Solution representation learning in transfer evolutionary optimization

Lim, Ray Chee Chuan

2021

Lim, R. C. C. (2021). Solution representation learning in transfer evolutionary optimization. Doctoral thesis, Nanyang Technological University, Singapore.
<https://hdl.handle.net/10356/153156>

<https://hdl.handle.net/10356/153156>

<https://doi.org/10.32657/10356/153156>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 21:24:09 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

**SOLUTION REPRESENTATION LEARNING
IN TRANSFER EVOLUTIONARY
OPTIMIZATION**

RAY LIM CHEE CHUAN

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

2021

SOLUTION REPRESENTATION LEARNING IN TRANSFER EVOLUTIONARY OPTIMIZATION

RAY LIM CHEE CHUAN

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

A thesis submitted to the Nanyang Technological University

in partial fulfilment of the requirement for the degree of

Doctor of Philosophy

2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

01/11/2021

• • • • •

Date

Ray

.....

Ray Lim Chee Chuan

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

02/11/2021

.....

Date

NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU
NTU NTU NTU NTU NTU NTU NTU NTU

.....

Professor Ong Yew Soon

Authorship Attribution Statement

This thesis contains material from two papers published in the following peer-reviewed journals in which I am listed as an author.

The first journal paper is published as R. Lim, A. Gupta, Y. S. Ong, L. Feng and A. N. Zhang, “Non-Linear Domain Adaptation in Transfer Evolutionary Optimization,” *Cognitive Computation*, vol. 13, pp. 290-307, Feb. 2021. DOI: 10.1007/s12559-020-09777-7.

The contributions of the co-authors are as follows:

- Prof. Ong Yew Soon, Dr. Abhishek Gupta and I had discussed and set the research direction. The core idea of this work was to establish a novel probabilistic model-based transfer evolutionary optimization framework that encompasses a solution representation learning module – to facilitate the development of novel algorithms capable of inducing positive transfers across optimization tasks characterized by discrepancies between their original search spaces, while simultaneously curbing the threat of negative inter-task interactions.
- Dr. Gupta and I formalized the notion of solution representation learning in probabilistic model-based transfer evolutionary optimization, and put forward a principled framework for guiding the development of novel algorithms in this respect.
- I conducted the computational experiments and analyzed the numerical results.
- I prepared the manuscript drafts. The manuscript was revised by Dr. Gupta and Prof Ong, who both provided constructive advice for improvements.
- Dr. Feng Liang provided a relevant transfer optimization algorithm for performance comparison with our proposed algorithm.
- Dr. Zhang Nengsheng Allan provided guidance and support in the manuscript preparation.

Abstract

The human cognitive ability to learn with experience is a masterpiece of natural evolution that has yet to be fully duplicated in computational and artificial intelligence systems. When presented with a new task, our brain has the natural tendency to retrieve and reuse knowledge priors acquired from related experiences, thereby speeding up our problem-solving process. In the modern era of data-driven optimization, fuelled by growing amounts of data and seamless information transmission technologies, it is becoming increasingly important for machines to embody the ability to learn from experiences as well. To this end, a recent computational paradigm known as *transfer evolutionary optimization* (TrEO) has emerged to encompass methods that leverage knowledge priors from various *source* optimization tasks to boost the convergence performance in a new but related *target* task. Despite the potential for performance speed-up, the effectiveness of existing TrEO algorithms in actual practice could be hampered by discrepancies between the original search spaces of source and target problems – a common scenario when solving black-box optimization problems of diverse properties, where little is known about their search landscapes *a priori*. In particular, *heterogeneous source and target dimensionalities* or *a lack of overlap between their optimized search distributions* may give rise to *unaligned solution representations* that conceal useful inter-task relationships. This could in turn cause two undesired issues in TrEO, namely, *the scarcity of beneficial positive transfers*, or *the increase of harmful negative transfers*. Even though methodological advances have been made, for instance, the state-of-the-art probabilistic model-based transfer approach for curbing negative inter-task interactions, *there is currently a lack of approaches capable of jointly addressing the two predominant issues in TrEO*.

Taking the cue, this thesis presents a novel study on *solution representation learning in probabilistic model-based TrEO* for inducing greater *alignment* (of solution representations) and hence positive transfers between distinct optimization tasks that bear discrepancies in their original search spaces, while simultaneously curbing the

occurrence of negative transfers. A formalization of solution representation learning in TrEO is established. To this end, the importance and *motivations* of solution representation learning for uncovering useful but hidden inter-task relationships are conceived. Following from the motivations, a principled perspective of *solution representation learning via search space mappings in TrEO* is presented, where a streamlined *definition* is first given. Accordingly, the *source-to-target map* is proposed as a category of spatial mappings by which source tasks are mapped to a transformed and well-aligned solution representation space, such that the discrepancies between various sources and the target task are reduced – in this category, the target search space serves as the transformed (or common) solution representation space. In particular, we outline the *probabilistic formulation of TrEO inclusive of a source-to-target map* that flexibly allows various source-to-target spatial mapping functions to be incorporated. To complete the formalization, a *novel TrEO algorithmic framework* is put forward, endowed with a *solution representation learning module* to induce positive transfers, while *synergized* with a *probabilistic model-based inter-task similarity capture mechanism* to simultaneously mitigate the threat of negative transfers.

Based on the proposed framework, novel TrEO algorithms are developed in the context of *single-objective* and *multi-objective* optimization. Under each context, case studies are presented for *continuous* and *discrete* optimization domains, in which *distinct methodologies for constructing source-to-target spatial mappings are designed for learning well-aligned solution representations*. A range of empirical experiments are conducted to verify the efficacy of the resultant TrEO algorithmic instantiations, using benchmark optimization problems as well as practical tasks in inventory planning, engineering design and graph-based vehicle route planning. The numerical results demonstrate superior convergence rates and solution qualities achieved by the proposed algorithms, in comparison to state-of-the-art TrEO baselines. Further analyses of the results reveal the effectiveness of the source-to-target spatial mappings, and provide insights into the *synergy* between learning solution representations and capturing source-target similarities.

The work of this thesis subsequently sheds light on several promising future research directions for TrEO algorithms that improve their efficacy by learning from various sources of past problem-solving experiences.

Acknowledgements

The pursuit of this Ph.D. degree is a journey of learning about both science and life. I want to express my sincere gratitude to the people who have guided, helped and encouraged me throughout my Ph.D. study.

First of all, I wish to express my utmost gratitude to my supervisor and mentor, Professor Ong Yew Soon. Prof. Ong has shared significant advice and guidance for my research career and professional development. Beyond work, he is also very supportive of my personal growth. I truly appreciate how Prof. Ong has helped me persevere through the research challenges, and inspired a better version of myself.

I am deeply grateful to have Dr. Zhang NengSheng Allan as my co-supervisor from A*STAR. Dr. Zhang has generously shared his knowledge and experiences in research that helped improve my work and research perspectives. I appreciate that he has always been supportive of my research direction, career and personal development.

I wish to express deep gratitude to Dr. Abhishek Gupta for being a great mentor and research partner. Abhishek has spent generous amounts of time to have inspiring and thought-provoking discussions with me, which helped elevate my research skills and the quality of my research output. Furthermore, I say thank you to my mentors, Dr. Zhou Lei, Dr. Joo Byung Jun and Dr. He Tiantian, for the numerous intellectual discussions that helped refine my research ideas and works.

I thank my sponsor A*STAR for the financial support of my Ph.D. study. I am also grateful to my scholarship officers from the A*STAR Graduate Academy, for the moral and administrative support that they have provided during my course of study. Besides, I wish to thank my research group members and friends for kindly sharing their knowledge and for the good times that we had together during my Ph.D. journey: Alvin Chan, Nick Zhang, Qu Xinghua, Bali Kavitesh, Da Bingshui, Chit Lin Su, Hanh Phan, Suttinee Sawadsitang, Dr. Feng Liang, Dr. Wang Zhenkun, Joey Lin, Zhang

Cong and Stephen Zhang. I also thank Mr. Kesavan Asaithambi for all the technical support that he has rendered.

Finally, I want to express my deepest gratitude to my wife, Samantha, and my family members for their love, kind understanding and unconditional support throughout my Ph.D. journey.

Contents

Abstract	i
Acknowledgements	iii
Contents	v
List of Figures	viii
List of Tables	xi
1 Introduction	1
1.1 Research Background	1
1.1.1 Motivations	2
1.2 Research Objectives	3
1.2.1 List of Publications	5
1.3 Thesis Organization	6
2 Literature Review	8
2.1 Transfer Learning	8
2.2 Traditional Evolutionary Algorithms	9
2.3 The Transfer Evolutionary Optimization (TrEO) Paradigm	10
2.3.1 Early TrEO Methods	11
2.3.2 Toward Adaptive TrEO Methods	13
2.3.3 Other Practical Applications and Multi-Objective Problem-Solving in TrEO	15
2.4 Chapter Summary	17
3 A Formalization of Solution Representation Learning in TrEO	19
3.1 Probabilistic View of TrEO	20
3.1.1 Preliminaries	20
3.1.2 Probabilistic Formulation of TrEO	21
3.2 Motivating Solution Representation Learning in TrEO	24
3.3 Solution Representation Learning via Search Space Mappings in TrEO	27

3.3.1	Source-to-Target Map: M_S	28
3.4	A Novel Framework for Solution Representation Learning in Probabilistic Model-Based TrEO.	30
3.4.1	The Solution Representation Learning (SRL) Module	30
3.4.2	Probabilistic Model-Based Transfer.	32
3.4.2.1	Probabilistic Model-Based Evolutionary Search	33
3.4.2.2	Inter-Task Similarity Capture Mechanism	34
3.4.3	Summary of Algorithmic Framework	36
3.5	Chapter Summary	38
4	TrEO with Source-to-Target Map: TrEO+M_S	40
4.1	The TrEO+ M_S Algorithm.	41
4.2	Case Study in Discrete Optimization: A 0-1 Knapsack Problem Exemplar	44
4.2.1	Methodology for Constructing M_S	45
4.2.2	Experimental Study	47
4.2.2.1	Experimental Setup	47
4.2.2.2	Results and Discussion.	48
4.3	Case Studies in Continuous Optimization	49
4.3.1	Methodology for Learning M_S	50
4.3.2	A Study of Synthetic Benchmark Functions.	53
4.3.2.1	Experimental Specifications	54
4.3.2.2	Results and Discussion.	55
4.3.3	A Real-World Example on Multi-Location Inventory Planning	58
4.3.3.1	Experimental Setting.	59
4.3.3.2	Results and Discussion.	60
4.4	Chapter Summary.	63
5	Multi-Objective TrEO with Source-to-Target Map: MOTrEO+M_S	65
5.1	Multi-Objective Optimization.	66
5.2	The MOTrEO+ M_S Algorithm.	67
5.2.1	Motivations.	67
5.2.2	Algorithm Summary	68
5.3	Case Studies in Multi-Objective Continuous Optimization.	70
5.3.1	Methodology for Learning M_S	71

5.3.2	A Study of Complex Benchmark Tasks	75
5.3.2.1	Experimental Specifications	75
5.3.2.2	Results and Discussion	77
5.3.3	A Real-World Example on Multi-Objective Vehicle Crashworthiness Design	81
5.3.3.1	Experimental Setup.	82
5.3.3.2	Results and Discussion	82
5.4	A Practical Case Study in Multi-Objective Vehicle Route Planning .	84
5.4.1	Problem Formulations	85
5.4.1.1	Source Task: Truck Delivery.	86
5.4.1.2	Target Task: Multi-Objective Drone Delivery. .	86
5.4.2	Methodology for Constructing M_S	87
5.4.3	Experimental Study.	91
5.4.3.1	Source and Target Instances	91
5.4.3.2	Experimental Configuration	92
5.4.3.3	Results and Discussion.	92
5.5	Chapter Summary	94
6	Conclusions and Future Work	96
6.1	Conclusions	96
6.2	Future Research Directions	98
6.2.1	Target-to-Source Map	100
6.2.2	Source-cum-Target Maps	102
	Abbreviations	105
	Notations of Chapter 3	107
	Notations of Chapter 4	109
	Notations of Chapter 5	112
	Notations of Chapter 6	116
	Bibliography	119

List of Figures

1.1	A visualization of the proposed TrEO framework that comprises a solution representation learning module to induce positive transfers, as well as a source-target similarity capture mechanism to simultaneously mitigate negative transfers	4
2.1	Block diagrams that visually summarize how the present thesis takes motivation from the shortcomings discovered in notable state-of-the-art adaptive TrEO methods. (a) Adaptive model-based TrEO. (b) TrEO with adaptive learning across heterogeneous problems. (c) A novel framework proposed for solution representation learning in adaptive probabilistic model-based TrEO	18
3.1	An example where a search space mapping $M(x) = -x$ applied to the source solutions induces strong (positive) correlation between the transformed source function $f_S(M(x))$ and the target fitness function $f_T(x)$, resulting in greater overlap between their respective optimized search distributions	25
3.2	Illustration of the proposed source-to-target map for SRL in TrEO. .	29
3.3	Graphical illustration of the proposed framework, featuring the SRL module and the probabilistic model-based transfer mechanism as essential components of a resultant TrEO algorithm	32
4.1	Illustrative flowchart of the proposed TrEO+ M_S algorithm, featuring the possibility of an <i>offline</i> construction or an <i>online</i> learning of the source-to-target mapping M_S . The solution representation learning process has been highlighted	43
4.2	Convergence trends and transfer coefficients achieved for target task “target_KP_sc” (top row), “target_KP_wc” (central row), and “target_KP_uc” (bottom row). The central and rightmost columns depict the transfer coefficients obtained by the TrEO+ M_S and the baseline AMTEA, respectively. All results are averaged over 30 independent runs of each optimizer. The shaded region spans one standard deviation on either side of the mean	49

4.3	Convergence trends for the target shifted Sphere function, and the transfer coefficients obtained by the AMTEA and the TrEO+ M_S in <i>Case 1</i> : $d_T = d_S$ (left column), <i>Case 2</i> : $d_T < d_S$ (central column), and <i>Case 3</i> : $d_T > d_S$ (rightmost column). Source data comes from the multimodal Griewank function. The shaded region spans one standard deviation on either side of the mean (based on 30 independent runs)	56
4.4	Convergence trends for the target multi-location inventory planning task, and the transfer coefficients obtained by the AMTEA and the TrEO+ M_S in <i>Case 1</i> : $d_T = d_S$ and $\mu_T = \mu_S$ (left column), <i>Case 2</i> : $d_T = d_S$ but $\mu_T \neq \mu_S$ (central column), and <i>Case 3</i> : $d_T > d_S$ and $\mu_T \neq \mu_S$ (rightmost column). The shaded region spans one standard deviation on either side of the mean (based on 30 independent runs).	61
4.5	Performance gap between the TrEO+ M_S and the AMTEA for the multi-location inventory cost minimization problem. The gap is measured by subtracting the objective values achieved by TrEO+ M_S from those achieved by AMTEA in Figure 4.4. The plot shows that the performance boost in the TrEO+ M_S is more significant as the extent of source-target domain mismatch increases	62
5.1	(a) The preference relationship between solutions in a multi-objective setting (assuming objective function maximization). (b) Source and target populations monotonically aligned according to decreasing order of preference.	72
5.2	An illustration of the proposed methodology for learning well-aligned solution representations in multi-objective continuous search spaces, using a two-layer feedforward neural network.	74
5.3	Convergence trends for the complex multi-objective benchmark tasks in Table 5.2. Results are for distinct problem configurations: “high similarity” (leftmost column), “medium similarity” (central column), “low similarity” (rightmost column), “complete intersection” (top row), “partial intersection” (middle row), and “no intersection” (bottom row). The shaded regions span one standard deviation on either side of the mean (based on 20 independent runs).	78
5.4	The transfer coefficients obtained by the AMTEA and the MOTrEO+ M_S in three representative cases. (a) CIHS. (b) PILS. (c) NILS	79
5.5	The transfer coefficients deduced in the MOTrEO+ M_S are large only when the learned mapping is effective in inducing positive transfers. The effectiveness of M_S learned is measured by the average Euclidean distance between the transformed source solutions and their corresponding nearest target Pareto optimal solution. For instance, the effectiveness of M_S is considered to be “high”, “medium” or “low” when the distances are “< 0.3”, “0.3 – 0.5” or “> 0.5”, respectively	80

5.6	Convergence trends for the target multi-objective vehicle crashworthiness design optimization task and the transfer coefficients learned by the AMTEA and the MOTrEO+ M_S in <i>Case 1</i> : $d_T = d_{S1}, d_T = d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$ (left column), <i>Case 2</i> : $d_T > d_{S1}, d_T > d_{S2}$ and $\mu_T = \mu_{S1}, \mu_T \neq \mu_{S2}$ (central column), and <i>Case 3</i> : $d_T = d_{S1}, d_T > d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$ (rightmost column)	84
5.7	The importance of solution representation alignment in the effectiveness of knowledge transfers between two distinct vehicle routing tasks, featuring the overlap of their optimal sub-routes. (a) The known optimal routes of source graph G . (b) The (<i>a priori</i> unknown) optimal routes of target graph H	88
5.8	A graphical illustration of the proposed methodology to learn well-aligned solution representations for distinct VRPs. (a) The original source and target graphs. (b) The source-to-target transformation M_S in Step 1. (c) The source node index reassignments in Step 2.	90
5.9	Convergence trends for the target multi-objective optimization task of drone delivery route planning, and the transfer coefficients obtained by the AMTEA and the MOTrEO+ M_S , for target instances Golden_17 (left column), X-n214 (central column) and CMT4 (rightmost column). The shaded regions span one standard deviation on either side of the mean (based on 20 independent runs)	94
6.1	Illustration of the <i>target-to-source map</i> as a possible category of spatial mappings for SRL in TrEO	101
6.2	Illustration of the <i>source-cum-target maps</i> as a possible category of spatial mappings for SRL in TrEO	103

List of Tables

4.1	Experimental results for the target shifted Sphere function, at the end of 5000 function evaluations. In bold are the best results with 95% confidence level based on the Wilcoxon signed-rank test. Results are reported based on 30 independent runs of all optimizers.	55
4.2	Experimental results for the target multi-location inventory planning task, at the end of 3000 function evaluations. In bold are the best results with 95% confidence level based on the Wilcoxon signed-rank test. Results are reported based on 30 independent runs of all optimizers.	60
5.1	Complex multi-objective benchmark tasks and their properties. . . .	76
5.2	Experimental results for the multi-objective benchmark tasks after 50,000 function evaluations. In bold are the best results (values with 95% confidence level based on the Wilcoxon Signed-Rank Test are marked with “*”). Results are reported based on 20 independent runs of all optimizers	77
5.3	Experimental results for the multi-objective vehicle crashworthiness design tasks after 300 function evaluations. In bold are the best results (values with 95% confidence level based on the Wilcoxon Signed-Rank Test are marked with “*”). Results are reported based on 20 independent runs of all optimizers.	83
5.4	Experimental results for the target multi-objective drone delivery route planning tasks after 10,000 function evaluations. In bold are the best results (values with 95% confidence level based on the Wilcoxon Signed-Rank Test are marked with “*”). Results are reported based on 20 independent runs of all optimizers.	93

Chapter 1

Introduction

1.1 Research Background

The human cognitive ability to learn with experience is a masterpiece of natural evolution that has yet to be fully duplicated in computational and artificial intelligence systems. When presented with a new task, our brain has the natural tendency to retrieve and reuse knowledge priors acquired from related experiences, thereby speeding up our problem-solving process [1, 2]. The idea of leveraging *knowledge transfer* to enhance the performance of artificial intelligence systems has earlier been explored in the machine learning literature under the label of *transfer learning* [3, 4]. The notion of transfer learning is based on exploiting existing data from past experiences to boost the learning efficacy in related new tasks.

Now, in the modern era of data-driven optimization that is fuelled by growing amounts of data (knowledge) and seamless information transmission technologies, it is becoming increasingly important for intelligent optimizers to embody the ability to learn from experiences as well [5, 6, 7]. To this end, a relatively recent computational paradigm known as *transfer evolutionary optimization* (TrEO) has emerged to encompass methods that leverage knowledge priors from various *source* optimization tasks to boost the convergence performance in a new but related *target* task [7, 8].

Although the development of TrEO is still in its nascent stages, some methodological advances have been made. On one hand, *sequential transfer* methods [8, 9, 10] have catered to the reuse of knowledge priors acquired from past problem-solving experiences; much similar to the notion of transfer learning in the machine learning context. On the other hand, *evolutionary multitasking* [11, 12] is yet another family of methods that gives rise to the possibility for mutual knowledge exchange

among multiple optimization problems tackled in unison. *In this thesis, we focus on the sequential transfer of knowledge in TrEO, which exploits source data available from a knowledge base consisting of previously optimized source tasks.*

What logically follows is that real-world problems seldom exist in isolation, especially given the common circumstances of new but similar tasks arising on a periodic basis; Section 2.3 shall highlight several examples of application areas that involve such repeated problem-solving. The TrEO paradigm thus takes this wisdom to store and reuse source data instead of solving newly encountered problems from scratch each time. However, one of the main challenges of TrEO lies in the ability to leverage *relevant* knowledge priors which are beneficial for the target (i.e., current) problem at hand.

1.1.1 Motivations

A typical assumption in many of the existing TrEO methods is that the source and target tasks share significant overlap of feature (e.g., dimensionality) in their originally defined search spaces, as well as the optimum distribution of solutions in those spaces [13]. As a consequence of this assumption being true, transferring knowledge from the source task leads to *positive transfers* which enhance performance of the target search. In contrast, when the transferred solutions (i.e., knowledge) turn out to be useless for the target task, the onus is placed on evolutionary selection to gradually sieve out the irrelevant information. The extra effort spent in sieving naturally impedes the target search, leading to a phenomenon that has come to be referred to as *negative transfers* in optimization [7, 8].

Despite the potential for performance speed-up, the effectiveness of TrEO algorithms in actual practice could be hampered by the scarcity of beneficial positive transfers or the occurrence of harmful negative transfers. Specifically, in many applications it is possible that useful inter-task relationships are concealed by *discrepancies* between the source and target tasks – characterized by (i) *heterogeneous source and target dimensionalities*, or (ii) *a lack of overlap between optimized source and target solution distributions*, or both; thus manifesting in *unaligned solution representations* that may cause reduced positive transfers or even increased negative

transfers [14]. This concern (i.e., presence of unaligned solution representations) is especially crucial when TrEO methods are used to solve complex and real-world optimization tasks of diverse problem dimensionality and characteristics.

It is then imperative to develop TrEO approaches that are capable of handling the aforementioned discrepancies between the source and target tasks; for example, search space transformation methods for *learning well-aligned solution representations* that reduce such discrepancies and hence induce positive inter-task transfers. Moreover, equipping algorithms with the ability to mitigate negative influences from dissimilar sources is another crucial aspect in TrEO. Although there are some adaptive TrEO algorithms designed recently for those purposes, most have *solely* focused either on (i) designing spatial transformation methodologies to increase positive transfers, or on (ii) curbing detrimental negative transfers from unrelated sources. Nevertheless, *we contend that it is vital to tackle the two issues concurrently in order to tap the desired potential of TrEO*. The present TrEO literature, however, reveals *a lack of approaches capable of jointly addressing both issues*.

Taking the above cue, this thesis presents a novel study on *solution representation learning in probabilistic model-based TrEO* for inducing greater *alignment* (of solution representations) and hence positive transfers, as well as for simultaneously mitigating negative transfers between distinct optimization tasks that bear discrepancies in their original solution representation (or search) spaces.

1.2 Research Objectives

The work of this thesis takes motivation from the research issues found in the current TrEO literature as indicated in the previous section. The following points outline the research objectives of this thesis, clearly stating how the identified research issues are addressed.

- **Objective 1:** Formalize the notion of solution representation learning in probabilistic model-based TrEO, by first detailing its *motivations* and importance. Following from the motivations, present a principled perspective of *solution representation learning via search space mappings in TrEO*, which first gives a

streamlined *definition* of this notion. Accordingly, propose the *source-to-target map* as a category of spatial mappings by which source tasks are mapped (or transformed) to the target search space (which serves as a common solution representation space), such that the discrepancies between various sources and the target task are reduced. In particular, to outline the *probabilistic formulation of TrEO with a source-to-target map* that flexibly allows various source-to-target spatial mapping functions to be incorporated. Thereafter, put forward a *novel TrEO algorithmic framework* that is (i) endowed with a *solution representation learning module* to induce positive transfers through learning well-aligned solution representations between source and target tasks, and (ii) equipped with a *probabilistic model-based source-target similarity capture mechanism* to simultaneously mitigate negative transfers; the framework thus provide guidance in developing adaptive algorithms capable of jointly addressing the two predominant issues in TrEO; see Section 1.1.1. Figure 1.1 gives a visual illustration of the proposed framework.

FIGURE 1.1. A visualization of the proposed TrEO framework that comprises a solution representation learning module to induce positive transfers, as well as a source-target similarity capture mechanism to simultaneously mitigate negative transfers.

- **Objective 2:** Following from the proposed framework, develop *a novel probabilistic model-based TrEO algorithm with source-to-target spatial mappings*. Further, demonstrate the algorithm’s *generalizability* to discrete and continuous optimization domains that are of distinct solution representation spaces and problem structures. To this end, design methodologies for constructing source-to-target spatial mappings in each domain, with the aim of uncovering useful but hidden inter-task relationships – specifically, guide the design by the following desired abilities to (i) *handle arbitrarily heterogeneous dimensionalities*, and (ii) *induce greater overlap between the optimized search distributions of source and target tasks*. This is so that well-aligned solution representations can be learned, hence positive transfers can be induced.
- **Objective 3:** Showcase how the concepts in “Objective 2” are similarly applicable to the context of *multi-objective optimization*. In tandem, introduce *a novel multi-objective probabilistic model-based TrEO algorithm with source-to-target mapping*, which is notably one of the first multi-objective TrEO solvers capable of jointly activating beneficial positive transfers (through solution representation learning) and curbing detrimental negative transfers (by probabilistic mixture modeling).
- **Objective 4:** Validate the resultant TrEO algorithmic instantiations through a range of *experimental case studies*, considering well-defined benchmark optimization problems as well as practical TrEO applications of interest such as operations research and engineering design tasks. Show that by virtue of the *synergy* between solution representation learning and inter-task similarity capture, the proposed TrEO algorithms are capable of inducing positive transfers while reducing negative transfers, thus bringing about faster search convergence to solutions of high quality.

1.2.1 List of Publications

The research objectives detailed above have been achieved and affirmed in the following peer-reviewed publications written by the author of this thesis:

- **Published:** A journal article that has been accepted in September 2020 for publication in *Cognitive Computation*, with the title “Non-Linear Domain Adaptation in Transfer Evolutionary Optimization” [13].
- **Published:** A journal article that has been accepted in March 2021 for publication in *IEEE Access*, with the title “Solution Representation Learning in Multi-Objective Transfer Evolutionary Optimization” [14].

1.3 Thesis Organization

The rest of this thesis is organized as follows:

- Chapter 2 provides the literature review of related works and research advances within the scope of this thesis. The notion of transfer learning and its relevance to the present thesis are first described. A historical overview of traditional evolutionary optimizers is then provided, revealing both their advantages and limitations which consequently motivate the emergence of TrEO algorithms. This is followed by a comprehensive review of the TrEO paradigm, spanning early and recent adaptive TrEO methods, as well as their practical applications and shortcomings. In addition, the prospective multi-objective TrEO research is discussed.
- Chapter 3 establishes the formalization of solution representation learning in probabilistic model-based TrEO, with the aim of addressing the limitations of recent methods. A probabilistic view of TrEO is first provided, with an emphasis on a state-of-the-art probabilistic model-based transfer approach (with an inter-task similarity capture mechanism) for curbing negative transfers. The shortcomings of the aforementioned approach are then detailed, which serve as the motivations for learning well-aligned solution representations in TrEO. Thereafter, a principled perspective of solution representation learning via search space mappings in TrEO is presented, where a streamlined definition is first given. Accordingly, the source-to-target map is proposed as a category of spatial mappings aimed at reducing the discrepancies between various sources and the target task (hence inducing positive inter-task transfers). In particular, the probabilistic formulation of TrEO with a

source-to-target map is outlined. Finally, a novel TrEO algorithmic framework is put forward, which is endowed with a solution representation learning module (for activating positive transfers) and synergized with a probabilistic model-based transfer mechanism (for simultaneously curbing negative transfers).

- Chapter 4 first develops a novel probabilistic model-based TrEO algorithm with source-to-target map, namely, the $\text{TrEO}+M_S$, in accordance with the framework in Chapter 3. Thereafter, case studies are presented for the discrete and continuous optimization domains, in which distinct methodologies for constructing source-to-target spatial mappings are designed to learn well-aligned solution representations. The resultant algorithmic instantiations are verified using a range of numerical experiments, including a 0-1 Knapsack Problem as a discrete decision-making exemplar, continuous benchmark optimization tasks, and a real-world example on multi-location inventory planning.
- Chapter 5 showcases that the proposed framework from Chapter 3 is applicable to the context of multi-objective TrEO as well. Firstly, a brief background on multi-objective optimization is given, and the limitations of existing multi-objective TrEO algorithms are highlighted. To overcome the limitations, a novel multi-objective TrEO algorithm with source-to-target map, known as the $\text{MOTrEO}+M_S$, is developed. Case studies are then presented for the multi-objective optimization domains of continuous tasks and a class of graph-based combinatorial problems, in which corresponding source-to-target spatial transformation methodologies are designed for learning well-aligned solution representations. Under the case studies, numerical experiments are conducted to verify the efficacy of the respective algorithmic instantiations. The experimental studies span complex benchmark functions, as well as real-world multi-objective optimization tasks in engineering design and intelligent logistics planning involving unmanned aerial vehicles.
- Chapter 6 concludes this thesis with a summary of research outcomes and analyses, and suggests several research directions for future work.

Chapter 2

Literature Review

In this chapter, we review existing works in the literature that are relevant to the realm of *transfer evolutionary optimization* (TrEO). We first briefly describe (i) the notion of transfer learning in the machine learning context and (ii) the popular use of evolutionary algorithms in *traditional* problem-solving. Following from the two topics, we highlight several features that inspire and act as the enabler of the *TrEO paradigm*, which is discussed subsequently. To this end, we look at some of the early realizations of TrEO methods and identify their limitations; this brings us to the discussion of several recent *adaptive* TrEO approaches that aim to overcome the limitations their earlier (non-adaptive) counterparts. Accordingly, we reveal the shortcomings of the existing adaptive TrEO algorithms, as well as highlight the emerging research in *multi-objective* TrEO that serve as the motivations of the work in the remainder of this thesis.

2.1 Transfer Learning

In the machine learning context of *predictive analytics*, the idea of leveraging available data from an existing (source) task or *domain* (i.e., a feature space and its density distribution) to boost the learning performance of a new (target) task or domain has been widely explored under the label of transfer learning [3, 4, 15, 16, 17]. The definition of transfer learning, according to [4], is stated as,

Definition 2.1 (*Transfer Learning*): Given a learning task $\mathcal{T}_S$ defined in a *source* domain $\mathcal{D}_S$, and a learning task $\mathcal{T}_T$ defined in a *target* domain $\mathcal{D}_T$, transfer learning aims to improve the learning of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$.

Various aspects of transfer learning have been studied in the machine learning literature, under the guidance of Definition 2.1. Among a plethora of techniques, those of particular interest in this thesis include (i) *online transfer learning* [18, 19] – during the learning process of the target task $\mathcal{T}_T$, source knowledge (comprising $\mathcal{D}_S$ and $\mathcal{T}_S$) is exploited on the fly, and (ii) *domain adaptation* [20, 21] – when learning a task in the target domain $\mathcal{D}_T$, data or density distribution model from a different but related source domain $\mathcal{D}_S$ is extracted and adapted to $\mathcal{D}_T$.

Taking our attention back to the context of *search and optimization*, the notion of leveraging knowledge transfer to enhance convergence performance is in part inspired by the widespread study of transfer learning in the context of predictive analytics. In the past, while some optimization algorithms may attempt to jump start their search processes by simply reusing existing solutions (which will be further elaborated in Section 2.3.1 on “Early TrEO Methods”), many (non-TrEO) solvers typically start their search from scratch without exploiting knowledge from related past experiences. Take evolutionary optimizers as example. Despite their popularity as general black-box optimization solvers, they are traditionally designed to optimize every newly encountered problem from ground zero. The next section hence provides a historical overview of *traditional (non-TrEO) evolutionary algorithms*, and highlights several of their features that (in parallel with the concept of transfer learning) inspire the TrEO paradigm subsequently.

2.2 Traditional Evolutionary Algorithms

Evolutionary algorithms (EAs) are population-based search procedures that are a popular choice for solving a variety of practical problems. Besides being able to obtain reasonable approximations of optimal solutions, they are known to be effective for tackling complex optimization problems with highly non-linear, non-differentiable, or even black-box objectives and constraint functions [22, 23].

Typically, EAs for optimization problem-solving can generally be classified into instance-based or model-based optimization methods [24]. Instance-based EAs generate new candidate solutions by performing recombination and/or mutation on the current population of solutions, whereas model-based EAs do so by building and

sampling a model (e.g., probabilistic model) that captures the underlying distribution of solutions in the current population [25]. Due to the *generality* and *algorithmic simplicity* of EAs, they have been widely used with much success in a plethora of real-world applications [26, 27, 28].

Despite their advantages, the largely problem-independent nature of their search operators implies that EAs may sometimes be slow to converge to optimal (or even near-optimal) solutions. Moreover, the prevalence of slow convergence when solving complex real-world problems in resource-constrained settings is undesirable, especially among those traditional (non-TrEO) EAs that are designed to solve each newly encountered optimization task from scratch (without any attempt to boost their search performance by learning from related problem-solving experiences) [8]. These issues have in fact ignited the opportunity for knowledge transfers in evolutionary optimization; we shall review some early realizations of knowledge transfer methods in Section 2.3.1, followed by more recent adaptive approaches in Section 2.3.2.

Given (i) *the rich pool of promising techniques and results evident in the transfer learning literature*, (ii) *the flexibility of EAs in the hybridization between problem-solving and learning strategies*, and (iii) *the realization that many optimization tasks seldom exist in isolation*, there has been increasing research interest in the design of EAs that could take advantage of available *knowledge priors* relevant to the current task at hand to boost its search performance. This has led to the emergence of the *TrEO paradigm*, which will be elaborated in greater detail in the next section.

2.3 The Transfer Evolutionary Optimization (TrEO) Paradigm

Inspired by the potential for performance speed-up of EAs through knowledge transfer, *transfer evolutionary optimization (TrEO)* has emerged quite recently as a computational paradigm which encompasses evolutionary solvers that exploit useful information from various source tasks to enhance optimization efficiency on a related target task [7, 8]. With reference to [8], the definition of TrEO can be generally stated as,

Definition 2.2 (*Transfer Evolutionary Optimization*): Given a source optimization task $\mathcal{T}_S$ defined in a search space $\mathcal{X}_S$, and a target optimization task $\mathcal{T}_T$ defined in a space $\mathcal{X}_T$, TrEO aims to improve the solution of the target objective function $f_T(\cdot)$ of $\mathcal{T}_T$ using the knowledge in $\mathcal{X}_S$ and $\mathcal{T}_S$.

Similar to transfer learning, the TrEO paradigm takes motivation from the human cognitive ability to speed up problem-solving processes by learning from related experiences [29, 30]. Furthermore, the competitive business landscapes and increasing problem complexities of today, as well as the growing amounts of data generated and technologies for seamless information transmission [5, 6, 31, 32], altogether incentivize the idea of *rapid* optimization made feasible by TrEO. As also mentioned in Section 1.1, there exists practical application areas that involve repeated problem-solving such as production and inventory planning [33, 34, 35], vehicle routing [36, 37], and personnel scheduling [38], to name a few – which are likely to benefit from knowledge transfers (between related and recurring tasks) to achieve rapid optimization as well; Section 2.3.3 shall further highlight other applications that are receiving research attention in this paradigm.

Under the TrEO label, novel EAs (known as *TrEO algorithms, methods or approaches* hereafter) have been designed to incorporate inter-task learning mechanisms, which enable the exploitation of useful information from related tasks. In what follows, we review some of the early realizations of TrEO algorithms and identify their limitations.

2.3.1 Early TrEO Methods

Although the TrEO paradigm is still in its nascent stages, a number of methodological developments have been made. We note that some of the earliest TrEO methods are based on the direct reuse of high quality solutions from related source tasks – such approaches can also be referred to as *direct genetic transfer* [8, 39]. A case-based reasoning approach is proposed to reuse previously obtained optimal solutions when solving new scheduling tasks [40]. Other similar methods have been proposed to enhance the target search performance through direct genetic transfers (from past experiences into a related target problem) – including the direct injection of

intermediate source solutions [41], and the reuse of source solutions to bias the re-planning of robotic navigation in dynamic environments [42]. In [43], a knowledge base of source problems is put through case-by-case assessment to select the most useful source solutions, which are then reused to guide the target search for several classes of combinatorial optimization problems.

A study has shown that directly exploiting prior model structures obtained from past optimization runs can increase the likelihood of discovering useful model structures in model-based EAs, which enable faster convergence in new but related problems [44]. A bootstrapping approach is designed to use a previously obtained population of solutions as biases for the current search, when optimizing competitive coevolution tasks in a robotic controller application [45, 46]. In [9], a culture-inspired mechanism is proposed to select and inject the most relevant past experiences, which is shown to achieve high quality initialization of solutions in newly encountered graph-based routing problems. Some studies have also demonstrated that algorithmic performance can be improved by extracting and reusing genetic programming code fragments from lower-level solutions; such as when solving learning classifier systems [47, 48], large-scale Boolean problems [16], and images classification tasks [17].

Despite the promising outcomes demonstrated by early realizations of TrEO algorithms, they are generally reliant on a practitioner’s domain expertise for *manually* determining the most related solutions or tasks to be exploited; the typical assumption being that similar tasks often give rise to similar solutions [41]. The crux of the TrEO paradigm thus lies in harnessing such useful knowledge, which in turn translates to beneficial *positive transfers* that boost the target convergence performance. In contrast, in cases where the transferred knowledge turns out to be useless for the target task, the onus is placed on evolutionary selection to gradually sieve out the irrelevant information [8]. The extra effort spent in sieving naturally impedes the target search, a phenomenon that has come to be referred to as *negative transfer* in optimization. Besides, the manual selection of appropriate sources will become increasingly inefficient as the size of the knowledge base grows with experiences [49, 50]. Furthermore, in practice, useful inter-task relationships could remain hidden in the source problem’s originally defined search space – this causes reduced positive

transfers, hence renders methods that execute direct knowledge transfers (as discussed above) to be less effective for tapping the desired potential in TrEO.

More recently, there has been increasing research attention given toward devising *adaptive TrEO methods* capable of overcoming the aforementioned limitations of their earlier (non-adaptive) counterparts. The next part of this section discusses some notable works that aim to further enhance the performance advantage of TrEO algorithms via adaptation strategies.

2.3.2 Toward Adaptive TrEO Methods

Section 2.3.1 revealed several pivotal limitations of early TrEO algorithms, thus paving the way toward recent developments of adaptive methods. We have previously noted that (i) *the occurrence of harmful negative transfers is a key concern in the TrEO setting*, and (ii) *beneficial source-target information could remain hidden in the problems' originally defined search spaces*.

On one hand, *there is a handful of research dedicated to curbing the influence of negative transfers in TrEO by automatically adapting the extent of knowledge transmission based on some form of perceived inter-task similarity*. Notably, the *adaptive (probabilistic) model-based transfer* is a prominent state-of-the-art TrEO approach [7] proposed for the aforementioned purpose. Accordingly, an adaptive model-based transfer EA (AMTEA) is designed with the ability to suppress undesirable negative transfers from dissimilar sources [10]. The AMTEA captures the degree of source-target similarity via the optimal online stacking of their respective probabilistic models [51], and uses this similarity to determine the amount of source priors transferred. Enhanced convergence has been demonstrated by AMTEA when compared to non-adaptive case injected EAs, during the optimization of continuous tasks as well as a class of discrete problems.

A few similar works can be found in the recent TrEO literature. In [52], a multisource selective transfer framework is developed to exploit the most relevant knowledge priors based on the similarity between the centroid distributions of distinct problems. A multifactorial EA with transfer parameter estimation is proposed to

reduce negative transfer between multiple tasks tackled concurrently, which has improved convergence characteristics by learning optimal inter-task crossover probabilities on the fly [53]. Likewise, some studies have also shown to reduce negative transfers through automatically adapting the intensity of knowledge exchange based on task relatedness [54, 55, 56].

On the other hand, *there exist several TrEO methods that explore ways to increase positive transfers between distinct optimization problems through adaptive transformations of the search spaces*. A noteworthy example is the TrEO search with adaptive learning across heterogeneous problems [57]. This approach learns linear transformations of solutions in continuous search spaces, and shows promising results for transferring the transformed solutions when solving a range of multi-objective benchmark problems as well as a real-world case study in manufacturing. Other similar methods of linear transformations are proposed to promote positive transfers between multiple optimization tasks solved concurrently [58, 59]. In [60], a common representation scheme is designed to facilitate seamless and positive inter-task transfers across two distinct graph-based routing problems, where the structural similarities between the problems are first identified and then leveraged.

At this juncture, we have seen how recent TrEO methods have adopted adaptive strategies to overcome the limitations of their earlier and non-adaptive counterparts. However, we note that *these adaptive TrEO methods have solely focused either on (i) the curbing of negative inter-task interactions, or (ii) strategies to increase beneficial positive transfers*. Recall the AMTEA [10] for instance, it has shown that convergence performance can be enhanced by automatically adapting the extent of information transmission to mitigate negative transfers from dissimilar sources. Despite the potential for performance speed-up, *the effectiveness of such TrEO algorithms in actual practice could be hampered by unaligned solution representations that conceal useful inter-task relationships, which may reduce positive transfers or even increase negative transfers* – solely focusing on the curbing of negative transfers will then result in the conservative cancellation of knowledge transfers (possibly causing the scarcity of beneficial positive transfers), which undermines the efficacy of such TrEO methods (this shall be further elaborated in Chapter 3).

In contrast, take the TrEO search with adaptive learning [57] as example, where linear transformations of solutions are learned in a bid to increase positive transfers. Such algorithms have the potential to address the aforementioned issue of unaligned solution representations, through learning transformations that could uncover any useful information hidden in the problem's originally defined space. Even so, *these TrEO methods are susceptible to performance impedance in cases of inaccurately learned transformations, which could be caused by a lack of sufficient target search data required for the learning process* – solely focusing on increasing positive transfers without any mechanism to guard against harmful negative transfers may also undermine the efficacy of such TrEO algorithms.

The above shortcomings of existing adaptive TrEO algorithms reveal *a lack approaches capable of inducing positive transfers while simultaneously mitigating the threat of negative transfers; an algorithmic framework that jointly addresses both issues is vital for tapping the desired potential of TrEO*. This motivates the works of the present thesis, which will be unveiled in subsequent chapters of this thesis.

2.3.3 Other Practical Applications and Multi-Objective Problem-Solving in TrEO

Following from the methodological advances of TrEO algorithms discussed in Section 2.3.1 and Section 2.3.2, there is an increasing number of works devoted to investigating the effectiveness of these algorithms in a multitude of real-world applications.

Among a variety of practical applications in TrEO, operations research is one of the popular topics with promising results found, for instance, in intelligent logistics planning [9, 11, 60, 61, 62, 63]. Complex engineering-related tasks have also received much attention from TrEO researchers; works are found for aircraft design [64], process design [57, 65, 66, 67], robotics [45, 46], controller design [10][53] and satellite design [52], to name a few. Other diverse areas that have benefitted from TrEO methods include but are not limited to binary decision making [10], software engineering [68], machine learning [17, 69] and healthcare diagnosis [70].

Besides optimization tasks that involve a single objective function, there exist a significant number of applications considering multiple objectives in problem-solving which are known as multi-objective problems (MOPs). Multi-objective EAs (MOEAs), by virtue of their population-based search strategy, have gained popularity for solving MOPs as they are able to reasonably approximate a set of alternative solutions in the multi-objective optimization setting within a single run. Some of the most widely used MOEAs today include the Pareto dominance-based NSGA-II [71, 72] and SPEA2 [73, 74], the decomposition-based MOEA/D [75, 76], and the preference-based PBEA [77], to name a few. Just like traditional EAs, many existing MOEAs start their search from scratch without any attempt to exploit related source knowledge from past optimization experiences. As a consequence, such (non-TrEO) MOEAs often (i) succumb to premature convergence to sub-optimal solutions, and (ii) require more computational resources than desired (in order to achieve reasonable solution qualities).

Owing to the complexity of MOPs and their ubiquity in the real-world, the potential for performance speed-up offered by the TrEO paradigm becomes attractive to MOEAs. Moreover, given (i) the methodological advances of TrEO algorithms, and (ii) the promising results obtained in a variety of practical applications, there is now growing research attention placed on *multi-objective problem-solving in TrEO* as well. To this end, several TrEO algorithms have recently been applied to boost multi-objective optimization performance through leveraging knowledge from related problems. In the remainder of this section, we briefly review some of the research works in *multi-objective TrEO*.

A multi-objective covariance matrix adaptation evolution strategy is designed to directly inject information from best solutions found so far into the target search [78]. In [65], helper tasks are leveraged when solving multi-objective production optimization problems. A multi-objective multifactorial EA is proposed for mutual knowledge exchange among multiple MOPs tackled in unison [79, 80]; this method is subsequently improved in [81] which attempts to reduce negative transfers by learning optimal inter-task crossover probabilities on the fly. Likewise, a selective transfer framework is developed to automatically select the most relevant sources when solving MOPs [52]. Some studies have similarly designed data-driven multi-objective

TrEO methods to curtail negative influences from dissimilar sources [82, 83]. *Despite the growing research in multi-objective TrEO, we note that majority of the works have solely focused on methods to curb negative transfers; there has been little or no work done in jointly addressing (i) the occurrence of detrimental negative transfers, and (ii) the scarcity of beneficial positive transfers in such a setting, as similarly highlighted in Section 2.3.2.*

2.4 Chapter Summary

This chapter presented a comprehensive literature review of the transfer evolutionary optimization (TrEO) paradigm. It was first described how the notion of transfer learning and the flexibility of evolutionary solvers in incorporating learning strategies have led to the emergence of TrEO algorithms. Some of the early realizations of TrEO methods and their limitations were then discussed. Accordingly, the shift toward the recent developments of adaptive TrEO methods and their shortcomings were elaborated. In particular, we revealed the lack of approaches capable of jointly addressing two predominant issues in TrEO, namely, (i) the occurrence of harmful negative transfers, and (ii) the scarcity of beneficial positive transfers. This was followed by a brief review of other practical applications as well as the prospective multi-objective problem-solving in TrEO.

Figure 2.1 depicts block diagrams that visually summarize the main contribution of the present thesis in comparison to notable state-of-the-art adaptive TrEO methods. Specifically, Chapter 3 will propose a novel framework for solution representation learning (SRL) in adaptive probabilistic model-based TrEO, as depicted in Figure 2.1(c) – the framework serves to facilitate the development of new TrEO algorithms capable of synergistically addressing the shortcomings of (i) the adaptive model-based TrEO [10] which solely focuses on the curbing of negative inter-task interactions as illustrated in Figure 2.1(a), as well as (ii) the TrEO search with adaptive learning across heterogeneous problems [57] which solely tries to increase positive transfers without explicitly guarding against negative transfers as shown in Figure 2.1(b).

FIGURE 2.1. Block diagrams that visually summarize how the present thesis takes motivation from the shortcomings discovered in notable state-of-the-art adaptive TrEO methods. (a) Adaptive model-based TrEO [10]. (b) TrEO with adaptive learning across heterogeneous problems [57]. (c) A novel framework proposed for solution representation learning in adaptive probabilistic model-based TrEO.

With the aim of addressing the key *research gaps* (highlighted in Section 2.3.2 and Section 2.3.3), and guided by the concept in Figure 2.1(c), our framework provides a principled guide for developing new adaptive TrEO algorithms that flexibly allows (i) different methodologies to be designed for SRL in different optimization domains, and (ii) various forms of probabilistic model-based transfer mechanism to be incorporated for distinct solution representation spaces; concrete examples of novel algorithmic instantiations based on our framework shall be given in Chapter 4 and Chapter 5. Accordingly, the subsequent chapters of this thesis will detail the research contributions as outlined in Section 1.3.

Chapter 3

A Formalization of Solution Representation Learning in Transfer Evolutionary Optimization

This chapter formalizes the notion of *solution representation learning (SRL)* in *probabilistic model-based TrEO*¹, with the aim of addressing the limitations of recent adaptive TrEO methods as highlighted in Section 2.3.2. To this end, a *novel and principled framework* is put forward for developing TrEO algorithms capable of inducing beneficial positive transfers while curbing negative inter-task interactions concurrently. The framework encompasses (i) an SRL module that utilizes *search space mappings to learn well-aligned solution representations* (for activating positive transfers), and (ii) a synergistic *probabilistic model-based transfer mechanism to capture source-target similarities* that are used to adapt the extent of knowledge transmission (for simultaneously mitigating negative transfers).

For a coherent formalization, the rest of this chapter is organized as follows. Section 3.1 first gives an overview of the probabilistic model-based TrEO approach for curbing negative transfers. Section 3.2 then discusses how the narrow focus on the curbing of negative transfers potentially causes the conservative cancellation of knowledge transfers in the presence of discrepancies between source and target tasks; thus motivates the importance of SRL in TrEO. Section 3.3 presents a principled perspective of SRL via search space mappings in TrEO, where a streamlined definition is first given. Accordingly, the *source-to-target map* is proposed as a category of spatial mappings aimed at inducing positive inter-task transfers, and the probabilistic formulation of TrEO with a source-to-target map is outlined. Section 3.4 describes the proposed framework for SRL in probabilistic model-based TrEO. Finally, Section 3.5 summarizes the chapter.

¹ The work in this chapter has been published in [13].

3.1 Probabilistic View of TrEO

In Chapter 2, we have described the limitations of traditional evolutionary optimizers that are typically designed to solve optimization problems from scratch without leveraging past problem-solving experiences. This ushers in the emerging TrEO paradigm, under which new breeds of algorithms are developed to exploit knowledge priors from previously solved source tasks to boost the optimization efficiency of the target task at hand. We have also seen how the performance of early TrEO methods can be adversely affected by the occurrence of negative transfers; thereby guiding some recent adaptive methods toward adapting the extent of knowledge transmission based on source-target similarities. In what follows, we discuss the probabilistic model-based TrEO, which is a state-of-the-art approach equipped with the salient ability to automatically curb negative influences from dissimilar sources [10].

3.1.1 Preliminaries

Consider a *series* of T distinct optimization tasks denoted as $\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_T$, with objective (fitness) functions $f_1, f_2, \dots, f_T$, respectively. At this stage, we make the explicit *assumption* that all the fitness functions are predefined in the same solution representation space $\mathcal{X}$. (This assumption shall be relaxed in Section 3.3 on “Solution Representation Learning via Search Space Mappings in TrEO”.) Any task $\mathcal{T}_i$ can then be stated as,

$$\max_{\mathbf{x} \in \mathcal{X}} f_i(\mathbf{x}), \quad (3.1)$$

where $\mathbf{x} \in \mathcal{X}$ is a candidate solution vector in space $\mathcal{X}$. Note that we consider the maximization of objective function without loss of generality, and constraint functions are intentionally ignored for the simplicity of exposition. When carrying out optimization via probabilistic model-based search algorithms, Eq. (3.1) is effectively reformulated as,

$$\max_{p_i(\mathbf{x})} \int_{\mathcal{X}} f_i(\mathbf{x}) \cdot p_i(\mathbf{x}) \cdot d\mathbf{x}. \quad (3.2)$$

Here, $p_i(\mathbf{x})$ represents the probability distribution of a population $P_i = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ of candidate solutions for $\mathcal{T}_i$. A population P_i^* – with distribution $p_i^*(\mathbf{x})$ – of *good* (near-optimal) solutions for $\mathcal{T}_i$, thus satisfies the following,

$$\int_{\mathcal{X}} f_i(\mathbf{x}) \cdot p_i^*(\mathbf{x}) \cdot d\mathbf{x} \geq f_i^* - \varepsilon_i, \quad (3.3)$$

where f_i^* represents the globally optimum fitness value, and $\varepsilon_i(> 0)$ is a small convergence threshold. Notice that letting ε_i be zero could lead to the unsolicited case of a degenerate distribution $p_i^*(\mathbf{x})$.

Given the above, in this thesis, the domain $\mathcal{D}$ of an optimization task $\mathcal{T}$ is deemed to consist of two components: its solution representation space $\mathcal{X}$, and the probability (or search) distribution of good solutions $p^(\mathbf{x})$ in that space; i.e., $\mathcal{D} = \{\mathcal{X}, p^*(\mathbf{x})\}$.*

This expands on a previous description of optimization domains presented in [8], where the second component (i.e., $p^*(\mathbf{x})$) had been excluded; see Definition 2.2 in Section 2.3. However, referring to [4], the present thesis regards the optimized search distribution as a necessary component to fully characterize the domain of an optimization task. It is worth noting that such a concept of what constitutes the domain $\mathcal{D} = \{\mathcal{X}, p^*(\mathbf{x})\}$ of an optimization task forms an integral component of the probabilistic model-based TrEO that follows.

3.1.2 Probabilistic Formulation of TrEO

In the sequential transfer optimization setting, it is assumed that when addressing *target* task $\mathcal{T}_T$, we have access to the data (including the optimized search distribution models) drawn from $T - 1$ *source* tasks $\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{T-1}$ that have been previously encountered. The crux of the TrEO paradigm lies in harnessing related knowledge priors (from the sources) in the form of search biases that could accelerate the target search. The modified formulation of the target optimization task from the standpoint of probabilistic model-based search then becomes [7],

$$\max_{w_1, w_2, \dots, w_T, p_T(x)} \int_{\mathcal{X}} f_T(\mathbf{x}) \cdot \left[\sum_{s=1}^{T-1} w_s \cdot p_s^*(\mathbf{x}) + w_T \cdot p_T(\mathbf{x}) \right] \cdot d\mathbf{x}, \quad (3.4)$$

where $f_T(\mathbf{x})$ is the target objective function, $p_T(\mathbf{x})$ and $p_s^*(\mathbf{x})$ are the target and optimized source search distributions, respectively; $\sum_{i=1}^T w_i = 1$, and $w_i \geq 0 \forall i \in \{1, 2, \dots, T\}$. The coefficients $w_1, w_2, \dots, w_T$ in the probability mixture model $\left[\sum_{s=1}^{T-1} w_s \cdot p_s^*(\mathbf{x}) + w_T \cdot p_T(\mathbf{x}) \right]$ (also known as the *target mixture model*) specify and adapt the extent to which the sources influence the target search – and are therefore also referred to as *transfer coefficients*.

If there exists at least one source, say $\mathcal{T}_{s'} \in \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{T-1}\}$, whose optimized search distribution $p_{s'}^*(\mathbf{x})$ significantly overlaps with $p_T^*(\mathbf{x})$ – which we denote as $p_{s'}^*(\mathbf{x}) \approx p_T^*(\mathbf{x})$ – then the *source-target similarity* is captured by assigning $w_{s'} \rightarrow 1$, and sampling from the resultant mixture model will yield good solutions near the optimal of the target task $\mathcal{T}_T$ (i.e., giving objective values close to f_T^*) with high probability. On the contrary, if a particular source $\mathcal{T}_{s''} \in \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{T-1}\}$ appears substantially dissimilar (i.e., $p_{s''}^*(\mathbf{x}) \not\approx p_T^*(\mathbf{x})$), then the perceived lack of source-target similarity is also captured by assigning $w_{s''} \rightarrow 0$ which cancels knowledge transfers from source $\mathcal{T}_{s''}$. In principle, sampling the mixture model in Eq. (3.4) curbs the occurrence of negative inter-task interactions.

Based on our previous assumption in Section 3.1.1 – all source and target tasks are defined in the same search space $\mathcal{X}$ – the target mixture model in Eq. (3.4) can be built through the optimal stacking of source and target models in $\mathcal{X}$; providing the transfer coefficients values. In order to effectively resolve Eq. (3.4), it is important to be able to deduce accurate transfer coefficients online, based on the data generated during the course of the target search. It is to this end that a state-of-the-art probabilistic model-based TrEO algorithm has recently been proposed in [10]; specifying that the transfer coefficients in the t^{th} iteration can be obtained by solving the following out-of-sample log-likelihood maximization program:

$$\max_{w_1, w_2, \dots, w_T} \sum_{\forall \mathbf{x} \in P_T(t)} \log \left(\sum_{s=1}^{T-1} w_s \cdot p_s^*(\mathbf{x}) + w_T \cdot p_T(\mathbf{x}|t) \right), \quad (3.5)$$

where $P_T(t)$ and $p_T(\mathbf{x}|t)$ are the (current) target population data and search distribution model, respectively, at iteration t . Note that Eq. (3.5) can be solved to optimality via the stacked density estimation procedure [51] that utilizes the classical expectation-maximization algorithm [84]. The detailed proof of how solving Eq. (3.5) theoretically leads to the curtailment of negative inter-task transfers is given in [10].

A probabilistic model-based TrEO algorithm progresses by iteratively generating populations of candidate solutions through sampling the built target mixture models. Hence, solving Eq. (3.5) provides the transfer coefficients w_s that capture the source-target similarity, and subsequently adapts the extent to which the sources influence the target search. A larger w_s suggests that the corresponding source’s optimized search distribution $p_s^*(\mathbf{x})$ overlaps more significantly with the target population, thus supports source-to-target knowledge transfers at little threat of harmful interactions. In contrast, a smaller w_s reflects the dissimilarity in the source and target population distributions, thereby preemptively curtailing the transfer of solutions from the corresponding source task $\mathcal{T}_s \in \{\mathcal{T}_1, \mathcal{T}_2, \dots, \mathcal{T}_{T-1}\}$.

In practice, however, the underlying target optimum distribution $p_T^*(\mathbf{x})$ is typically *a priori* unknown, which is what makes it challenging to decipher the extent of knowledge transfers from various source probabilistic models. On one hand, assigning a high transfer coefficient to an inappropriate source model will lead to the sampling of inferior solutions for the target task. On the other hand, not assigning a sufficiently high transfer coefficient to a relevant source model will lead to the loss of useful solutions that could help boost the target search. The challenge faced is even greater in black-box optimization settings where little can be said with certainty about inter-task similarities.

As highlighted in Chapter 1 and Chapter 2, discrepancies between source and target tasks could take the form of (i) heterogeneous dimensionalities, or (ii) a lack of overlap between optimized search distributions, or both; manifesting in unaligned solution representations that conceal useful inter-task similarities. When this happens, conservative cancellation of knowledge transfers is inevitable among existing probabilistic model-based TrEO methods that focus solely on the mitigation of negative inter-task interactions – hence, motivating the notion of solution representation learning as a tool for inducing greater alignment (of solution representations) and hence positive transfers in TrEO.

3.2 Motivating Solution Representation Learning in TrEO

The success of probabilistic model-based transfer has been demonstrated for a range of benchmark examples where there exists at least one source task whose optimized search distribution overlaps with that of the target. In these cases, solutions sampled from the related source model survive in the target population (due to the survival of the fittest principle in EAs), and, in turn, drive up the associated transfer coefficient in Eq. (3.4). The opposite occurs for source models that are not related to the target, pre-emptively curbing the threat of negative transfer.

In contrast, consider a situation in which there is not a single source whose domain $\mathcal{D}_S = \{\mathcal{X}, p_S^*(\mathbf{x})\} \forall S \in \{1, 2, \dots, T-1\}$ matches that of the target domain $\mathcal{D}_T = \{\mathcal{X}, p_T^*(\mathbf{x})\}$. Even assuming the same predefined search space $\mathcal{X}$, their optimized search distributions do not overlap in that space; i.e., $p_T^*(\mathbf{x}) \not\approx p_S^*(\mathbf{x}) \forall S \in \{1, 2, \dots, T-1\}$. In such cases, the probability mixture modelling in Eq. (3.4) will serve to suppress the transfer of solutions from the various sources to the target task. It is at this juncture we note that *the lack of significant search distribution overlap does not necessarily imply the absence of useful information in a source*. We argue that *there may exist inter-task relationships that although remain hidden in the problems' originally defined search spaces, can be revealed under certain search space transformations*. Overlooking the latent relationships would potentially lead to the loss of useful solutions from the source that could help boost the target optimization search – we refer to this undesirable situation as *the conservative cancellation of knowledge transfers, which potentially causes the scarcity of beneficial positive transfers*.

As a substantiation of our claim, take the example of the synthetic source and target fitness functions (denoted as $f_S(\mathbf{x})$ and $f_T(\mathbf{x})$, respectively) in Figure 3.1, where the 1-dimensional search space is characterized by the variable bounds $[-1, 1]$. Given the same search space, the source domain $\mathcal{D}_S$ and target domain $\mathcal{D}_T$ are distinguished by their respective optimized search distributions, $p_S^*(\mathbf{x})$ and $p_T^*(\mathbf{x})$. As is clear from Figure 3.1, the two functions are negatively correlated (one decreases in the direction in which the other increases), which indicates that $p_S^*(\mathbf{x})$ and $p_T^*(\mathbf{x})$ will be focused at opposite ends of the feasible space. In this example, the dissonance in optimized

distributions gives rise to a scenario of *source-target domain mismatch*, i.e., $p_S^*(x) \neq p_T^*(x) \Rightarrow \mathcal{D}_S \neq \mathcal{D}_T$, manifesting in *unaligned solution representations*. Therefore, directly sampling solutions from $p_S^*(x)$ for the purpose of solving the target task would lead to *negative transfers*. But, as also shown in Figure 3.1, when a simple *search space mapping* $M(x) = -x$ is applied to the source solutions, we can *learn well-aligned solutions that induce a strong (positive) correlation* between the functions $f_S(M^{-1}(x))$ and $f_T(x)$. Sampling the resultant search distribution of mapped source solutions (obtained by transforming optimized source data P_S^* to $M(P_S^*)$) will then yield near-optimal solutions for the target. The utility of such spatial mappings extends beyond the 1-D case, as highlighted by the following proposition.

FIGURE 3.1. An example where a search space mapping $M(x) = -x$ applied to the source solutions induces strong (positive) correlation between the transformed source function $f_S(M^{-1}(x))$ and the target fitness function $f_T(x)$, resulting in greater overlap between their respective optimized search distributions.

Proposition 3.1: *Given a pair of continuous fitness functions $f_a(x)$ and $f_b(x)$ where their candidate solutions are vectors of dimensionality d , i.e., $x \in \mathcal{X} \subseteq \mathbb{R}^d$, if M_a is an invertible mapping such that $f_a(M_a^{-1}(x_1)) < f_a(M_a^{-1}(x_2)) \Rightarrow f_b(x_1) < f_b(x_2)$ for any x_1 and x_2 , then their respective optimized search distributions satisfy $p_{M_a}^*(x) \approx p_b^*(x)$ as $\varepsilon_a, \varepsilon_b \rightarrow 0$ (see Eq. (3.3)).*

For an example beyond the 1-D case, consider a TrEO setting in which high quality solutions from a source task (defined in a space $\mathcal{X}_S \subseteq \mathbb{R}^{d_S}$ of dimensionality d_S) are transferred to a target problem (defined in a space $\mathcal{X}_T \subseteq \mathbb{R}^{d_T}$ of dimensionality d_T). In practice, it is common that the *source and target dimensionalities are arbitrarily heterogeneous* (e.g., $d_T > d_S$ or $d_T \leq d_S$), resulting in yet another scenario of *source-target domain mismatch*, i.e., $d_S \neq d_T \Rightarrow \mathcal{X}_S \neq \mathcal{X}_T \Rightarrow \mathcal{D}_S \neq \mathcal{D}_T$. Some simplistic approaches to deal with such circumstances have surfaced in the literature; for instance, padding additional variables to and removing excess variables from the (i) solution representations [57], or (ii) probabilistic models [10] of the source task. Nevertheless, we again argue that these simple approaches (i) are unlikely to suffice in unveiling beneficial inter-task relationships (when $p_S^*(\mathbf{x}_S) \approx p_T^*(\mathbf{x}_T)$), or (ii) could lead to the loss of useful source-target information (when $d_T \leq d_S$) – eventually causing the conservative cancellation of knowledge transmission hence the scarcity of positive transfers. One possible way to overcome this is to similarly *utilize search space mappings (or transformations) to uncover any useful but hidden inter-task relationships*.

In this thesis, *the notion of solution representation learning (SRL) in TrEO thus comprises search space mappings that transform solutions to a common and well-aligned solution representation space, in which the discrepancies between various sources and the target task are reduced*. In particular, spatial mappings or transformations capable of (i) *handling arbitrarily heterogeneous source and target dimensionalities*, and (ii) *inducing greater overlap between their respective optimized search distributions* – or in other words, capable of inducing greater alignment (of solution representations) and hence positive transfers – are a crucial component of SRL in TrEO. *The success of SRL however depends on the design of appropriate methodologies for constructing effective search space mapping functions in particular optimization domains of interest* (e.g., the mapping $M(\mathbf{x}) = -\mathbf{x}$ seen in Figure 3.1).

On one hand, in areas where sufficient domain expertise is available, a hand-crafted methodology for the *offline* construction of the mapping function is possible. On the other, for general black-box optimization settings with a scarceness of domain knowledge, a methodology could be designed for the *online* learning of appropriate mapping functions by utilizing the data generated during the course of the target

search. *Proposals of methodologies for constructing effective spatial mappings in distinct optimization domains shall be given in Chapter 4 and Chapter 5.* Moreover, we contend that *the synergy between SRL and adapting the extent of knowledge transfer (through inter-task similarity capture) is essential for tapping the desired performance boosts in TrEO.* Taking the cue, the next section presents a principled perspective of SRL (via search space mappings) in probabilistic model-based TrEO.

3.3 Solution Representation Learning via Search Space Mappings in TrEO

Section 3.2 has elaborated on the motivations and importance of solution representation learning (SRL) in uncovering useful but hidden inter-task relationships; we have also seen the scope for utilizing search space mappings (or transformations) to learn well-aligned solution representations that induce positive transfers. Following from that, this section presents a principled perspective of SRL via search space mappings in probabilistic model-based TrEO. To this end, *a streamlined definition of SRL in TrEO* is first given.

From this point, we *relax* our assumption from Section 3.1.1 that the tasks are predefined in the same search space. However, we simplify our exposition by limiting to the case of a *single source*. Scaling to a larger number of sources can be carried out as in [85]. With the above and Proposition 3.1 in mind, the notion of SRL (via search space mappings) in TrEO can be condensed into a streamlined definition as follows; see also Section 3.2.

Definition 3.1 (*Solution representation learning in TrEO*): *Given source optimization task $\mathcal{T}_S$ with domain $\mathcal{D}_S = \{\mathcal{X}_S, p_S^*(\mathbf{x}_S)\}$, and target task $\mathcal{T}_T$ with domain $\mathcal{D}_T = \{\mathcal{X}_T, p_T^*(\mathbf{x}_T)\}$, where $\mathcal{X}_S \neq \mathcal{X}_T$ or/and $p_S^*(\mathbf{x}_S) \neq p_T^*(\mathbf{x}_T)$, solution representation learning aims to help boost the optimization of $\mathcal{T}_T$ through search space mappings (or transformations) that induce positive knowledge transfers across tasks in the transformed solution representation space.*

It is worth reiterating that in optimization problems, the optimum target search distribution $p_T^*(\mathbf{x}_T)$ would not be known in advance. What is more, task-specific data

may not be available prior to the onset of the search; this marks a key distinguishing factor between the notion of solution representation learning in TrEO unveiled in this thesis and the corresponding concept of *domain adaptation* prevalent in the machine learning literature (where a plethora of methods have been proposed under the assumption of source and target data availability [86]). In what follows, we firm up the principled perspective of SRL (via search space mappings) in probabilistic model-based TrEO by presenting the *source-to-target map* and its mathematical formulation from the standpoint of probabilistic model-based search and transfer.

3.3.1 Source-to-Target Map: M_S

Here we propose the *source-to-target map* M_S as a category of spatial mappings by which source tasks are mapped to a transformed and well-aligned solution representation space, such that the discrepancies between various sources and the target task are reduced – in this category, the target search space serves as the transformed (or common) solution representation space. With reference to the example in Figure 3.1 (which also transforms the source task), we note that distinct source-to-target mapping functions may be constructed for the purpose of learning well-aligned solution representations. Accordingly, we outline *the probabilistic formulation of TrEO with a source-to-target map that flexibly allows various source-to-target mapping functions to be incorporated in this setting*.

Let the search space of source task $\mathcal{T}_S$ be $\mathcal{X}_S \subseteq \mathbb{R}^{d_S}$ and that of the target task $\mathcal{T}_T$ be $\mathcal{X}_T \subseteq \mathbb{R}^{d_T}$. In general, $\mathcal{X}_S \neq \mathcal{X}_T$ and $d_S \neq d_T$. Modifying Eq. (3.4), the probabilistic formulation of TrEO inclusive of a source-to-target map M_S is stated as,

$$\begin{aligned} \max_{w_S, w_T, p_T(\mathbf{x}_T)} \quad & \int_{\mathcal{X}_T} f_T(\mathbf{x}_T) \cdot [w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T)] \cdot d\mathbf{x}_T, \\ \text{s. t.}, \quad & w_S + w_T = 1 \text{ and } w_S, w_T \geq 0. \end{aligned} \quad (3.6)$$

Here, $\mathbf{x}_T \in \mathcal{X}_T$, $M_S: \mathcal{X}_S \rightarrow \mathcal{X}_T$, and $p_{M_S}^*$ is the transformation of $p_S^*(\mathbf{x}_S)$ under mapping function M_S where $\mathbf{x}_S \in \mathcal{X}_S$. The purpose of introducing M_S in Eq. (3.6) is to stimulate $p_{M_S}^*(\mathbf{x}_T) \approx p_T^*(\mathbf{x}_T)$, or $d_S = d_T$, or both; in other words, M_S aims to

overcome source-target domain mismatch scenarios described in Section 3.2. Therefore, sampling the target mixture model $[w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T)]$ yields good solutions for the target task with high probability when $w_S > 0$; see Section 3.1.2.

FIGURE 3.2. Illustration of the proposed source-to-target map for SRL in TrEO.

An illustration of the source-to-target map M_S is provided in Figure 3.2. Notably, the TrEO formulation in Eq. (3.6) flexibly allows distinct source-to-target mapping functions to be incorporated in this setting. For instance, in a recent work, M_S has been realized in the form of a *linearized* transformation function from $\mathcal{X}_S$ to $\mathcal{X}_T$ [57]; i.e.,

$$M_S^L(\mathbf{x}_S) = A \cdot \mathbf{x}_S + \mathbf{b}, \quad (3.7)$$

where matrix $A \in \mathbb{R}^{d_T \times d_S}$, and $\mathbf{b} \in \mathbb{R}^{d_T}$ is a bias vector.

While a linear map is workable when $d_T \leq d_S$, it does not suffice when $d_T > d_S$. This is because the transformed source distribution $p_{M_S^L}^*$ becomes degenerate when $d_T > d_S$; although $p_{M_S^L}^*$ is defined in the d_T -dimensional space $\mathcal{X}_T$, it is supported only on a lower d_S -dimensional subspace. As discussed in Section 3.2, an ad-hoc workaround of padding the source model with extra variables was proposed to compensate for the difference between d_T and d_S [10]. A *more general approach* however – one that can automatically cope with both $d_T \leq d_S$ as well as $d_T > d_S$

without resulting in a degenerate distribution – is to let M_S be non-linear. Novel methodologies for learning non-linear source-to-target mapping functions will be proposed subsequently in Chapter 4 and Chapter 5 of this thesis.

Having established a principled perspective of solution representation learning in probabilistic model-based TrEO, the next section puts forward a novel framework for developing adaptive TrEO algorithms capable of (i) inducing greater alignment (of solution representations) via the utility of spatial mappings, and hence positive transfers, while (ii) simultaneously curbing negative transfers between source and target tasks.

3.4 A Novel Framework for Solution Representation Learning in Probabilistic Model-Based TrEO

This section first details the *solution representation learning (SRL) module*, an essential component within our novel framework that utilizes search space mappings to learn well-aligned solution representations (for inducing positive transfers).

3.4.1 The Solution Representation Learning (SRL) Module

When a *sequential transfer iteration* – determined by a *transfer interval* parameter Δ – is launched in a resultant TrEO algorithm, the SRL module within our framework is first activated to (i) *construct the spatial mappings*, and (ii) *transform the probabilistic model(s)*.

Let us consider *the case of incorporating source-to-target spatial mappings in the SRL module*. With reference to Eq. (3.6), the target mixture model $[w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T)]$ consists of the transformed source model $p_{M_S}^*(\mathbf{x}_T)$, which is the transformation of the original source model $p_S^*(\mathbf{x}_S)$ under the mapping function M_S . Here, the transformed model $p_{M_S}^*(\mathbf{x}_T)$ is obtained via *simple Monte Carlo approximation* as shown in Algorithm 3.1. For this case, the SRL module aims to induce positive inter-task transfers by facilitating (i) $p_{M_S}^*(\mathbf{x}_T) \approx p_T^*(\mathbf{x}_T)$ (i.e.,

increased overlap between optimized source and target search distributions), or (ii) $d_S = d_T$, or both, through source-to-target search space transformations. In other cases, nevertheless, *the SRL module flexibly allows different forms of search space mappings to be incorporated as well*; Section 6.2 shall unveil two other possible categories of spatial mappings that may be conceived on the basis of distinct forms of transformations involved.

Algorithm 3.1 The transformation of probabilistic model via simple Monte Carlo approximation

Input: Original source model $p_S^*(\mathbf{x}_S)$, source-to-target mapping function M_S

1. **If** search space mappings take the form of *Source-to-Target Map* M_S
 2. Sample $p_S^*(\mathbf{x}_S)$ to obtain N solutions $\mathbf{x}_S$ in $\mathcal{X}_S$
 3. Apply $M_S: \mathcal{X}_S \rightarrow \mathcal{X}_T$ on each $\mathbf{x}_S$ to obtain N transformed source solutions $\mathbf{x}_T = M_S(\mathbf{x}_S)$
 4. Build a probabilistic model of the transformed source solutions to obtain the transformed source model $p_{M_S}^*(\mathbf{x}_T)$
 5. **Output:** Transformed source model $p_{M_S}^*(\mathbf{x}_T)$
 6. **End If**
-

Thereafter, the source and target models are put through stacked density estimation (i.e., mixture modelling) [51], leading to probabilistic model-based knowledge transfer that occurs alongside a source-target similarity capture mechanism (which mitigates negative inter-task interactions) – see Figure 3.3 for a graphical illustration of the proposed framework. On this account, we shall next describe the *probabilistic model-based transfer mechanism* as another essential component of our framework.

FIGURE 3.3. Graphical illustration of the proposed framework, featuring the SRL module and the probabilistic model-based transfer mechanism as essential components of a resultant TrEO algorithm.

3.4.2 Probabilistic Model-Based Transfer

It is worth noting that our framework executes knowledge transfer iterations based on probabilistic model-based evolutionary search procedure, whereas non-transfer iterations proceed according to standard recombination-based evolutionary search; see

Figure 3.3 – this will be further elaborated in Section 3.4.3. In what follows, we provide a brief description of the probabilistic model-based evolutionary search.

3.4.2.1 Probabilistic Model-Based Evolutionary Search

Probabilistic model-based evolutionary search is a derivative-free optimization procedure that guides a population of candidate solutions towards the global optimum f^* of a fitness function $f(\mathbf{x})$ by iterating over the steps below [87].

Step 1: Define an initial search distribution model $p(\mathbf{x}|t = 0)$ over the search space $\mathcal{X}$ of optimization task $\mathcal{T}$.

Step 2: At iteration t , sample and evaluate a set $P'(t)$ of N candidate solutions (referred to as *offspring*) from the current model $p(\mathbf{x}|t - 1)$.

Step 3: Based on the evaluated fitness, select a set of $n(\leq N)$ *parent solutions* from $P'(t)$; denoted as $P(t)$.

Step 4: Build probabilistic model $p(\mathbf{x}|t)$ of the parent population $P(t)$.

Step 5: Increment the iteration count and return to Step 2 until a termination condition is met.

A well-known result and proof supporting the asymptotic global convergence of the above steps is presented in [88], which is summarized in the theorem below.

Theorem 3.1 (Zhang and Muhlenbein [88]): *Let $p(\mathbf{x}|t = 0)$ be positive and continuous everywhere in $\mathcal{X}$. Assuming $N \rightarrow \infty$, if the probabilistic model $p(\mathbf{x}|t)$ – from which offspring $P'(t + 1)$ are sampled – captures the true underlying distribution of the parent population $P(t)$, then the probabilistic model-based search algorithm satisfies,*

$$\lim_{t \rightarrow \infty} \int_{\mathcal{X}} f(\mathbf{x}) \cdot p(\mathbf{x}|t) \cdot d\mathbf{x} = f^*. \quad (3.11)$$

The main takeaway from Theorem 3.1 is that global convergence happens when the model $p(\mathbf{x}|t)$ captures the true distribution of the parent population $P(t)$. It is this insight that is utilized in the probabilistic model-based TrEO approach to deduce accurate transfer coefficients in Eq. (3.4). Specifically, this approach aims to sample candidate solutions from a built *target mixture model* (comprising both source and target models) that more accurately approximates the true distribution of the target $P(t)$, which in turn facilitates the target search convergence [10].

3.4.2.2 Inter-Task Similarity Capture Mechanism

The target mixture model built in existing probabilistic model-based TrEO algorithms depends on the *transfer coefficients* obtained by solving Eq. (3.5) [10]. Particularly, the transfer coefficients act as an *inter-task similarity capture mechanism* to curb the occurrence of negative transfers from seemingly dissimilar sources (possibly due to hidden source-target relationships); see Section 3.1.2. These algorithms make the simplifying assumption that the various source and target tasks have been predefined in a search space $\mathcal{X}$, hence they do not include the construction of the solution representation space itself.

In contrast, the SRL module of the proposed framework enables the transformations of tasks to different solution representation spaces in which useful but hidden source-target relationships can potentially be uncovered. Moreover, our framework encompasses both the SRL module as well as the probabilistic model-based transfer mechanism – thus it is capable of (i) the activation of positive transfers by learning well-aligned solution representations, and (ii) the simultaneous mitigation of negative transfers (that may also arise from inaccurately learned solution representations) through capturing source-target similarities. Following our framework’s joint consideration of the two predominant issues in TrEO, the mathematical program in Eq. (3.5) to deduce the transfer coefficients (i.e., optimal mixture modelling) can be reformulated based on the category of spatial mappings employed in the SRL module.

Given a source probabilistic model $p_S^*(\mathbf{x}_S)$ and an intermediate target distribution $p_T(\mathbf{x}|t)$ at any iteration t , the *stacked density estimation* procedure [51] is used to

deduce the transfer coefficients under search space mappings (or transformations) as follows.

As a substantiation of the above, let us again consider *the case where the framework's SRL module incorporates a source-to-target map M_S* . In this case, the probabilistic model-based transfer mechanism within our framework determines the transfer coefficients by maximizing the modified log-likelihood program below,

$$\max_{w_S, w_T} \sum_{\forall \mathbf{x}_T \in P_T(t)} \log(w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T|t)), \quad (3.12)$$

where $p_{M_S}^*(\mathbf{x}_T)$ refers to the transformation of $p_S^*(\mathbf{x}_S)$ under M_S , and $P_T(t)$ is the target task's parent population at iteration t .

Furthermore, when *multiple source models* $\{p_1^*(\mathbf{x}_1), p_2^*(\mathbf{x}_2), \dots, p_{T-1}^*(\mathbf{x}_{T-1})\}$ – where $\mathbf{x}_1 \in \mathcal{X}_1, \mathbf{x}_2 \in \mathcal{X}_2, \dots, \mathbf{x}_{T-1} \in \mathcal{X}_{T-1}$ – are involved, we can decipher the transfer coefficients (under source-to-target search space mappings) by solving the multi-source formulation of Eq. (3.12) as follows,

$$\max_{w_1, w_2, \dots, w_T} \sum_{\forall \mathbf{x}_T \in P_T(t)} \log \left(\sum_{s=1}^{T-1} w_s \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T|t) \right). \quad (3.13)$$

We note that the mathematical programs in Eq. (3.12) and Eq. (3.13) can also be solved to optimality by the classical *expectation-maximization* (EM) algorithm [84]. The procedure is not reproduced herein for the sake of brevity; for a detailed description readers are referred to [7]. At this juncture, it is worth mentioning that *the probabilistic model-based transfer mechanism can also flexibly accommodate different forms of search space mappings*; Section 6.2 shall unveil the mathematical programs to deduce the transfer coefficients for two other possible categories of spatial mappings (based on distinct transformations involved).

3.4.3 Summary of Algorithmic Framework

Section 3.4.1 and Section 3.4.2 elaborated on the *synergy* between the *solution representation learning module* (highlighted as blue in Algorithm 3.2) and the *probabilistic model-based source-target similarity capture mechanism* (highlighted as red in Algorithm 3.2) within the proposed framework; see Figure 3.3 for a graphical illustration. Our framework therefore supports the seamless development of novel probabilistic model-based TrEO algorithms with solution representation learning (via search space mappings). Algorithm 3.2 summarizes how the novel framework leads to resultant TrEO algorithms capable of (i) activating positive transfers through spatial mappings that learn well-aligned solution representations, and (ii) simultaneously curbing the occurrence of negative transfers based on a probabilistic model-based source-target similarity capture mechanism.

Algorithm 3.2 takes as input the source data and model from a previously optimized source task $\mathcal{T}_S$, the target task $\mathcal{T}_T$, and a user-defined transfer interval Δ . The TrEO search starts by randomly sampling and evaluating a population of initial solutions (lines 1–3). During non-transfer iterations (i.e., $\text{mod}(t, \Delta) \neq 0$) which execute canonical EA search, standard crossover and mutation operators are applied on the parent population $P_T(t)$ to generate the offspring population $C_T(t)$ (lines 5–6).

When sequential transfer iterations are triggered by the parameter Δ (i.e., $\text{mod}(t, \Delta) = 0$), the spatial mappings are first constructed and then applied to learn new solution representations (lines 8–9). Thereafter, the corresponding search distribution models are transformed by simple Monte Carlo approximation (line 10); see an example procedure in Algorithm 3.1 where source-to-target spatial mappings are utilized in the SRL module within the proposed framework. Accordingly, the target mixture model is built (by deducing the transfer coefficients through stacked density estimation [51] – solving Eq. (3.12) will obtain the coefficients when source-to-target mapping functions are used in lines 8–10), and sampled to generate $C_T(t)$ (lines 11–12). The new parent population $P_T(t + 1)$ for the next iteration is selected based on the specified selection procedure, and the entire evolutionary search process repeats until the stopping condition is satisfied (lines 14–16).

Algorithm 3.2. A novel framework for solution representation learning via search mappings in probabilistic model-based TrEO

Input: Source task $\mathcal{T}_S$; target task $\mathcal{T}_T$; transfer interval Δ

Output: Data and optimized search distribution for target task $\mathcal{T}_T$

```

1: Set iteration  $t = 0$ 
2: Initialize the TrEO search by randomly generating a parent population  $P_T(t)$ 
3: Evaluate fitness of solutions in  $P_T(t)$ 
4: While stopping condition not met do
5:   If  $\text{mod}(t, \Delta) \neq 0$  then

       /* Canonical EA search iteration */

6:     Generate the offspring population  $C_T(t)$  by applying standard crossover
       and/or mutation operators on  $P_T(t)$ 
7:   Else

       /* Sequential transfer iteration */

8:     Construct the spatial mappings using  $\mathcal{T}_S$  and  $\mathcal{T}_T$ 
9:     Learn new solution representations by applying the constructed mappings
10:    Transform the probabilistic models using learned solution representations
11:    Obtain the transfer coefficients, then build the target mixture model
12:    Sample offspring  $C_T(t)$  from the target mixture model
13:   End if
14:   Evaluate fitness of solutions in  $C_T(t)$ 
15:   Select new parent population  $P_T(t + 1)$ 
16:   Set iteration  $t = t + 1$ 
17: End while

```

An important facet of the resultant TrEO algorithms is that they intertwine probabilistic model-based search with conventional crossover and/or mutation based evolutionary updates. This, as suggested in [89], helps to reduce the (albeit small) computational cost associated with repeated building of the probability mixture

models. Notably, *the workflow of Algorithm 3.2 (i) is not limited to any specific methodology for the construction of spatial mappings* in line 8 – our framework gives the possibility of *offline construction* as well as *online learning* of effective mapping functions (this shall be elaborated in Section 4.1), and (ii) *flexibly allows different forms of search space mappings to be incorporated within the SRL module and the probabilistic model-based transfer mechanism* (see Section 3.4.1 and Section 3.4.2, respectively) – this shall be further discussed in Section 6.2.

Despite the generality of our proposed framework, it is worth noting that technical challenges may still arise when applying the framework to design algorithmic instantiations in different optimization domains. On one hand, in cases where the existence of unique problem structures or information (i.e., domain knowledge) allows the offline construction of search space mappings, a foreseeable challenge lies in (i) identifying useful domain knowledge to be extracted, and (ii) determining how the extracted information is exploited to design effective methodologies for specific applications. On the other hand, when such domain knowledge is unavailable, the challenge is to devise methodologies that achieve fast online learning of effective spatial mappings – where the learning process (which utilizes generated search data) is nested within the evolutionary optimization procedure – with the aim of unveiling any beneficial but hidden inter-task relationships. We shall showcase novel methodologies that feature both the offline construction as well as the online learning of effective mapping functions in Chapter 4 and Chapter 5 of this thesis.

3.5 Chapter Summary

In this chapter, a formalization of solution representation learning (SRL) in probabilistic model-based TrEO was established. We demonstrated how state-of-the-art probabilistic model-based transfer approaches may fail to exploit useful but hidden inter-task similarities (in the presence of source-target domain mismatch that manifests in unaligned solution representations), which in turn lead to the conservative cancellation of knowledge transfers. This precisely revealed the importance of SRL in uncovering such latent similarities, as well as motivated the need to furnish a principled perspective of SRL (via search space mappings) in TrEO. For this reason,

the notion of SRL in TrEO is first consolidated into a streamlined definition. We then proposed the source-to-target map as a category of spatial mappings aimed at learning well-aligned solution representations hence inducing positive inter-task transfers; in parallel, the probabilistic formulation of TrEO with a source-to-target map was outlined.

Subsequently, a novel framework was presented for SRL in probabilistic model-based TrEO, intended for guiding the development of adaptive algorithms that are endowed with the following salient features:

- An SRL module that flexibly allows various search space mapping functions to be incorporated (for inducing positive transfers).
- A probabilistic model-based transfer mechanism which captures source-target similarities (for simultaneously mitigating negative transfers).

Following from the formalized notion of SRL in TrEO, the next chapter proposes a novel probabilistic model-based TrEO algorithm with source-to-target map (labelled as the TrEO+ M_S), and experimentally verifies distinct instantiations of the proposed algorithm in the domains of discrete and continuous optimization, respectively.

Chapter 4

TrEO with Source-to-Target Map: TrEO+ M_S

Chapter 3 has put forward a novel algorithmic framework for solution representation learning in probabilistic model-based TrEO (presented in Section 3.4), with the aim of facilitating the development of algorithms that are capable of (i) inducing positive transfers, while (ii) concurrently reducing negative transfers. *In accordance with the framework, we contend that jointly considering these capabilities can further boost convergence performance when compared to focusing solely on one capability at a time* (as was done in state-of-the-art adaptive TrEO algorithms; see Section 2.3.2). As a first stepping stone to investigate the applicability of our framework, this chapter proposes a *probabilistic model-based TrEO algorithm with source-to-target map*², which we denote as the TrEO+ M_S hereafter. To verify the efficacy of the TrEO+ M_S , case studies in the domains of discrete and continuous optimization are considered. In each domain, we design novel methodologies for constructing source-to-target spatial mappings; the methods give rise to distinct TrEO+ M_S algorithmic instantiations for learning well-aligned solution representations in the corresponding domains. The performance advantage of our algorithms is then demonstrated through a range of computational experiments.

The rest of this chapter is organized as follows. Based on our framework, Section 4.1 outlines the proposed TrEO+ M_S algorithm. On one hand, Section 4.2 investigates the efficacy of the TrEO+ M_S in the domain of discrete optimization, looking at the 0-1 Knapsack problem as a practical exemplar of binary-coded decision making. On another hand, Section 4.3 examines the efficacy of the TrEO+ M_S in the domain continuous optimization, spanning benchmark functions and a real-world example on multi-location inventory planning. Section 4.4 then summarizes the chapter.

² The work in this chapter has been published in [13].

4.1 The TrEO+ M_S Algorithm

The solution representation learning module embedded in our TrEO framework aims to induce greater alignment (of solution representations) and hence positive inter-task transfers in cases of source-target domain mismatch, characterized either by (i) *arbitrarily heterogeneous source and target dimensionalities* (i.e., $d_S \neq d_T \Rightarrow \mathcal{X}_S \neq \mathcal{X}_T \Rightarrow \mathcal{D}_S \neq \mathcal{D}_T$), or (ii) *a lack of overlap between optimized source and target search distributions* (i.e., $p_S^*(\mathbf{x}_S) \not\approx p_T^*(\mathbf{x}_T) \Rightarrow \mathcal{D}_S \neq \mathcal{D}_T$), or both; see Section 3.2. Referring to the framework in Section 3.4 as well as Definition 3.1, this section proposes the TrEO+ M_S , a novel probabilistic model-based TrEO algorithm which utilizes source-to-target map $M_S: \mathcal{X}_S \rightarrow \mathcal{X}_T$ to learn well-aligned solution representations that help induce (i) $d_S = d_T$, or (ii) $p_{M_S}^*(\mathbf{x}_T) \approx p_T^*(\mathbf{x}_T)$, or both. A pseudocode of the TrEO+ M_S is provided in Algorithm 4.1. It is worth noting that the proposed TrEO+ M_S accounts for *multiple sources*, which is an extension of the exposition in Section 3.3 and Section 3.4 that considered only a single source.

The inputs to Algorithm 4.1 are a knowledge base of source model(s) $p_S^*(\mathbf{x}_S)$ and data drawn from $T - 1$ source tasks $\mathcal{J}_S \forall S \in \{1, \dots, T - 1\}$, the objective function f_T of the target task $\mathcal{J}_T$ and its solution representation space $\mathcal{X}_T$, and the transfer interval Δ . The TrEO search starts by sampling and evaluating a population of N random initial solutions which form the parent population $P_T(t)$ (lines 1–3). When canonical EA search is executed at non-transfer iterations (i.e., $\text{mod}(t, \Delta) \neq 0$), standard crossover and mutation operators are applied on $P_T(t)$ to generate the offspring population $C_T(t)$ (lines 5–6).

When the user-defined transfer interval parameter Δ triggers the sequential transfer iterations (i.e., $\text{mod}(t, \Delta) = 0$), we first construct the source-to-target spatial mapping M_S to learn new solution representations for source task $\mathcal{J}_S \forall S \in \{1, \dots, T - 1\}$ (lines 8–9). Subsequently, each original source model $p_S^*(\mathbf{x}_S)$ is transformed via simple Monte Carlo approximation to obtain the corresponding transformed model $p_{M_S}^*(\mathbf{x}_T)$ (line 10); see Algorithm 3.1. The transfer coefficients are then obtained based on the current target model $p_T(\mathbf{x}_T|t)$ and $p_{M_S}^*(\mathbf{x}_T) \forall S \in \{1, 2, \dots, T - 1\}$, after which the target mixture model is built and sampled to generate $C_T(t)$ (lines 11–12). The next iteration proceeds after selecting the new parent population $P_T(t + 1)$ from

$P_T(t) \cup C_T(t)$ based on elitism, and the entire optimization process repeats until the stopping condition is satisfied (lines 14–16).

Algorithm 4.1. Pseudocode of the proposed TrEO+ M_S algorithm

Input: Source database; f_T ; $\mathcal{X}_T$; transfer interval Δ

Output: Data and optimized search distribution for target $\mathcal{T}_T$

- 1: Set $t = 0$
 - 2: Sample N initial solutions uniformly at random from $\mathcal{X}_T$ to form $P_T(t)$
 - 3: Evaluate fitness of solutions $\mathbf{x}_T$ in $P_T(t)$ w.r.t. f_T
 - 4: **While** stopping condition not met **do**
 - 5: **If** $\text{mod}(t, \Delta) \neq 0$ **then**
 - 6: Generate $C_T(t)$ by applying standard crossover and/or mutation operators on $P_T(t)$
 - 7: **Else**
 - 8: Construct the mapping $M_S \forall S \in \{1, 2, \dots, T-1\}$ using available source priors and knowledge from $\mathcal{T}_T$
 - 9: Apply M_S to learn new solution representations for source $\mathcal{T}_S, \forall S \in \{1, 2, \dots, T-1\}$
 - 10: Transform $p_S^*(\mathbf{x}_S), \forall S \in \{1, 2, \dots, T-1\}$, using the corresponding M_S ; refer to Algorithm 3.1
 - 11: Obtain $w_1, w_2, \dots, w_T$ by solving Eq. (3.12) or Eq. (3.13), based on $p_{M_S}^*(\mathbf{x}_T) \forall S \in \{1, 2, \dots, T-1\}$ and $p_T(\mathbf{x}_T|t)$
 - 12: Build the target mixture model $[\sum_{S=1}^{T-1} w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T|t)]$ and sample N solutions to get $C_T(t)$
 - 13: **End if**
 - 14: Evaluate fitness of solutions $\mathbf{x}_T$ in $C_T(t)$ w.r.t. f_T
 - 15: Select N solutions from $P_T(t) \cup C_T(t)$ based on elitism, to form $P_T(t+1)$
 - 16: Set $t = t + 1$
 - 17: **End while**
-

FIGURE 4.1. Illustrative flowchart of the proposed TrEO+ M_S algorithm, featuring the possibility of an offline construction or an online learning of the source-to-target mapping M_S . The solution representation learning process has been highlighted.

Notably, the workflow of Algorithm 4.1 is not limited to any specific construction of source-to-target mapping M_S (in line 8). On one hand, in areas where sufficient domain expertise is available, an *offline* specification of M_S is possible. On the other hand, in the general setting of black-box optimization tasks with scarceness of domain knowledge, the data generated during the course of the target optimization search could be utilized for learning appropriate mapping functions M_S *online*. In both scenarios, the utility of solution representation learning (via the mapping M_S) enables

the transformation of the source models to the target search space. Figure 4.1 shows an illustrative flowchart of the TrEO+M_S. Moreover, Algorithm 4.1 provides the *flexibility to incorporate various probabilistic models to represent search distributions in distinct optimization domains* (e.g., discrete and continuous optimization) for its source-target similarity capture mechanism (executed in lines 11–12).

As shall be experimentally shown in subsequent sections, the above enhancements in our algorithm are vital to achieving a better balance between the curbing of negative transfers and the activation of positive transfers. To this end, we present case studies in the domains of discrete and continuous optimization, along with their respective TrEO+M_S algorithmic instantiations based on novel methodologies for constructing the source-to-target spatial mappings M_S. Computational experiments are conducted to showcase the efficacy of our methods in each case study.

4.2 Case Study in Discrete Optimization: A 0-1 Knapsack Problem Exemplar

In this section, we consider the 0-1 knapsack problem (KP) [90] as an illustrative case study of binary-coded decision making, to showcase the efficacy of the TrEO+M_S algorithm when applied in a discrete optimization domain. Due to its ubiquity in various practical applications, the KP has been widely studied over the years in computer science and operations research, with a number of heuristics and meta-heuristic optimization procedures proposed – the most prominent among them being *Dantzig’s greedy approximation algorithm* [91].

Let there be a set of m items each of which has a value v_i and a weight q_i , $i \in \{1, \dots, m\}$, and a knapsack of capacity Q . Then, the 0-1 KP aims to select a combination of items so that the total profit of the selected items is maximized without violating the knapsack’s capacity constraint. The overall mathematical program is stated as,

$$\max_{x \in \{0,1\}^m} \sum_{i=1}^m x_i \cdot v_i,$$

$$\text{s. t.,} \quad \sum_{i=1}^m q_i \cdot x_i \leq Q \text{ and } x_i \in \{0,1\} \forall i, \quad (4.1)$$

where x_i is the binary variable indicating the selection of item i (i.e., $x_i = 1$ if item i is selected and $x_i = 0$ otherwise) and a candidate solution is represented by a binary vector $\mathbf{x} = (x_1, x_2, \dots, x_m)^T$.

4.2.1 Methodology for Constructing M_S

According to Dantzig's greedy approximation algorithm [91], given a KP instance, the first step is to compute the *efficiency ratio* of each of its items as,

$$r_i = \frac{v_i}{q_i} \quad \forall i \in \{1, 2, \dots, m\}. \quad (4.2)$$

The items are then inserted into the knapsack in *decreasing order* of efficiency, until the capacity constraint is violated. Although this algorithm does not guarantee optimality, it is able to provide good solutions in practice.

Dantzig's algorithm offers us the general insight that even across distinct KP instances (say, a source KP_S defined in search space $\mathcal{X}_S$ and a target KP_T defined in search space $\mathcal{X}_T$), items with high efficiency ratio are similarly treated. Specifically, the item with the highest efficiency among all remaining items in a KP instance is preferentially considered for insertion into the knapsack. Nevertheless, it is worth noting that in TrEO, *unaligned solution representations can cause items with high efficiency ratios in source KP_S to correspond to items with low efficiency ratios in target KP_T* . This leads to a scenario of source-target domain mismatch, whereby there is *a lack of overlap between optimized source and target search distributions* (i.e., $p_S^*(\mathbf{x}_S) \not\approx p_T^*(\mathbf{x}_T) \Rightarrow \mathcal{D}_S \neq \mathcal{D}_T$); see Section 3.2. Thus, simply transferring knowledge between distinct KP instances in TrEO may result in reduced beneficial positive transfers or even increased harmful negative transfers.

The above insights motivate our proposed methodology for the construction of a source-to-target spatial mapping M_S , with the aim of inducing greater alignment of solution representations and hence positive transfers between distinct 0-1 KP instances. The mapping is defined as,

$$M_S(\mathbf{x}_S) = \mathbf{A}_{\mathcal{P}} \cdot \mathbf{x}_S, \quad (4.3)$$

where $\mathbf{x}_S \in \mathcal{X}_S$ and $\mathbf{A}_{\mathcal{P}}$ is a *permutation matrix* for reordering items in KP_S such that their efficiency ratios (positively and ordinaly) correlate with those of the items in KP_T .

To demonstrate how a permutation matrix $\mathbf{A}_{\mathcal{P}}$ can be handcrafted for the aforementioned purpose, let us consider a simple numerical example as follows. Firstly, we have a source KP_S of 4 items that are characterized by their respective (i) values $\mathbf{v}_S = (1,2,3,4)$, (ii) weights $\mathbf{q}_S = (1,1,1,1)$, and (iii) efficiency ratios $\mathbf{r}_S = (1,2,3,4)$. Secondly, the target KP_T also consists of 4 items that are characterized by their corresponding (a) values $\mathbf{v}_T = (4,2,3,1)$, (b) weights $\mathbf{q}_T = (1,1,1,1)$, and (c) efficiency ratios $\mathbf{r}_T = (4,2,3,1)$. Thirdly, we assume that both KP_S and KP_T have the same maximum capacity $Q = 7$. Based on the source's originally defined search space, transferring high quality source solutions (e.g., $\mathbf{x}_S^* = (0,0,1,1)^T$) to the target search is unlikely to induce positive transfers owing to the lack of alignment between their items' efficiency ratios. The mapping M_S in Eq. (4.3) can learn well-aligned source solution representations that increase positive transfers when the $\mathbf{A}_{\mathcal{P}}$ is defined as,

$$\mathbf{A}_{\mathcal{P}} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix},$$

where,

$$M_S(\mathbf{x}_S^*) = \mathbf{A}_{\mathcal{P}} \cdot \mathbf{x}_S^* = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix},$$

such that the source items' efficiency ratios under mapping M_S becomes $\mathbf{r}_{M_S} = \mathbf{r}_T = (4,2,3,1)$, and $M_S(\mathbf{x}_S^*) = \mathbf{x}_T^* = (1,0,1,0)^T$ hence inducing $p_{M_S}^*(\mathbf{x}_T) \approx p_T^*(\mathbf{x}_T)$.

Given the insights in this exemplar, we note that M_S can be constructed offline and subsequently fixed throughout the target search; see Figure 4.1. Moreover, we contend that the transformation in Eq. (4.3) will facilitate the learning of well-aligned solution representations that unveil any useful but hidden source-target relationships. The combination of our proposed methodology and a *univariate marginal probabilistic model* (to represent search distributions in discrete binary spaces [92]) results in a TrEO+ M_S algorithmic instantiation for this case study. In what follows, we conduct numerical experiments to verify the efficacy of the resultant algorithm.

4.2.2 Experimental Study

4.2.2.1 Experimental Setup

Three optimizers are considered in this case study. These are: (i) a *canonical evolutionary algorithm* (CEA) without transfer, (ii) the baseline AMTEA without solution representation learning [10], and (iii) the proposed TrEO+ M_S . The parameter configurations of the three optimizers are kept consistent for fairness. They employ the uniform crossover operator (with crossover probability of 1.0), bit-flip mutation (with probability $1/m$), a population size of $N = 50$ binary-coded solutions. Furthermore, the same greedy solution repair procedure from [93] has been incorporated in the CEA, AMTEA and TrEO+ M_S to ensure capacity constraint satisfaction. We set the transfer interval $\Delta = 2$ in both AMTEA and TrEO+ M_S , at which stacked density estimation [51] is launched to build the optimal mixture of source and target probabilistic models. Each component of the mixture model takes the form of a univariate marginal distribution [92].

Here, all the source and target KP instances are generated independently at random as proposed in [93]. A KP instance can be classified either as *strongly correlated* (sc), *weakly correlated* (wc) or *uncorrelated* (uc), depending on the relationship between its items' values and weights. Our experiments are designed such that we have access to optimized search distribution models drawn from multiple source tasks, viz., “source_KP_sc”, “source_KP_wc” and “source_KP_uc”, while solving the target tasks “target_KP_sc”, “target_KP_wc” and “target_KP_uc”. The capacity of the knapsack is *restrictive* in all the instances, which is to say that very few items can be inserted into it.

4.2.2.2 Results and Discussion

Figure 4.2 depicts all the averaged convergence trends as well as the transfer coefficients obtained by the TrEO+ M_S and AMTEA. Results in the leftmost column clearly show that our algorithm is able to converge significantly faster than the baseline AMTEA, which in turn performs better than the CEA. In order to explain this outcome, we draw attention to the transfer coefficients (these are plotted in the central and rightmost columns of Figure 4.2 for the TrEO+ M_S and AMTEA, respectively). Even though all the KP instances are generated independently at random, intuition suggests that source and target tasks that belong to the same class of KP instances (i.e., sc, wc, or uc) are likely to share the most in common.

Our intuition is closely borne out by the transfer coefficients of the TrEO+ M_S . As can be seen in the central column of Figure 4.2 (i.e., in Figure 4.2(b), (e), (h)), our algorithm correctly assigns higher transfer coefficient values to source models that belong to the same class as the target task. As discussed in Section 3.2, the observed outcome is achieved by the transformed source solution representations of Eq. (4.3) which successfully uncover beneficial inter-task relationships that remain hidden in the source problem’s originally search space. This manifests in well-aligned solution representations that induce greater overlap between optimized source and target search distributions, and hence, higher transfer coefficient values being deduced for the more relevant sources. As a result, TrEO+ M_S ’s convergence rates are enhanced by positive transfers from the most related source tasks.

In contrast, as seen in the rightmost column of Figure 4.2 (i.e., in Figure 4.2(c), (f), (i)), the AMTEA (without solution representation learning) tends to wrongly assign higher transfer coefficient values to source tasks that belong to a class different from that of the target; e.g., in the case of the *uncorrelated* “target_KP_uc”, the transfer coefficient assigned to the *strongly correlated* “source_KP_sc” is highest in the AMTEA; see Figure 4.2(i). As a consequence, even though the AMTEA is able to receive some positive transfer from the various sources (which explains its performance in comparison to CEA), it is unable to fully tap the desired potential of TrEO. The TrEO+ M_S successfully overcomes this limitation by virtue of solution representation learning (via the source-to-target spatial mapping M_S) which induces positive transfers.

FIGURE 4.2. Convergence trends and transfer coefficients achieved for target task “target_KP_sc” (top row), “target_KP_wc” (central row), and “target_KP_uc” (bottom row). The central and rightmost columns depict the transfer coefficients obtained by the TrEO+M_S and the baseline AMTEA, respectively. All results are averaged over 30 independent runs of each optimizer. The shaded region spans one standard deviation on either side of the mean.

4.3 Case Studies in Continuous Optimization

While the previous section demonstrated the performance advantage of the TrEO+M_S for discrete optimization (considering the 0-1 Knapsack Problem as exemplar), there exist many practical problems in the domain of continuous optimization as well. With

this in mind, this section investigates the applicability of the TrEO+ M_S for boosting the performance of optimization tasks defined in *continuous search spaces*. Specifically, we shall showcase the efficacy of the TrEO+ M_S algorithm when *the source-to-target mapping M_S is learned online using the source and target search data*; see Figure 4.1.

4.3.1 Methodology for Learning M_S

Consider a TrEO setting in which high quality solutions from a source task (defined in a continuous search space $\mathcal{X}_S$ of dimensionality d_S) are transformed using a source-to-target mapping M_S , and transferred to a target problem (defined in a continuous space $\mathcal{X}_T$ of size d_T). Existing works have shown promising results of learning transformed source solution representations through linear mapping functions [57, 94]. Although such linear transformations work well when $d_T \leq d_S$, the source’s optimized search distribution model $p_{M_S}^*(\mathbf{x}_T)$ in the linearly transformed space becomes degenerate when $d_T > d_S$ – as $p_{M_S}^*(\mathbf{x}_T)$ will be supported only on a lower d_S -dimensional subspace; see also Section 3.3.1. Importantly, the target mixture model in Algorithm 4.1 can no longer be built given degenerate priors.

To address the aforementioned issue, some simplistic approaches have been proposed in the literature, including padding additional variables to the solution representations [57] or probabilistic models [10] of the source task. However, in many cases of unaligned solution representations causing $p_S^*(\mathbf{x}_S) \not\approx p_T^*(\mathbf{x}_T)$, these simple approaches are unlikely to suffice in unveiling beneficial inter-task relationships; thereby reducing the potential for positive transfers. In order to handle arbitrarily heterogeneous source and target dimensionalities (i.e., $d_T > d_S$ and $d_T \leq d_S$) as well as to increase the overlap between their optimized search distributions, we thus propose a methodology for learning well-aligned solution representations in continuous spaces via a *fast* and *non-linear* source-to-target spatial mapping M_S . Under the proposed non-linear mapping, the transformed source model $p_{M_S}^*(\mathbf{x}_T)$ will not be degenerate even when $d_T > d_S$; hence, making possible the activation of positive inter-task transfers according to Algorithm 4.1.

Taking a cue from [57], we assume that the parent population $P_S(t) = \{\mathbf{x}_{S,1}, \mathbf{x}_{S,2}, \dots, \mathbf{x}_{S,N}\}$ of a source task $\mathcal{T}_S$ (with objective function f_S) had been cached at every iteration t of an EA, during its optimization in the past. In the ongoing target task $\mathcal{T}_T$ (with objective function f_T), let its parent solutions at the current (t -th) iteration be $P_T(t) = \{\mathbf{x}_{T,1}, \mathbf{x}_{T,2}, \dots, \mathbf{x}_{T,N}\}$. The solutions in $P_S(t)$ and $P_T(t)$ are *sorted* in decreasing order of their fitness evaluations, resulting in a monotonic relationship between the two sets; i.e., $f_S(\mathbf{x}_{S,i}) > f_S(\mathbf{x}_{S,j}) \Rightarrow f_T(\mathbf{x}_{T,i}) \geq f_T(\mathbf{x}_{T,j})$, and vice versa. Consequently, for a large population size N , building a mapping $M_S(\mathbf{x}_S)$ from $P_S(t)$ to $P_T(t)$ will locally, within the support of distribution $p_T(\mathbf{x}_T|t)$, induce strong (positive) ordinal correlation between $f_S(M_S^{-1}(\mathbf{x}_T))$ and $f_T(\mathbf{x}_T)$; note that M_S^{-1} is a hypothetical inversion that need not be explicitly derived in practice. Assuming the global optima $\mathbf{x}_S^*$ and $\mathbf{x}_T^*$ to be contained within the support of $p_S(\mathbf{x}_S|t)$ and $p_T(\mathbf{x}_T|t)$, respectively, we can then invoke Proposition 3.1 to approximate $p_{M_S}^*(\mathbf{x}_T) \approx p_T^*(\mathbf{x}_T)$ in $\mathcal{X}_T$.

For any source-target pair, our proposed source-to-target spatial mapping M_S takes the form of a *two-layer feedforward neural network*. The parameters $\boldsymbol{\theta}$ of this network are learned for mapping $P_S(t)$ to $P_T(t)$ by optimizing against the objective,

$$\min_{\boldsymbol{\theta}} \sum_{j=1}^N \|\mathbf{x}_{T,j} - M_S(\mathbf{x}_{S,j}; \boldsymbol{\theta})\|_2^2 + \lambda \|\boldsymbol{\theta}\|_2^2, \quad (4.4)$$

where the second term is an L_2 regularization of the parameters. Note that as the learning of M_S is to be nested in step 8 of Algorithm 4.1, it is important for the learning to be fast in order to keep the computational overhead to a minimum.

With that in mind, consider the two-layer neural network M_S to be formed by the composition,

$$M_S(\mathbf{x}_S; \boldsymbol{\theta}) = \boldsymbol{\psi}_2(\boldsymbol{\psi}_1(\mathbf{x}_S)), \quad (4.5)$$

in which,

$$\boldsymbol{\psi}_1(\mathbf{x}_S) = \sigma(\mathbf{A}_1 \cdot \mathbf{x}_S + \mathbf{b}), \quad (4.6)$$

and,

$$\boldsymbol{\psi}_2(\boldsymbol{\psi}_1(\mathbf{x}_S)) = \mathbf{A}_2 \cdot \boldsymbol{\psi}_1(\mathbf{x}_S). \quad (4.7)$$

Let $\boldsymbol{\theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2]$ in Eq. (4.5), such that $\boldsymbol{\theta}_1$ serves as the entries of matrix $\mathbf{A}_1 \in \mathbb{R}^{d_h \times d_s}$ and the bias vector $\mathbf{b} \in \mathbb{R}^{d_h}$ in Eq. (4.6). Likewise, in Eq. (4.7), $\boldsymbol{\theta}_2$ serves as the entries of matrix $\mathbf{A}_2 \in \mathbb{R}^{d_T \times d_h}$; d_h is the number of neurons in the hidden layer of the neural network, and σ symbolizes the non-linear sigmoidal activation function. In the spirit of [95], the elements of $\boldsymbol{\theta}_1$ are randomly assigned (by sampling from the standard normal distribution) and then fixed. Thereafter, entries of $\mathbf{A}_2$ (i.e., $\boldsymbol{\theta}_2$) can be obtained using the *closed-form expression*,

$$\mathbf{A}_2 = P_T \cdot [\boldsymbol{\psi}_1(P_S)]^{\text{tr}} \cdot (\boldsymbol{\psi}_1(P_S) \cdot [\boldsymbol{\psi}_1(P_S)]^{\text{tr}} + \lambda \cdot I_{d_h})^{-1}, \quad (4.8)$$

where $[\cdot]^{\text{tr}}$ indicates a matrix transpose, λ is the regularization coefficient and I_{d_h} is an identity matrix of size d_h . Given the closed-form expression of Eq. (4.8), the computational complexity of learning the proposed non-linear mapping M_S is comparable to its linear counterpart (see Section 3.3.1).

$\mathbf{A}_2$ is updated online in step 8 of Algorithm 4.1 based on population datasets $P_S(t)$ and $P_T(t)$ at iteration t . Given continuous search spaces, the probabilistic models $p_S^*(\mathbf{x}_S)$, $p_{M_S}^*(\mathbf{x}_T)$, and $p_T(\mathbf{x}_T|t)$ are realizable in the form of multi-variate Gaussian distributions. Combining the proposed methodology for learning M_S and the Gaussian probability distribution, we acquire a TrEO+ M_S algorithmic instantiation that is operable for continuous search spaces where $\mathcal{X}_S \subseteq \mathbb{R}^{d_s}$ and $\mathcal{X}_T \subseteq \mathbb{R}^{d_T}$. Once the non-linear mapping M_S is learned for a source task $\mathcal{T}_S$, the transformation of its model $p_S^*(\mathbf{x}_S)$ to $p_{M_S}^*(\mathbf{x}_T)$ is achieved in our implementation via simple Monte Carlo approximation; see the procedure in Algorithm 3.1. For differentiable mapping functions, as is the case with neural networks, alternative analytical procedures for the non-linear transformation of Gaussians, offering greater computational speedup, are discussed in [96].

In the subsequent parts of this section, we experimentally verify the effectiveness of the resultant algorithm through case studies spanning continuous benchmark functions as well as a real-world example on multi-location inventory planning.

4.3.2 A Study of Synthetic Benchmark Functions

Here, we choose two synthetic test functions to demonstrate the efficacy of the proposed $\text{TrEO}+M_S$. The *Griewank function* is used as the source task, whereas the *Sphere function* serves as the target task [97, 98]. Both of these are *minimization* problems, and their mathematical expressions are given in Eq. (4.9) and Eq. (4.10), respectively.

$$f_S(\mathbf{x}_S) = \frac{1}{4000} \sum_{i=1}^{d_S} x_{S,i}^2 - \prod_{i=1}^{d_S} \cos\left(\frac{x_{S,i}}{\sqrt{i}}\right) + 1, \quad (4.9)$$

$$f_T(\mathbf{x}_T) = \sum_{j=1}^{d_T} (x_{T,j} - 10)^2, \quad (4.10)$$

where $x_{S,i} \in [0, 100]$ for $i = 1, \dots, d_S$, and $x_{T,j} \in [0, 100]$ for $j = 1, \dots, d_T$.

Notice that the first term in the Griewank function is the same as the Sphere, which lends the two functions similar global character (thus giving rise to the scope of knowledge transfer between them). However, while the Griewank function has multiple local optima (due to the second term in Eq. (4.9)), the Sphere function is unimodal. What is more, in our formulation of Eq. (4.10), the global optimum of the Sphere function has been *shifted* to $\{10\}^{d_T}$, while that of the Griewank function is at $\{0\}^{d_S}$. The shifted distribution of good solutions implies $p_S^*(\mathbf{x}_S) \neq p_T^*(\mathbf{x}_T) \Rightarrow \mathcal{D}_S \neq \mathcal{D}_T$.

Furthermore, we setup three different TrEO cases, namely, (i) *Case 1*: $d_T = d_S$, (ii) *Case 2*: $d_T < d_S$, and (iii) *Case 3*: $d_T > d_S$, using the test functions above. In all three cases we set $d_T = 20$, whereas $d_S = 20, 30, 10$, respectively. Recall that when $d_T > d_S$, a linear source-to-target mapping does not directly apply due to the

occurrence of a degenerate distribution. Even if $d_T = d_S$ (i.e., $\mathcal{X}_S = \mathcal{X}_T$), we have $p_S^*(\mathbf{x}_S) \neq p_T^*(\mathbf{x}_T)$ due to the shifted optimum of the Sphere function, which arguably renders TrEO algorithms without solution representation learning less effective.

4.3.2.1 Experimental Specifications

Five optimizers are considered for comparison, including state-of-the-art TrEO as well as non-TrEO methods. These are: (i) the *exponential natural evolution strategies* (xNES) which is a powerful continuous optimization algorithm for small-to-medium scale problems [99], (ii) the CEA without knowledge transfer, (iii) CEA+ M_S which is a non-linear extension of the TrEO search with adaptive learning across heterogeneous problems [57], (iv) the baseline AMTEA without solution representation learning [10], and (v) our proposed TrEO+ M_S . All optimizers are real-coded with a population size of $N = 50$. Apart from the xNES, all other algorithms employ uniform crossovers, Gaussian mutation (with probability 0.04) and tournament selection. We set the transfer interval to $\Delta = 2$, at which stacked density models with Gaussian components are learned in the AMTEA and the TrEO+ M_S [51], whereas transformed source solutions are directly injected into the CEA+ M_S . For xNES, its default parameter settings are used, with the population initialization chosen so as to facilitate fair comparison with its counterparts.

The same non-linear source-to-target mapping M_S , as described in Section 4.3.1, is used in both CEA+ M_S and TrEO+ M_S . In the two-layer feedforward neural network that constitutes M_S , we set the number of neurons in the hidden layer as $d_h = 2d_T$. The value of d_h should be set sufficiently large as the randomization of the first layer of the neural mapping function can lead to several redundant features. Any adverse impact of the redundant features is mitigated by the regularization term in Eq. (4.4). In our implementation, the regularization coefficient is randomized in the range [1, 1000]. The source optimization data was collected by running the CEA on the respective source tasks.

4.3.2.2 Results and Discussion

A summary of the results returned by the five optimizers (averaged over 30 runs) at the end of 5000 function evaluations is provided in Table 4.1. In all the cases, the TrEO algorithms (which execute canonical evolutionary search during non-transfer iterations) are able to surpass the standard CEA by harnessing solutions transferred from the source.

TABLE 4.1. Experimental results for the target shifted Sphere function, at the end of 5000 function evaluations. In bold are the best results with 95% confidence level based on the Wilcoxon signed-rank test. Results are reported based on 30 independent runs of all optimizers.

Problem Configuration	Algorithms	Target Objective Values			Q_{better} [100]
		Average	Best	Worst	
$d_T = d_S$	xNES	96.22	55.62	140.43	1.0000
	CEA	550.18	122.90	2104.94	1.0000
	CEA+ M_S	117.44	76.35	167.99	1.0000
	AMTEA	23.31	0.03	72.79	0.7944
	TrEO+ M_S	2.30	0.01	24.28	-
$d_T < d_S$	xNES	96.22	55.62	140.43	1.0000
	CEA	550.18	122.90	2104.94	1.0000
	CEA+ M_S	110.70	53.41	182.63	1.0000
	AMTEA	8.94	0.02	68.76	0.8011
	TrEO+ M_S	0.96	0.00	9.45	-
$d_T > d_S$	xNES	96.22	55.62	140.43	1.0000
	CEA	550.18	122.90	2104.94	1.0000
	CEA+ M_S	121.36	53.38	231.07	1.0000
	AMTEA	39.60	0.50	215.56	0.9822
	TrEO+ M_S	0.63	0.01	10.11	-

The highlight of Table 4.1 is that the TrEO+ M_S performs significantly better than all the other algorithms. This is attributed to the fact that our algorithm accesses the complementary benefits of probabilistic model-based transfer and non-linear solution representation learning through M_S . The CEA+ M_S on the other hand does not

explicitly guard against negative transfers, which tends to slow down its convergence rate in the presence of inaccurately learned source-to-target transformations. This is evident when $\text{CEA}+M_S$ is consistently outperformed by xNES , AMTEA and $\text{TrEO}+M_S$ in all three cases. Even though the xNES performs significantly better than CEA , its solution quality is surpassed by both AMTEA and $\text{TrEO}+M_S$. Comparing the baseline AMTEA with the CEA in Table 4.1, the former clearly benefits from a boost in performance as a consequence of the source induced search bias. However, the final objective values reached by the AMTEA are far from optimal. In contrast, the scope for non-linear source-to-target mapping in the $\text{TrEO}+M_S$ allows it to adapt to source-target domain mismatches, approaching near-optimal solutions more consistently.

FIGURE 4.3. Convergence trends for the target shifted Sphere function, and the transfer coefficients obtained by the AMTEA and the $\text{TrEO}+M_S$ in *Case 1*: $d_T = d_S$ (left column), *Case 2*: $d_T < d_S$ (central column), and *Case 3*: $d_T > d_S$ (rightmost column). Source data comes from the multimodal Griewank function. The shaded region spans one standard deviation on either side of the mean (based on 30 independent runs).

The superior convergence trends of our proposed algorithm are depicted in Figure 4.3. To this end, we draw attention to the larger convergence performance boost achieved by the TrEO+M_S in Case 3 ($d_T > d_S$), which is a direct result of a larger scope for solution representation learning to unveil beneficial but hidden source-target information; see Figure 4.3(c). Besides that, Figure 4.3 also shows the transfer coefficients patterns obtained by AMTEA and TrEO+M_S. Interestingly, our algorithm is able to obtain substantial values of w_S for Case 3, while the AMTEA is unable to capture any noticeable source-target similarity – this confirms that the TrEO+M_S successfully induces greater alignment (of solution representations) and hence positive transfers, which also explains the larger performance gap; see Figure 4.3(f).

In addition to the above, we conduct an additional quantitative analysis using a probabilistic performance comparison metric, denoted herein as Q_{better} ; readers are referred to [100] for a detailed description of this metric. Roughly speaking, Q_{better} measures the empirical probability that the TrEO+M_S achieves better target objective values relative to a baseline algorithm. The closer Q_{better} is to 1.0, the greater the confidence we have in the superior performance of our proposed algorithm.

Based on the Q_{better} values in Table 4.1, we can claim that TrEO+M_S performs better than xNES, CEA and CEA+M_S with high confidence across the 30 independent runs. When considering the Q_{better} values relative to AMTEA, the $d_T > d_S$ case gives the largest probability of 98.2% that AMTEA is outperformed by our algorithm. Furthermore, Table 4.1 shows that TrEO+M_S’s best and worst objective values are consistently smaller than those of the other algorithms. These outcomes strongly support the utility of the non-linear source-to-target mappings M_S learned in our algorithm for inducing positive transfers, while simultaneously mitigating negative inter-task interactions through the probabilistic model-based source-target similarity capture mechanism. Our results also suggest that the solution representation learning module in the TrEO+M_S is able to uncover hidden inter-task relationships, which leads to enhanced target convergence rate in accordance with the discussions in Section 3.2.

4.3.3 A Real-World Example on Multi-Location Inventory Planning

Next, we consider a practical case study of multi-location inventory planning under demand uncertainty, for validating the proposed $\text{TrEO}+M_S$ algorithm in a real-world application. The plausible *source-target domain mismatch* scenarios (which could give rise to *unaligned solution representations*) in this study stem from the fact that the *number of locations* (i.e., problem dimensionalities), as well as the *expected demand patterns of products* (i.e., probability distributions of good solutions), could frequently change over time.

In complex supply chains, *inventory redistribution* has been studied to cope with the imbalance between realized demands and inventory levels. In this study, we account for instantaneous lateral transshipments for inventory redistribution in a multi-location inventory system consisting of a single distribution centre and multiple retailers. At every decision-making step, the optimal order-up-to replenishment level at each retailer is to be determined so that the total expected cost (which includes the inventory holding cost c_h , shortage or backlogging cost c_b , and lateral transshipment cost c_{lt}) is minimized. Lateral transshipments occur when the shortage of inventory at one retailer can be counterbalanced by a surplus of inventory at another retailer. A detailed description and mathematical model of the problem can be found in [101]. Briefly, the cost minimization program can be stated as,

$$\min_{\mathbf{x}} \quad \mathbb{E}_{D \sim p(D)}[(c_h + c_b + c_{lt})|\mathbf{x}, D], \quad (4.11)$$

where $\mathbf{x} \in \mathbb{R}_{\geq 0}^d$ is the order-up-to replenishment level at each of d retailers, and $p(D)$ is the uncertain demand distribution with mean $\boldsymbol{\mu}$ and variance Σ .

The above multi-location inventory planning problem gives rise to a range of possible scenarios in which the efficacy of TrEO algorithms can be tested. In the experiments that follow, we consider *three contrasting cases with increasing discrepancies between source and target tasks* (accompanied by increasing scope for solution representation learning in the $\text{TrEO}+M_S$). (i) In the simplest case, the number of retailers in the source and the target optimization tasks are the same (i.e., $d_T = d_S$)

and the expected demands at the retailers do not change over time (we denote this as $\mu_T = \mu_S$). (ii) In the second case, while the number of retailers remains the same (i.e., $d_T = d_S$), the expected demands at the retailers differ in the source and target optimization tasks (i.e., $\mu_T \neq \mu_S$); this could reflect a practical scenario where different products sold by the same retailers face different demand patterns. (iii) Finally, we consider a third case that involves an increase in the number of retailers over time (i.e., $d_T > d_S$), as well as a change in the expected demands at the retailers (i.e., $\mu_T \neq \mu_S$). Recall that when $d_T > d_S$, a linear source-to-target mapping does not directly apply in conjunction with probabilistic formulations of TrEO; see Section 3.3.1.

The three cases described above can accordingly be summarized as, *Case 1*: $d_T = d_S$ and $\mu_T = \mu_S$, *Case 2*: $d_T = d_S$ but $\mu_T \neq \mu_S$, and *Case 3*: $d_T > d_S$ and $\mu_T \neq \mu_S$. In all cases we set $d_T = 20$, whereas $d_S = 10$ (when $d_T > d_S$) or 20 (when $d_T = d_S$).

4.3.3.1 Experimental Setting

In our implementation of the multi-location inventory planner, given any candidate solution $x_T \in \mathbb{R}_{\geq 0}^{d_T}$ in an EA, nested linear programs (LPs) must be solved (using an off-the-shelf LP solver) to obtain corresponding optimal lateral transshipments under various demands realizations sampled from $p(D_T)$ [101]. Thus the fitness evaluation of every candidate solution is quite expensive, to a degree that the marginal computational overhead in learning the non-linear source-to-target mapping M_S and subsequent stacked density estimation in the TrEO+ M_S is negligible.

Given the above, we consider the same five optimizers used for comparison in the synthetic benchmark study of Section 4.3.2. They are the CEA, the xNES [99], the CEA+ M_S [57], the AMTEA [10] and the proposed TrEO+ M_S . The evolutionary operators and parameter settings for each optimizer are also configured in a similar manner as Section 4.3.2.

4.3.3.2 Results and Discussion

A summary of the results returned by the five optimizers (averaged over 30 runs) at the end of 3000 function evaluations is provided in Table 4.2. As in previous case studies, we find that the various TrEO algorithms are able to outperform the standard CEA by exploiting the search biases obtained from the source optimization data. The source data used for this purpose is collected by running the CEA on the respective source tasks. In two out of the three cases, the TrEO+M_S is found to be significantly better than all its counterparts (with 95% confidence based on the Wilcoxon signed-rank test). Here although xNES achieves a better target objective value than CEA, its solution quality is outperformed by all TrEO algorithms.

TABLE 4.2. Experimental results for the target multi-location inventory planning task, at the end of 3000 function evaluations. In bold are the best results with 95% confidence level based on the Wilcoxon signed-rank test. Results are reported based on 30 independent runs of all optimizers.

Problem Configuration	Algorithms	Target Objective Values			Q_{better} [100]
		Average	Best	Worst	
$d_T = d_S$ ($\mu_T = \mu_S$)	xNES	6.10 e3	4.93 e3	6.95 e3	0.9978
	CEA	6.96 e3	4.88 e3	10.35 e3	0.9978
	CEA+M _S	4.18 e3	3.49 e3	4.99 e3	0.8544
	AMTEA	3.45 e3	2.38 e3	4.68 e3	0.5467
	TrEO+M _S	3.37 e3	2.51 e3	5.15 e3	-
$d_T = d_S$ ($\mu_T \neq \mu_S$)	xNES	6.10 e3	4.93 e3	6.95 e3	1.0000
	CEA	6.96 e3	4.88 e3	10.35 e3	1.0000
	CEA+M _S	4.15 e3	3.44 e3	5.31 e3	0.8222
	AMTEA	4.19 e3	3.69 e3	5.05 e3	0.8533
	TrEO+M _S	3.53 e3	2.66 e3	4.81 e3	-
$d_T > d_S$ ($\mu_T \neq \mu_S$)	xNES	6.10 e3	4.93 e3	6.95 e3	0.9978
	CEA	6.96 e3	4.88 e3	10.35 e3	0.9967
	CEA+M _S	4.08 e3	3.18 e3	6.20 e3	0.8833
	AMTEA	4.06 e3	3.55 e3	5.29 e3	0.8511
	TrEO+M _S	3.45 e3	2.57 e3	5.32 e3	-

For *Case 1*, where $d_T = d_S$ and $\mu_T = \mu_S$, AMTEA performs competitively with $\text{TrEO}+M_S$ as there is little difference between the source and target domains (i.e., $\mathcal{D}_S \approx \mathcal{D}_T$); thus, there is little to be gained by the solution representation learning module of our algorithm. However, as the source and target domain mismatch gradually increases from *Case 2* (where $d_T = d_S$ but $\mu_T \neq \mu_S$) to *Case 3* (where $d_T > d_S$ and $\mu_T \neq \mu_S$), the performance boost achieved by the $\text{TrEO}+M_S$ becomes increasingly apparent; see Figure 4.4 and Figure 4.5.

FIGURE 4.4. Convergence trends for the target multi-location inventory planning task, and the transfer coefficients obtained by the AMTEA and the $\text{TrEO}+M_S$ in *Case 1*: $d_T = d_S$ and $\mu_T = \mu_S$ (left column), *Case 2*: $d_T = d_S$ but $\mu_T \neq \mu_S$ (central column), and *Case 3*: $d_T > d_S$ and $\mu_T \neq \mu_S$ (rightmost column). The shaded region spans one standard deviation on either side of the mean (based on 30 independent runs).

Comparisons of convergence trends for the target task are presented in Figure 4.4. One of the key highlights of Figure 4.4 is the transfer coefficients obtained by the AMTEA and our algorithm. In Figure 4.4(d) for *Case 1*, the coefficients patterns are somewhat similar. In contrast, in Figure 4.4(e), (f) for *Case 2* and *Case 3*, respectively, $\text{TrEO}+M_S$ assigns significantly higher transfer coefficient values to the source

probabilistic model. This observation clearly demonstrates the utility of solution representation learning (via the source-to-target spatial mapping M_S) for inducing positive inter-task transfers in cases of increasing source-target domain mismatch.

As elaborated in Section 3.2, the performance advantage of the TrEO+ M_S is attributed to the transformed source solution representations of Eq. (4.5), which successfully uncovers useful source-target relationships that remain hidden in the problem’s originally defined search space. Accordingly, greater alignment (of solution representations) and hence positive transfers are induced, and thereby, allowing larger source-target similarities (quantified by w_S values) to be captured in the TrEO+ M_S . On the other hand, narrow focus on the curbing of negative transfers in the baseline AMTEA causes the conservative attenuation of knowledge transfers – as is reflected by the much lower transfer coefficient values that the algorithm deciphers; see Figure 4.4(e), (f). The performance gap between TrEO+ M_S and AMTEA is depicted in Figure 4.5 – this result further highlights the performance boosts in the TrEO+ M_S due to the synergy between its solution representation learning module and inter-task similarity capture mechanism.

FIGURE 4.5. Performance gap between the TrEO+ M_S and the AMTEA on the multi-location inventory cost minimization problem. The gap is measured by subtracting the objective values achieved by TrEO+ M_S from those achieved by AMTEA in Figure 4.4. The plot shows that the performance boost in the TrEO+ M_S is more significant as the extent of source-target domain mismatch increases.

4.4 Chapter Summary

This chapter demonstrated the applicability of our framework for solution representation learning in probabilistic model-based TrEO, specifically for inducing greater alignment (of solution representations) and hence positive transfers in the context of *single-objective optimization*. To this end, we proposed the TrEO+ M_S , which is a novel probabilistic model-based TrEO algorithm with source-to-target map, aimed at addressing source-target domain mismatch scenarios characterized by (i) arbitrary discrepancies between source and target search space dimensionality, or (ii) a lack of overlap between their optimized search space distributions, or both.

Distinct case studies to verify the effectiveness of the TrEO+ M_S were presented in the domains of *discrete* and *continuous* optimization problems. On one hand, for an illustrative case study of binary-coded decision making tasks, we considered the 0-1 Knapsack Problem. A novel methodology for constructing source-to-target spatial mappings was proposed to induce positive transfers in this exemplar, hence brought about a unique TrEO+ M_S algorithmic instantiation. On another hand, Chapter 3 revealed that non-linear source-to-target maps offer a more generalizable form of solution representation learning (as opposed to their linear counterpart) in continuous search spaces. Accordingly, a novel methodology was proposed for the fast learning of non-linear source-to-target mapping functions, giving rise to a TrEO+ M_S algorithmic instantiation for continuous optimization. In this domain, we considered case studies spanning synthetic benchmark functions as well as a real-world example on multi-location inventory planning.

The noteworthy feature of the resultant TrEO+ M_S algorithmic instantiations was their ability to activate positive inter-task transfers while simultaneously curbing the occurrence of negative transfers. In particular, the solution representation learning module (via source-to-target mappings M_S) embedded in our algorithms had learned well-aligned solution representations that uncovered useful but hidden source-target information; this resulted in increased positive transfers. Furthermore, the probabilistic model-based source-target similarity capture mechanism had *synergistically* acted as a safety net which sieved out negative inter-task interactions (possibly due to inaccurately learned solution representations). This had been affirmed in our

experimental results for all the case studies, in which our algorithms showed *significant performance boosts even relative to existing state-of-the-art TrEO algorithms*.

In addition to optimization tasks with just a single objective function as explored in this chapter, we note the growing interest of TrEO researchers and practitioners in the context of *multi-objective optimization* – where problems are defined by multiple (i.e., 2 or 3) objective functions. Therefore, the next chapter puts forward a novel multi-objective probabilistic model-based TrEO algorithm with source-to-target map (labelled as the MOTrEO+M_S), and experimentally validates distinct instantiations of the proposed algorithm in the domains of continuous optimization and a class of graph-based combinatorial problems, respectively.

Chapter 5

Multi-Objective TrEO with Source-to-Target Map: MOTrEO+ M_S

In accordance with the novel algorithmic framework (presented in Section 3.4), Chapter 4 proposed the probabilistic model-based TrEO algorithm with source-to-target map (TrEO+ M_S). Promising results with enhanced convergence rate and final solution quality were attained in the domains of discrete and continuous optimization, for which case studies spanning benchmark problems as well as practical examples were considered. We note that the TrEO+ M_S was developed in the context of single-objective optimization for tackling tasks with scalar-valued objective functions only. However, in many cases, practitioners often need to consider multiple objective functions in the search and optimization process, especially when solving real-world problems. At this juncture, it is imperative for us to showcase the applicability of our framework in the context of *multi-objective optimization*³.

This chapter is organized as follows. Section 5.1 first presents the fundamental concepts of multi-objective optimization. Based on our framework, Section 5.2 motivates and outlines the proposed *multi-objective probabilistic model-based TrEO algorithm with source-to-target map (MOTrEO+ M_S)*. On one hand, Section 5.3 examines the efficacy of the proposed MOTrEO+ M_S in the domain of multi-objective continuous optimization, considering complex benchmark tasks and a real-world example on vehicle crashworthiness design. On another hand, Section 5.4 investigates the effectiveness of the MOTrEO+ M_S in a class of graph-based combinatorial optimization, looking at a practical case study in the multi-objective route planning of unmanned aerial vehicles. Section 5.5 then summarizes the chapter.

³ The work in this chapter has been published in [14].

5.1 Multi-Objective Optimization

With reference to Chapter 3, the mathematical statement of Eq. (3.2) gives an optimization formulation with a scalar-valued objective function (i.e., $f(\mathbf{x})$). However, many real-world problems often involve multiple objectives of interest (that are possibly conflicting), hence giving rise to the field of *multi-objective optimization* [102]. The optimization of a target task $\mathcal{T}_T$ with m objectives can be formulated as follows [103]:

$$\max_{\mathbf{x}_T} \mathbf{F}_T(\mathbf{x}_T) = (f_T^1(\mathbf{x}_T), f_T^2(\mathbf{x}_T), \dots, f_T^m(\mathbf{x}_T)), \quad (5.1)$$

where $\mathbf{x}_T = (x_{T,1}, \dots, x_{T,d_T})$ is a d_T -dimensional solution vector in search space $\mathcal{X}_T$ and $\mathbf{F}_T$ is a vector of m target objective functions $f_T^i(\mathbf{x}_T), i \in \{1, \dots, m\}$. For the sake of brevity, optional constraint functions are intentionally omitted in Eq. (5.1).

Conflicting objectives in a *multi-objective problem (MOP)* can generally not be concurrently optimized, thus introducing trade-offs and the concept of *Pareto dominance* [104]. This concept implies that a solution $\mathbf{x}_{T,a}$ is preferred over another solution $\mathbf{x}_{T,b}$ (or in other words, $\mathbf{x}_{T,a}$ dominates $\mathbf{x}_{T,b}$) in a multi-objective setting, i.e.,

$$\mathbf{x}_{T,a} \prec \mathbf{x}_{T,b},$$

iff $\forall i \in \{1, \dots, m\}: f_T^i(\mathbf{x}_{T,a}) \geq f_T^i(\mathbf{x}_{T,b})$, and $\exists j \in \{1, \dots, m\}: f_T^j(\mathbf{x}_{T,a}) > f_T^j(\mathbf{x}_{T,b})$.

Consequently, a solution $\mathbf{x}_T^*$ is considered to be *Pareto optimal* if it is not dominated by any other solutions in $\mathcal{X}_T$. For this reason, multi-objective optimizers typically solve for a *set* of alternative solutions that constitute the *Pareto-optimal set*, instead of returning just a single optimal solution [105] – see Section 2.3.3 for a background on how *traditional multi-objective evolutionary algorithms (MOEAs)* gained popularity for solving MOPs, as well as their shortcomings which ushered in the prospective *multi-objective TrEO*.

5.2 The MOTrEO+ M_S Algorithm

5.2.1 Motivations

Given the promising results that have been obtained in single-objective TrEO, there is now growing research attention placed on multi-objective TrEO tasks as well. To this end, several TrEO algorithms have recently been applied to boost multi-objective optimization performance through leveraging knowledge from related problems [10, 52, 57, 66, 78, 81, 82]; see also Section 2.3.3. We note that despite the potential for performance speed-up, the effectiveness of multi-objective TrEO algorithms in actual practice could also be hampered by the *occurrence of harmful negative transfers* or the *scarcity of beneficial positive transfers* – particularly *due to the presence of arbitrarily heterogeneous source and target dimensionalities, or the lack of overlap between their respective optimized search distributions, or both* (manifesting in unaligned solution representations).

This unveils the importance of (i) solution representation learning for inducing greater alignment (of solution representations) and hence positive transfers between distinct MOPs that bear discrepancies in their original search spaces (resulting in source-target domain mismatch $\mathcal{D}_S \neq \mathcal{D}_T$ scenarios; see Section 3.2), and (ii) capturing source-target similarities for reducing negative inter-task transfers. Besides that, existing multi-objective TrEO works have *solely* focused either on (a) *methods to increase positive transfers* (e.g., the multi-objective variant of TrEO search with learning across heterogeneous problems [57]), or on (b) *techniques to curb detrimental negative transfers from unrelated sources* (e.g., the AMTEA for solving MOPs [10]). To the best of our knowledge, *there is little previous work in the TrEO literature on synergizing these dual effects in the context of multi-objective optimization.*

It is with the aim of filling the above research gap and achieving the desired performance boost that we propose the *multi-objective probabilistic model-based TrEO algorithm with source-to-target map*, labelled as the MOTrEO+ M_S , which emphasizes the joint effect of activating positive transfers while simultaneously guarding against negative transfers when solving MOPs. In particular, the MOTrEO+ M_S learns well-aligned solution representations via *source-to-target spatial*

mappings M_S (within the embedded solution representation learning module; see Section 3.4.1) with the goal of inducing increased positive correlations across performance indicators in the multi-dimensional objective spaces of distinct optimization tasks. The proposed algorithm is therefore envisioned to be capable of handling (i) *arbitrarily heterogeneous source and target dimensionalities*, and (ii) *a lack of overlap between optimized source and target search distributions*; see Section 3.2.

5.2.2 Algorithm Summary

The proposed MOTrEO+ M_S is detailed in Algorithm 5.1, according to the novel framework established in Section 3.4. Our algorithm assumes (without loss of generality) the popular *NSGA-II* [72] to be the base MOEA. Hence, the concepts of *non-dominated sorting* and *crowding distances* to evaluate the relative fitness of candidate solutions in multi-dimensional objective spaces are adopted herein. It is also worth noting that the proposed MOTrEO+ M_S accounts for *multiple sources*, which is again an extension of the exposition in Section 3.3 and Section 3.4 which considered only a single source.

The inputs to Algorithm 5.1 include m target objective functions $\mathbf{F}_T(\mathbf{x}_T) = (f_T^1(\mathbf{x}_T), f_T^2(\mathbf{x}_T), \dots, f_T^m(\mathbf{x}_T))$, a database of knowledge priors (data and optimized probabilistic models) drawn from $T - 1$ source tasks $\mathcal{T}_S \forall S \in \{1, \dots, T - 1\}$, the search space $\mathcal{X}_T$ of target task $\mathcal{T}_T$, and the transfer interval Δ . Given the inputs, the MOTrEO+ M_S proceeds as follows. At the start of optimization, a target population $P_T(t = 0)$ of size N is initialized (lines 1–2). The fitness of each solution $\mathbf{x}_T \in \mathcal{X}_T$ is evaluated with respect to the target objectives $\mathbf{F}_T$ (line 3). Based on their objective values, the relative fitness of candidate solutions is ascertained via non-dominated sorting and crowding distances across all subsequent iterations of the algorithm. During *non-transfer* iterations, standard crossover and mutation operators are applied on the parent populations $P_T(t)$ to generate the offspring population $C_T(t)$ (lines 5–6).

When the *sequential transfer* iterations are triggered by the user-defined parameter Δ , the source-to-target mapping M_S is first constructed using both the knowledge from $\mathcal{T}_T$ and the information extracted from the source database (lines 7–

Algorithm 5.1 Pseudocode of the proposed MOTrEO+ M_S algorithm**Input:** Source database; $\mathbf{F}_T$; $\mathcal{X}_T$; Δ **Output:** Pareto set and optimized search distribution for $\mathcal{T}_T$

1. Set $t = 0$
2. Initialize a parent population $P_T(t)$ of N solutions in $\mathcal{X}_T$
3. Evaluate each $\mathbf{x}_T$ in $P_T(t)$ w.r.t. $\mathbf{F}_T$
4. **While** stopping condition not met **do**
5. **If** $\text{mod}(t, \Delta) \neq 0$ **then**
6. Generate offspring population $C_T(t)$ from $P_T(t)$ using crossover and/or mutation operators
7. **Else**
8. Construct $M_S: \mathcal{X}_S \rightarrow \mathcal{X}_T \forall S \in \{1, 2, \dots, T-1\}$, using knowledge from the source database and $\mathcal{T}_T$
9. Learn new solution representations for $\mathcal{T}_S$ using $M_S \forall S \in \{1, \dots, T-1\}$
10. Transform each $p_S^*(\mathbf{x}_S) \forall S \in \{1, 2, \dots, T-1\}$ using the corresponding $M_S \rightarrow$ refer to Algorithm 3.1
11. Obtain $w_1, w_2, \dots, w_T$ by solving Eq. (3.12) or Eq. (3.13), then build the target mixture model $[\sum_{S=1}^{T-1} w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T|t)]$ using $p_{M_S}^*(\mathbf{x}_T) \forall S \in \{1, 2, \dots, T-1\}$ and $p_T(\mathbf{x}_T|t)$
12. Sample the mixture model to obtain offspring $C_T(t)$
13. **End If**
14. Evaluate each $\mathbf{x}_T$ in $C_T(t)$ w.r.t. $\mathbf{F}_T$
15. Elitist selection of top N solutions from $P_T(t) \cup C_T(t)$ based on non-dominated sorting and crowding distances, to form $P_T(t+1)$
16. Set $t = t + 1$
17. **End While**

8). The M_S is then applied to learn new solution representations for each source task, and the transformed source models $p_{M_S}^*(\mathbf{x}_T)$ are obtained subsequently (line 9–10); see the procedure in Algorithm 3.1 for achieving this via simple Monte Carlo approximation. Accordingly, the target mixture model comprising each $p_{M_S}^*(\mathbf{x}_T) \forall S \in$

$\{1, 2, \dots, T-1\}$ and the current target model $p_T(\mathbf{x}_T|t)$ is built, followed by the sampling of the mixture model to generate $C_T(t)$ (lines 11–13). The new parent population $P_T(t+1)$ for the next iteration is selected from the combined populations $P_T(t) \cup C_T(t)$, based on an elitist selection strategy (lines 14–16). The entire evolutionary search process repeats until the stopping condition is satisfied.

We note that the probabilistic model-based source-target similarity capture mechanism (executed in lines 11–12) of the MOTrEO+ M_S flexibly allows various models (representing search distributions in distinct optimization domains, e.g., continuous and combinatorial optimization) to be incorporated for the purpose of sieving out the threat of negative transfers. The sieving mechanism even serves as a safeguard in cases of inaccurately learned representations, which could be caused by the lack of sufficient target search data required for the learning process. Furthermore, other existing MOEAs can also be implemented into the MOTrEO+ M_S as the base multi-objective optimizer, by simply modifying the evolutionary operators and selection steps. It is also worth noting that Algorithm 5.1 permits both the offline construction as well as the online learning of the mapping M_S in a similar manner as Algorithm 4.1; see Figure 4.1.

In the subsequent sections, we present case studies in the domain of continuous optimization and a class of graph-based combinatorial problems, along with their respective MOTrEO+ M_S algorithmic instantiations based on novel methodologies for constructing the source-to-target spatial mappings M_S . Computational experiments are also conducted to showcase the efficacy of our methods.

5.3 Case Studies in Multi-Objective Continuous Optimization

This section considers case studies in the domain of multi-objective continuous optimization, where the source and target solution representation spaces are real-valued (i.e., $\mathcal{X}_S \subseteq \mathbb{R}^{d_S}$ and $\mathcal{X}_T \subseteq \mathbb{R}^{d_T}$). We first propose a methodology for learning well-aligned solution representations in multi-objective continuous spaces. Using this method and a multi-variate Gaussian distribution model, we then obtain the resultant

MOTrEO+ M_S algorithmic instantiation for multi-objective TrEO in continuous spaces. To verify the effectiveness of our algorithm, we conduct experimental case studies spanning complex benchmark tasks as well as a real-world example on multi-objective vehicle crashworthiness design.

5.3.1 Methodology for Learning M_S

Let us consider a multi-objective TrEO setting in which knowledge priors from a source task $\mathcal{T}_S$ (of dimensionality d_S) are first transformed via a source-to-target mapping M_S , then transferred to a target task $\mathcal{T}_T$ (of dimensionality d_T). We recall that several recent methods in the TrEO literature have proposed linear mapping functions in their attempt to increase positive transfers between tasks in continuous search spaces [57, 94]; see Section 2.3.2. However, we have discussed in Section 3.3.1 that such linear mappings only work well when $d_T \leq d_S$, whereas they result in degenerate source model(s) $p_{M_S}^*(\mathbf{x}_T)$ when $d_T > d_S$ – as $p_{M_S}^*(\mathbf{x}_T)$ will be supported only on a lower d_S -dimensional subspace. As a consequence of degenerate priors when linear mappings are used in cases of $d_T > d_S$, the target mixture model in Algorithm 5.1 can no longer be built.

To address the aforementioned issue, as highlighted in Section 5.2, the presented MOTrEO+ M_S algorithm is envisioned to be capable of handling either (i) arbitrarily heterogeneous source and target dimensionalities (i.e., $d_T \leq d_S$ and $d_T > d_S$), or (ii) a lack of overlap between their optimized search distributions (i.e., $p_S^*(\mathbf{x}_S) \not\approx p_T^*(\mathbf{x}_T)$), or both. Accordingly, we propose a methodology for the *fast* learning of *non-linear source-to-target spatial mappings*, which we denote hereafter as $M_S: \mathcal{X}_S \rightarrow \mathcal{X}_T$, that aim to cope with heterogeneous scenarios of multi-objective TrEO where $d_T \neq d_S$ without leading to degenerate distributions, as well as cases of $p_S^*(\mathbf{x}_S) \not\approx p_T^*(\mathbf{x}_T)$. Moreover, returning to Eq. (3.6), we see that the proposed category of solution representation learning in TrEO (i.e., source-to-target maps) does not require the construction of an inverse mapping function. Thus, the learning complexity of source-to-target mappings is likely to be lower when compared to other possible forms of solution representation learning in TrEO; see Section 6.2 for more information.

Our proposed non-linear spatial mapping function M_S takes the form of a *two-*

layer feedforward neural network as follows,

$$M_S(\mathbf{x}_S; \boldsymbol{\theta}) = \boldsymbol{\psi}_2(\boldsymbol{\psi}_1(\mathbf{x}_S)), \quad (5.2)$$

where $\boldsymbol{\theta}$ represents the neural network parameters, while $\boldsymbol{\psi}_1$ and $\boldsymbol{\psi}_2$ are the first and second layer transformations, respectively.

FIGURE 5.1. (a) The preference relationship between solutions in a multi-objective setting (assuming objective function maximization). (b) Source and target populations monotonically aligned according to decreasing order of preference.

In our method, the mapping M_S is learned using the target and source population data at designated knowledge transfer iterations. Taking motivation from [57], we assume that the source data $P_S(t) = \{\mathbf{x}_{S,1}, \dots, \mathbf{x}_{S,N}\}$ of population size N had been archived at every iteration t of an MOEA, when solving the source task $\mathcal{T}_S$ of dimensionality d_S . Different from the methodology proposed in Section 4.3.1, here we adopt the preference relationship between solutions based on the concepts of *nondominated front (NF)* and *crowding distance (CD)* [72] that are lexicographically considered in a multi-objective optimization setting; given the common knowledge of these terms in the associated literature, we refrain from providing a detailed discussion about them herein for the sake of brevity. Accordingly, a solution $\mathbf{x}_1$ is preferred over another solution $\mathbf{x}_2$ if any one of the following conditions holds true [79]: (i) $NF_1 < NF_2$ or (ii) $NF_1 = NF_2$ and $CD_1 > CD_2$; see Figure 5.1(a) for a pictorial depiction.

This preference relationship shall be assumed hereafter for evaluating and ranking solutions in $P_S(t)$ as well as in the target population (as was also done during the evolutionary search process in Algorithm 5.1).

During the optimization of a target task $\mathcal{T}_T$ of dimensionality d_T , let the target population data at iteration t be $P_T(t) = \{\mathbf{x}_{T,1}, \dots, \mathbf{x}_{T,N}\}$. A necessary step for learning a mapping M_S between $P_S(t)$ and $P_T(t)$ is to first *sort* the solutions in both $P_S(t)$ and $P_T(t)$ according to the *preference relationship* defined above. This induces a *monotonic alignment between the source and target population sets*, as shown in Figure 5.1(b). Preferred solutions in the source data are matched with those in the target data, and vice versa, thereby facilitating the discovery of source-to-target mappings that tend to induce a high ordinal correlation (on the preference relationship scale) across tasks in the transformed solution representation space. Assuming the Pareto sets of the source and target tasks to be located in the regime (i.e., not far from the underlying distribution) of $P_S(t)$ and $P_T(t)$, respectively, transforming Pareto optimal solutions of the source with M_S could then help generate high quality solutions to the target as well.

Given the motivation and data alignment procedure above, the M_S in Eq. (5.2) is learned through the following two-layer transformations:

$$\boldsymbol{\psi}_1(\mathbf{x}_S) = \sigma(\mathbf{A}_1 \cdot \mathbf{x}_S + \mathbf{b}), \quad (5.3)$$

and,

$$M_S = \boldsymbol{\psi}_2(\boldsymbol{\psi}_1(\mathbf{x}_S)) = \mathbf{A}_2 \cdot \boldsymbol{\psi}_1(\mathbf{x}_S), \quad (5.4)$$

where matrices $\mathbf{A}_1 \in \mathbb{R}^{d_h \times d_S}$ and $\mathbf{A}_2 \in \mathbb{R}^{d_T \times d_h}$ contain the neural network's weighting terms, $\mathbf{b}$ is a bias vector, and σ is the non-linear sigmoidal activation function; d_h represents the number of neurons in the hidden layer.

In the spirit of [95] supporting fast learning, we randomly assign and then fix the elements of $\mathbf{A}_1$ in Eq. (5.3). Thereafter, the elements of $\mathbf{A}_2$ in Eq. (5.4) are obtained using the following *closed-form expression*:

$$\mathbf{A}_2 = P_T \cdot [\boldsymbol{\psi}_1(P_S)]^{\text{tr}} \cdot (\boldsymbol{\psi}_1(P_S) \cdot [\boldsymbol{\psi}_1(P_S)]^{\text{tr}} + \lambda \cdot I_{d_h})^{-1}, \quad (5.5)$$

where $[\cdot]^{\text{tr}}$ symbolizes a matrix transpose, I_{d_h} is an identity matrix of size d_h , and λ is the L_2 regularization coefficient of network parameters. Figure 5.2 gives an illustration of the solution representation learning process via our proposed M_S .

FIGURE 5.2. An illustration of the proposed methodology for learning well-aligned solution representations in multi-objective continuous search spaces, using a two-layer feedforward neural network.

In line with Algorithm 5.1, the learned M_S is applied to obtain the transformed source model $p_{M_S}^*(\mathbf{x}_T)$ via simple Monte Carlo approximation. The resultant MOTrEO+ M_S algorithmic instantiation for multi-objective TrEO in continuous spaces is attained by actualizing $p_{M_S}^*(\mathbf{x}_T) \forall S \in \{1, 2, \dots, T-1\}$ and the current target model $p_T(\mathbf{x}_T|t)$ as *multi-variate Gaussian distributions*. The target and transformed source models are then used in the stacked density estimation procedure [51] to build the target mixture model $[\sum_{S=1}^{T-1} w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T|t)]$ in the target search space $\mathcal{X}_T$. In what follows, the performance advantage of our algorithm is demonstrated through experimental case studies.

5.3.2 A Study of Complex Benchmark Tasks

Here we first showcase the effectiveness of the MOTrEO+ M_S using complex multi-objective benchmark problems characterized by *varying degree of discrepancies* between the source and target tasks [106].

5.3.2.1 Experimental Specifications

Table 5.1 shows the problem configuration (leftmost column) of nine multi-objective TrEO tasks considered in this study, and their properties. The source and target dimensionalities are set to $d_S = 10$ and $d_T = 20$, respectively. The four-letter problem configuration represents the known degree of discrepancies between the source and target tasks (i.e., $\mathcal{T}_S$ and $\mathcal{T}_T$, respectively). Each configuration comprises the *extent of intersection (overlap) between the Pareto sets* of $\mathcal{T}_S$ and $\mathcal{T}_T$, as well as a *degree of similarity between the fitness landscapes* of $\mathcal{T}_S$ and $\mathcal{T}_T$ (quantified as γ). For instance, we have: (i) CIHS: *complete* intersection with *high* similarity; (ii) PIMS: *partial* intersection with *medium* similarity; and (iii) NILS: *no* intersection with *low* similarity. Readers are referred to [106] for the full details of these benchmark problems. We expect that a smaller extent of intersection between source-target Pareto optimal solutions (i.e., “no” < “partial” < “complete” intersection), and a lower degree of similarity in the fitness landscapes (i.e., “low” < “medium” < “high” similarity) will increase the scope for solution representation learning in the MOTrEO+ M_S to unveil useful but hidden inter-task relationships.

Four optimizers are considered for comparison in this case study. They are: (i) *the non-dominated sorted genetic algorithm II* (NSGA-II) as the base optimizer without knowledge transfer [72], (ii) NSGA-II+ M_S which is the multi-objective variant of the TrEO search with adaptive learning across heterogeneous problems [57] adapted using the proposed non-linear mapping function M_S from Section 5.3.1, (iii) AMTEA which is the state-of-the-art probabilistic model-based TrEO algorithm without solution representation learning [10], and (iv) our proposed MOTrEO+ M_S . All search populations of size $N = 50$ consist of real-coded solutions. The optimizers employ simulated binary crossover, polynomial mutation (with probability $1/d_T$) and tournament selection. The transfer interval is set to $\Delta = 2$, at which target mixture

TABLE 5.1. Complex multi-objective benchmark tasks and their properties [106].

Problem Configuration	γ	Task	Pareto Set	Properties
CIHS	0.970	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,i} = 0, i = 2:10$	concave, unimodal, separable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $x_{T,j} = 0, j = 2:20$	concave, unimodal, separable
CIMS	0.520	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,i} = 1, i = 2:10$	concave, multimodal, nonseparable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $(x_{T,2}, \dots, x_{T,20})^T = s_{cm}$	concave, unimodal, nonseparable
CILS	0.070	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,i} = 0, i = 2:10$	concave, multimodal, separable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $x_{T,j} = 0, j = 2:20$	convex, multimodal, nonseparable
PIHS	0.990	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,i} = 0, i = 2:10$	convex, unimodal, separable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $(x_{T,2}, \dots, x_{T,20})^T = s_{ph}$	convex, multimodal, separable
PIMS	0.550	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $(x_{S,2}, \dots, x_{S,10})^T = s_{pm}$	concave, unimodal, nonseparable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $x_{T,j} = 0, j = 2:20$	concave, multimodal, nonseparable
PILS	0.002	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,i} = 0, i = 2:10$	concave, multimodal, nonseparable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $(x_{T,2}, \dots, x_{T,20})^T = s_{pl}$	concave, multimodal, nonseparable
NIHS	0.940	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,i} = 0, i = 2:10$	concave, multimodal, nonseparable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $x_{T,j} = 1, j = 2:20$	convex, unimodal, separable
NIMS	0.510	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,2} \in [0,1]$ $x_{S,i} = 1, i = 3:10$	concave, multimodal, nonseparable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $x_{T,2} \in [0,1]$ $x_{T,j} = 0, j = 3:20$	concave, unimodal, nonseparable
NILS	0.001	$\mathcal{J}_S$	$x_{S,1} \in [0,1]$ $x_{S,2} \in [0,1]$ $(x_{S,3}, \dots, x_{S,10})^T = s_{nl}$	concave, multimodal, nonseparable
		$\mathcal{J}_T$	$x_{T,1} \in [0,1]$ $x_{T,j} = 0, j = 3:20$	concave, multimodal, nonseparable

models with multi-variate Gaussian components are built and sampled in the AMTEA and the MOTrEO+ M_S . For the two-layer neural network in the NSGA-II+ M_S and the

MOTrEO+ M_S , we set the number of neurons in the hidden layer to $d_h = 2d_T$. The *inverted generational distance* (IGD) [107] is used as the multi-objective performance indicator to quantify the solution quality returned by all algorithms. A single-run of each algorithm is terminated after 50,000 function evaluations.

5.3.2.2 Results and Discussion

TABLE 5.2. Experimental results for the multi-objective benchmark tasks after 50,000 function evaluations. In bold are the best results (values with 95% confidence level based on the Wilcoxon Signed-Rank Test are marked with “*”). Results are reported based on 20 independent runs of all optimizers.

Problem Configuration	Average IGD Values			
	NSGA-II	NSGA-II + M_S	AMTEA	MOTrEO+ M_S
CIHS	0.27	0.29	0.01	0.01
CIMS	8.01	8.96	4.45	4.57
CILS	12.16	13.52	9.41	8.32
PIHS	3.02	3.84	2.48	2.46
PIMS	243.22	126.34	74.76	26.43*
PILS	20.37	20.14	20.23	1.04*
NIHS	100.02	97.51	87.97	84.44
NIMS	8.78	7.19	7.36	3.33*
NILS	20.30	20.29	20.31	7.11*

Table 5.2 summarizes the average IGD values returned by the four optimizers after 20 independent runs. The MOTrEO+ M_S has outperformed the other algorithms in 8 out of 9 cases, among which it has achieved significant difference in 4 cases. The convergence trends are illustrated in Figure 5.3 for all the cases, while the transfer coefficients patterns are shown in Figure 5.4 for three representative cases, namely, CIHS, PILS and NILS. Looking at CIHS, PIHS and NIHS with high inter-task similarity even in the original solution representation spaces, our expectation is closely borne out by the almost equivalent convergence rates between the AMTEA and the MOTrEO+ M_S , as well as their comparable transfer coefficients obtained; see Figure 5.3(a), (d), (g), and Figure 5.4(a). This observation is explained by the high similarities

between source and target function landscapes, which leaves little to be gained through learning solution representations in the MOTrEO+ M_S .

FIGURE 5.3. Convergence trends for the complex multi-objective benchmark tasks in Table 5.2. Results are for distinct problem configurations: “high similarity” (leftmost column), “medium similarity” (central column), “low similarity” (rightmost column), “complete intersection” (top row), “partial intersection” (middle row), and “no intersection” (bottom row). The shaded regions span one standard deviation on either side of the mean (based on 20 independent runs).

FIGURE 5.4. The transfer coefficients obtained by the AMTEA and the MOTrEO+ M_S in three representative cases. (a) CIHS. (b) PILS. (c) NILS.

In contrast, the MOTrEO+ M_S significantly outperforms the other algorithms in PIMS, PILS, NIMS and NILS. The results are attributed to the increased degree of discrepancies between the source and target tasks in those cases, giving rise to the scope for solution representation learning to unveil hidden source-target relationships. Moreover, the multimodality of the target tasks in PILS and NILS increases their difficulty substantially, which further incentivizes the solution representation learning process in the MOTrEO+ M_S . The performance advantage of our algorithm is clearly demonstrated in its significantly higher convergence rates and transfer coefficients obtained; see Figure 5.3(f), (i) and Figure 5.4(b), (c), respectively.

Positive transfers are empirically identified when an algorithm's convergence rate is boosted. Hence, we can say that the MOTrEO+ M_S 's performance advantage is attributed to the presence of positive transfers, which in turn imply larger transfer coefficients. The theoretical developments in [10] in fact show that the deciphering of transfer coefficients in probability mixture model-based TrEO explicitly reduces negative inter-task interactions, relative to the no-transfer case. Take PILS and NILS as examples. We observe that the MOTrEO+ M_S 's enhanced convergence rates are accompanied by larger transfer coefficients, whereas the AMTEA's slower convergence is accompanied by near-zero transfer coefficients; as shown in Figure 5.3 and Figure 5.4. For NILS in particular, the synergistic effect of activating positive transfers while jointly mitigating the threat of negative transfers is demonstrated in Figure 5.5, where larger transfer coefficients are obtained only when the learned mapping M_S is effective. In a nutshell, observing both (a) an enhanced convergence rate, and (b) larger transfer coefficients obtained, are good indicators of the target

search benefitting from positive transfers from the available sources.

FIGURE 5.5. The transfer coefficients deduced in the MOTrEO+ M_S are large only when the learned mapping is effective in inducing positive transfers. The effectiveness of M_S learned is measured by the average Euclidean distance between the transformed source solutions and their corresponding nearest target Pareto optimal solution. For instance, the effectiveness of M_S is considered to be “high”, “medium” or “low” when the distances are “ < 0.3 ”, “ $0.3 - 0.5$ ” or “ > 0.5 ”, respectively.

To further affirm that the effectiveness of the MOTrEO+ M_S is indeed by virtue of positive transfers induced by the source-to-target mapping M_S (and is not a spurious artifact of introducing diversity through knowledge transfers from external sources), we consider an additional MOEA that transfers the solutions sampled from a *random source model* (RSM), which we label as the MOTrEO(RSM). The RSM takes the form of a *uniform distribution defined over the target search space*, which is solely intended for introducing diversity into the target population. It is worth noting that in the MOTrEO(RSM), the amount of knowledge transfers is determined by the transfer coefficients obtained in the MOTrEO+ M_S , which ensures that both algorithms receive the same extent of solution transfers from external sources. Results show that the convergence trends of the MOTrEO(RSM) are only comparable to those of the NSGA-II (without knowledge transfers) for all the problem configurations, which are consistently worse than those of the MOTrEO+ M_S ; see Figure 5.3. Therefore, the

efficacy of our proposed algorithm in inducing positive transfers across distinct multi-objective continuous problems is verified in the experimental results.

5.3.3 A Real-World Example on Multi-Objective Vehicle Crashworthiness Design

Following from the promising results shown for benchmark MOPs, we now showcase the effectiveness of the proposed MOTrEO+ M_S in a real-world engineering application of vehicle crashworthiness design.

Crashworthiness is an important quality of vehicle engineering designs that aims to reduce human injury in the event of collision impact. The vehicle crashworthiness design optimization is often a computationally expensive problem that involves multiple objectives. In this case study, we consider the minimization of *three* critical objectives, namely, the vehicle mass, the jerk, and the vehicle toeboard intrusion. Readers can refer to [108] for the mathematical model and a detailed description of this design problem.

Here, the multi-objective minimization program can be simply stated as:

$$\begin{aligned} \min \quad & F(\mathbf{x}) = (f_{mass}(\mathbf{x}), f_{jerk}(\mathbf{x}), f_{intrusion}(\mathbf{x})), \\ \text{s. t.}, \quad & \mathbf{x} = (x_1, \dots, x_d)^T \text{ and } 1mm \leq x_j \leq 3mm, \forall j, \end{aligned} \quad (5.6)$$

where $f_{mass}(\mathbf{x})$, $f_{jerk}(\mathbf{x})$ and $f_{intrusion}(\mathbf{x})$ are the objective functions for vehicle mass, jerk and toeboard intrusion, respectively, $\mathbf{x}$ is a candidate solution vector, d is the number of design variables (i.e., dimensionality) and x_j refers to the j^{th} design variable.

In real-world settings, it is possible that the source and target tasks bear arbitrary discrepancies in their original search spaces, causing source-target domain mismatch (see Section 3.2). For instance, (a) having different number of design variables (i.e., $d_S \neq d_T$), and (b) lacking one-to-one alignment between source and target design variables – we refer to this scenario as $\mu_S \neq \mu_T$; take for example, $\mathbf{x}_S = (x_3, x_1, x_2)^T$

and $\mathbf{x}_T = (x_1, x_2, x_3, x_4, x_5)^T$. Such misalignments generally reduce the overlap of optimized search distributions between the source and target problems, which could lead to reduced positive transfers or even increased negative transfers.

Despite the presence of the aforementioned discrepancies, we contend that our proposed algorithm, the MOTrEO+ M_S , by virtue of automatically learning well-aligned solution representations, is able to uncover useful inter-task relationships and induce positive transfers. To verify this, we conduct experiments using distinct cases of the problem in Eq. (5.6) characterized by arbitrary source-target discrepancies as described next.

5.3.3.1 Experimental Setup

We consider three *multi-source transfer* scenarios, each involving two source tasks (i.e., $\mathcal{T}_{S1}$ and $\mathcal{T}_{S2}$) as follows: (i) *Case 1*: $d_T = d_{S1}, d_T = d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$, (ii) *Case 2*: $d_T > d_{S1}, d_T > d_{S2}$ and $\mu_T = \mu_{S1}, \mu_T \neq \mu_{S2}$, and (iii) *Case 3*: $d_T = d_{S1}, d_T > d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$. The same four optimizers as Section 5.3.2.1 are considered for comparison. The population size in this case study is set to $N = 15$ [108]. A single run of each algorithm is terminated after 300 function evaluations. The average IGD values are computed based on 20 independent runs, using the Pareto front which is approximated beforehand with 20,000 function evaluations of all the four optimization algorithms under comparison.

5.3.3.2 Results and Discussion

Table 5.3 gives a summary of the experimental results that are averaged over 20 independent runs for all the three cases. As seen in Table 5.3, the MOTrEO+ M_S consistently performs better than the other algorithms in terms of final solution quality attained. Equipped with both the solution representation learning module as well as the source-target similarity capture mechanism, our algorithm achieves performance enhancement through a favourable balance between the activation of positive transfers and the concurrent reduction of negative transfers. In contrast, without any source

induced search bias, the base (non-TrEO) NSGA-II is consistently outperformed by the other optimizers.

TABLE 5.3. Experimental results for the multi-objective vehicle crashworthiness design tasks after 300 function evaluations. In bold are the best results (values with 95% confidence level based on the Wilcoxon Signed-Rank Test are marked with “*”). Results are reported based on 20 independent runs of all optimizers.

Problem Configuration	Average IGD Values			
	NSGA-II	NSGA-II + M_S	AMTEA	MOTrEO+ M_S
$d_T = d_{S1}, d_T = d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$	0.415	0.403	0.411	0.373*
$d_T > d_{S1}, d_T > d_{S2}$ and $\mu_T = \mu_{S1}, \mu_T \neq \mu_{S2}$	0.415	0.377	0.386	0.361
$d_T = d_{S1}, d_T > d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$	0.415	0.400	0.401	0.359*

Comparing the AMTEA and the MOTrEO+ M_S in Table 5.3, relative to the NSGA-II, the AMTEA shows smaller improvements in solution quality. The presence of unaligned source-target solution representations in each case substantiates the scope for the designed source-to-target spatial mapping methodology to unveil hidden inter-task relationships. The superiority of our algorithm is manifested in its higher convergence rates and larger transfer coefficients obtained; see Figure 5.6(a), (b), (c) and Figure 5.6(d), (e), (f), respectively. What is more, despite executing the same solution representation learning module, the MOTrEO+ M_S outperforms the NSGA-II+ M_S in all cases. The NSGA-II+ M_S is not equipped with the probabilistic model-based inter-task similarity capture mechanism, thereby unable to adaptively control the extent of knowledge transfers from the various sources to the target task.

The numerical results of this case study confirm the efficacy of our proposed MOTrEO+ M_S in inducing positive transfers between distinct real-world MOPs defined

in continuous search spaces. However, we note that other than continuous optimization problems, there exists a plethora of real-world applications of MOPs in domains of discrete and combinatorial optimization. This brings us to the next section where another novel algorithmic instantiation of the MOTrEO+ M_S is put forward for a class of vehicle route planning combinatorial tasks.

FIGURE 5.6. Convergence trends for the target multi-objective vehicle crashworthiness design optimization tasks and the transfer coefficients obtained by the AMTEA and the MOTrEO+ M_S in *Case 1*: $d_T = d_{S1}, d_T = d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$ (left column), *Case 2*: $d_T > d_{S1}, d_T > d_{S2}$ and $\mu_T = \mu_{S1}, \mu_T \neq \mu_{S2}$ (central column), and *Case 3*: $d_T = d_{S1}, d_T > d_{S2}$ and $\mu_T \neq \mu_{S1}, \mu_T \neq \mu_{S2}$ (rightmost column).

5.4 A Practical Case Study in Multi-Objective Vehicle Route Planning

This section considers a practical case study in which the multi-objective route planning of unmanned aerial vehicles (UAVs) benefit from past experiences of

optimizing the routes of ground-based vehicles. These graph-based tasks belong to a class of combinatorial optimization problems referred to as *vehicle routing problems* (VRPs), which are typically characterized by *permutation-based combinatorial solution representation spaces*. Accordingly, we highlight the importance of solution representation alignment in the effectiveness of knowledge transfers between distinct VRPs. To this end, we propose a novel methodology to learn well-aligned solution representations via source-to-target spatial transformations in the corresponding combinatorial search spaces, with the aim of unveiling beneficial inter-task graph structural similarities. Using this method and a permutation-based probabilistic model, we then obtain the resultant MOTrEO+ M_S algorithmic instantiation for multi-objective TrEO in those combinatorial spaces. Computational experiments are conducted to verify that our method can effectively utilize knowledge priors from previously optimized ground-based VRPs, to speed up the multi-objective optimization of UAVs route planning.

5.4.1 Problem Formulations

Recent years have seen the growing use of UAVs (e.g., drones) as an alternative to ground vehicles (e.g., trucks) for last-mile delivery, as drones can be deployed more swiftly at lower operating costs. *Drone delivery* route planning commonly minimizes multiple objectives such as the total distance travelled and the number of drones needed to satisfy customer demands [109]. For these reasons, it is important to rapidly optimize multi-objective drone delivery problems under essential operational requirements. Taking motivation from the above, our case study aims to boost the multi-objective optimization efficiency of drone delivery routing tasks, by exploiting knowledge priors from related *truck delivery* VRPs.

We note that it is common practice to solve UAV route planning tasks as 2-D variants of ground-based VRPs [109]. This is due to similar problem characteristics involved, such as minimizing the total route distance under the constraints of capacity and travel distance limits. In view of the aforesaid, we choose the well-studied truck delivery route planning problems as the *source* tasks in this study. Our choice is further motivated by the abundance of past problem-solving experiences accumulated

in the domain of truck delivery planning [110]. The following formulates the source and target tasks for our case study.

5.4.1.1 Source Task: Truck Delivery

We define the source task on an undirected graph $G = (V, E)$, where $V = \{v_i\}$ is the set of vertices (i.e., customer nodes) and $E = \{e_{ij}\}$ is the set of edges connecting vertices v_i and v_j ($i \neq j$), for $i, j \in \{1, \dots, d_S\}$, while d_S is the number of customers (i.e., source dimensionality). The single depot is represented by vertex v_0 which stations K number of trucks. The vehicles are allowed to do multiple trips that start and end at the depot, and they are constrained by delivery capacity Q and maximum travel distance limit L per trip. Each customer i has a known deterministic demand $c_{S,i}$ and a distance $z_{S,i}$ from the depot. They are served by vehicle k in route r , where $k \in \{1, \dots, K\}$ and $r \in \{1, \dots, R\}$. Let $D_{k,r}$ be the distance travelled and $W_{k,r}$ be the weight capacity served by vehicle k in route r . The objective and constraint functions are then formulated as follows:

$$\begin{aligned} \min f_S &= \sum_{r=1}^R \sum_{k=1}^K D_{k,r}, \\ \text{s.t.}, D_{k,r} &\leq L \text{ and } W_{k,r} \leq Q, \forall k, \forall r, \end{aligned} \quad (5.7)$$

where f_S is the source objective function to minimize the total distance travelled by every truck after serving all customer demands.

5.4.1.2 Target Task: Multi-Objective Drone Delivery

We define the target task on an undirected graph $H = (U, A)$, where $U = \{u_i\}$ is the set of vertices and $A = \{a_{ij}\}$ is the set of edges connecting vertices u_i and u_j ($i \neq j$), for $i, j \in \{1, \dots, d_T\}$, such that d_T is the number of customer nodes (i.e., target dimensionality). The single depot is represented by vertex u_0 , at which each drone starts and ends a route b . All the drones have a delivery capacity q , a maximum travel

distance l per route and a maximum flight range from the depot h . Any customer j has a known deterministic demand $c_{T,j}$ and a distance $z_{T,j}$ from the depot. They can only be served once by a drone y in route b , where $y, b \in \{1, \dots, B\}$. We assume that the drones are not allowed to do multiple trips due to their limited power capacity that requires time-consuming battery recharging. As such, a total of B number of drones is needed to complete B number of routes. Further, let $D_{y,b}$ be the distance travelled and $W_{y,b}$ be the payload served by drone y in route b . We formulate the target objective and constraint functions as follows:

$$\begin{aligned} \min_{\mathbf{x}_T} \quad & \mathbf{F}(\mathbf{x}_T) = (f_T^1(\mathbf{x}_T), f_T^2(\mathbf{x}_T)), \\ \text{s.t.}, \quad & f_T^1 = B, \quad f_T^2 = \sum_{y,b}^B D_{y,b}, \end{aligned} \quad (5.8)$$

$$q \ll Q, l \ll L, W_{y,b} \leq q, D_{y,b} \leq l \text{ and } z_{T,j} \leq h, \forall y, \forall b, \forall j,$$

where $\mathbf{x}_T$ is a target candidate solution, f_T^1 and f_T^2 are the target objective functions to minimize the number of drones and the total flight distance, respectively. Eq. (5.8) accounts for the essential operating constraints of real commercial drones including smaller payload (i.e., $q \ll Q$), shorter flight distance per route (i.e., $l \ll L$) and limited flight range from the depot (i.e., $z_{T,j} \leq h$) [111].

5.4.2 Methodology for Constructing M_S

One of the most widely used solution representations in VRPs is the *permutation* of customer node indices to determine the order of visiting the customers. *This permutation-based solution representation is dependent on the index number assigned to each customer node, as well as the graph-based geometric distribution of nodes.* Furthermore, we note that different real-world vehicle routing tasks are likely to have different customer node indices and geometric distributions, *despite possibly sharing structurally similar optimal routes or sub-routes.* This is likely to give rise to unaligned solution representations between distinct routing tasks, which could in turn

reduce the effectiveness of TrEO algorithms. While the work in [112] has shown promising preliminary findings of transferring knowledge between graph-based routing tasks that share structural overlaps, there is a lack of a principled methodology for overcoming the abovementioned scenarios.

FIGURE 5.7. The importance of solution representation alignment in the effectiveness of knowledge transfers between two distinct vehicle routing tasks, featuring the overlap of their optimal sub-routes. (a) The known optimal routes of source graph G . (b) The (*a priori* unknown) optimal routes of target graph H .

To illustrate, consider the optimal routes of distinct source and target vehicle routing tasks as seen in Figure 5.7. The routes, when encoded as permutation-based solution representations, could appear as $\mathbf{x}_S = [0, 1, 2, 3, 4, 0, 5, 6, 7, 8, 0, 9, 10, 11, 12, 0]$ and $\mathbf{x}_T = [0, 1, 5, 0, 4, 6, 8, 3, 0, 9, 2, 7, 0]$ for the source and target VRPs, respectively [113]. Figure 5.7 entails some structural overlaps of source-target optimal sub-routes, which can be identified by the coloured customer nodes (for illustration purpose). Despite the existence of such overlaps, the source and target solution representations appear to be completely different because of the arbitrary indices assigned to their customer nodes. This suggests that directly sampling and transferring solutions from the source task in Figure 5.7(a) to the target instance in Figure 5.7(b) is unlikely to result in positive transfers, or it may even cause negative transfers. Following from the above, it is clear that *greater alignment of solution representations* is crucial to the success of knowledge transfers between distinct VRPs.

With that in mind, we propose a novel methodology to learn well-aligned solution representations in such permutation-based combinatorial spaces, with the aim of inducing positive transfers between distinct vehicle routing tasks. Specifically, our method learns *new source solution representations* via search space transformations as follows.

Step 1: A *source-to-target spatial mapping*, denoted as M_S , is first constructed to *match* as closely as possible, the source and target problems' graph-based distributions of customer nodes. Under the mapping function M_S , the source graph is first scaled to a geometric size that is comparable with the target. The source is then translated and rotated to align the centroid and principal components, respectively, of its customer distribution with those of the target graph. The overall source-to-target search space transformation through M_S is mathematically defined below,

$$M_S(\mathbf{C}_{S,i}) = \alpha \mathbf{A} \cdot \mathbf{C}_{S,i} + \mathbf{b}, \forall i, \\ \text{s.t.}, \alpha = \frac{\sum_{j=1}^{d_T} z_{T,j} / d_T}{\sum_{i=1}^{d_S} z_{S,i} / d_S}, \mathbf{A} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix}, \quad (5.9)$$

where $\mathbf{C}_{S,i} \in \mathbb{R}^{2 \times 1}$ is the Cartesian coordinates vector of a source node $i \in \{1, \dots, d_S\}$, $\alpha \in \mathbb{R}^+$ is a *scaling* constant, $\mathbf{A}$ is an *angular rotation* matrix, θ is the angle between the principal components of the source and target graphs, and $\mathbf{b} \in \mathbb{R}^{2 \times 1}$ is a *translation* vector.

Step 2: *Reassign* the index numbers of all the source nodes by copying the indices from the corresponding nearest target nodes. If the source has more nodes than the target (i.e., $d_S > d_T$), randomly assign a number $i \in \{1, \dots, d_S\}$ without replacement to the remaining $(d_S - d_T)$ nodes. The new source solution representations are thus learned.

Our proposed method is based on the intuition that the transformations in Eq. (5.9) of Step 1, together with Step 2, will lead to a *closer alignment of customer nodes indices and graph-based nodes distributions between the source and target VRPs*. As a consequence, the *learned source solution representations will be more likely to capture any overlap of optimal routes or sub-routes between the source and target*

tasks. Figure 5.8 illustrates the solution representation learning via our proposed method. Given the original source and target graphs in Figure 5.8(a) that may appear to lack transferrable sub-routes, the M_S in Step 1 attempts to match their geometric distributions of customer nodes as shown in Figure 5.8(b). Step 2 of our approach is depicted in Figure 5.8(b), (c), where each source node is reassigned a new index number based on the geometrically nearest target node.

FIGURE 5.8. A graphical illustration of the proposed methodology to learn well-aligned solution representations for distinct VRPs. (a) The original source and target graphs. (b) The source-to-target transformation M_S in Step 1. (c) The source node index reassignments in Step 2.

Besides, it is worth noting that the mapping M_S in Eq. (5.9) preserves graph-based shortest paths, hence our method does not change the optimal routes of the source in the transformed space. This is proven in *Lemma 1* below.

Lemma 1: The source-to-target transformations in Eq. (5.9) preserve the source problem's original graph structure and shortest paths.

Proof: Firstly, the angular rotation, translation and compression of a graph are rigid transformations that lead to direct isometry. Secondly, the linear combination of isometries is also an isometry. Since isometry preserves geometric structure [114], it is hence proven that the mapping M_S in Eq. (5.9) preserves the source task's original graph structure and shortest paths.

In line with Algorithm 5.1, the constructed M_S is subsequently applied to obtain the transformed source model $p_{M_S}^*(\mathbf{x}_T)$ via simple Monte Carlo approximation. We obtain the resultant MOTrEO+ M_S algorithmic instantiation for multi-objective TrEO in the corresponding combinatorial search spaces by actualizing $p_{M_S}^*(\mathbf{x}_T) \forall S \in \{1, 2, \dots, T-1\}$ and the current target model $p_T(\mathbf{x}_T|t)$ as *edge histogram-based probability distributions* [115] of permutation-coded solution representations. The target and transformed source models are then used in the stacked density estimation procedure [51] to build the target mixture model $[\sum_{S=1}^{T-1} w_S \cdot p_{M_S}^*(\mathbf{x}_T) + w_T \cdot p_T(\mathbf{x}_T|t)]$ in the target search space $\mathcal{X}_T$. In what follows, the performance advantage of our algorithm is demonstrated through numerical experiments. The following experimental study serves to verify the efficacy of our algorithm in boosting the performance of the target multi-objective optimization task of drone delivery route planning defined in Eq. (5.8).

5.4.3 Experimental Study

5.4.3.1 Source and Target Instances

We generate vehicle routing instances from several well-known VRP datasets [116, 117, 118]. For the truck delivery (source) instances, we either consider the original datasets or modify the datasets by removing an arbitrary number of customer nodes. The dimensionalities of the source and target problems, denoted as d_S and d_T , respectively, are moderately large with up to a few hundred customers. In our implementation, we generate new multi-objective drone delivery (target) instances according to the following procedure: (i) create a new target instance using only the Cartesian coordinates of customer nodes from the original VRP dataset; (ii) arbitrarily transform the geometric distribution of customer nodes and randomly assign an index number to each node; (iii) set the drone travel distance limit $l = 50$, and randomly generate the flight range h between [5,10] in L_2 -norm distances, the payload limit q between [20,200] in kilograms, as well as the customer demands $c_{T,j}$ between [1,5] in kilograms, for $j = 1, \dots, d_T$; (iv) remove customer nodes located beyond the range h . The drone constraints and customer demands values are motivated by the operating requirements of real commercial drones.

5.4.3.2 Experimental Configuration

Four optimizers are considered for comparison in this our experiments. They are: (i) the base NSGA-II [72], (ii) a multi-objective variant of the edge histogram-based sampling algorithm (EHBSA) which is a probabilistic model-based optimizer (without knowledge transfer) for combinatorial problems [119], (iii) the AMTEA without solution representation learning [10], and (iv) our proposed MOTrEO+ M_S . All search populations of size $N = 100$ consist of permutation-coded solutions. Excluding EHBSA, the other algorithms execute optimized crossover (with probability 0.75) along with inversion or swap sequence mutation (with probability 0.2) [120]. For EHBSA, the sampling without template strategy is used with its default parameter settings [115].

The transfer interval is set to $\Delta = 5$, at which target mixture models with edge histogram-based components [115] are built and sampled in both the AMTEA and the MOTrEO+ M_S , while our algorithm also incorporates the solution representation learning methodology described in Section 5.4.2. The solution quality achieved by all the optimizers is measured using the IGD performance metric [107]. A single-run of each optimizer is terminated after 10,000 function evaluations, whereas the true Pareto front is approximated at the end of 100,000 function evaluations of NSGA-II.

5.4.3.3 Results and Discussion

The average IGD values achieved by the four optimizers after 20 independent runs are provided in Table 5.4. Figure 5.9 depicts the convergence trends of all the optimizers and the transfer coefficients obtained by the AMTEA and the MOTrEO+ M_S , for 3 representative target instances. The highlight of Table 5.4 is that the MOTrEO+ M_S performs significantly better than all the other algorithms for all the six cases. Moreover, the convergence rates of the MOTrEO+ M_S are significantly enhanced in the early stages of the optimization processes, as seen in Figure 5.9(a), (b), (c). These outcomes are attributed to the activation of positive transfers in our algorithm, made possible by well-aligned solution representations learned through the proposed source-to-target mapping M_S . For example, the larger w_S values obtained by the MOTrEO+ M_S in Figure 5.9(d), (e), (f) show that it is able to uncover beneficial source-target structural overlap,

which is exploited to boost the target search. The superior performance of the MOTrEO+ M_S is also explained by its ability to achieve a complementary balance between inducing positive transfers and mitigating harmful negative transfers.

TABLE 5.4. Experimental results for the target multi-objective drone delivery route planning tasks after 10,000 function evaluations. In bold are the best results (while values with 95% confidence level based on the Wilcoxon Signed-Rank Test are marked with “*”). Results are reported based on 20 independent runs of all optimizers.

Problem Instance	d_S	Average IGD Values			
		NSGA-II	EHBSA	AMTEA	MOTrEO+ M_S
M-n151 ($d_T = 144$)	135, 140	175.37	170.63	184.17	10.38*
X-n214 ($d_T = 200$)	205, 213	130.50	131.45	143.76	6.38*
Golden_17 ($d_T = 222$)	210, 240	334.22	366.31	127.59	3.05*
Golden_18 ($d_T = 282$)	290, 300	346.67	308.07	208.09	70.46*
CMT4 ($d_T = 133$)	130, 140, 150	185.63	211.40	192.00	0.35*
X-n162 ($d_T = 143$)	140, 150, 161	184.35	173.71	198.44	57.61*

In contrast, the AMTEA achieves second best performance in 2 out of 6 cases, but it performs the worst in 3 out of 6 cases (i.e., M-n151, X-n214, X-n162). Without the solution representation learning module, the AMTEA’s performance is adversely impacted by the extra efforts required to sieve out negative transfers caused by unaligned source-target solution representations. The AMTEA is also less capable of facilitating positive transfers; as reflected by the smaller w_S values obtained in Figure 5.9(d), (e), (f). These observations affirm the efficacy of our method in uncovering useful routes or sub-routes that remain hidden in the original search spaces of source tasks. The proposed MOTrEO+ M_S is indeed more effective at leveraging related truck delivery routing experiences to enhance the performance of multi-objective route

optimization of UAVs for drone delivery planning. Therefore, our results in this case study clearly demonstrates the utility of solution representation learning for inducing positive transfers between distinct but related vehicle routing tasks.

FIGURE 5.9. Convergence trends for the target multi-objective optimization task of drone delivery route planning, and the transfer coefficients obtained by the AMTEA and the MOTrEO+ M_S , for target instances Golden_17 (left column), X-n214 (central column) and CMT4 (rightmost column). The shaded regions span one standard deviation on either side of the mean (based on 20 independent runs).

5.5 Chapter Summary

This chapter showcased the applicability of our framework for solution representation learning in TrEO in the context of multi-objective optimization. To this end, we proposed the MOTrEO+ M_S , which is a multi-objective probabilistic model-based TrEO algorithm with source-to-target map, aimed at inducing positive inter-task transfers by overcoming discrepancies between the original search spaces of source and target MOPs. Thereafter, case studies to validate the efficacy of the MOTrEO+ M_S were presented in the domains of multi-objective continuous optimization tasks as well

as a class of vehicle route planning combinatorial tasks. Under each case study, novel source-to-target search space transformation methodologies were put forward to learn well-aligned solution representations that unveil useful but hidden source-target relationships; these methods, when flexibly coupled with appropriate probability distribution models of the corresponding (continuous or combinatorial) search spaces, had given rise to distinct MOTrEO+ M_S algorithmic instantiations.

The salient feature of the resultant algorithms was their ability to promote positive transfers while simultaneously mitigating the threat of negative transfers. A range of empirical experiments was conducted to verify the efficacy of our algorithms. In addition to rigorous comparisons on complex multi-objective optimization benchmarks, we also considered practical examples of multi-objective engineering design as well as the route planning of unmanned aerial vehicles. The computational results affirmed that our proposed algorithms were able to achieve a favourable balance between the activation of positive transfers (from available source tasks) and the mitigation of negative inter-task interactions, which synergistically led to enhanced convergence rate and solution quality for the multi-objective target problems.

Chapter 6

Conclusions and Future Work

This chapter concludes the present thesis with a summary of main contributions arising from the research work of the author. Several future research directions are suggested thereafter for TrEO algorithms that improve their efficacy by learning from various sources of past problem-solving experiences.

6.1 Conclusions

This thesis is motivated by the core idea of learning well-aligned solution representations in TrEO as the means to induce positive transfers between distinct optimization tasks that bear discrepancies in their original search spaces. In parallel, the importance of synergizing solution representation learning with source-target similarity capture is emphasized as well, for the purpose of simultaneously guarding against the threat of negative transfers. The following summarizes the main contributions of this thesis, which are organized according to chapters.

Chapter 3 has formalized the notion of solution representation learning (SRL) in probabilistic model-based TrEO, on the grounds of a lack of approaches capable of jointly addressing two predominant issues in TrEO: (i) the scarcity of beneficial positive transfers, and (ii) the occurrence of harmful negative transfers. The state-of-the-art probabilistic model-based transfer approach for curbing negative transfers is first discussed, followed by the revelation of its shortcoming (that is envisioned to be overcome by SRL) – the narrow focus on curbing negative influences can lead to the conservative cancellation of knowledge transfers, when source-target domain mismatch scenarios (manifesting in unaligned solution representations) arise either from (i) arbitrarily heterogeneous source and target dimensionalities, or (ii) a lack of

overlap between their optimized search distributions, or both. To overcome the aforementioned challenges, a principled perspective of SRL (via search space mappings) in TrEO is established, where a streamlined definition of this notion is first given. Accordingly, the source-to-target map is proposed as a category of spatial mappings aimed at learning well-aligned solution representations and hence inducing positive inter-task transfers; in parallel, the probabilistic formulation of TrEO with a source-to-target map is outlined. Thereafter, a novel framework for SRL in probabilistic model-based TrEO is put forward to guide the development of TrEO algorithms that are capable of inducing greater alignment (of solution representations) and hence positive transfers, while concurrently mitigating negative inter-task interactions.

In accordance with the proposed framework, Chapter 4 developed a probabilistic model-based TrEO algorithm with source-to-target map, denoted as the $\text{TrEO}+M_S$. In particular, source-to-target spatial mappings M_S are first constructed (either offline or online), which are applied to learn well-aligned source solutions that are in turn used to transform the source models; thus promoting positive transfers from source to target tasks. Case studies are then presented in the domains of discrete (i.e., the 0-1 Knapsack Problem exemplar) and continuous optimization. Under each optimization domain, a novel methodology for constructing M_S is proposed, giving rise to a distinct algorithmic instantiation of the $\text{TrEO}+M_S$ for the corresponding domain. Computational experiments comprising binary decision-making Knapsack problems, continuous benchmark tasks and a real-world example on multi-location inventory planning are conducted to verify the efficacy of the resultant $\text{TrEO}+M_S$ algorithms. Results have demonstrated the performance advantage of the proposed methods, showing significant improvements in convergence rates and solution qualities even when compared to state-of-the-art TrEO algorithms.

In Chapter 5, the applicability of the novel framework is extended to the context of multi-objective TrEO. Accordingly, a multi-objective probabilistic model-based TrEO algorithm with source-to-target map is proposed and labeled as the $\text{MOTrEO}+M_S$. The salient feature of the $\text{MOTrEO}+M_S$ lies in its capability to induce increased positive correlations (through SRL) across performance indicators in multi-dimensional objective space of distinct optimization tasks. The embedded SRL module

which learns well-aligned solution representations (via source-to-target spatial mappings M_S) that unveil useful but hidden source-target relationships, acts in conjunction with the probabilistic model-based inter-task similarity capture mechanism, jointly facilitating increased positive transfers and reduced negative transfers. The efficacy of the MOTrEO+ M_S is showcased using case studies spanning the domains of continuous optimization and a class of graph-based combinatorial problems. Similarly, for each domain, a distinct algorithmic instantiation of the MOTrEO+ M_S is realized based on the proposed methodology for constructing M_S in the corresponding domain. The resultant algorithmic instantiations have shown substantial performance boosts across complex multi-objective benchmark tasks, as well as practical examples on multi-objective engineering design and route planning of unmanned aerial vehicles. Besides demonstrating enhanced convergence rates and solution qualities, further analyses of the results have also revealed the effectiveness of the source-to-target spatial mappings as well as insights into the synergy between learning solution representations and capturing source-target similarities.

The work of this thesis subsequently sheds light on several promising future research directions for TrEO algorithms that improve their efficacy by learning from various (and possibly heterogeneous) sources of past problem-solving experiences.

6.2 Future Research Directions

This thesis has established a novel framework for solution representation learning (SRL) in probabilistic model-based TrEO, paving the way for the design of evolutionary optimizers that can better leverage useful but possibly hidden search biases from various sources. Along this line, we suggest several promising research ideas for future explorations.

A practical future research direction is to further expand the applicability of the proposed TrEO+ M_S and MOTrEO+ M_S algorithms to *a wider variety of real-world optimization problems*. For instance, it is interesting to delve deeper into different classes of discrete or combinatorial optimization problems that are characterized by a rich diversity of search spaces and solution representations. While the discrete 0-1 Knapsack Problem and the graph-based combinatorial vehicle routing tasks have been

studied in this thesis, a plethora of other problems remains to be explored (potentially demanding new methodologies for constructing effective spatial mappings). Furthermore, there exist practical problems such as those characterized by *computationally expensive* (e.g., simulation-based) function evaluations or resource-constrained settings, which could receive significant savings in computing resource utilization by tapping on SRL in TrEO – recent advances in transfer Bayesian optimization algorithms [121] provided some initial pathways to tackle source-target domain mismatch in such cases as well.

Improving the *scalability* of the framework for SRL in probabilistic model-based TrEO is one foreseeable research direction for the purpose of encouraging widespread adoption. From a practitioner’s perspective, given the ever growing amount of past problem-solving data available, it is of vital importance to (i) automatically extract source information that are more relevant to the target task, and (ii) reduce the storage of redundant or less useful source data. As an example, consider the two-layer neural network methodologies (for learning well-aligned solution representations in continuous search spaces) utilized in the TrEO+ M_S and the MOTrEO+ M_S . These methods currently require the source population data at every search iteration to be stored, which can potentially implode the source knowledge base to the extent of expensive big datasets consideration in TrEO [85, 122]. This example highlights the significance of exploring ways to *improve the utilization efficiency of the available knowledge priors*, especially in practical TrEO settings involving many source tasks.

Imminent information technologies for optimization and learning [123, 124, 125, 126] such as cloud computing and the internet of things, serve to motivate another promising research direction in future – the fusion of such technologies and adaptive TrEO algorithms (e.g., TrEO+ M_S and MOTrEO+ M_S) for the deployment of *on-demand problem-solving*. For instance, imagine a scenario of multiple users requesting access to *cloud-based optimization solvers* for solving distinct but possibly related tasks. Such a scenario is not just common in practice, it also provides the scope for knowledge transfers between various users (i.e., sources of information) that would result in mutual benefits like enhanced convergence performance and reduced computational requirements. Given the diverse problem characteristics posed by different users of cloud-based solvers, a likely challenge in this case is the occurrence

of source-target domain mismatch (manifesting in unaligned solution representations) which could lead to decreased positive transfers or even increased negative transfers. To this end, the utility of the formalized SRL in probabilistic model-based TrEO comes in handy to overcome the aforementioned challenge. A research aspect of interest in this case would be the adaptive selection as well as sharing of spatial mappings according to the domains or configurations of optimization tasks being solved. The above notion of TrEO is visualized as a form of *generalized optimization intelligence (GOI)*, encompassing intelligent machines with the ability to automatically learn from various experiences when solving newly encountered tasks.

Furthermore, alternative categories of spatial mappings may be conceived on the basis of distinct forms of transformations involved [13]. For instance, the *target-to-source map* is one possible category of spatial mappings to be explored, by which the target task is mapped to the source search space. The joint transformation of both the source and target tasks to a third (and common) solution representation space gives another possible category of spatial mappings – we refer to this category as the *source-cum-target maps*. In a similar manner as the *source-to-target map* (discussed in Section 3.3.1), spatial mappings of the two possible categories also transform tasks (originally defined in different search spaces, i.e., $\mathcal{X}_S \neq \mathcal{X}_T$) to a common and well-aligned solution representation space in which positive transfers can be induced. Notably, *the notion of SRL in TrEO defined in this thesis is seamlessly extensible to encompass different categories (or forms) of spatial mappings*; see Definition 3.1. What is more, the novel framework flexibly allows different forms of search space mappings to be incorporated within its SRL module and probabilistic model-based transfer mechanism; see Section 3.4 and Algorithm 3.2.

In what follows, we shall briefly discuss the prospects of (i) the target-to-source map, and (ii) the source-cum-target maps, as well as outline the probabilistic formulations of TrEO inclusive of the respective maps.

6.2.1 Target-to-Source Map

Let the search space of source task $\mathcal{T}_S$ be $\mathcal{X}_S \subseteq \mathbb{R}^{d_S}$ and that of the target task $\mathcal{T}_T$ be $\mathcal{X}_T \subseteq \mathbb{R}^{d_T}$. In general, $\mathcal{X}_S \neq \mathcal{X}_T$ and $d_S \neq d_T$. In accordance with Eq. (3.4), the

probabilistic formulation of TrEO inclusive of a target-to-source map for a *single source* case can be written as,

$$\begin{aligned} \max_{w_S, w_T, p_T(x_T)} \int_{\mathcal{X}_S} f_T(M_T^{-1}(x_S)) \cdot [w_S \cdot p_S^*(x_S) + w_T \cdot p_{M_T}(x_S)] \cdot dx_S, \\ \text{s. t. , } w_S + w_T = 1 \text{ and } w_S, w_T \geq 0. \end{aligned} \quad (6.1)$$

Above, M_T is the target-to-source map where $M_T: \mathcal{X}_T \rightarrow \mathcal{X}_S$, and p_{M_T} is obtained by transforming the argument of $p_T(x_T)$ via M_T . For applications where problem sizes increase over time (i.e., $d_T > d_S$), it is feasible for the mapping function to be linear under the formulation of Eq. (6.1), as p_{M_T} will not be degenerate in $\mathcal{X}_S$; thus, enabling building and sampling of the target mixture model $[w_S \cdot p_S^*(x_S) + w_T \cdot p_{M_T}(x_S)]$.

The purpose of introducing M_T in Eq. (6.1) is to stimulate $p_S^*(x_S) \approx p_{M_T}^*(x_S)$, where $p_{M_T}^*$ is the transform of the *a priori* unknown $p_T^*(x_T)$ – or in other words, to induce strong (positive) correlation between $f_T(M_T^{-1}(x_S))$ and $f_S(x_S)$. Given such a transformation, samples drawn from the resultant mixture model, when mapped back to $\mathcal{X}_T$ via M_T^{-1} as depicted in Figure 6.1, will yield good quality target solutions (with high probability when $p_S^*(x_S) \approx p_{M_T}^*(x_S)$).

FIGURE 6.1. Illustration of the *target-to-source map* as a possible category of spatial mappings for SRL in TrEO.

According to Proposition 3.1, one way to attain the desired outcome is to learn M_T and M_T^{-1} so that a strong ordinal correlation is induced between $f_S(\mathbf{x}_S)$ and $f_T(M_T^{-1}(\mathbf{x}_S))$. Here, M_T^{-1} marks a slight abuse of notation as M_T may not be precisely invertible. Nevertheless, we assume that in constructing M_T and M_T^{-1} , the minimization of the reconstruction error of the composition $(M_T^{-1} \circ M_T)(\mathbf{x}_T)$ is accounted for. This can be achieved by optimizing against the following objective [127],

$$\min_{M_T, M_T^{-1}} \lambda_1 \cdot \sum_{\forall \mathbf{x}_T \in P_T} \|\mathbf{x}_T - (M_T^{-1} \circ M_T)(\mathbf{x}_T)\|_2^2 - \lambda_2 \cdot \Gamma(f_S(\mathbf{x}_S); f_T(M_T^{-1}(\mathbf{x}_S))), \quad (6.2)$$

where P_T is the target population of solutions, $\Gamma(f_S; f_T)$ is an estimate of the correlation between the source and target fitness functions [128], while λ_1 and λ_2 are weighting terms.

An important distinction when compared to the source-to-target map (in Section 3.3.1) is in the need to explicitly define an inverse map M_T^{-1} , which would increase the complexity of learning well-aligned solution representations using the target-to-source map. Finally, when the SRL module of our framework (in Algorithm 3.2) employs target-to-source spatial mappings, the probabilistic model-based transfer mechanism can obtain the transfer coefficients by modifying Eq. (3.5) and solving the log-likelihood program (via the EM algorithm [84]) below,

$$\max_{w_S, w_T} \sum_{\forall \mathbf{x}_S \in P_S(t)} \log(w_S \cdot p_S^*(\mathbf{x}_S) + w_T \cdot p_{M_T}(\mathbf{x}_S|t)), \quad (6.3)$$

where $p_{M_T}(\mathbf{x}_S|t)$ and $P_S(t)$ are the transformed target model and the population of solutions in the source's solution representation space, respectively, at iteration t .

6.2.2 Source-cum-Target Maps

Imagine a special case where the behaviours of the source and target fitness functions are correlated with respect to a small subset of variables (or features) in their overall

search spaces. An illustration is depicted in Figure 6.2, where $\mathcal{X}_C \subseteq \mathbb{R}^{d_C}$ represents the *common lower-dimensional subspace* of dimensionality d_C . With reference to Eq. (3.4), the probabilistic formulation of TrEO inclusive of source-cum-target maps for a *single source* case (involving the joint transformation of source and target search spaces to $\mathcal{X}_C$) can be expressed as,

$$\begin{aligned} \max_{w_S, w_T, p_T(x_T)} \int_{\mathcal{X}_C} f_T(M_T^{C^{-1}}(x_C)) \cdot [w_S \cdot p_{M_S^C}^*(x_C) + w_T \cdot p_{M_T^C}(x_C)] \cdot dx_C, \\ \text{s. t.}, w_S + w_T = 1 \text{ and } w_S, w_T \geq 0. \end{aligned} \quad (6.4)$$

FIGURE 6.2. Illustration of the *source-cum-target maps* as a possible category of spatial mappings for SRL in TrEO.

Here, M_S^C and M_T^C are the source-cum-target maps where $M_S^C: \mathcal{X}_S \rightarrow \mathcal{X}_C$, $M_T^C: \mathcal{X}_T \rightarrow \mathcal{X}_C$, and $p_{M_T^C}$, $p_{M_S^C}^*$ are the respective transformed target and optimized source distributions in $\mathcal{X}_C$. We assume that M_T^C is *approximately* invertible, which is achievable by minimizing the reconstruction error of the composition $(M_T^{C^{-1}} \circ M_T^C)(x_T)$ in a similar manner as in Eq. (6.2). Hence, we aim to learn the spatial mappings M_S^C , M_T^C and $M_T^{C^{-1}}$ such that a strong ordinal correlation is induced between

$f_S(M_S^{C-1}(\mathbf{x}_C))$ and $f_T(M_T^{C-1}(\mathbf{x}_C))$ (where M_S^{C-1} is a hypothetical inversion that need not be explicitly derived in practice).

As a delineative example application of source-cum-target maps, we may consider M_S^C and M_T^C to take the form of *feature selection* operations expressible as $M_S^C(\mathbf{x}_S) = \mathbf{x}_S \odot \boldsymbol{\beta}_S^C$ and $M_T^C(\mathbf{x}_T) = \mathbf{x}_T \odot \boldsymbol{\beta}_T^C$, where $\mathbf{x} \odot \boldsymbol{\beta}$ is the Hadamard product (i.e., element-wise multiplication) between $\mathbf{x}$ and $\boldsymbol{\beta}$, while $\boldsymbol{\beta}_S^C \in \{0,1\}^{d_S}$ and $\boldsymbol{\beta}_T^C \in \{0,1\}^{d_T}$ are binary indicator vectors whose entries are 1 for the selected subset of features and 0 otherwise [129]. Since both $\boldsymbol{\beta}_S^C$ and $\boldsymbol{\beta}_T^C$ map solutions to the common space $\mathcal{X}_C$ of dimensionality d_C , they must satisfy $\|\boldsymbol{\beta}_S^C\|_0 = \|\boldsymbol{\beta}_T^C\|_0 = d_C$.

Lastly, when source-cum-target spatial mappings are used in the SRL module of our framework (in Algorithm 3.2), the probabilistic model-based transfer mechanism deciphers the transfer coefficients by maximizing the modified log-likelihood program below (which can also be solved via the EM algorithm [84]),

$$\max_{w_S, w_T} \sum_{\forall \mathbf{x}_C \in P_C(t)} \log(w_S \cdot p_{M_S^C}^*(\mathbf{x}_C) + w_T \cdot p_{M_T^C}(\mathbf{x}_C|t)), \quad (6.5)$$

where $p_{M_S^C}^*(\mathbf{x}_C)$ is the transformed source model, while $p_{M_T^C}(\mathbf{x}_C|t)$ and $P_S(t)$ refer to the transformed target search distribution and the population of solutions in the common solution representation space $\mathcal{X}_C$, respectively, at iteration t .

Following from the discussions in Section 6.2.1 and Section 6.2.2, as well as taking guidance from the framework in Algorithm 3.2, another promising future research direction is to propose new probabilistic model-based TrEO algorithms that incorporate either *target-to-source map* or *source-cum-target maps* for learning well-aligned solution representations (and hence activating positive transfers). It would be constructive to (i) investigate the efficacies of such algorithms in different source-target domain mismatch scenarios, and (ii) study the behaviour or impact of distinct categories of search space mappings on particular optimization domains or problem configurations.

Abbreviations

AMTEA	Adaptive Model-Based Transfer Evolutionary Algorithm
CD	Crowding Distance
CEA	Canonical Evolutionary Algorithm
CEA+ M_S	The TrEO Search with Adaptive Learning Across Heterogeneous Problems
CIHS	Complete Intersection with High Similarity
CILS	Complete Intersection with Low Similarity
CIMS	Complete Intersection with Medium Similarity
EA	Evolutionary Algorithm
EHBSA	Edge Histogram-Based Sampling Algorithm
EM	Expectation-Maximization
GOI	Generalized Optimization Intelligence
IGD	Inverted Generational Distance
KP	Knapsack Problem
LP	Linear Program
MOEA	Multi-Objective Evolutionary Algorithm
MOP	Multi-Objective Problem
MOTrEO(RSM)	A Multi-Objective TrEO Algorithm that Transfers Solutions from a Random Source Model

MOTrEO+ M_S	The Proposed Multi-Objective Probabilistic Model-Based TrEO Algorithm with Source-to-Target Map
NF	Nondominated Front
NIHS	No Intersection with High Similarity
NILS	No Intersection with Low Similarity
NIMS	No Intersection with Medium Similarity
NSGA-II	Nondominated Sorted Genetic Algorithm-II
NSGA-II+ M_S	A Multi-Objective Variant of the TrEO Search with Adaptive Learning Across Heterogeneous Problems
PIHS	Partial Intersection with High Similarity
PILS	Partial Intersection with Low Similarity
PIMS	Partial Intersection with Medium Similarity
RSM	Random Source Model
SRL	Solution Representation Learning
TrEO	Transfer Evolutionary Optimization
TrEO+ M_S	The Proposed Probabilistic Model-Based TrEO Algorithm with Source-to-Target Map
UAV	Unmanned Aerial Vehicle
VRP	Vehicle Routing Problem
xNES	Exponential Natural Evolution Strategies

Notations of Chapter 3

T	Total number of distinct optimization tasks considered in series
$\mathcal{T}_i$	The i -th optimization task
$\mathcal{X}$	The universal search space encompassing the search spaces of all T optimization tasks
$\mathbf{x}$	A candidate solution vector in space $\mathcal{X}$
$f_i(\mathbf{x})$	Objective function of the i -th optimization task
$p_i(\mathbf{x})$	The probability distribution of candidate solutions for task $\mathcal{T}_i$ in space $\mathcal{X}$
P_i	A population of candidate solutions for task $\mathcal{T}_i$
$p_i^*(\mathbf{x})$	The probability distribution of good (near-optimal) solutions for task $\mathcal{T}_i$ in space $\mathcal{X}$
P_i^*	A population of good (near-optimal) solutions for task $\mathcal{T}_i$
f_i^*	The globally optimum objective function value for task $\mathcal{T}_i$
ε_i	A small convergence threshold for task $\mathcal{T}_i$
$\mathcal{D}_i$	Domain of an optimization task $\mathcal{T}_i$
w_i	Transfer coefficient deduced for task $\mathcal{T}_i$
$p_i(\mathbf{x} t)$	The probability distribution of candidate solutions for task $\mathcal{T}_i$ in space $\mathcal{X}$ at iteration t

$P_i(t)$	A population of candidate solutions for task $\mathcal{T}_i$ in space $\mathcal{X}$ at iteration t
M_i	A search space mapping that maps task $\mathcal{T}_i$ from its originally defined space to a transformed solution representation space
M_i^{-1}	A reverse mapping that maps task $\mathcal{T}_i$ from the transformed solution representation space to its originally defined space
$\mathcal{X}_i$	The originally defined search space of task $\mathcal{T}_i$
$\mathbf{x}_i$	A candidate solution vector in space $\mathcal{X}_i$
d_i	The search space dimensionality of task $\mathcal{T}_i$
$p_i^*(\mathbf{x}_i)$	The probability distribution of good (near-optimal) solutions for task $\mathcal{T}_i$ in space $\mathcal{X}_i$
$M_i(\mathbf{x}_i)$	A candidate solution vector $\mathbf{x}_i$ for task $\mathcal{T}_i$ in the transformed solution representation space under mapping M_i
$p_{M_i}^*(\mathbf{x}_j)$	The probability distribution of good (near-optimal) solutions for task $\mathcal{T}_i$ in space $\mathcal{X}_j$ under mapping M_i
$\mathbf{A}$	A matrix of real-numbered elements
$\mathbf{b}$	A bias vector of real-numbered elements
N	The total number of solutions in the entire search population
n	The number of solutions in a selected parent population
Δ	Transfer interval
$\mathcal{C}_T(t)$	Offspring population of target task $\mathcal{T}_T$ at iteration t

Notations of Chapter 4

$\mathcal{T}_S$	Source optimization task
$\mathcal{X}_S$	The search space of $\mathcal{T}_S$
$\mathbf{x}_S$	A solution vector in space $\mathcal{X}_S$
d_S	The dimensionality of $\mathcal{T}_S$
$\mathcal{D}_S$	The domain of $\mathcal{T}_S$
$p_S^*(\mathbf{x}_S)$	The probability distribution of good (near-optimal) solutions for $\mathcal{T}_S$ in its originally defined space $\mathcal{X}_S$
$\mathcal{T}_T$	Target optimization task
$\mathcal{X}_T$	The search space of $\mathcal{T}_T$
$\mathbf{x}_T$	A solution vector in space $\mathcal{X}_T$
d_T	The dimensionality of $\mathcal{T}_T$
$\mathcal{D}_T$	The domain of $\mathcal{T}_T$
$p_T^*(\mathbf{x}_T)$	The probability distribution of good (near-optimal) solutions for $\mathcal{T}_T$ in its solution representation space $\mathcal{X}_T$
M_S	Source-to-target search space mapping
$p_{M_S}^*(\mathbf{x}_T)$	The probability distribution of good (near-optimal) solutions for source task $\mathcal{T}_S$ in the target space $\mathcal{X}_T$ under mapping M_S

f_T	Target objective function
N	The total number of solutions in the target search population
$P_T(t)$	Target search population at iteration t
Δ	Transfer interval
$C_T(t)$	Offspring population of target task $\mathcal{T}_T$ at iteration t
$p_T(\mathbf{x}_T t)$	The probability distribution of candidate solutions for target task $\mathcal{T}_T$ in space $\mathcal{X}_T$ at iteration t
w_S	Transfer coefficient deduced for source task $\mathcal{T}_S$
w_T	Transfer coefficient deduced for target task $\mathcal{T}_T$
KP	Knapsack problem
m	Total number of items of a KP
v_i	Value of the i -th item of a KP
q_i	Weight of the i -th item of a KP
Q	Total knapsack capacity of a KP
r_i	Efficiency ratio of the i -th item of a KP
$M_S(\mathbf{x}_S)$	A source solution vector $\mathbf{x}_S$ for task $\mathcal{T}_S$ in the transformed solution representation space under mapping M_S
$\mathbf{A}_{\mathcal{P}}$	A permutation matrix for reordering items in a source KP
f_S	Source objective function
M_S^{-1}	A hypothetical inversion that maps source task $\mathcal{T}_S$ from $\mathcal{X}_T$ to $\mathcal{X}_S$
$\mathbf{x}_S^*$	A good (near-optimal) solution for $\mathcal{T}_S$ in its originally defined space $\mathcal{X}_S$

$\mathbf{x}_T^*$	A good (near-optimal) solution for $\mathcal{T}_T$ in space $\mathcal{X}_T$
$\boldsymbol{\theta}$	Parameters of a two-layer feedforward neural network
$P_S(t)$	Source search population at iteration t
$\boldsymbol{\psi}_k$	The k -th layer of transformation in a two-layer feedforward neural network
$\mathbf{A}$	A matrix of real-numbered elements
$\mathbf{b}$	A bias vector of real-numbered elements
σ	The non-linear sigmoidal activation function
d_h	Number of neurons in the hidden layer of a two-layer feedforward neural network
λ	The L_2 regularization coefficient
I_{d_h}	An identity matrix of size d_h
Q_{better}	The empirical probability that the TrEO+ M_S achieves better target objective values relative to a baseline algorithm
$p(D)$	The demand distribution with mean $\boldsymbol{\mu}$ and variance Σ of the multi-location inventory planning problem
c_h	Inventory holding cost
c_b	Inventory backlogging cost
c_{lt}	Inventory lateral transshipment cost
$\boldsymbol{\mu}_S$	The expected demands at the retailers of source task $\mathcal{T}_S$
$\boldsymbol{\mu}_T$	The expected demands at the retailers of target task $\mathcal{T}_T$

Notations of Chapter 5

$f_T^i(\mathbf{x}_T)$	The i -th target objective function
m	Number of target objective functions
$\mathbf{F}_T$	A vector of m target objective functions
$\mathcal{X}_T$	The target search space
$\mathbf{x}_T$	A solution vector in space $\mathcal{X}_T$
$\mathbf{x}_T^*$	A good (near-Pareto optimal) target solution in space $\mathcal{X}_T$
$\mathcal{D}_S$	The source domain
$\mathcal{D}_T$	The target domain
M_S	Source-to-target search space mapping
Δ	Transfer interval
$\mathcal{T}_T$	Target optimization task
N	The total number of solutions in the target search population
$P_T(t)$	Target search population at iteration t
$C_T(t)$	Offspring population of target task $\mathcal{T}_T$ at iteration t
$\mathcal{T}_S$	Source optimization task
$\mathcal{X}_S$	The search space of $\mathcal{T}_S$

$\mathbf{x}_S$	A solution vector in space $\mathcal{X}_S$
$p_S^*(\mathbf{x}_S)$	The probability distribution of good (near-Pareto optimal) solutions for $\mathcal{T}_S$ in its originally defined space $\mathcal{X}_S$
$p_{M_S}^*(\mathbf{x}_T)$	The probability distribution of good (near-Pareto optimal) solutions for $\mathcal{T}_S$ in the target space $\mathcal{X}_T$ under mapping M_S
$p_T(\mathbf{x}_T t)$	The probability distribution of candidate solutions for target task $\mathcal{T}_T$ in space $\mathcal{X}_T$ at iteration t
w_S	Transfer coefficient deduced for source task $\mathcal{T}_S$
w_T	Transfer coefficient deduced for target task $\mathcal{T}_T$
d_S	The dimensionality of $\mathcal{T}_S$
d_T	The dimensionality of $\mathcal{T}_T$
$p_T^*(\mathbf{x}_T)$	The probability distribution of good (near-Pareto optimal) solutions for $\mathcal{T}_T$ in its solution representation space $\mathcal{X}_T$
$M_S(\mathbf{x}_S)$	A source solution vector $\mathbf{x}_S$ for task $\mathcal{T}_S$ in the transformed solution representation space under mapping M_S
θ	Parameters of a two-layer feedforward neural network
ψ_k	The k -th layer of transformation in a two-layer feedforward neural network
$P_S(t)$	Source search population at iteration t
$\mathbf{A}$	A matrix of real-numbered elements
$\mathbf{b}$	A vector of real-numbered elements
σ	The non-linear sigmoidal activation function

d_h	Number of neurons in the hidden layer of a two-layer feedforward neural network
λ	The L_2 regularization coefficient
I_{d_h}	An identity matrix of size d_h
γ	Similarity between the fitness landscapes of $\mathcal{T}_S$ and $\mathcal{T}_T$ in the complex multi-objective benchmark tasks
μ_S	The alignment of source design variables in the multi-objective vehicle crashworthiness design tasks
μ_T	The alignment of target design variables in the multi-objective vehicle crashworthiness design tasks
G	Source graph of truck delivery task
V	Set of vertices in source graph G
E	Set of edges in source graph G
K	Total number of trucks
Q	Truck delivery capacity
L	Truck travel distance limit
R	Total number of truck routes
$D_{k,r}$	The distance travelled by truck k in route r
$W_{k,r}$	The weight capacity served by truck k in route r
f_S	The source objective function
H	Target graph of drone delivery task
U	Set of vertices in target graph H

A	Set of edges in target graph H
q	Drone delivery capacity
l	Drone travel distance limit
h	Drone flight range
B	Total number of drones and drone routes
$D_{y,b}$	The distance travelled by drone y in route b
$W_{y,b}$	The payload served by drone y in route b
$\mathbf{C}_{s,i}$	The Cartesian coordinates vector of a source node i
α	A graph scaling constant
θ	The angle between the principle components of source and target graphs

Notations of Chapter 6

M_S	Source-to-target search space mapping
$\mathcal{X}_S$	The source search space
$\mathcal{X}_T$	The target search space
$\mathcal{T}_S$	Source optimization task
d_S	The dimensionality of $\mathcal{T}_S$
$\mathcal{T}_T$	Target optimization task
d_T	The dimensionality of $\mathcal{T}_T$
f_S	Source objective function
$\mathbf{x}_S$	A solution vector in space $\mathcal{X}_S$
$p_S^*(\mathbf{x}_S)$	The probability distribution of good (near-optimal) solutions for $\mathcal{T}_S$ in its originally defined search space $\mathcal{X}_S$
w_S	Transfer coefficient deduced for source task $\mathcal{T}_S$
f_T	Target objective function
$\mathbf{x}_T$	A solution vector in space $\mathcal{X}_T$
$p_T(\mathbf{x}_T)$	The probability distribution of candidate solutions for target task $\mathcal{T}_T$ in its originally defined search space $\mathcal{X}_T$
M_T	Target-to-source search space mapping

$p_{M_T}(\mathbf{x}_S)$	The probability distribution of candidate solutions for target task $\mathcal{T}_T$ in the source space $\mathcal{X}_S$ under mapping M_T
w_T	Transfer coefficient deduced for target task $\mathcal{T}_T$
$p_T^*(\mathbf{x}_T)$	The <i>a priori</i> unknown probability distribution of good (near-optimal) solutions for $\mathcal{T}_T$ in space $\mathcal{X}_T$
$p_{M_T}^*(\mathbf{x}_S)$	The <i>a priori</i> unknown probability distribution of good (near-optimal) solutions for $\mathcal{T}_T$ in source space $\mathcal{X}_S$ under mapping M_T
M_T^{-1}	A reverse mapping that maps task $\mathcal{T}_T$ from the source solution representation space $\mathcal{X}_S$ to its originally defined space $\mathcal{X}_T$
$\Gamma(f_S; f_T)$	An estimate of the correlation between the source and target objective functions
λ_i	The i -th weighting term
P_T	Population of solutions in the target space $\mathcal{X}_T$
$P_S(t)$	Population of solutions in the source space $\mathcal{X}_S$ at iteration t
$p_{M_T}(\mathbf{x}_S t)$	The probability distribution of candidate solutions for target task $\mathcal{T}_T$ in the source space $\mathcal{X}_S$ under mapping M_T at iteration t
$\mathcal{X}_C$	The common solution representation space
d_C	The dimensionality of $\mathcal{X}_C$
$\mathbf{x}_C$	A solution vector in space $\mathcal{X}_C$
M_S^C	Source-cum-target search space mapping that maps source task $\mathcal{T}_S$ from its original space $\mathcal{X}_S$ to the common space $\mathcal{X}_C$
M_T^C	Source-cum-target search space mapping that maps target task $\mathcal{T}_T$ from its original space $\mathcal{X}_T$ to the common space $\mathcal{X}_C$

$p_{M_S^C}^*(\mathbf{x}_C)$	The probability distribution of good (near-optimal) solutions for $\mathcal{T}_S$ in the common space $\mathcal{X}_C$ under mapping M_S^C
$p_{M_T^C}(\mathbf{x}_C)$	The probability distribution of candidate solutions for target task $\mathcal{T}_T$ in the common space $\mathcal{X}_C$ under mapping M_T^C
$M_S^{C^{-1}}$	A reverse mapping that maps source task $\mathcal{T}_S$ from the common space $\mathcal{X}_C$ to its originally defined space $\mathcal{X}_S$
$M_T^{C^{-1}}$	A reverse mapping that maps target task $\mathcal{T}_T$ from the common space $\mathcal{X}_C$ to its originally defined space $\mathcal{X}_T$
$\boldsymbol{\beta}$	A binary indicator vector for feature selection
$p_{M_T^C}(\mathbf{x}_C t)$	The probability distribution of candidate solutions for $\mathcal{T}_T$ in the common space $\mathcal{X}_C$ under mapping M_T^C at iteration t

Bibliography

- [1] A. Newell, J. C. Shaw, and H. A. Simon, “Elements of a theory of human problem solving,” *Psychological review*, vol. 65, no. 3, pp. 151, 1958.
- [2] T. Ward, and A. Plagnol, “Defined by Our Past,” in *Cognitive Psychodynamics as an Integrative Framework in Counselling Psychology and Psychotherapy*, pp. 69–88, Palgrave Macmillan, Cham, 2019.
- [3] S. Thrun, “Is learning the n-th thing any easier than learning the first?,” in *Advances in Neural Information Processing Systems*, pp. 640–646, 1996.
- [4] S. J. Pan, and Q. Yang, “A survey on transfer learning,” *IEEE Transactions on Knowledge and Data Engineering*, volume 22, issue 10, pages 1345–1359, 2010.
- [5] Y. Jin, H. Wang, T. Chugh, D. Guo, and K. Miettinen, “Data-driven evolutionary optimization: An overview and case studies,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 3, pp. 442–458, 2019.
- [6] Y. S. Ong, and A. Gupta, “AIR 5: Five Pillars of Artificial Intelligence Research,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 3, no. 5, pp. 411–415, Oct. 2019.
- [7] A. Gupta and Y. S. Ong. *Memetic Computation: The Mainspring of Knowledge Transfer in a Data-Driven Optimization Era*, vol. 21. Springer, 2018.
- [8] 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*, volume 2, issue 1, pages 51–64, 2018.

- [9] L. Feng, Y. S. Ong, A. H. Tan, and I. W. Tsang, "Mememes as building blocks: A case study on evolutionary optimization + transfer learning for routing problems," *Memetic Computing*, vol. 7, no. 3, pp. 159–180, 2015.
- [10] B. Da, A. Gupta, and Y. S. Ong, "Curbing negative influences online for seamless transfer evolutionary optimization," *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4365–4378, Aug. 2018.
- [11] A. Gupta, Y. S. Ong, and L. Feng, "Multifactorial evolution: toward evolutionary multitasking," *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 343–357, Jun. 2016.
- [12] A. Gupta, B. Da, Y. Yuan, Y. S. Ong, "On the emerging notion of evolutionary multitasking: A computational analog of cognitive multitasking," In *Recent Advances in Evolutionary Multi-Objective Optimization*, New York, NY, USA: Springer, pp. 139–157, 2017.
- [13] R. Lim, A. Gupta, Y. S. Ong, L. Feng and A. N. Zhang, "Non-Linear Domain Adaptation in Transfer Evolutionary Optimization," *Cognitive Computation*, vol. 13, pp. 290–307, Feb. 2021.
- [14] R. Lim, L. Zhou, A. Gupta, Y. S. Ong and A. N. Zhang, "Solution Representation Learning in Multi-Objective Transfer Evolutionary Optimization," *IEEE Access*, vol. 9, pp. 41844–41860, Mar. 2021.
- [15] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.
- [16] 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, Aug. 2014.
- [17] M. Iqbal, B. Xue, H. Al-Sahaf, and M. Zhang, "Cross-domain reuse of extracted knowledge in genetic programming for image classification," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 4, pp. 569–587, Jan. 2017.

- [18] P. Zhao, and S. C. Hoi, “OTL: A framework of Online Transfer Learning,” in *Proceedings of the International Conference on Machine Learning (ICML)*, Jun. 2010, pp. 1231–1238.
- [19] L. Ge, J. Gao, and A. Zhang, “OMS-TL: A framework of online multiple source transfer learning,” in *Proceedings of the ACM International Conference on Information & Knowledge Management*, Oct. 2013, pp. 2423–2428.
- [20] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, “Domain adaptation via transfer component analysis,” *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.
- [21] X. Glorot, A. Bordes, and Y. Bengio, “Domain adaptation for large-scale sentiment classification: A deep learning approach,” in *Proceedings of the International Conference on Machine Learning (ICML)*, Jan. 2011, pp. 513–520.
- [22] T. Back, *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, 1996.
- [23] D. Thierens. *Analysis and design of genetic algorithm*. Ph.D. Dissertation, 1995.
- [24] J. Quinlan. Combining instance-based and model-based learning. In *Proceedings of the Tenth International Conference on Machine Learning*, pages 236–243, 1993.
- [25] M. Pelikan, D. Goldberg, F. G. Lobo, “A survey of optimization by building and using probabilistic models,” *Computational Optimization and Applications*, vol. 21, pp. 5-20, 2002.
- [26] R. A. Mahale and S. D. Chavan, “A survey: evolutionary and swarm based bio-inspired optimization algorithms,” *International Journal of Scientific and Research Publications*, vol. 2, no. 12, pp. 1–6, 2012.
- [27] A. Gupta and Y. S. Ong, “Back to the roots: Multi-x evolutionary computation,” *Cognitive Computation*, vol. 11, no. 1, pp. 1–17, 2019.
- [28] W. Xu, J. X. Xu, D. He, K. C. Tan, “An Evolutionary Constraint-Handling Technique for Parametric Optimization of a Cancer Immunotherapy Model,”

- IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 3, no. 2, pp. 151–62, 2019.
- [29] D. N. Perkins and G. Salomon, “Transfer of learning,” *International Encyclopedia of Education*, vol. 2, pp. 6452–6457, 1992.
- [30] L. Bai, Y. S. Ong, T. He, and A. Gupta, “Multi-task gradient descent for multi-task learning,” *Memetic Computing*, vol. 12, no. 4, pp. 355–369, 2020.
- [31] C. He, Y. Tian, H. Wang, and Y. Jin, “A repository of real-world datasets for data-driven evolutionary multiobjective optimization,” *Complex & Intelligent Systems*, vol. 6, pp. 189–197, 2020.
- [32] T. Chugh, K. Sindhya, J. Hakanen, and K. Miettinen, “A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms,” *Soft Computing*, vol. 23, no. 9, pp. 3137–3166, 2019.
- [33] K. M. Calhoun, R. F. Deckro, J. T. Moore, J. W. Chrissis, and J. C. Van Hove, “Planning and re-planning in project and production scheduling,” *Omega*, vol. 30, no. 3, pp. 155–170, 2002.
- [34] D. Ivanov, A. Pavlov, and B. Sokolov, “Optimal distribution (re) planning in a centralized multi-stage supply network under conditions of the ripple effect and structure dynamics,” *European Journal of Operational Research*, vol. 237, no. 2, pp. 758–770, 2014.
- [35] L. He, and P. Jiang, “A framework for planing and solving production problems with manufacturing knowledge reuse,” in *IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE)*, Apr. 2019, pp. 103–107.
- [36] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia, “A review of dynamic vehicle routing problems,” *European Journal of Operational Research*, vol. 225, no. 1, pp. 1–11, 2013.
- [37] K. K. H. Ng, C. K. Lee, S. Z. Zhang, K. Wu, and W. Ho, “A multiple colonies artificial bee colony algorithm for a capacitated vehicle routing problem and re-routing strategies under time-dependent traffic congestion,” *Computers &*

- Industrial Engineering*, vol. 109, pp. 151–168, 2017.
- [38] B. Maenhout, and M. Vanhoucke, “A perturbation matheuristic for the integrated personnel shift and task re-scheduling problem,” *European Journal of Operational Research*, vol. 269, no. 3, pp. 806–823, 2018.
- [39] B. Koçer and A. Arslan, “Genetic transfer learning,” *Expert Systems with Applications*, vol. 37, no. 10, pp. 6997–7002, 2010.
- [40] P. Cunningham and B. Smyth, “Case-based reasoning in scheduling: reusing solution components,” *International Journal of Production Research*, vol. 35, no. 11, pp. 2947–2962, 1997.
- [41] S. J. Louis and J. McDonnell, “Learning with case-injected genetic algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 4, pp. 316–328, 2004.
- [42] S. R. Martin, S. E. Wright, and J. W. Sheppard, “Offline and online evolutionary bi-directional RRT algorithms for efficient re-planning in dynamic environments,” in *IEEE International Conference on Automation Science and Engineering*, Sep. 2007, pp. 1131–1136.
- [43] M. Kaedi and N. Ghasem-Aghaee, “Biasing Bayesian optimization algorithm using case based reasoning,” *Knowledge-Based Systems*, vol. 24, no. 8, pp. 1245–1253, 2011.
- [44] M. Pelikan, M. W. Hauschild, and P. L. Lanzi, “Transfer Learning, Soft Distance-Based Bias, and the Hierarchical BOA,” in *International Conference on Parallel Problem Solving from Nature (PPSN)*, Sep. 2012, pp. 173–183.
- [45] S. Israel and A. Moshaiiov, “Bootstrapping aggregate fitness selection with evolutionary multi-objective optimization,” in *International Conference on Parallel Problem Solving from Nature (PPSN)*, Sep. 2012, pp. 52–61.
- [46] A. Moshaiiov and A. Tal, “Family bootstrapping: A genetic transfer learning approach for onsetting the evolution for a set of related robotic tasks,” in *IEEE Congress on Evolutionary Computation*, Jul. 2014, pp. 2801–2808.
- [47] M. Iqbal, W. N. Browne, and M. Zhang, “Extracting and using building blocks

- of knowledge in learning classifier systems,” in *Proceedings of the Genetic and Evolutionary Computation Conference*, Jul. 2012, pp. 863–870.
- [48] I. M. Alvarez, W. N. Browne, and M. Zhang, “Human-inspired scaling in learning classifier systems: Case study on the n-bit multiplexer problem set,” in *Proceedings of the Genetic and Evolutionary Computation Conference*, Jul. 2016, pp. 429–436.
- [49] R. Meuth, M. H. Lim, Y. S. Ong and D. C. Wunsch, “A proposition on memes and meta-memes in computing for higher-order learning,” *Memetic Computing*, vol. 1, no. 2, pp. 85–100, 2009.
- [50] M. Salamó and M. López-Sánchez, “Adaptive case-based reasoning using retention and forgetting strategies,” *Knowledge-Based Systems*, vol. 24, no. 2, pp. 230–247, 2011.
- [51] P. Smyth and D. Wolpert, “Stacked density estimation,” In *Advances in Neural Information Processing Systems*, Denver, Colorado, USA, 1998, pp. 668–674.
- [52] J. Zhang, W. Zhou, X. Chen, W. Yao, and L. Cao, “Multi-source selective transfer framework in multi-objective optimization problems,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, pp. 424–438, Jul. 2019.
- [53] 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*, vol. 24, no. 1, pp. 69–83, Feb. 2020.
- [54] X. Zheng, A. K. Qin, M. Gong, and D. Zhou, “Self-regulated evolutionary multi-task optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 1, pp. 16–28, Feb. 2020.
- [55] C. Lyu, Y. Shi, and L. Sun, “A Novel Multi-Task Optimization Algorithm Based on the Brainstorming Process,” *IEEE Access*, vol. 8, pp. 217134–217149, Dec. 2020.
- [56] L. Zhou, L. Feng, K. C. Tan, J. Zhong, Z. Zhu, K. Liu, and C. Chen, “Toward

- adaptive knowledge transfer in multifactorial evolutionary computation,” *IEEE Transactions on Cybernetics*, In Press.
- [57] L. Feng, Y. S. Ong, S. Jiang, and A. Gupta, “Autoencoding evolutionary search with learning across heterogeneous problems,” *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 5, 760–772, Oct. 2017.
- [58] K. K. Bali, A. Gupta, L. Feng, Y. S. Ong, and P. S. Tan, “Linearized domain adaptation in evolutionary multitasking,” in *IEEE Congress on Evolutionary Computation (CEC)*, 2017, pp. 1295–1302.
- [59] L. Feng, L. Zhou, J. Zhong, A. Gupta, Y. S. Ong, K. C. Tan, and A. K. Qin, “Evolutionary multitasking via explicit autoencoding,” *IEEE Transactions on Cybernetics*, vol. 49, no. 9, pp. 3457–3470, Sept. 2019.
- [60] L. Feng, Y. S. Ong, M. H. Lim, and I. W. Tsang, “Memetic search with interdomain learning: A realization between CVRP and CARP,” *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 5, pp. 644–658, Oct. 2015.
- [61] L. Feng, L. Zhou, A. Gupta, J. Zhong, Z. Zhu, K. C. Tan, and K. Qin, “Solving generalized vehicle routing problem with occasional drivers via evolutionary multitasking,” *IEEE Transactions on Cybernetics*, In Press.
- [62] M. A. Ardeh, Y. Mei and M. Zhang, “A Novel Genetic Programming Algorithm with Knowledge Transfer for Uncertain Capacitated Arc Routing Problem,” in *Pacific Rim International Conference on Artificial Intelligence*, Aug. 2019, pp. 196–200.
- [63] J. MacLachlan, Y. Mei, J. Branke, and M. Zhang, “Genetic programming hyper-heuristics with vehicle collaboration for uncertain capacitated arc routing problems,” *Evolutionary computation*, vol. 28, no. 4, pp. 563–593, 2020.
- [64] A. T. W. Min, R. Sagarna, A. Gupta, Y. S. Ong, and C. K. Goh, “Knowledge transfer through machine learning in aircraft design,” *IEEE Computational Intelligence Magazine*, vol. 12, no. 4, pp. 48–60, Nov. 2017.
- [65] C. Yang, J. Ding, Y. Jin, C. Wang, and T. Chai, “Multitasking multiobjective

- evolutionary operational indices optimization of beneficiation processes,” *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 3, pp. 1046–1057, Jul. 2019.
- [66] A. T. W. Min, Y. S. Ong, A. Gupta, and C. K. Goh, “Multiproblem surrogates: transfer evolutionary multiobjective optimization of computationally expensive problems,” *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 1, pp. 15–28, Feb. 2019.
- [67] Z. Wang, X. Wang, “Multiobjective multifactorial operation optimization for continuous annealing production process,” *Industrial & Engineering Chemistry Research*, vol. 58, no. 41, pp. 19166–19178, 2019.
- [68] R. Sagarna and Y. S. Ong, “Concurrently searching branches in software tests generation through multitask evolution,” in *IEEE Symposium Series on Computational Intelligence*, Dec. 2016, pp. 1–8.
- [69] Y. Bi, B. Xue, and M. Zhang, “An effective feature learning approach using genetic programming with image descriptors for image classification [Research Frontier],” *IEEE Computational Intelligence Magazine*, vol. 15, no. 2, pp. 65–77, May 2020.
- [70] Q. U. Ain, H. Al-Sahaf, B. Xue, and M. Zhang, “Generating knowledge-guided discriminative features using genetic programming for melanoma detection,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, In Press.
- [71] K. Deb, A. Pratap, S. Agarwal, and T. A. M. T. Meyarivan, “A fast and elitist multiobjective genetic algorithm: NSGA-II,” *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
- [72] L. M. Pang, H. Ishibuchi, and K. Shang, “NSGA-II With Simple Modification Works Well on a Wide Variety of Many-Objective Problems,” *IEEE Access*, vol. 8, pp. 190240–190250, Oct. 2020.
- [73] E. Zitzler, M. Laumanns, and L. Thiele, “SPEA2: Improving the strength Pareto evolutionary algorithm,” *TIK-report*, vol. 103, 2001.
- [74] S. Larraín, L. Pradenas, I. Pulkkinen, and F. Santander, “Multiobjective

- optimization of a continuous kraft pulp digester using SPEA2,” *Computers & Chemical Engineering*, vol. 143, 107086, 2020.
- [75] 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, Dec. 2007.
- [76] T. C. Wang, R. T. Liaw, and C. K. Ting, “MOEA/D using covariance matrix adaptation evolution strategy for complex multi-objective optimization problems,” in *IEEE Congress on Evolutionary Computation (CEC)*, 2016, pp. 983–990.
- [77] K. Li, M. Liao, K. Deb, G. Min, and X. Yao, “Does Preference Always Help? A Holistic Study on Preference-Based Evolutionary Multi-Objective Optimisation Using Reference Points,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 6, pp. 1078–1096, Dec. 2020.
- [78] O. R. Castro, A. Pozo, J. A. Lozano, and R. Santana, “Transfer weight functions for injecting problem information in the multi-objective CMA-ES,” *Memetic Computing*, vol. 9, no. 2, pp. 153–180, 2017.
- [79] A. Gupta, Y. S. Ong, L. Feng, and K. C. Tan, “Multiobjective multifactorial optimization in evolutionary multitasking,” *IEEE Transactions on Cybernetics*, vol. 47, no. 7, pp. 1652–1665, Jul. 2017.
- [80] H. T. T. Binh, N. Q. Tuan, and D. C. T. Long, “A multi-objective multi-factorial evolutionary algorithm with reference-point-based approach,” in *IEEE Congress on Evolutionary Computation*, 2019, pp. 2824–2831.
- [81] K. K. Bali, A. Gupta, Y. S. Ong, and P. S. Tan, “Cognizant multitasking in multiobjective multifactorial evolution: MO-MFEA-II,” *IEEE Transactions on Cybernetics*, vol. 51, no. 4, pp. 1784–1796, Apr. 2021.
- [82] C. Yang, J. Ding, Y. Jin, and T. Chai, “Offline data-driven multiobjective optimization: Knowledge transfer between surrogates and generation of final solutions,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, pp. 409–423, Jun. 2020.

- [83] J. Lin, H. L. Liu, B. Xue, M. Zhang, and F. Gu, "Multi-objective multi-tasking optimization based on incremental learning," *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 5, pp. 824–838 Oct. 2020.
- [84] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.
- [85] M. Shakeri, A. Gupta, Y. S. Ong, C. Xu, A. N. Zhang, "Coping with Big Data in Transfer Optimization," In *IEEE International Conference on Big Data*, Dec. 2019, pp. 3925–3932.
- [86] S. Sun, H. Shi, Y. Wu, "A survey of multi-source domain adaptation," *Information Fusion*, vol. 24, pp. 84–92, 2015.
- [87] M. S. Krejca and C. Witt, "Theory of estimation-of-distribution algorithms," *Theory of Evolutionary Computation*, Cham, Switzerland, Springer, pp. 405–442, 2020.
- [88] 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, Apr. 2004.
- [89] S. Baluja and S. Davies, "Fast probabilistic modeling for combinatorial optimization," In *Proceedings of the AAAI Conference on Artificial Intelligence*, Jul. 1998, pp. 469–476.
- [90] L. Feng, A. Gupta, and Y. S. Ong, "Compressed representation for higher-level meme space evolution: a case study on big knapsack problems," *Memetic Computing*, vol. 11, no. 1, pp. 3–17, 2019.
- [91] G. B. Dantzig, "Discrete-Variable Extremum Problems," *Operations Research*, vol. 5, no. 2, pp. 266–288, 1957.
- [92] H. Mühlenbein, "The equation for response to selection and its use for prediction," *Evolutionary Computation*, vol. 5, no. 3, pp. 303–346, 1997.
- [93] Z. Michalewicz and J. Arabas, "Genetic algorithms for the 0/1 knapsack problem," in *Proceedings of the International Symposium on Methodologies for Intelligent Systems*, Oct. 1994, pp. 134–143.

- [94] X. Xue, K. Zhang, K. C. Tan, L. Feng, J. Wang, G. Chen, X. Zhao, L. Zhang and J. Yao, "Affine Transformation-Enhanced Multifactorial Optimization for Heterogeneous Problems," *IEEE Transactions on Cybernetics*, 2020.
- [95] N. Y. Liang, G. B. Huang, P. Saratchandran and N. Sundararajan, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE Transactions on Neural Networks*, vol. 17, no. 6, pp. 1411–1423, 2006.
- [96] G. Hendeby and F. Gustafsson, "On nonlinear transformations of Gaussian distributions," *Technical Report from Automatic Control*, Linköping University, Sweden, 2007.
- [97] M. Locatelli, "A note on the Griewank test function," *Journal of Global Optimization*, vol. 25, no. 2, pp. 169–174, 2003.
- [98] B. Da, Y. S. Ong, L. Feng, A. K. Qin, A. Gupta, Z. Zhu, C. K. Ting, K. Tang, and X. Yao, "Evolutionary multitasking for single-objective continuous optimization: Benchmark problems, performance metric, and baseline results," *CoRR*, abs/1706.03470, 2017.
- [99] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters and J. Schmidhuber, "Natural evolution strategies," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 949–980, 2014.
- [100] W. J. Gomes, A. T. Beck, R. H. Lopez and L. F. Miguel, "A probabilistic metric for comparing metaheuristic optimization algorithms," *Structural Safety*, vol. 70, pp. 59–70, 2018.
- [101] S. Chartnuyom, M. K. Lee, L. Luong and R. Marian, "Multi-location inventory system with lateral transshipments and emergency orders," in *IEEE International Conference on Industrial Engineering and Engineering Management*, Dec 2007, pp. 1594–1598.
- [102] K. Taha, "Methods That Optimize Multi-Objective Problems: A Survey and Experimental Evaluation," *IEEE Access*, vol. 8, pp. 80855–80878, Apr. 2020.
- [103] Z. Wang, Y. S. Ong, and H. Ishibuchi, "On scalable multiobjective test problems with hardly dominated boundaries," *IEEE Transactions on*

- Evolutionary Computation*, vol. 23, no. 2, pp. 217–231, Apr. 2019.
- [104] C. A. Coello, “An updated survey of GA-based multiobjective optimization techniques,” *ACM Computing Surveys (CSUR)*, vol. 32, no. 2, pp. 109–143, 2000.
- [105] C. A. C. Coello, G. B. Lamont, and D. A. Van Veldhuizen, *Evolutionary algorithms for solving multi-objective problems*, New York: Springer, vol. 5, pp. 79–104, 2007.
- [106] 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 multiobjective continuous optimization: Benchmark problems, performance metrics and baseline results,” *Technical Report*, 2017. *arXiv preprint arXiv:1706.02766*.
- [107] S. Jiang, Y. S. Ong, J. Zhang, and L. Feng, “Consistencies and contradictions of performance metrics in multiobjective optimization,” *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2391–2404, Dec. 2014.
- [108] X. Liao, Q. Li, X. Yang, W. Zhang, and W. Li, “Multiobjective optimization for crash safety design of vehicles using stepwise regression model,” *Structural and multidisciplinary optimization*, vol. 35, no. 6, pp. 561–569, 2008.
- [109] K. Dorling, J. Heinrichs, G. G. Messier, and S. Magierowski, “Vehicle routing problems for drone delivery,” *IEEE Transactions on Systems, Man, and Cybernetics, Systems*, vol. 47, no. 1, pp. 70–85, Jan. 2017.
- [110] B. L. Golden, S. Raghavan, and E. A. Wasil, *The Vehicle Routing Problem: Latest Advances and New Challenges*, vol. 43, Springer Science & Business Media, 2008.
- [111] I. Hong, M. Kuby, and A. T. Murray, “A range-restricted recharging station coverage model for drone delivery service planning,” *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 198–212, 2018.
- [112] R. Lim, Y. S. Ong, Hanh T. H. Phan, A. Gupta and A. N. Zhang, “Can route planning be smarter with transfer optimization?,” In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, Jul. 2019, pp. 318–319.

- [113] K. C. Tan, L. H. Lee, Q. L. Zhu, K. Ou, "Heuristic method for vehicle routing problem with time windows," *Artif. Intell. Eng.*, vol. 15, pp. 281–295, 2001.
- [114] H. Coxeter, *Introduction to geometry*. New York: John Wiley & Sons, 1969.
- [115] S. Tsutsui, "Probabilistic model-building genetic algorithms in permutation representation domain using edge histogram," in *International Conference on Parallel Problem Solving from Nature*, Sept. 2002, pp. 224–233.
- [116] N. Christofides, A. Mingozzi and P. Toth, "The vehicle routing problem," in: *Combinatorial Optimization*, eds. N. Christofides, N. Mingozzi, A. Toth and C. Sandi (Wiley, Chichester, 1979).
- [117] B. L. Golden, E. A. Wasil, J. P. Kelly, and I. M. Chao, "The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results," in *Fleet management and logistics*. Springer, Boston, MA, pp. 33–56, 1998.
- [118] E. Uchoa, D. Pecin, A. Pessoa, M. Poggi, T. Vidal, and A. Subramanian, "New benchmark instances for the capacitated vehicle routing problem," *Eur. J. Oper. Res.*, vol. 257, no. 3, pp. 845–858, 2017.
- [119] S. Tsutsui and G. Wilson, "Solving capacitated vehicle routing problems using edge histogram based sampling algorithms," in *IEEE Congress on Evolutionary Computation (CEC)*, 2004, pp. 1150–1157.
- [120] H. Nazif and L. S. Lee, "Optimised crossover genetic algorithm for capacitated vehicle routing problem," *Applied Mathematical Modelling*, vol. 36, no. 5, pp. 2110–2117, 2012.
- [121] A. T. W. Min, A. Gupta and Y. S. Ong, "Generalizing Transfer Bayesian Optimization to Source-Target Heterogeneity," *IEEE Transactions on Automation Science and Engineering*, 2020. In Press.
- [122] Y. Zhai, Y. S. Ong, and I. W. Tsang, "The emerging "big dimensionality"," *IEEE Computational Intelligence Magazine*, vol. 9, no. 3, pp. 14–26, 2014.
- [123] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, and A. Ghalsasi, "Cloud

- computing—The business perspective,” *Decision support systems*, vol. 51, no. 1, pp. 176–189, 2011.
- [124] J. Xu, E. Huang, C. H. Chen, and L. H. Lee, “Simulation optimization: A review and exploration in the new era of cloud computing and big data,” *Asia-Pacific Journal of Operational Research*, vol. 32, no. 3, 1550019, 2015.
- [125] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, “Imagenet: A large-scale hierarchical image database,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2009, pp. 248–255.
- [126] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [127] E. Kodirov, T. Xiang, S. Gong, “Semantic autoencoder for zero-shot learning,” In *IEEE Conference on Computer Vision and Pattern Recognition*, Jul. 2017, pp. 3174–3183.
- [128] L. Zhou, L. Feng, J. Zhong, Z. Zhu, B. Da, Z. Wu, “A study of similarity measure between tasks for multifactorial evolutionary algorithm,” In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, Jul. 2018, pp. 229–230.
- [129] W. Y. Deng, A. Lendasse, Y. S. Ong, I. W. Tsang, L. Chen, Q. H. Zheng, “Domain adaption via feature selection on explicit feature map,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 4, pp.1180–1190, 2019.