


Article

Decomposition Is All You Need: Single-Objective to Multi-Objective Optimization towards Artificial General Intelligence

Wendi Xu ^{1,2,3,4,†}, Xianpeng Wang ^{1,2,3,†}, Qingxin Guo ^{1,2,3,*,†} , Xiangman Song ^{1,2,3,†}, Ren Zhao ^{1,2,3,†}, Guodong Zhao ^{1,2,3}, Dakuo He ^{1,2,3,4,*}, Te Xu ^{1,2,3}, Ming Zhang ^{5,6} and Yang Yang ^{1,2,3,*}

- ¹ College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; 1910280@stu.neu.edu.cn (W.X.); wangxianpeng@ise.neu.edu.cn (X.W.); songxiangman@ise.neu.edu.cn (X.S.); zhaoren@mail.neu.edu.cn (R.Z.); zhaoguodong@mail.neu.edu.cn (G.Z.); xute@mail.neu.edu.cn (T.X.)
 - ² Key Laboratory of Data Analytics and Optimization for Smart Industry, Ministry of Education, Shenyang 110819, China
 - ³ Frontier Sciences Center for Industrial Intelligence and Systems Optimization, Ministry of Education, Shenyang 110819, China
 - ⁴ Institute of Industrial Artificial Intelligence and Optimization, Northeastern University, Shenyang 110819, China
 - ⁵ Key Laboratory for Radio Astronomy, Chinese Academy of Sciences, Nanjing 210000, China; zhang.ming@xao.ac.cn
 - ⁶ University of Chinese Academy of Sciences, Beijing 100000, China
- * Correspondence: guoqingxin@ise.neu.edu.cn (Q.G.); hedakuo@ise.neu.edu.cn (D.H.); yangyang@ise.neu.edu.cn (Y.Y.)
- † These authors contributed equally to this work.



Citation: Xu, W.; Wang, X.; Guo, Q.; Song, X.; Zhao, R.; Zhao, G.; He, D.; Xu, T.; Zhang, M.; Yang, Y. Decomposition Is All You Need: Single-Objective to Multi-Objective Optimization towards Artificial General Intelligence. *Mathematics* **2023**, *11*, 4390. <https://doi.org/10.3390/math11204390>

Academic Editors: Huajun Tang, Jun Wu, Huimin Jiang, Ivan Lai and Shih-Wei Lin

Received: 31 August 2023
Revised: 12 October 2023
Accepted: 19 October 2023
Published: 23 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: As a new abstract computational model in evolutionary transfer optimization (ETO), single-objective to multi-objective optimization (SMO) is conducted at the macroscopic level rather than the intermediate level for specific algorithms or the microscopic level for specific operators; this method aims to develop systems with a profound grasp of evolutionary dynamic and learning mechanism similar to human intelligence via a “decomposition” style (in the abstract of the well-known “Transformer” article “Attention is All You Need”, they use “attention” instead). To the best of our knowledge, it is the first work of SMO for discrete cases because we extend our conference paper and inherit its originality status. In this paper, by implementing the abstract SMO in specialized memetic algorithms, key knowledge from single-objective problems/tasks to the multi-objective core problem/task can be transferred or “gathered” for permutation flow shop scheduling problems, which will reduce the notorious complexity in combinatorial spaces for multi-objective settings in a straight method; this is because single-objective tasks are easier to complete than their multi-objective versions. Extensive experimental studies and theoretical results on benchmarks (1) emphasize our decomposition root in mathematical programming, such as Lagrangian relaxation and column generation; (2) provide two “where to go” strategies for both SMO and ETO; and (3) contribute to the mission of building safe and beneficial artificial general intelligence for manufacturing via evolutionary computation.

Keywords: evolutionary transfer optimization; green scheduling; transfer learning; artificial general intelligence; mathematical programming; system optimization; carbon neutrality

MSC: 68T01

1. Introduction

Artificial general intelligence [1–20] (AGI) for language and vision includes the well-known Chat GPT and ViT, respectively. Both Chat GPT and ViT are empowered by

Transformers. In a Transformer, one of its main features is **attention** design (a Transformer is **A**ttentive, **R**esidual, and **H**ierarchical, which has deep connections with the three “A-R-H” ideas in our visual AGI framework of **WARSHIP** [9]). In addition to the **attention** design in the structural/architectural aspect, the **transferring** mechanism in the functional aspect is also common and important in Transformers, spanning several facets of cognition, learning, and decision making.

Insisting on or preserving the **transferring** mechanism in the functional aspect is necessary when we choose potential tools of AGI [1–3] to solve evolutionary tasks. We believe that evolutionary transfer optimization (ETO) is quite promising. Inspired by the **decomposition** styles in applied mathematics and computational intelligence (CI), we develop a new **transferring** system or computational model of ETO: the single-objective to multi-objective optimization (SMO) model, which belongs to the third kind of “complex optimization” in the ETO survey [20]. In the tradition of CI, the abbreviations of single-objective optimization and multi-objective optimization are SOO and MOO, respectively. Both SOO and MOO are not new, whereas both ETO and SMO are new.

The works on SMO by our groups can be “gathered” together in a Chinese lantern diagram (Figure 1) as follows: the conference paper of “Meets”, future works named as “xMeets” and journal papers of “iMeets” [7], “rMeets” [8], and “eMeets”. Actually, eMeets is this paper here and is an extension of Meets [6], thus inheriting the originality status of Meets.

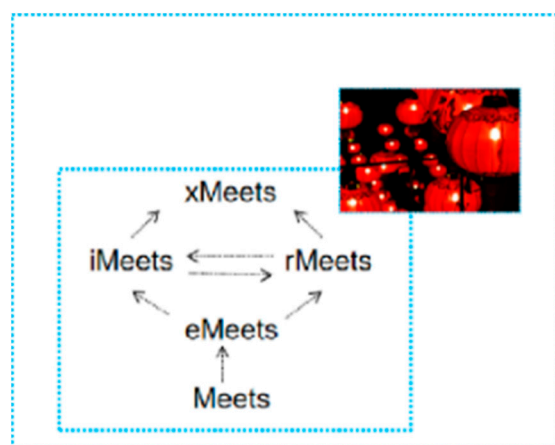


Figure 1. In the background of **industrial intelligence** [1,2], the lantern is derived from the tree, and the tree is rooted in the “Tai ji” [2] (the same as [7]). As Chinese lanterns symbolize red ornaments for a festive atmosphere, the lantern above is dedicated to the establishment of our lab at the Frontier Science Center for Industrial Intelligence and System Optimization, Ministry of Education, established since 2021, which is a **national**-level center.

Our main contributions are as follows:

1. To the best of our knowledge, this is the **first** work of SMO in discrete cases. And, it is also the **first** work of ETO for the permutation flow shop scheduling problem (PFSP).
2. Two “where to go” strategies of SMO provide the rough strokes of SMO, endowing the great significance of eMeets.
3. An extensive study of different combinations of operators shows the functional parts of our SMO framework towards AGI.

The related works are as follows. The most related work to our paper here is [5]. To the best of their knowledge, their paper was the first paper (they did not use the concept of SMO) to boost the task of evolutionary multi-objective optimization via key knowledge transferred or learned from the corresponding problems of single-objective cases. In our paper here, we choose to roughly follow the same abstract framework of [5], such as transferring or learning experience or knowledge via the direct injection of external solution populations from the source evolutionary task of all single objective evolutionary

tasks to a corresponding multi-objective target evolutionary task “every G generation, where G is the gap” [6]. In the following section, another four important and representative related works are given. The first one is an ETO for multitasking optimization (MTO) in discrete cases. A method of the multifactorial evolutionary algorithm was deployed to explore the optimization power for cases in evolutionary multitasking, which can serve as the key engine for simultaneous optimizations in the case of multiple permutation-based problems for applications of supply chain networks [10]. The second case is an ETO for MTO in the discrete case of routing. In the work [11], a framework of memetic computing was deployed, which can learn and transfer knowledge memes that traverse two different but related optimization domains to enhance the search work. The third case is “machine learning based intelligent optimization for PFSP” [12]. In [12], the machine learning based memetic algorithm was used for the case of PFSP under a multi-objective setting. The method is named ML-MOMA, and one of its main developments is the setting of a local search via ideas of machine learning. In addition to the algorithms mentioned above, another method in intelligent optimization that is like simulated annealing can also solve PFSP optimization as follows. The last case is “residual learning based intelligent optimization simulated annealing for PFSP” [13]. In order to solve PFSP, ref. [13] uses an improved algorithm of simulated annealing, which is deployed with the setting of residual learning. The neighborhood is well defined in the optimization of PFSP.

2. Materials and Methods

2.1. Test Problem: PFSP

In the test problem of PFSP, all jobs in the shop must undergo a series of machine operations. Usually, these operations must be performed on every job in the same order, implying that every job must follow the same route. “The machines are then assumed to be in a series” [6]. Often, each queue to be processed is assumed to be under the discipline of the first in first out (FIFO) rule, that is, each job cannot pass another while in an arranged queue. As for the objectives, we choose makespan (Cmax) and total flow time (TFT) as the optimization goals towards optimality.

To tackle the test problem of PFSP, we need the **3 tools** of transferred knowledge, science clustering, and building block hypothesis (BBH, fundamental genetic algorithms, and memetic algorithms theory). For **transferred knowledge**, we find that positional BB dominates other BBs. From [6], it could be summarized that positional BBs help improve the optimization of Cmax and also improve the optimization of other objectives in an inherent way; this highlights the significance of positional BBs. In **science clustering**, we focus on mining the positional BB via unsupervised learning. For PFSP here, we implemented “science clustering” (we named it this way because it is published in the literature of Science journal) via the hamming distance metric. To our surprise, the hamming distance also focuses on the work of mining positional BBs. For **BB hypothesis**, we are mainly concerned with Goldberg’s decomposition theory (7 steps). The building block hypothesis is mainly based on Goldberg’s decomposition theory, which contains 7 steps as follows: (1) “know what genetic algorithms process” [14]—BBs; (2) solve optimization problems at hand that have the difficulty of bounded BBs; (3) ensure that the supply state of raw BBs is indeed adequate; (4) ensure that the “market share” [14] of superior building blocks can increase; (5) know the “BB takeover” [14] and the computational models that characterize convergence times; (6) ensure that genetic algorithms usually make the decisions of BBs well; and (7) ensure that good mixing in BBs is achieved.

2.2. The Framework across Tasks: SMO

2.2.1. Typical 3 Tasks: 2 Subtasks Boost the Core Task

We firstly propose the definition of SMO via the case study of ETO_PFSPP, that is, an ETO for PFSP. Two subtasks boost the core task when solving PFSP.

2.2.2. 5 Bags, 5*2 Groups, 5*2*4 Tasks for SMO: e.g., Bag 1: Group 1, t1_wc (t_wc 1.0, t_wc 1.1), t2_wc, and t2e_wc; Group 2, t1_nc (t_nc 1.0, t_nc 1.1), t2_nc, and t2e_nc

The overview is in Figure 2. In ETO_PFSP, for each bag, we set two task groups.

Bag 0 is full of Group 1 and Group 2. “Group 1 owns 4 tasks, namely, t1_wc including two sub tasks (t_wc 1.0, t_wc 1.1), t2_wc and t2e_wc, where “wc” means with clustering and “e” is external transferring from t1_wc, sharing the same toolkit of W-X-L (only probabilities vary in X, more is in Figures 3 and 4). All above is the same for group 2 of t1_nc (t_nc 1.0, t_nc 1.1), t2_nc and t2e_nc, except that no clustering (named “nc”) is in W” [6]. In Bag 0, each case calculates the measure of the hamming distance in the job permutation from the 15th one to the last job (just test the special distribution of the positional BB).

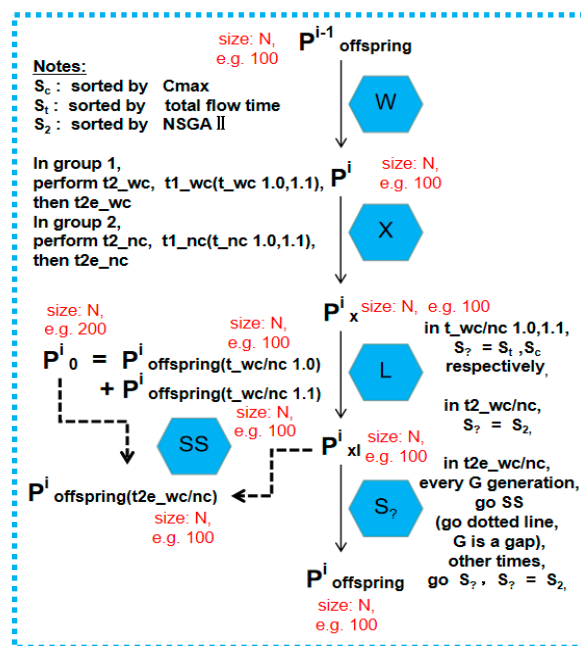


Figure 2. Overview of an abstract SMO framework implemented in memetic algorithms [8].

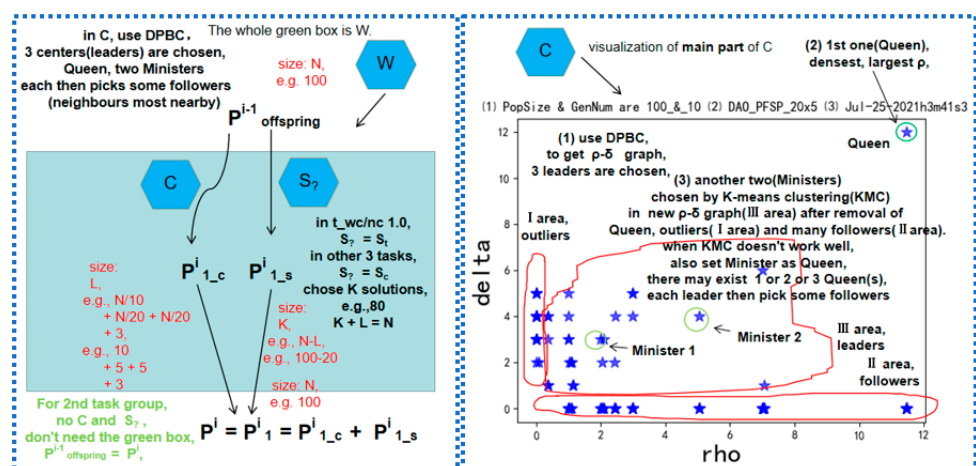


Figure 3. The details of W and C are as listed above. Here, we choose the parents P_i . DPBC is in C. You can see a typical ρ - δ (rho-delta) graph in the simulation, where ρ denotes the local density and δ denotes “the minimum distance between one sample point and any other one with higher density” [8]. Each star is an individual or a solution to solve PFSP in SMO, and 3 red lines characterize 3 areas in the solution space.

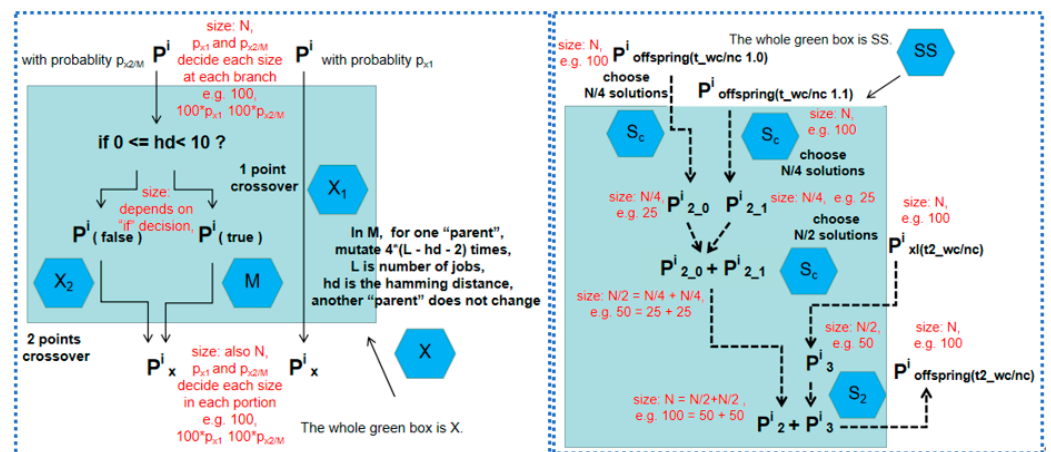


Figure 4. Details of X and SS. “*” in the figure is a multiplication sign in the abstract SMO. M may help us to overcome premature convergence [8].

Bag 1 stores Group 3 and Group 4. Furthermore, the MWT modifications are set (3 modifications of “MWT” are added: (1) just remove M(M) in X; (2) study the whole (W) chromosome, calculate the hamming distance from the first job to last job instead of the 15th job to the last job in Bag 0; and (3) task 1.0 choose TFT(T) as the objective for local search, while all tasks in Bag 0 are set with Cmax for the local search). Then, just remove L from Group 1 and Group 2 in Bag 0 to achieve Group 3 and Group 4 for Bag 1. Then, **Bag 2** stores Group 5 and Group 6. Furthermore, the MWT modifications are set. Then, just remove L from Group 1 and Group 2 in Bag 0 to achieve Group 5 and Group 6 for Bag 2. **Bag 3** is occupied by Group 7 and Group 8. Furthermore, the MWT modifications are set. Just remove X from Group 1 and Group 2. Finally, **Bag 4** is occupied by Group 9 and Group 10. Moreover, the MWT modifications are set. Just remove S from group 1 and group 2.

W-X-L deploys a special operator to choose the parents (W), a crossover (X) operator, and a local search (L) operator. It is worth mentioning that the family of tasks above shares the same initial (I) population (random) for a fair comparison. The phase of selection (S) differs. For S, we use NSGAI, some sorting methods using the Cmax objective or TFT objective, and so on. Therefore, many shared parts above from both problem and algorithm sides are elaborately constituted towards a harmony test bed for a well-defined SMO (Figures 2–4).

3. Results

3.1. Experimental Setup

We choose well-studied open PFSP benchmarks, that is, tai01 (20 × 5), tai42 (50 × 10), and VFR100_20_1 (100 × 20), e.g., the symbol of 20 × 5 denotes 20 jobs and 5 machines.

The following parameters of ETO_PFSF are set: “N is the size of population, set 100, and number of generations is 100. In t1_wc/nc 1.0 and 1.1, [p_{x1}, p_{x2/m}] (seen in operator X in Figure 4) are [0.3, 0.7] and [0.1, 0.9], respectively. For t2_wc/nc and t2e_wc/nc, it’s [0.2, 0.8]. For reference points, tai01 takes range (2500, 1000) to normalize Cmax, and (25,000, 10,000) to normalize TFT; tai42 uses (4200, 2500) and (120,000, 80,000); and instance 3 picks (10,000, 5000) and (550,000, 350,000). The gap G is 2. The base-line size of P₁₂ is 50, modified by a factor K1 (e.g., K1 = 0.6, 50*K1 = 50*0.6 = 30). For P₁₁, the base-line size is 20 + H. And 20 is also adapted by K2 (e.g., K2 = 0.6, 20*K2 = 20*0.6 = 12), H may be 0, 1, 2 or 3, depending on the solutions with equal distance at cutting distance” [6]. By varying (1*3) the setup of [K1, K2] from vectors of [1, 0.6], [0.6, 0.6] to [1, 1] in each instance (*3), we obtain 9 (1*3*3) cases (in Figures 5–7, we only show 2/3 representative cases, and all 9 cases are presented in Table 1), and each case owns 20 total independent runs. In each run, we perform a simulation of eight tasks, that is, the two task groups in Section 2.2.

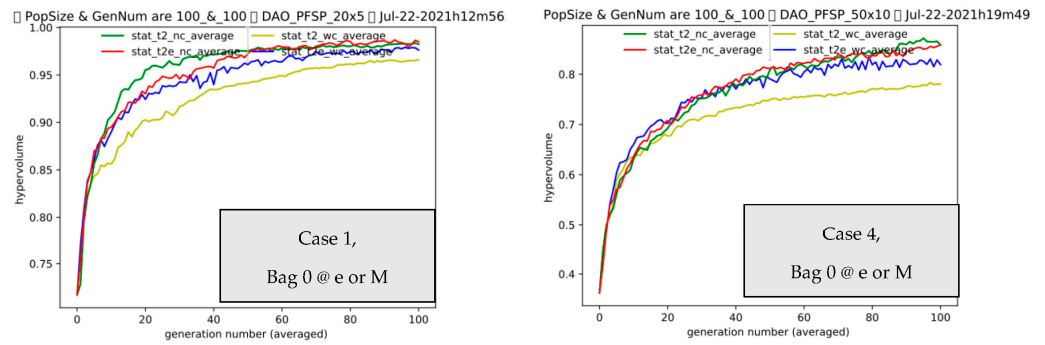


Figure 5. Cases 1 and 4 in Bag 0 are shown above, respectively. We choose representative data of Cases 1 and 4 to save space. “e or M” means eMeets or Meets. Notes: (1) “stat” means statistics of hypervolume. (2) Actually, t2_nc is the baseline NSGAI without both clustering and transferring.

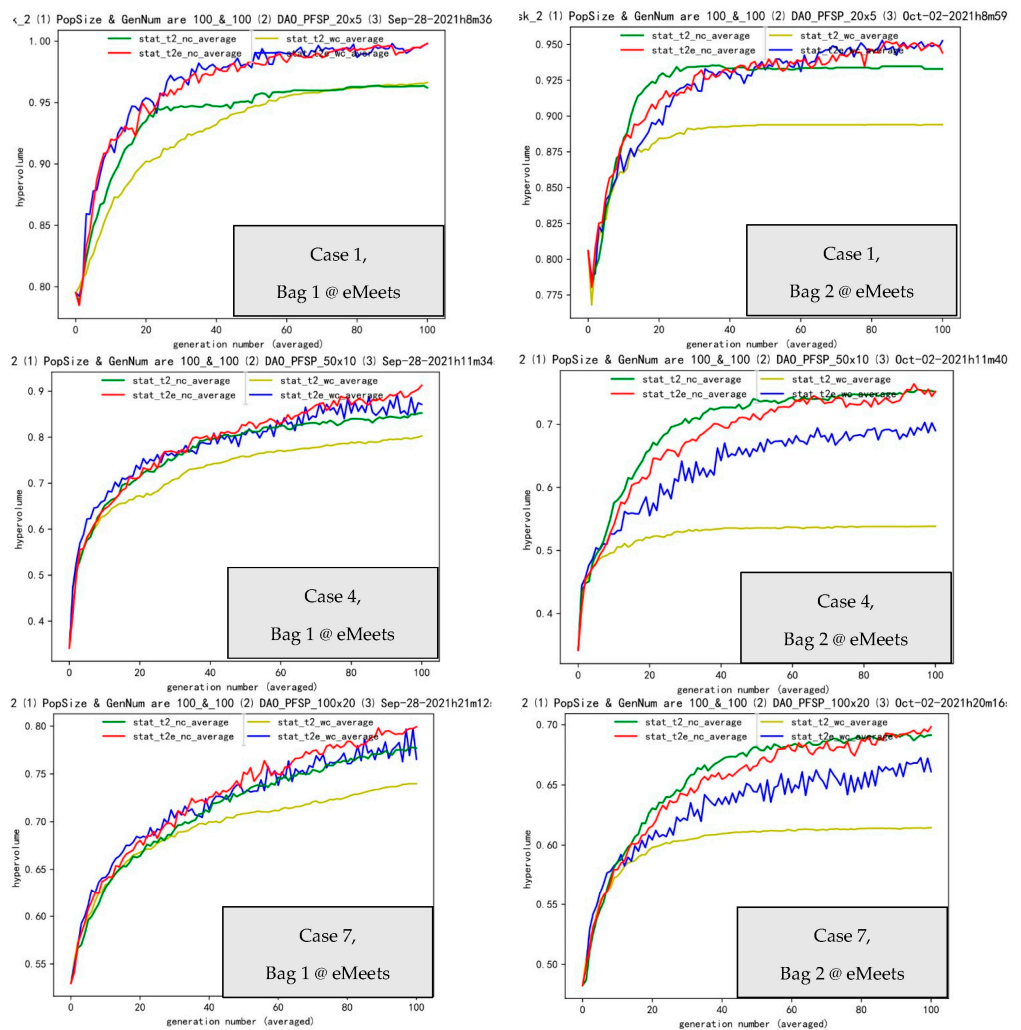


Figure 6. Cases 1, 4, and 7 in Bags 1 and 2 are shown above, respectively. We choose representative data of Cases 1, 4, and 7 to save space.

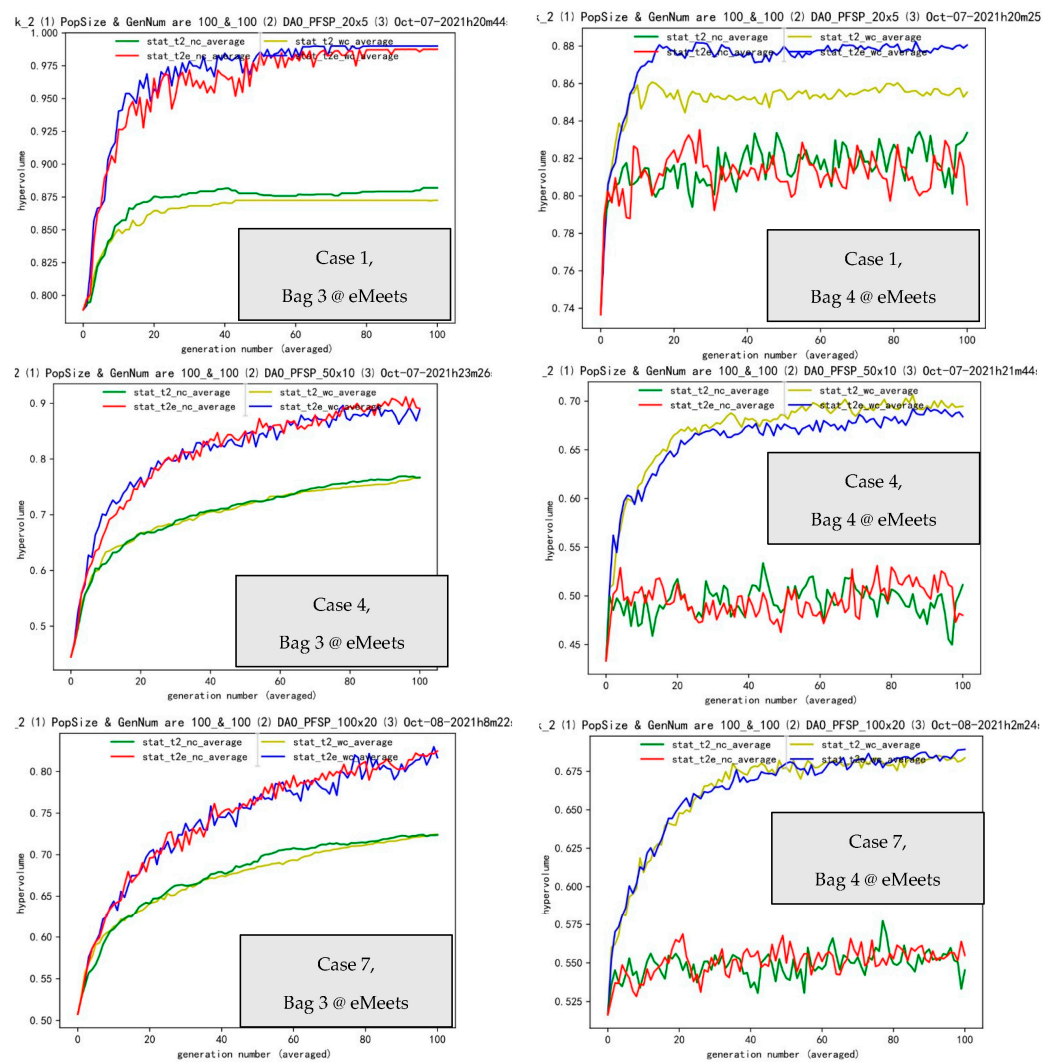


Figure 7. Cases 1, 4, and 7 for Bags 3 and 4 are shown above, respectively.

Table 1. Comparisons of bags.

	Operators	T12	T34
Bag 0	Wxls, XLS, no MWT	e(5ee,4e)	ie(6ie,3=)
Bag 1	Wxls, XLS, +MWT	ee(8ee,1e)	e(4ee,5e)
Bag 2	WXS, XS, +MWT	ee(9ee)	ie(4e,5ie)
Bag 3	WLS, LS, +MWT	ee(9ee)	ee(9ee)
Bag 4	WXL, XL, +MWT		ec

3.2. Simulations and Comparisons

In every case, overall, we evolve 5 (Bags) *100 (generations per task) *4 (4 tasks in each group) *2 (wc/nc, that is wc or nc) *20 (independent runs) = 80,000 generations! Furthermore, the whole size of the solution space or search space is a factorial (for example, 20×5 , means 20! solutions), imposing a tremendous computational challenge for obtaining the 5 (Bags) *9 results or cases (for Cases 7, 8, and 9 in Bag 0, each case takes a long time of 30+ hours, even on a server).

For **Bag 0** [6], between the t2_wc and t2e_wc pair, there is always positive transferring occurring in Figure 5, which tends to validate the successful part of effectiveness (e) in the SMO. Meanwhile, ineffectiveness exists in the comparison between t2_nc and t2e_nc.

These observations can be seen in the details of Table 1 below (in the table, “T12” means the comparison between $t2_wc$ and $t2e_wc$ and “T34” means the comparison between $t2_nc$ and $t2e_nc$. e means effectiveness, ee is great effectiveness, ie stands for ineffectiveness, and ec tells us the effectiveness of the clustering).

Then, we provide a more systematic study of each operator in **Bag 1, 2, 3, and 4**.

For **Bag 1**, between the $t2_wc$ and $t2e_wc$ pair, there is almost always an obvious positive transferring effectiveness in Figure 6, which tends to validate that there is a great effectiveness (ee) part in our ETO_PFSF or SMO framework. However, as for $t2_nc$ and $t2e_nc$, both great effectiveness and normal effectiveness (e) exist. These results can be summed up in Table 1 below.

Then, in **Bag 2** below, we focus on the systems based on Bag 1 but without a local search or memes. Here are another nine cases as follows (we choose the representative data of Cases 1, 4, and 7 to save paper space). As for Bag 2, between the pair of $t2_wc$ and $t2e_wc$, there is definitely an obvious positive transferring effectiveness in Figure 6, which tends to validate a great effectiveness (ee) part in our framework. However, for $t2_nc$ and $t2e_nc$, both normal effectiveness (e) and ineffectiveness (ie) can be seen. These observations are summed up below in the Table 1.

Moving forward to **Bag 3**, we study just the local search tools/sub-systems (we choose representative Cases 1, 4, and 7 to help save space). In Bag 3, regardless of whether it is between the tasks of $t2_wc$ and $t2e_wc$ or between the pair of $t2_nc$ and $t2e_nc$, obvious positive transferring effectiveness exist in Figure 7, which tends to strongly validate great effectiveness (ee) in the SMO. These results are again summed up via Table 1.

Lastly, in **Bag 4**, we explore the W operator (we choose the three representative cases of Cases 1, 4, and 7 to save space). Bag 4 tells us that between $t2_wc$ and $t2e_wc$ and the pair of $t2_nc$ and $t2e_nc$, you can see only some rules in the SMO. The ec is quite obvious. The report is again summed up by Table 1.

4. Discussion

4.1. Simple Case Study of SMO: 4 Additional Findings

From Section 3 above, the findings we gather are as follows:

- (1) We conclude that the transfer effectiveness of the SMO framework is always quite obvious (Bag 4 is aimed to just test selection, not test transferring), especially under clustering conditions ($t2e_wc$ VS $t2_wc$).
- (2) By comparing Bags 1, 2, and 3, we believe that collaboration between X (genetic algorithms) and L are necessary.
- (3) The comparison of two green lines in Bag 3 tells us that clustering or W helps speed up the convergence.
- (4) W or clustering can serve as the selection, whose potentials are obvious in Bag 4 when comparing the group with the clustering/selection and when comparing the group without clustering/selection. In paper [15], clustering-based subset selection has also been proven as an effective selection method.

4.2. “Where to Go” Strategy about Decomposition: Decomposition Styles

Decomposition is all you need when we solve many tasks in applied mathematics and computational intelligence. Two well-known approaches of mathematical programming for combinatorial problems including PFSP are Lagrangian relaxation and column generation, which have been investigated by our groups for production and logistics systems in the steel and iron [2] industry. When used to tackle combinatorial problems like vehicle routing problems with time windows, both are deployed in decomposition styles. Lagrangian relaxation splits the original problem into independent subproblems and column generation turns the targeted problem into a master problem and a subproblem, which again is decomposed into several independent problems. Stepping into evolutionary or meta-heuristics methods, co-evolutionary algorithms are well known for their decomposition power. As for SMO, it roughly divides the pipeline into three components, task 2, task 1.0,

and task 1.1. For example, $t2e_wc = t2_wc + t_wc\ 1.0 + t_wc\ 1.1$, which is firstly proposed here. The subproblems and decomposition also are developed here. As fruitful results are already achieved by Lagrangian relaxation and column generation and as co-evolutionary algorithms may help the newborn ETO, attention to inspiration from these methods for transfer optimization should not be abandoned.

4.3. “Where to Go” Strategy about Transferring: Transfer Family

Both transfer learning and transfer optimization belong to the transfer family. The former includes four classes [16]: transfer learning for classification, transfer learning for regression, transfer learning for clustering, and transfer learning for reinforcement learning. Transfer optimization contains transfer Bayesian optimization (also, it is the type of transfer learning for regression) and evolutionary-based optimization of ETOs (including our SMO, MTO, etc.). In transfer learning, there are [16] transductive transfer learning, unsupervised transfer learning, and inductive transfer learning methods. For inductive transfer learning, the source and target domains are the same. And those two domains are also the same for our SMO framework, that is, our framework shares common properties and settings with inductive transfer learning; however, things are more complex in evolutionary scenarios as domains are not enough. What matters more are the landscapes that are built by domains and operators (“parameter-transfer approach” in transfer learning may relate to the guided crossover probability within the MTO setting). That is, we have already seen some deep connections between transfer learning and ETO. As transfer learning may inspire ETO, we should learn the lessons from transfer learning.

5. Conclusions

What we should highlight here are two of the many advantages of SMO: it is simple and general. Firstly, “simple yet effective” is usually a nice philosophy/rule for sciences, technologies, and engineering (**residual learning** is an outstanding case of a “simple” skip-connection). Keeping the simple rule firmly in our minds, our SMO aims to reduce the horrible complexity in multi-objective (even many objective) discrete spaces in a simple way via decomposition because single-objective tasks are easier than their multi-objective counterparts. Secondly, in contrast to narrow/limited artificial intelligence (AI) “which is created to excel in specific tasks or domains” [1], AGI systems in SMO frameworks “mimic humans’ general-purpose problem-solving abilities” [1].

We believe that a well-designed SMO with specific cases can reach state-of-the-art (SOTA) performance; in this paper, we only provide a rough idea of the abstract and macroscopic framework of SMO and a simple case study. The key problem of SOTA [8] is discussed further there for the abstract and macroscopic SMO.

The study of SMO here will not only push the boundaries of SMO by itself but also inspire multitasking optimization (the mainstream type in ETO now) and even other ETO members for the **transferring** mechanism. One of many most important reasons is that in the ETO survey [20], the authors provide the idea that for the third type of “complex optimization”, simpler artificial tasks sound interesting and encouraging. Following the idea above, we believe that single-objective tasks/versions are simpler and are easier starting points and important types for the entire ETO family.

In the future, many directions seem attractive and inspiring. It is quite important to decompose the knowledge learned/transferred, which to some extent is like the **decomposition** of different features in deep learning for the interpretability of AGI and AI [1–3,21,22].

Author Contributions: Conceptualization, Q.G., D.H., M.Z. and W.X.; methodology, R.Z., X.S. and G.Z.; software, W.X., X.S. and G.Z.; validation, R.Z., X.S. and G.Z.; formal analysis, W.X., X.S. and G.Z.; investigation, Y.Y., T.X. and G.Z.; resources, Y.Y., T.X. and G.Z.; data curation, Y.Y., T.X. and G.Z.; writing—original draft preparation, W.X. and G.Z.; writing—review and editing, D.H., X.W., Q.G., R.Z., X.S. and W.X.; visualization, Y.Y., T.X. and G.Z.; supervision, D.H., Q.G., R.Z. and Y.Y.; project

administration, D.H., Q.G., W.X. and R.Z.; funding acquisition, D.H., Q.G. and M.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (72101052) and the National Key Research and Development Program of China (2021YFC2902403).

Data Availability Statement: Not applicable.

Acknowledgments: The authors thank Zhou Han (DAO), Li Zuocheng (DAO), and Chen Yingping (from National Yang Ming Chiao Tung University) for their useful support and kindly help. Finally, the authors thank the reviewers and editor office for supportive help.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Latif, E.; Mai, G.; Nyaaba, M.; Wu, X.; Liu, N.; Lu, G.; Li, S.; Liu, T.; Zhai, X. Artificial General Intelligence (AGI) for Education. *arXiv* **2023**, arXiv:2304.12479.
2. Tang, L.; Meng, Y. Data analytics and optimization for smart industry. *Front. Eng. Manag.* **2021**, *8*, 157–171. [[CrossRef](#)]
3. Stone, P.; Brooks, R.; Brynjolfsson, E.; Calo, R.; Etzioni, O.; Hager, G.; Hirschberg, J.; Kalyanakrishnan, S.; Kamar, E.; Kraus, S.; et al. Artificial Intelligence and Life in 2030. Available online: <http://ai100.stanford.edu/2016-report> (accessed on 30 August 2023).
4. Gui, W.; Zeng, Z.; Chen, X.; Xie, Y.; Sun, Y. Knowledge-driven process industry smart manufacturing. *Sci. Sin. Inf.* **2020**, *50*, 1345. [[CrossRef](#)]
5. Huang, L.; Feng, L.; Wang, H.; Hou, Y.; Liu, K.; Chen, C. A preliminary study of improving evolutionary multi-objective optimization via knowledge transfer from single-objective problems. In Proceedings of the 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Toronto, ON, Canada, 11–14 October 2020.
6. Xu, W.; Wang, X. ETO meets scheduling: Learning key knowledge from single-objective problems to multi-objective problem. In Proceedings of the 2021 China Automation Congress (CAC), Beijing, China, 22–24 October 2021.
7. Xu, W.; Wang, X.; Guo, Q.; Song, X.; Zhao, R.; Zhao, G.; Yang, Y.; Xu, T.; He, D. Evolutionary Process for Engineering Optimization in Manufacturing Applications: Fine Brushworks of Single-Objective to Multi-Objective/Many-Objective Optimization. *Processes* **2023**, *11*, 693. [[CrossRef](#)]
8. Xu, W.; Wang, X.; Guo, Q.; Song, X.; Zhao, R.; Zhao, G.; Yang, Y.; Xu, T.; He, D. Gathering strength, gathering storms: Knowledge Transfer via Selection for VRPTW. *Mathematics* **2022**, *10*, 2888. [[CrossRef](#)]
9. Xu, W.; Zhang, M. Towards WARSHIP: Combining brain-inspired computing of RSH for image super resolution. In Proceedings of the 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems, Nanjing, China, 23–25 November 2018.
10. Yuan, Y.; Ong, Y.; Gupta, A.; Tan, P.; Xu, H. Evolutionary multitasking in permutation-based combinatorial optimization problems: Realization with TSP, QAP, LOP, and JSP. In Proceedings of the 2016 IEEE Region 10 Conference (TENCON), Singapore, 22–25 November 2016.
11. Feng, L.; Ong, Y.; Lim, M.; Tsang, I. Memetic search with interdomain learning: A realization between CVRP and CARP. *IEEE Trans. Evol. Comput.* **2015**, *19*, 644–658. [[CrossRef](#)]
12. Wang, X.; Tang, L. A machine-learning based memetic algorithm for the multi-objective permutation flowshop scheduling problem. *Comput. Oper. Res.* **2017**, *79*, 60–77. [[CrossRef](#)]
13. Li, Y.; Wang, C.; Gao, L.; Song, Y.; Li, X. An improved simulated annealing algorithm based on residual network for permutation flow shop scheduling. *Complex Intell. Syst.* **2020**, *7*, 1173–1183. [[CrossRef](#)]
14. Chen, Y. *Extending the Scalability of Linkage Learning Genetic Algorithms*; Studies in Fuzziness and Soft Computing; Springer: Berlin/Heidelberg, Germany, 2006.
15. Chen, W.; Ishibuchi, H.; Shang, K. Clustering-based subset selection in evolutionary multiobjective optimization. *arXiv* **2021**, arXiv:2108.08453.
16. Pan, S.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2010**, *22*, 1345–1359. [[CrossRef](#)]
17. Littman, M.; Ajunwa, I.; Berger, G.; Boutilier, C.; Currie, M.; Velez, F.; Hadfield, G.; Horowitz, M.; Isbell, C.; Kitano, H.; et al. *Gathering Strength, Gathering Storms: The One Hundred Year Study on Artificial Intelligence (AI100) Study Panel Report*; Stanford University: Stanford, CA, USA, 2021; Available online: <http://ai100.stanford.edu/2021-report> (accessed on 30 August 2023).
18. Antosz, K.; Machado, J.; Mazurkiewicz, D.; Antonelli, D.; Soares, F. Systems Engineering: Availability and Reliability. *Appl. Sci.* **2022**, *12*, 2504. [[CrossRef](#)]
19. Malozyomov, B.V.; Martyushev, N.V.; Konyukhov, V.Y.; Oparina, T.A.; Zagorodnii, N.A.; Efremenkov, E.A.; Qi, M. Mathematical Analysis of the Reliability of Modern Trolleybuses and Electric Buses. *Mathematics* **2023**, *11*, 3260. [[CrossRef](#)]
20. Tan, K.; Feng, L.; Jiang, M. Evolutionary transfer optimization—A new frontier in evolutionary computation research. *IEEE Comput. Intell. Mag.* **2021**, *16*, 22–33. [[CrossRef](#)]

21. Pan, Y.; Li, X.; Yang, Y. *A Content-Based Neural Reordering Model for Statistical Machine Translation*; Communications in Computer and Information Science; Springer: Singapore, 2017.
22. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is All you Need. *arXiv* **2023**, arXiv:1706.03762.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.