in *Asia-Pacific Conference on Simulated Evolution and Learning*. Springer, 2017, pp. 134–144.
- [23] K. K. Bali, Y.-S. Ong, A. Gupta, and P. S. Tan, "Multifactorial evolutionary algorithm with online transfer parameter estimation: Mfea-ii," *IEEE Transactions on Evolutionary Computation*, 2019.
- [24] L. Bao, Y. Qi, M. Shen, X. Bu, J. Yu, Q. Li, and P. Chen, "An evolutionary multitasking algorithm for cloud computing service composition," in *World Congress on Services*. Springer, 2018, pp. 130–144.
- [25] M. Iqbal, W. N. Browne, and M. Zhang, "Reusing building blocks of extracted knowledge to solve complex, large-scale boolean problems," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 465–480, 2014.
- [26] Y. Yuan, Y.-S. Ong, L. Feng, A. K. Qin, A. Gupta, B. Da, Q. Zhang, K. C. Tan, Y. Jin, and H. Ishibuchi, "Evolutionary multitasking for multi-objective continuous optimization: Benchmark problems, performance metrics and baseline results," *arXiv preprint arXiv:1706.02766*, 2017.
- [27] K. K. Bali, A. Gupta, L. Feng, Y. S. Ong, and T. P. Siew, "Linearized domain adaptation in evolutionary multitasking," in *Evolutionary Computation (CEC), 2017 IEEE Congress on*. IEEE, 2017, pp. 1295–1302.
- [28] D. T. Stuss and D. F. Benson, "Neuropsychological studies of the frontal lobes," *Psychological bulletin*, vol. 95, no. 1, p. 3, 1984.
- [29] C. A. C. Coello, G. B. Lamont, D. A. Van Veldhuizen *et al.*, *Evolutionary algorithms for solving multi-objective problems*. Springer, 2007, vol. 5.
- [30] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [31] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints," *IEEE transactions on evolutionary computation*, vol. 18, no. 4, pp. 577–601, 2013.
- [32] H. Jain and K. Deb, "An evolutionary many-objective optimization algorithm using reference-point based nondominated sorting approach, part ii: handling constraints and extending to an adaptive approach," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 602–622, 2013.
- [33] E. Zitzler, M. Laumanns, and L. Thiele, "Spear2: Improving the strength pareto evolutionary algorithm," *TIK-report*, vol. 103, 2001.

- [34] Q. Zhang and H. Li, "Moea/d: A multiobjective evolutionary algorithm based on decomposition," *IEEE Transactions on evolutionary computation*, vol. 11, no. 6, pp. 712–731, 2007.
- [35] K. Li, S. Kwong, Q. Zhang, and K. Deb, "Interrelationship-based selection for decomposition multiobjective optimization," *IEEE transactions on cybernetics*, vol. 45, no. 10, pp. 2076–2088, 2014.
- [36] S. Jiang and S. Yang, "An improved multiobjective optimization evolutionary algorithm based on decomposition for complex pareto fronts," *IEEE transactions on cybernetics*, vol. 46, no. 2, pp. 421–437, 2015.
- [37] S. Yao, Z. Dong, X. Wang, and L. Ren, "A multiobjective multifactorial optimization algorithm based on decomposition and dynamic resource allocation strategy," *Information Sciences*, vol. 511, pp. 18–35, 2020.
- [38] H. T. T. Binh, N. Q. Tuan, and D. C. T. Long, "A multi-objective multifactorial evolutionary algorithm with reference-point-based approach," in *2019 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2019, pp. 2824–2831.
- [39] J. C. Bean, "Genetic algorithms and random keys for sequencing and optimization," *ORSA journal on computing*, vol. 6, no. 2, pp. 154–160, 1994.
- [40] Q. Zhang and H. Muhlenbein, "On the convergence of a class of estimation of distribution algorithms," *IEEE Transactions on evolutionary computation*, vol. 8, no. 2, pp. 127–136, 2004.
- [41] A. Gupta and Y.-S. Ong, *Memetic Computation: The Mainspring of Knowledge Transfer in a Data-Driven Optimization Era*. Springer, 2018, vol. 21.
- [42] E. A. Williams and W. A. Crossley, "Empirically-derived population size and mutation rate guidelines for a genetic algorithm with uniform crossover," in *Soft computing in engineering design and manufacturing*. Springer, 1998, pp. 163–172.
- [43] M. Locatelli, "A note on the griewank test function," *Journal of global optimization*, vol. 25, no. 2, pp. 169–174, 2003.
- [44] D. Ackley, *A connectionist machine for genetic hillclimbing*. Springer Science & Business Media, 2012, vol. 28.
- [45] L. Rastrigin, "Extremal control systems," *Theoretical foundations of engineering cybernetics series*, vol. 3, 1974.
- [46] H. Mühlenbein, M. Schomisch, and J. Born, "The parallel genetic algorithm as function optimizer," *Parallel computing*, vol. 17, no. 6-7, pp. 619–632, 1991.
- [47] S. Jiang, Y.-S. Ong, J. Zhang, and L. Feng, "Consistencies and contradictions of performance metrics in multiobjective optimization," *IEEE transactions on cybernetics*, vol. 44, no. 12, pp. 2391–2404, 2014.
- [48] J. Tang, Y. Chen, Z. Deng, Y. Xiang, and C. P. Joy, "A group-based approach to improve multifactorial evolutionary algorithm," in *IJCAI*, 2018, pp. 3870–3876.
- [49] J. Ding, C. Yang, Y. Jin, and T. Chai, "Generalized multi-tasking for evolutionary optimization of expensive problems," *IEEE Transactions on Evolutionary Computation*, 2017.
- [50] M. Jun and R. D'Andrea, "Path planning for unmanned aerial vehicles in uncertain and adversarial environments," in *Cooperative control: models, applications and algorithms*. Springer, 2003, pp. 95–110.
- [51] M. Monwar, O. Semiari, and W. Saad, "Optimized path planning for inspection by unmanned aerial vehicles swarm with energy constraints," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.
- [52] J. Rubio-Hervas, A. Gupta, and Y.-S. Ong, "Data-driven risk assessment and multicriteria optimization of uav operations," *Aerospace Science and Technology*, vol. 77, pp. 510–523, 2018.
- [53] C. H. Reinsch, "Smoothing by spline functions," *Numerische mathematik*, vol. 10, no. 3, pp. 177–183, 1967.
- [54] E. Zitzler, D. Brockhoff, and L. Thiele, "The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2007, pp. 862–876.

Kavitesh Kumar BALI received the M.Sc. degree in computer science from the University of the South Pacific, Suva, Fiji, in 2016. He is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His current research interests include evolutionary computation, transfer optimization, and machine learning.

Abhishek GUPTA received his PhD in Engineering Science from the University of Auckland, New Zealand, in the year 2014. He currently serves as a Scientist in the Singapore Institute of Manufacturing Technology (SIMTech), at the Agency for Science, Technology and Research (A*STAR), Singapore. He has diverse research experience in the field of computational science, ranging from numerical methods in engineering physics, to topics in computational intelligence. His current research focus is in the development of memetic computation and probabilistic

model-based algorithms for automated knowledge extraction and transfer across optimization problems, with applications in cyber physical production systems and engineering design.

Yew-Soon ONG (M'99-SM'12-F'18) received a Ph.D. degree on artificial intelligence in complex design from the University of Southampton, U.K., in 2003. He is a President's Chair Professor of Computer Science at the Nanyang Technological University (NTU), and holds the position of Chief Artificial Intelligence Scientist at the Agency for Science, Technology and Research Singapore. At NTU, he currently serves as Director of the Data Science and Artificial Intelligence Research Center and Director of the Singtel-NTU Cognitive Artificial Intelligence Joint Lab. His current research interests is in artificial and computational intelligence. He is founding Editor-in-Chief of the IEEE Transactions on Emerging Topics in Computational Intelligence, Associate Editor of several IEEE Transactions and a Fellow of the IEEE. He has also received several IEEE outstanding paper awards, listed as a Thomson Reuters highly cited researcher and among the World's Most Influential Scientific Minds.

Puay Siew TAN leads the Manufacturing Control Tower™ (MCT™) as the Programme Manager which is responsible for the setup of Model Factory@SIMTech. Her research has been in the cross-field disciplines of Computer Science and Operations Research for cyber physical production system (CPPS) collaboration, in particular sustainable complex manufacturing and supply chain operations. To this end, she has been active in using context-aware and services techniques. Her present research interests include Complex Systems, specifically quantitative techniques for understanding disruptions propagation in networked supply chains and mitigation of risks caused by these disruptions.

techniques for understanding disruptions propagation in networked supply chains and mitigation of risks caused by these disruptions.