國立臺灣大學電機資訊學院電機工程學系

碩士論文

Department of Electrical Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

探討多目標組合優化中目標間的交互關係：多目標基因池最佳混合進化式演算法的加強變體

Investigation on Inter-relationship among Objectives for Multi-objective Combinatorial Optimization: an Enhanced Variant of MO-GOMEA

廖勗辰

Hsu-Chen Liao

指導教授: 于天立 博士

Advisor: Tian-Li Yu, Ph.D.

中華民國 112 年 7 月

July, 2023

doi:10.6342/NTU202303309

Acknowledgements

首先，我要特別感謝我的實驗室同學。感謝文忠和柏維，我們在同一研究領域中相互學習、互相幫助，使我能夠更好地理解與解決問題。我也要感謝柏儒和容均，他們總是在我需要打氣的時候給予我支持，讓我不會感到孤單或無力。另外，我要感謝麒仙學姊和庭碩學長，他們負責照料實驗室的各種事務，保證了我們工作的順利進行。

我也要向我的指導教授于天立教授表達我最誠摯的感謝。您的指導讓我在進行研究時能從多種角度思考問題，這對我的學術成長產生了深遠的影響。從您身上，我學到了如何以開放和創新的視角看待事物，這對我日後的研究有著無法估量的價值。您對人生的獨特見解，也讓我重新思考了我的生活方式和價值觀。

對於我的家人和女朋友，我也想表示最深的謝意。我的家人一直都在默默地支持我，他們的愛護和理解讓我能夠全心投入到這項研究中。我的女朋友，她總是在我側，提供我需要的支持，使我能夠在這個研究旅程中保持堅定不移的決心。

每一位在這裡提到的人都為我完成這項研究做出了巨大的貢獻。你們的支持和努力我將永遠記住。再次表示我最深的感謝。

doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

摘要

多目標基因池最佳混合進化式演算法 (MO-GOMEA) 是一種強大且無需參數的基於模型的基因演算法，擅長解決多目標組合優化問題。這篇研究論文從雙目標優化問題獲取啟發，進而提出自適應捐贈者選擇混合（ADSM）機制，此機制的構思是基於目標間錯綜複雜的相互關係。此機制設計特別針對群集導向和菁英導向混合對於目標在雙目標優化中的一致性、孤立性和不一致性改善的影響進行深入探討。將 ADSM 整合到 MO-GOMEA 中後，產生了一個新的變體，即 ADSM-MO-GOMEA。對多種基準問題的實證評估顯示，ADSM-MO-GOMEA 在反轉世代距離（IGD）和前沿佔用率（FO）兩個指標上均優於原始的 MO-GOMEA。在 ADSM-MO-GOMEA 超越 MO-GOMEA 的問題中，我們觀察到 IGD 指標平均提升約 10%，且在多目標背包問題的 FO 指標上也有顯著的提升。此外，本研究將範疇進一步擴大至涵蓋三至五個目標的問題，並考慮群集導向與菁英導向混合在改善目標一致性上的差異。在這種情況下，一致性的程度由顯示改善的目標數量決定。此延伸研究的實證結果揭示了，在處理具有更多目標的優化問題時，將一致性概念與修訂的 ADSM 結合起來將能有效改進原始 MO-GOMEA 的演化效率。

關鍵字：多目標最佳化、基因遺傳演算法、建構模塊、供體選擇

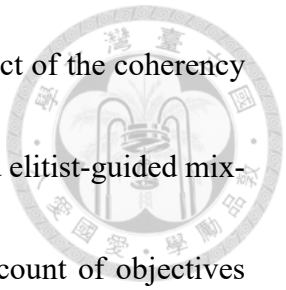
doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

Abstract

The multi-objective gene-pool optimal mixing evolutionary algorithm (MO-GOMEA) is a powerful, parameterless genetic algorithm that effectively solves multi-objective combinatorial optimization problems. Drawing insights from bi-objective optimization, this study introduced the adaptive donor selection mixing (ADSM) mechanism, devised in view of the intricate inter-relationship among objectives. It particularly focuses on the effects of cluster-guided and elitist-guided mixing on three types of improvements in objectives: coherent, solitary, and incoherent. The integration of ADSM into MO-GOMEA leads to the development of ADSM-MO-GOMEA. Empirical assessments on varied benchmark problems underscore that ADSM-MO-GOMEA outperforms its predecessor MO-GOMEA, excelling in both inverted generational distance (IGD) and front occupancy (FO) metrics. An average enhancement of about 10% in the IGD metric is observed where ADSM-MO-GOMEA outperforms the original, alongside a significant boost in the FO metric for the multi-objective knapsack problem. The research scope is further extended

doi:10.6342/NTU202303309



to tackle problems with three to five objectives, factoring in the impact of the coherency of improvement across objectives when employing cluster-guided and elitist-guided mixing. In this context, the degree of coherence is determined by the count of objectives that exhibit improvement. The empirical results from this broader study underscore the advantages of embedding the coherence concept within the refined ADSM in addressing optimization problems with a greater count of objectives.

Keywords: Multi-objective optimization, Model building, Genetic algorithm, Donor selection

doi:10.6342/NTU202303309

Contents

doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

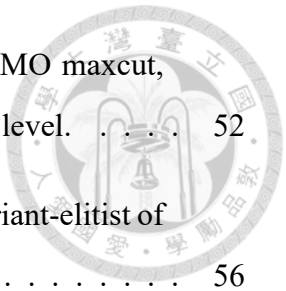
doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

List of Figures

doi:10.6342/NTU202303309

doi:10.6342/NTU202303309



doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

List of Tables

doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

Chapter 1 Introduction

Multi-objective optimization stands as a key research field centered on identifying a collection of Pareto-optimal solutions that provide a harmonious compromise between various objectives. This field finds extensive applications across diverse domains, encompassing engineering [20], operations research [1], and finance [31], among others. The presence of multiple competing objectives further complicates the resolution of these problems, as it becomes arduous to ascertain one optimal solution. Consequently, the pursuit of a set of non Pareto-dominated solutions is a viable approach, representing an optimal compromise among the competing objectives.

Research into high-dimensional multi-objective combinatorial optimization (MOCO) has been a prominent scholarly endeavor since the early 2000s [15, 26]. With the escalating deceptive characteristics that emerge with increasing problem sizes, researchers have centered their attention on harnessing the presence of nearly decomposable characteristics inherent in such problems for effective problem-solving strategies. Luong *et al.* proposed multi-objective gene-pool optimal mixing evolutionary algorithm (MO-GOMEA) [18, 19], incorporating the one of the latest model-building methodologies and corresponding recombination for MOCO. MO-GOMEA incorporates an elitist archive [17] to sustain the current approximation of the Pareto front, and it employs a clustering method akin to k-means clustering for segmenting the population. Following this, MO-GOMEA

doi:10.6342/NTU202303309

discerns inter-variable dependencies within each respective cluster by utilizing a linkage model and applies the variants of gene-pool optimal mixing (GOM) to each cluster individually. Empirical findings highlight the robust performance of MO-GOMEA across numerous metrics across various metrics in comparison to previous multi-objective model-building algorithms [26]. Moreover, it exhibits a scalability advantage for high-dimensional problems, outperforming binary versions of two advanced real-valued multi-objective optimization algorithms, multi-objective evolutionary algorithm based on decomposition (MOEA/D) [39] and non-dominated sorting genetic algorithm II (NSGA-II) [10]. These results highlight the critical role of exploiting variable dependencies to identify building blocks as potential partial solutions for high-dimensional problems [23, 34, 42]. Thus, taking building blocks into account collectively in recombination is crucial for advancing the existing approximation of the Pareto front. Although in single-objective optimization there has been substantial exploration of the contribution of building blocks [25, 32, 36], there remains scope for further investigation into how these building blocks affect objectives in the context of multi-objective optimization.

This thesis introduces a new model-based recombination mechanism designed for bi-objective scenarios, optimizing solutions through the exploitation of building blocks and their contributions to objective values. Guided by anticipated evolving regions for offspring and the current stage of evolution, the strategy for selecting donors within this mechanism effectively guides the progression towards the Pareto front. Incorporating this mixing mechanism into the framework of MO-GOMEA, we introduce a new variant. Empirical evaluations on benchmark problems underscore the effectiveness of our approach to achieve results that exhibit increased diversity and proximity to the Pareto front within a limited number of function evaluations (NFEs). Additionally, our extended study on

doi:10.6342/NTU202303309

problems with three to five objectives further supports the mixing strategy's enhancement over MO-GOMEA.

The thesis is organized as follows. Chapter 2 serves as an introduction to multi-objective combinatorial problems, examining benchmarks for high-dimensional scenarios and corresponding metrics, and also provides a synopsis of the MO-GOMEA with interleaved multi-start scheme (IMS). Chapter 3 presents the motivation and the construction process of the new mixing mechanism, detailing its structure and the way it is integrated into MO-GOMEA. Chapter 4 subsequently exhibits a series of empirical results and discussion on a variety of problem sets. Chapter 5 extends the research on the applicability of this mechanism to problems with three to five objectives. Finally, Chapter 6 concludes the thesis by summarizing the findings, presenting conclusions, and suggesting future research directions.

doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

Chapter 2 Background

This chapter serves as an introduction to MOCO, a challenging problem in high-dimensional multi-objective optimization. It discusses the relevant metrics, problem formulations, and existing solutions, with a specific focus on multi-objective genetic algorithms (MOGAs). Furthermore, an overview of the latest best-performing MO-GOMEA with IMS is provided, highlighting key components such as the elitist archive, k-leader-means clustering, IMS, and the model-based recombination mechanism. The recombination mechanism, encompassing critical components like GOM and forced improvement (FI), holds significant importance in this thesis, and its detailed workings are discussed extensively.

2.1 Multi-objective Combinatorial Optimization

MOCO refers to a specialized area of optimization that tackles problems characterized by combinatorial decision spaces and multiple objectives. These problems can be found in various domains, such as engineering, economics, social sciences, and computer science. MOCO problems, similar to other multi-objective optimization problems, present challenges due to the existence of multiple objectives that often conflict and compete with each other. For instance, in multi-objective knapsack problems, the objectives may in-

doi:10.6342/NTU202303309

volve maximizing the total value of selected items while minimizing the total weight. These conflicting objectives make the search for optimal solutions complex. Additionally, MOCO problems typically involve larger problem sizes with a high number of variables, which can introduce deceptive characteristics that hinder the effectiveness of MOGAs in finding optimal solutions.

2.1.1 Problem Formulation

Without loss of generality, a MOCO problem is defined as follows:

$$\underset{\mathbf{x}\in\mathcal{X}}{\text{maximize}} \quad \mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})]\,, \tag{2.1}$$

where:

- $\mathbf{x}$ is a specific solution or a decision variable vector.
- $\mathcal{X}$ is the combinatorial solution space, the set of all possible solutions $\mathbf{x}$.
- m is the number of objective functions.
- $f_i : \mathcal{X} \to \mathbb{R}$, for $i = 1, 2, \ldots, m$, are the objective functions. Each function f_i maps a solution $\mathbf{x} \in \mathcal{X}$ to a real number $f_i(\mathbf{x}) \in \mathbb{R}$ that quantifies the solution's quality concerning the i-th objective.

Solving a MOCO problem not only involves finding solutions that optimize the objectives but also requires these solutions to satisfy certain constraints. These constraints restrict the solution space, creating a feasible region $\mathcal{F}$ defined as:

doi:10.6342/NTU202303309

$$\mathcal{F} = \{\mathbf{x} \in \mathcal{X} | g_j(\mathbf{x}) \leq 0, \forall j\}\,, \tag{2.2}$$

where $g_j(\mathbf{x})$ are the constraint functions. Any solution that does not satisfy these constraints is not considered valid, irrespective of its performance on the objectives.

In the context of MOCO, a solution **x** is said to dominate another solution **y** (denoted as $\mathbf{x} \succ \mathbf{y}$) if and only if **x** is not worse than **y** in all objectives and **x** is strictly better than **y** in at least one objective:

$$\mathbf{x} \succ \mathbf{y} \iff (\forall i, f_i(\mathbf{x}) \geq f_i(\mathbf{y})) \wedge (\exists i, f_i(\mathbf{x}) > f_i(\mathbf{y}))\,. \tag{2.3}$$

The set of all non-dominated solutions within the solution space constitutes the Pareto set, denoted $\mathcal{PS}$:

$$\mathcal{PS} = \{\mathbf{x} \in \mathcal{F} |\ \nexists \mathbf{y} \in \mathcal{F}, \mathbf{y} \succ \mathbf{x}\}\,. \tag{2.4}$$

These solutions are mapped onto the objective space, resulting in what is commonly referred to as the Pareto front. Each point on the Pareto front represents an optimal balance among the objectives, none of which is categorically superior to the others. Consequently, the goal of solving MOCO problems is to identify the Pareto front, *i.e.*, to delineate the collection of potential solutions in the objective space that are not surpassed by any other feasible alternatives. These solutions represent the trade-offs among the objectives and provide the decision-maker with a set of options to choose from, according to their preferences.

doi:10.6342/NTU202303309

2.1.2 Performance Metrics

In order to evaluate the quality of the approximate Pareto front obtained by MOCO algorithms, various performance metrics have been proposed. In this section, we introduce two widely used metrics, front occupation (FO) and inverted generational distance (IGD).

FO is a metric that represents the number of solutions in the approximation set. A higher FO value is preferable as it indicates a more diverse and representative approximation set. Mathematically, FO can be expressed as follows:

$$FO(\mathcal{S}) = |\mathcal{S}| \,, \tag{2.5}$$

where $\mathcal{S}$ is the approximation set.

IGD is a performance metric that considers both accuracy and well-spreadness. It calculates the average distance in the objective space between a solution in the ground truth Pareto front and its closest approximation set solution. A lower IGD score is preferred, as it implies a more precise and diverse approximation set. Mathematically, IGD can be expressed as follows:

$$IGD(\mathcal{S},\, \mathcal{PF}) = \frac{1}{|\mathcal{PF}|} \sum_{p \in \mathcal{PF}} d(p, \mathcal{S}) \,, \tag{2.6}$$

where $\mathcal{PF}$ is the true Pareto front, $\mathcal{S}$ is the approximation set obtained by the algorithm, and $d(p, \mathcal{S})$ is the distance between solution p in $\mathcal{PF}$ and the closest solution in $\mathcal{S}$ in the objective space.

Overall, these two metrics offer quantitative measures to evaluate the quality of the approximation set. A lower IGD value indicates a closer approximation to the true Pareto

doi:10.6342/NTU202303309

front, and this metric is generally prioritized when comparing algorithms. In situations where multiple algorithms have similar IGD values, the FO metric becomes relevant. A higher FO value suggests better coverage of the Pareto front, indicating that the algorithm captures a larger portion of the true Pareto front.

2.2 Approach to Multi-objective Combinatorial Optimization

There are various ways to solve MOCO problems, including mathematical programming, genetic algorithm, swarm intelligence, and more. In this work, we focus on using a genetic algorithm (GA) to solve MOCO problems.

2.2.1 Multi-objective Genetic Algorithm

MOGAs have been extensively utilized for solving MOCO problems. However, the mechanisms employed in MOGAs are typically designed for real-valued problems and need to be adapted for binary problems. To address this, algorithms such as strength Pareto evolutionary algorithm 2 (SPEA2) [40] and NSGA-II modify the crossover and parent selection mechanisms to better suit MOCO problems by considering the ranking of Pareto optimality in the objective space. Relevant research that specifically address MOCO can be found in [35]. Alternatively, MOEA/D and its successors explore different scalarizing functions to handle MOCO by decomposing the multi-objective problem into multiple single-objective problems. These approaches can be studied in works such as [12, 39]. It is important to note that these methodologies do not explicitly exploit the interrelationships among variables, which may limit their effectiveness for larger problem sizes.

doi:10.6342/NTU202303309

2.2.2 Multi-objective Model-building Genetic Algorithm

In order to address high-dimensional MOCO problems that often exhibit deception characteristics in each objective, MOGAs use model-building techniques to uncover the linkage information between variables, which was previously used in single-objective optimization [5, 7, 32, 38]. Two notable examples of such MOGAs are the multi-objective hierarchical Bayesian optimization algorithm (mohBOA) [26] and MO-GOMEA [18, 19].

MohBOA employs a non-dominated crowding clustering method inspired by NSGA-II to divide the population into clusters. Within each cluster, it uses a hierarchical Bayesian model [24] to capture the dependencies between decision variables and generate offspring from the model. This technique efficiently guides the search process towards the Pareto front of high-dimensional MOCO problems. Additionally, mohBOA introduces a deceptive benchmark problem, trap5-inverse trap5, within the multi-objective context.

Within MO-GOMEA, the integration of one of the state-of-the-art model-building techniques called linkage tree [5, 32], alongside its associated recombination operator known as GOM, is a notable feature. In binary high-dimensional problem contexts, MO-GOMEA exhibits superior performance when compared to SPEA2, NSGA-II, and MOEA/D, highlighting the importance of discovering variable dependencies to solve high-dimension MOCO problems.

2.3 MO-GOMEA

In this thesis, we propose a new mixing mechanism based on MO-GOMEA. Consequently, we offer a detailed review of the MO-GOMEA with IMS, as proposed in 2017.

doi:10.6342/NTU202303309

The objective of this section is to elucidate the four fundamental components of MO-GOMEA, namely elitist archive, clustering, IMS and recombination, as well as the overall framework of MO-GOMEA. For illustrative purposes, the MO-GOMEA workflow is depicted in Figure 2.1, and the corresponding algorithm can be found in Algorithm 1.

Algorithm 1: MO-GOMEA

$\mathcal{P}$: population, $\mathcal{P}_i$: the i-th chromosome in $\mathcal{P}$,
O_i : offspring generated by the i-th chromosome in $\mathcal{P}$, E: elitist archive,
input : k: number of clusters, p: population size, ℓ: problem size,
output: E
$E \leftarrow$ empty set
$\mathcal{P} \leftarrow$ initialize population with p and ℓ randomly
$E \leftarrow$ update Elitist Archive with $\mathcal{P}$
while ¬`ShouldTerminate` **do**
  $\{C_0, \cdots, C_{k-1}\} \leftarrow$ divide $\mathcal{P}$ into k clusters
  $\{L_0, \cdots, L_{k-1}\} \leftarrow$ learn linkage model for each cluster
  for $i \leftarrow 0$ **to** $p-1$ **do**
    $C_i \leftarrow$ get the cluster of $\mathcal{P}_i$
    if C_i is extreme cluster **then**
      $O_i,\ E \leftarrow$ perform single-objective GOM on $\mathcal{P}_i$ with L_i and update E
    else
      $O_i,\ E \leftarrow$ perform multi-objective GOM on $\mathcal{P}_i$ with L_i and update E
    if `ShouldInvokeFI` **then**
      $O_i,\ E \leftarrow$ perform FI on $\mathcal{P}_i$ with C_i and L_i and update E
  $\mathcal{P} \leftarrow \{O_0,\ O_1, \cdots,\ O_{p-1}\}$
return E

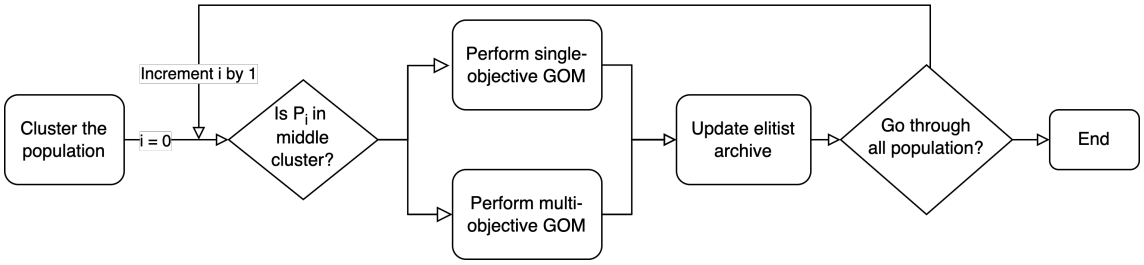


Figure 2.1: Flowchart of MO-GOMEA.

2.3.1 Elitist Archive

To handle the exponential size of the Pareto front, MO-GOMEA implements the adaptive grid discretization (AGD) version of the elitist archive. The original archive em-

doi:10.6342/NTU202303309

ploys rigid-grid discretization to divide the objective space into equal hypercubes, where each holds only one solution. New non-dominated solutions that occupy empty hypercubes or dominate existing solutions within the same hypercube are accepted, making the archive always a Pareto set. The AGD version introduces an archive size parameter and limits pruning frequency, making the archive more practical and computationally efficient while preserving its performance.

2.3.2 *K*-leader-means Clustering

To tackle the issue of variable dependency, MO-GOMEA incorporates the k-leader-means clustering technique, which allows for overlapping clusters. Initially, k leaders are selected based on a nearest-neighbor heuristic that maximizes their spread in the objective space. Then, the distances between each solution and the leaders are computed, and each cluster is assigned the c closest solutions. To increase the likelihood of overlapping clusters, MO-GOMEA sets the value of c to be $\frac{2}{k}|\mathcal{P}|$, where $\mathcal{P}$ represents the population.

By utilizing the k-leader-means clustering technique, MO-GOMEA effectively captures the linkage model of each segment of the population, considering the interdependencies among variables in a more accurate manner. After forming the clusters, they are subsequently classified into two categories: extreme clusters and middle clusters. Extreme clusters are identified as those exhibiting the highest mean objective value on each objective. Conversely, middle clusters encompass the remaining clusters that do not fall into the extreme category.

doi:10.6342/NTU202303309

2.3.3 Interleaved Multi-start Scheme

To achieve parameterless settings, MO-GOMEA employs the IMS scheme to manage population size and the number of clusters. IMS involves interleaving populations of varying sizes, with smaller populations starting earlier and running for more generations, while larger populations are introduced later. MO-GOMEA uses a population interleaved base of 2, beginning with an initial population $\mathcal{P}_0$ of size $N_0 = 8$ and cluster number $k_0 = m + 1$, where m is the number of objectives. Each successive population $\mathcal{P}_i$ follows the $[N_i = 2N_{i-1}, k_i = k_{i-1} + 1]$ scheme, doubling the population size N_i compared to the previous population and increasing the number of clusters k_i by one. During the optimization process, this parameterless approach adeptly adjusts the population size and the number of clusters. The workflow is shown at Figure 2.2 and Algorithm 2.

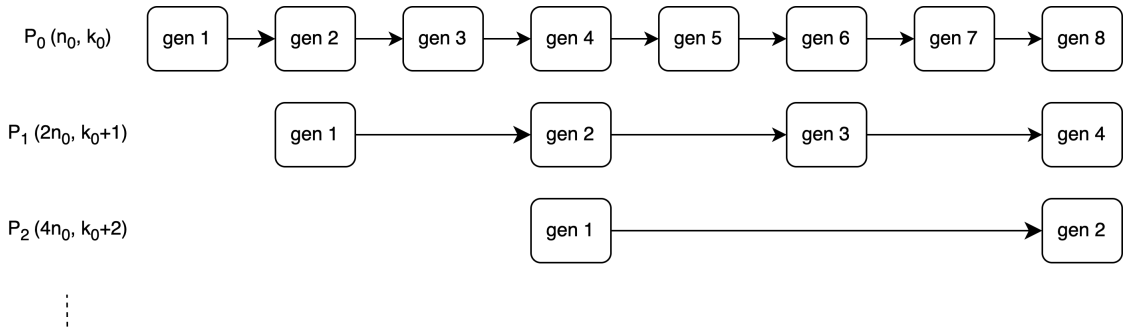


Figure 2.2: Flowchart of interleaved multi-start scheme. Each $\mathcal{P}_i(n_i, k_i)$ represents the i-th population of size n_i with k_i clusters. Populations commence at 2^i-th generation of $\mathcal{P}_0$ and proceed every 2^i generations, where 2 is the interleaved base.

2.3.4 Recombination

Within MO-GOMEA, two model-based mixing operators, namely GOM and FI, are employed. These operators utilize linkage models to guide the evolution process and are the focus of this thesis.

doi:10.6342/NTU202303309

Algorithm 2: Interleaved multi-start scheme

$\mathcal{P}_i$: i-th population,
g_i: generation of the i-th population,
input : b: interleaved base, N_0: initial population size,
k_0: initial cluster number
$\mathcal{P}_0 \leftarrow$ initialize population with N_0 and k_0
$g_0 \leftarrow 0$
$i \leftarrow 0$
while $\neg$`ShouldTerminate` **do**
  execute $\mathcal{P}_i$
  $g_i \leftarrow g_i + 1$
  if g_i *is divisible by* b **then**
    $i \leftarrow i + 1$
    if $i > currentPopulationNumber$ **then**
      $g_i \leftarrow 0$
      $\mathcal{P}_i \leftarrow$ initialize a new population with $2 \times N_{i-1}$ and $k_{i-1} + 1$
      $currentPopulationNumber \leftarrow currentPopulationNumber + 1$
  else
    $i \leftarrow 0$

At each generation, a linkage tree [5, 32] is constructed within each cluster to capture the interdependencies among the chromosomes and their variables. Subsequently, a family of subsets (FOS) [33] corresponding to the linkage tree provides candidate masks for the GOM mechanism. Donors are randomly chosen from within the receiver's cluster. In the extreme clusters, the original single-objective GOM [5, 6, 32] is applied to concentrate on evolving the corresponding objective. In the middle clusters, the receiver assesses each donor's mask from the FOS and adopts it under specific conditions: when the resulting offspring is Pareto-dominant to the receiver, aligns with the objectives of the receiver, or represents a novel non-dominated solution. This version of GOM is referred to as multi-objective GOM, and the algorithm for multi-objective GOM is outlined in Algorithm 3.

Alternatively, when neither single-objective nor multi-objective GOM succeeds in enhancing the receiver, the FI mechanism [5, 6, 18] is initiated. In this case, the model-based mixing framework remains unaltered, while the selection of donors is now directed from the elitist archive. Mixing operations are executed based on the linkage tree asso-

doi:10.6342/NTU202303309

ciated with the cluster of the receiver. Notably, the acceptance criteria for the resultant offspring deviate somewhat from the multi-objective GOM paradigm. Specifically, the resulting offspring is accepted it either forms a new non-dominated solution or it Pareto-dominates the receiver, which then concludes the FI process. Moreover, the FI mechanism is initiated if the elitist archive exhibits no alterations over a predetermined duration. The algorithm is outlined in Algorithm 3, and the whole workflow for recombination is also provided in Figure 2.3.

doi:10.6342/NTU202303309

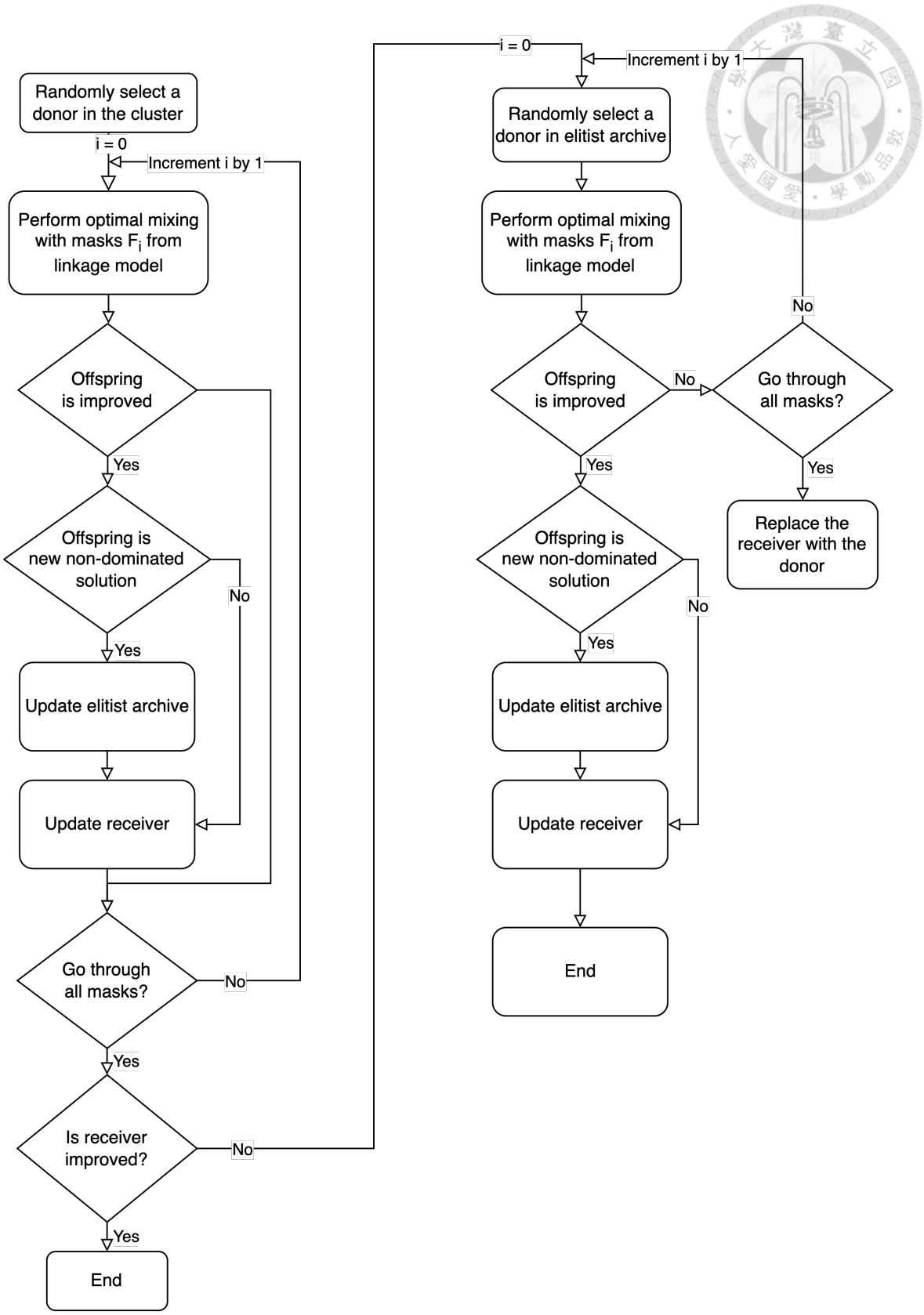


Figure 2.3: Flowchart of the recombination in MO-GOMEA.

doi:10.6342/NTU202303309

Algorithm 3: Multi-objective GOM

E: elitist archive, $\mathcal{P}$: population, ℓ : problem size,
input : C: cluster of donor,
L: linkage model,
R: receiver,
output: O
$\{F_0, \cdots, F_{\ell-1}\} \leftarrow$ get family of subsets from L
$O \leftarrow R$
for $i \leftarrow 0$ **to** $\ell - 1$ **do**
  $D \leftarrow$ randomly choose a donor from C with replacement
  $O \leftarrow$ adopt mask F_i from the receiver
  $f_O \leftarrow$ evaluate O using the objective function
  if *f_O meets the update condition of the elitist archive* **then**
    $E \leftarrow$ update elitist archive with O.
  if *O is not Pareto-dominated by E or O Pareto-dominates R or O and R have the same objective values* **then**
    $R \leftarrow O$
  else
    $O \leftarrow R$
return O

Algorithm 4: Forced improvement

E: elitist archive, $\mathcal{P}$: population, ℓ : problem size,
input : L: linkage model,
R: receiver,
output: O
$\{F_0, \cdots, F_{\ell-1}\} \leftarrow$ get family of subsets from L
$O \leftarrow R$
for $i \leftarrow 0$ **to** $\ell - 1$ **do**
  $D \leftarrow$ randomly choose a donor from E with replacement
  $O \leftarrow$ adopt mask F_i from the receiver
  $f_O \leftarrow$ evaluate O using the objective function
  if *f_O meets the update condition of the elitist archive* **then**
    $E \leftarrow$ update elitist archive with O.
  if *O is not Pareto-dominated by E or O Pareto-dominates R* **then**
    $R \leftarrow O$
    break the loop
  else
    $O \leftarrow R$
return O

doi:10.6342/NTU202303309

2.4 High-dimensional Multi-objective Combinatorial Optimization Benchmark Problems

In the context of high-dimensional MOCO, a number of benchmark problems have been introduced or adapted from single-optimization benchmark problems [18, 19, 26]. These include three scalable benchmark problems and two multi-objective non-deterministic polynomial-time complete (NPC) problems, thus effectively addressing various features such as deceptive characteristics, scalability, and real-world applicability.

2.4.1 Zeromax-onemax

The zeromax-onemax problem is a binary-coded problem, where the aim is to maximize the number of zeros and ones simultaneously in a binary string. It constitutes the first of the scalable benchmark problems, aiming to test the capability of algorithms in optimizing conflicting objectives. The objectives are given by:

$$f_{\text{onemax}}(\mathbf{x}) = \sum_{i=0}^{\ell-1} \mathbf{x}_i, \tag{2.7}$$

$$f_{\text{zeromax}}(\mathbf{x}) = \sum_{i=0}^{\ell-1} (1 - \mathbf{x}_i]) \,, \tag{2.8}$$

where $\mathbf{x}_i$ denotes the i^{th} bit in a binary string $\mathbf{x}$ and ℓ is the size of the problem.

doi:10.6342/NTU202303309

2.4.2 Trap5-inverse Trap5

The second scalable benchmark problem is trap5-inverse trap5, a multi-objective extension of the well-known trap5 problem [8]. The problem is defined as follows:

$$f_{\text{trap5}}(\mathbf{x}) = \sum_{i=0}^{\frac{\ell}{5}-1} f_{\text{trap5}}^{\text{sub}} \left(\sum_{j=5i}^{5i+4} \mathbf{x}_j \right), \tag{2.9}$$

$$f_{\text{inv-trap5}}(\mathbf{x}) = \sum_{i=0}^{\frac{\ell}{5}-1} f_{\text{inv-trap5}}^{\text{sub}} \left(\sum_{j=5i}^{5i+4} \mathbf{x}_j \right), \tag{2.10}$$

where $\mathbf{x}_i$ represents the i^{th} bit in a binary string and ℓ is the problem size. The subfunctions f^{sub}trap5 and f^{sub}inv-trap5 are defined as:

$$f_{\text{trap5}}^{\text{sub}}(u) = \begin{cases} 5, & \text{if } u = 5 \\ 4-u, & \text{if } u < 5 \end{cases}, \tag{2.11}$$

$$f_{\text{inv-trap5}}^{\text{sub}}(u) = \begin{cases} 5, & \text{if } u = 0 \\ u-1, & \text{if } u > 0 \end{cases}. \tag{2.12}$$

This problem is instrumental in evaluating an algorithm's proficiency of optimization algorithms in discerning the deceptive attributes and ability at detecting variable dependencies.

2.4.3 Leading Ones Trailing Zeros

The leading ones trailing zeros (LOTZ) problem is a binary-coded problem introduced to test the performance of algorithms with regard to conditional dependencies within

doi:10.6342/NTU202303309

the decision variables. The objectives are defined as:

$$f_{\text{LO}}(\mathbf{x}) = \sum_{i=0}^{\ell-1} \prod_{j=0}^{i} \mathbf{x}_j, \tag{2.13}$$

$$f_{\text{TZ}}(\mathbf{x}) = \sum_{i=0}^{\ell-1} \prod_{j=i}^{\ell-1} (1 - \mathbf{x}_j), \tag{2.14}$$

where $\mathbf{x}_i$ denotes the i^{th} bit in a binary string and ℓ is the size of the problem.

2.4.4 Multi-objective Knapsack

The multi-objective knapsack problem (MO knapsack) is one of the NPC problems utilized as a benchmark in this research. This problem holds significant practical implications, encapsulating scenarios such as capital budget management [28] and conservation strategies in ecology [14, 16]. The key challenge posed by the MO Knapsack problem is the identification of all feasible, non Pareto-dominated solutions, with the aim to maximize the profit within each knapsack without breaching the defined capacity limits. The problem is defined by the following objectives:

$$f_i(\mathbf{x}) = \sum_{j=0}^{\ell-1} p_{j,i}\mathbf{x}_j \quad \text{subject to} \sum_{j=0}^{\ell-1} w_{j,i}\mathbf{x}_j < c_i, \tag{2.15}$$

where $\mathbf{x}_j$ represents the j^{th} item in a binary string, ℓ is the size of the problem, $p_{j,i}$ is the profit of item j for knapsack i, $w_{j,i}$ is the weight of item j for knapsack i, and c_i is the capacity of knapsack i.

doi:10.6342/NTU202303309

2.4.5 Multi-objective Maxcut

The multi-objective maxcut problem (MO maxcut) is another NPC problem employed as a benchmark in this research. It presents a common graph partitioning task with myriad applications in fields such as very large-scale integration design and statistical physics [2]. The problem entails partitioning the vertices of a graph into two disjoint subsets, with the aim to maximize the sum of weights of the edges crossing the partition in a manner that simultaneously optimizes two objectives. It is defined by the following objectives:

$$f_i(\mathbf{x}) = \sum_{(v_m, v_n) \in E} \begin{cases} w_{(m,n,i)}, & \text{if } x_m \neq x_n \\ 0, & \text{otherwise} \end{cases}, \tag{2.16}$$

where $G = (V, E)$ is a graph formed by $\mathbf{x}$ and $w_{(m,n,i)}$ is the weight of edge (v_m, v_n) for maxcut i.

doi:10.6342/NTU202303309

doi:10.6342/NTU202303309

Chapter 3 The Proposed ADSM-MO-GOMEA

In this chapter, we commence by exploring the building block concept and its influence on the objectives in the context of multi-objective optimization. We subsequently assess and compare the effects of two different mixing strategies, elitist-guided and cluster-guided mixing, on the evolutionary process to gain insights into their distinct behavioral patterns. Additionally, we introduce mathematical derivations and an adaptive mechanism designed to enhance the performance. Building upon these findings, we put forward a new mixing mechanism and incorporate it into the framework of MO-GOMEA.

3.1 Investigation on Inter-relationship among Objectives

While MO-GOMEA's efficacy is widely recognized in multi-objective optimization, our empirical analyses hint at a tendency towards divergent evolution in its objective space. To quantify this divergence, we introduce a specific measure termed the angle of improvement and symbolized by θ_i. This measure is derived from the angle between the primary objective axis and the vector spanning from the receiver to the offspring, as depicted in Figure 3.1. To visualize the response of the successful improved offspring

doi:10.6342/NTU202303309

generated by MO-GOMEA to the angle of improvement, we present a histogram plot in Figure 3.2. The observed divergent distribution suggests that MO-GOMEA's evolution within the objective space is indeed characterized by divergence. It is essential to highlight that, for all the subsequent analyses, the objective values have been normalized to the range of $[0, 1]$. This normalization is implemented to alleviate any potential bias that could arise between the objectives.

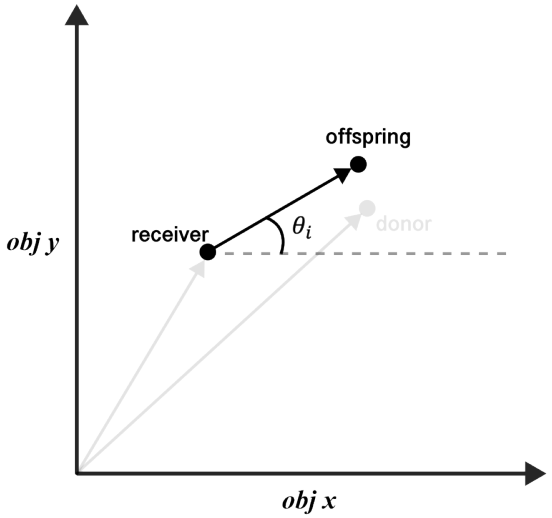


Figure 3.1: Visualization of angle of improvement, denoted as θ_i.

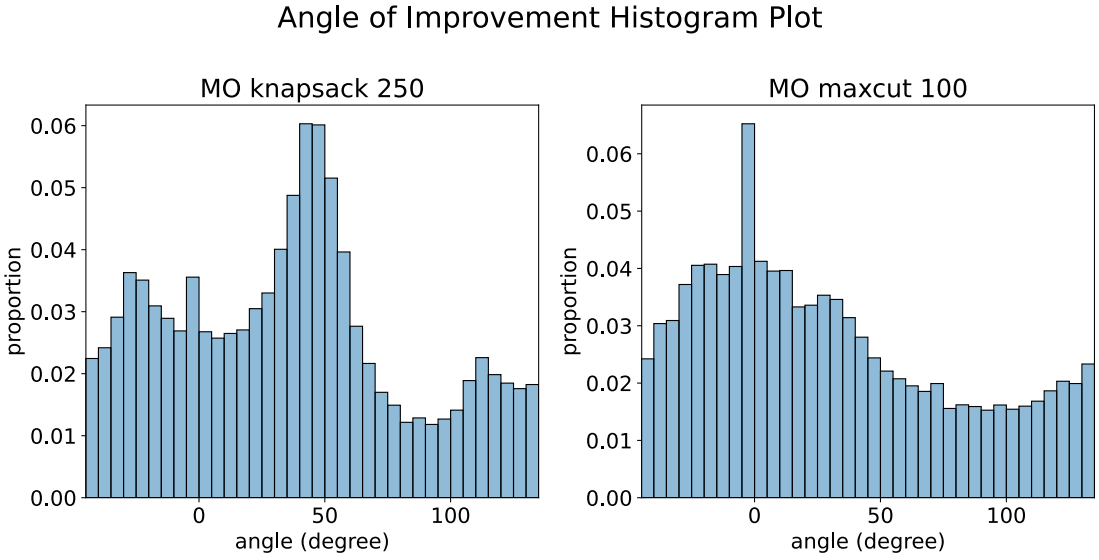


Figure 3.2: Distribution visualization for successful mixed offspring generaterd by MO-GOMEA with respect to the angle of improvement.

This observation motivates us to explore the potential advantages of incorporating a

doi:10.6342/NTU202303309

more directed evolution strategy, as depicted in Figure 3.3. Similar concepts of guided evolution can be found in real-valued multi-objective optimization, such as the utilization of reference points in algorithms like the non-dominated sorting genetic algorithm III [9]. We also draw inspiration from classical single-objective model-building genetic algorithms that have achieved success by leveraging promising genotypes derived from building blocks [25, 36]. Thus, we emphasize the importance of considering the contributions of building blocks in the multi-objective context, where exploiting the intricate inter-relationships between objectives can potentially enhance performance.

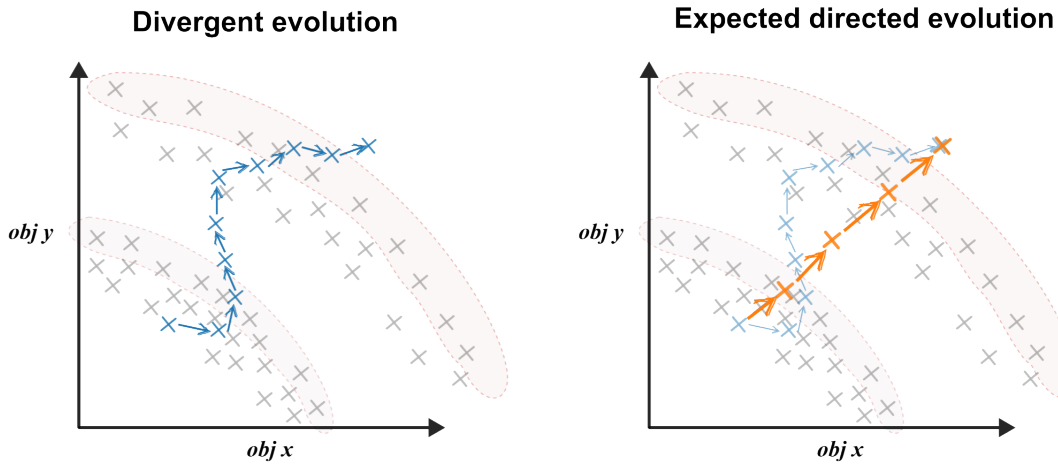


Figure 3.3: Example of divergent evolution and the contrasting concept of expected directed evolution.

3.2 Correlation between Building Blocks and Multi-objective Optimization

In the context of single-objective optimization, a clear positive connection has been observed between the building blocks and the fitness value, prompting the development of model-building genetic algorithms to tackle problems with decomposition characteristics [5, 11, 37]. Likewise, in mohBOA [26] and MO-GOMEA [18, 19], model-building

doi:10.6342/NTU202303309

techniques are utilized to address decomposable multi-objective problems, demonstrating a positive correlation between the building blocks and the progression of the Pareto front.

To elucidate this relationship, we propose a simplified problem with clear building blocks: optimizing an anti-virus system. This system aims to maximize virus coverage and minimize the time overhead of different system parts, given a specific cost constraint. Crucial factors in this scenario involve the diverse virus coverage offered by different brands and the increasing overhead from integrating system parts from multiple brands. In our model, a system includes database, server, and client components, with three brands (A, B, C) offering anti-virus solutions. We represent virus coverages using a chromosome c with genes $\{g1, g2, g3, g4, g5, g6, g7, g8, g9\}$, where each gene denotes the virus coverage of a specific brand for a particular system component, as outlined in Table3.1.

Table 3.1: Gene representation of virus coverage by brands across system components, where each g_i is paired with a set of viruses (v_x, v_y, v_z) that it covers.

	Brand A	**Brand B**	**Brand C**
Database	$g_1 : \{v_1, v_2, v_3\}$	$g_2 : \{v_2, v_3, v_4\}$	$g_3 : \{v_3, v_4, v_5\}$
Server	$g_4 : \{v_6, v_7, v_8\}$	$g_5 : \{v_7, v_8, v_9\}$	$g_6 : \{v_8, v_9, v_{10}\}$
Client	$g_7 : \{v_{11}, v_{12}, v_{13}\}$	$g_8 : \{v_{12}, v_{13}, v_{14}\}$	$g_9 : \{v_{13}, v_{14}, v_{15}\}$

Consequently, the genes from the same brands form the building blocks for the time objective, while the genes combining to cover more viruses become the building blocks for the coverage objective. Along the Pareto front, solutions combine these building blocks from both objectives, for example, two components from the same brand and one from a different one, as shown in Figure 3.4. This combination of building blocks is the only way to advance the front, hence establishing a new building block, further demonstrating the correlation between the building blocks and the progression of the Pareto front.

Building on this concept, we delve further into the examination of how the building blocks interact and contribute to the optimization process, exploring their inter-relationships

doi:10.6342/NTU202303309

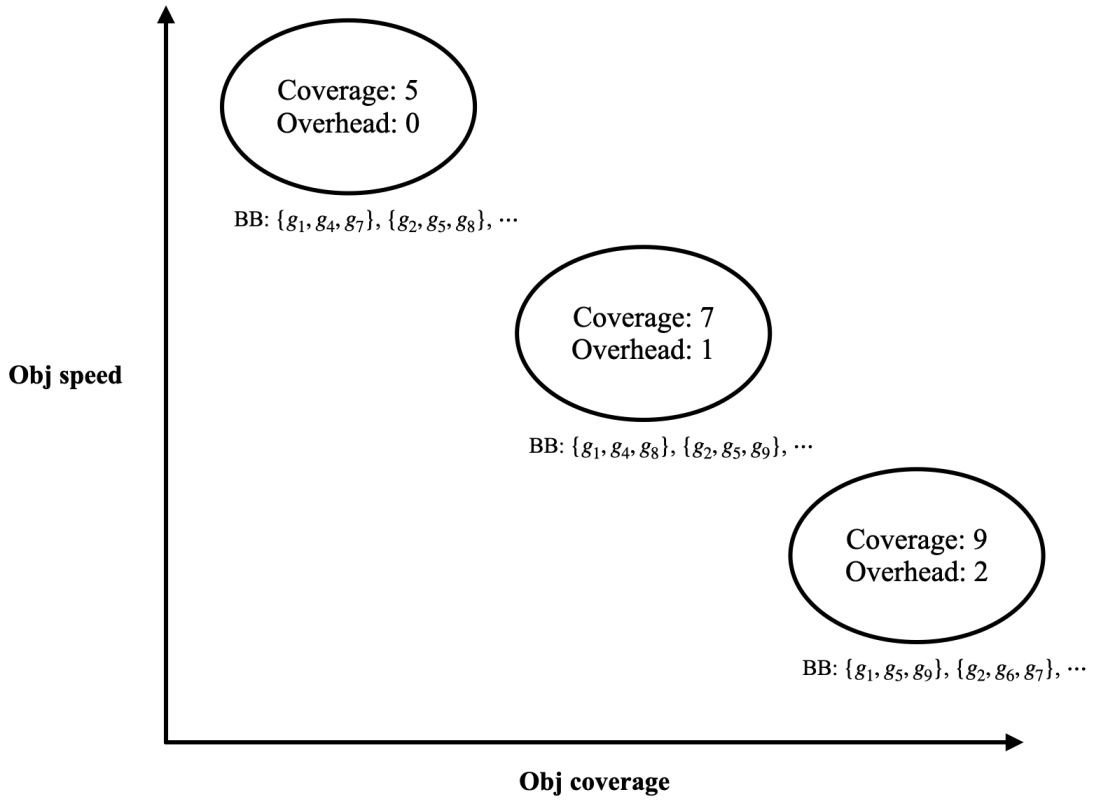


Figure 3.4: Pareto front for the anti-virus optimization, highlighting the differing composition of building blocks in the solutions along the front.

and implications for multi-objective optimization.

3.3 Cluster-guided Mixing and Elitist-guided Mixing

Leveraging the elitist archive, MO-GOMEA approximates the Pareto front, benefiting from its efficiency and ability to maintain a consistent size. Additionally, the elitist archive plays a critical role in overcoming stagnation through the use of FI. In this thesis, we suggest a new interpretation on the elitist archive, perceiving it as a reservoir of chromosomes that contain the currently favorable building blocks and their genotypes. We hypothesize that harnessing these beneficial genotypes can lead to a more directed and effective evolutionary progression.

doi:10.6342/NTU202303309

To investigate this idea further, we differentiate the mixing process into two categories: elitist-guided mixing, wherein the donor is sourced from the elitist archive, and cluster-guided mixing, where the donor is chosen from the same cluster as the receiver, as outlined in Figure 3.5. Consequently, we establish two distinct MO-GOMEA variants: the variant-cluster and the variant-elitist. In these variants, the recombination for the middle clusters is replaced. The variant-cluster exclusively employs multi-objective GOM, while the variant-elitist solely utilizes FI. With respect to the extreme clusters, we align the mixing mechanism with the previous research, utilizing the single-objective GOM. This strategy is validated by previous academic research that emphasizes the benefits of integrating a single-objective optimizer [4].

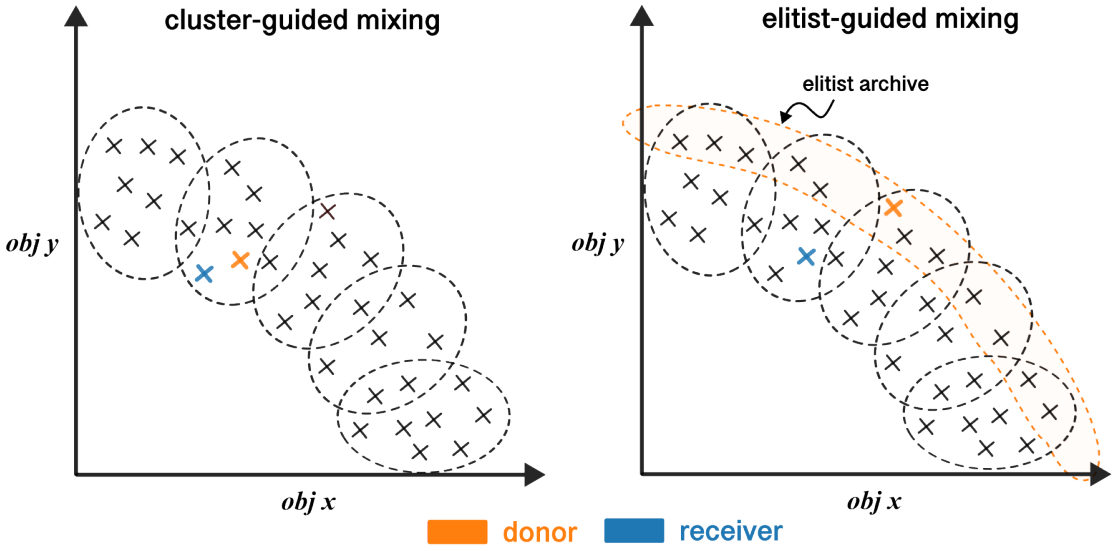


Figure 3.5: Illustrative examples for cluster-guided and elitist-guided mixings.

Regarding directed evolution, we present a concept based on the angle formed between the vector from the receiver to the donor and the vector from the receiver to the offspring. This concept is referred to as the evolving angle, as depicted in Figure 3.6. By examining the evolving angle, we can discern a more directed evolution with a smaller evolving angle. This suggests that the offspring is closer to the donor and implies a more directed evolution, guided by the donor.

doi:10.6342/NTU202303309

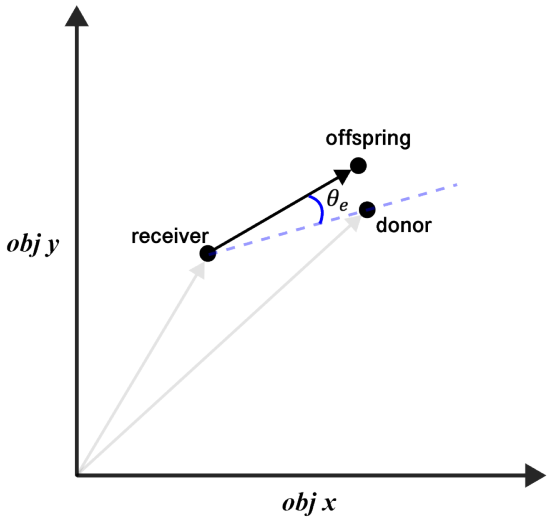


Figure 3.6: Visualization of evolving angle, denoted as θ_e.

To examine the evolving direction of two genetic algorithm variants, we execute a series of experiments, focusing on the distribution of evolving angles for successfully mixed offspring in relation to two NPC problems from our selected problem sets: MO knapsack and MO maxcut. To effectively represent this distribution, we adopt histogram plots, which are elucidated in Figure 3.7. The results reveal a clear contrast in the evolving angles of the two variants. For the variant-elitist, the angles are mainly concentrated within a narrow range close to zero, suggesting a more directed evolutionary trajectory towards the donor. On the other hand, the variant-cluster exhibits a more diverse and multimodal distribution, indicating less focused evolution.

doi:10.6342/NTU202303309

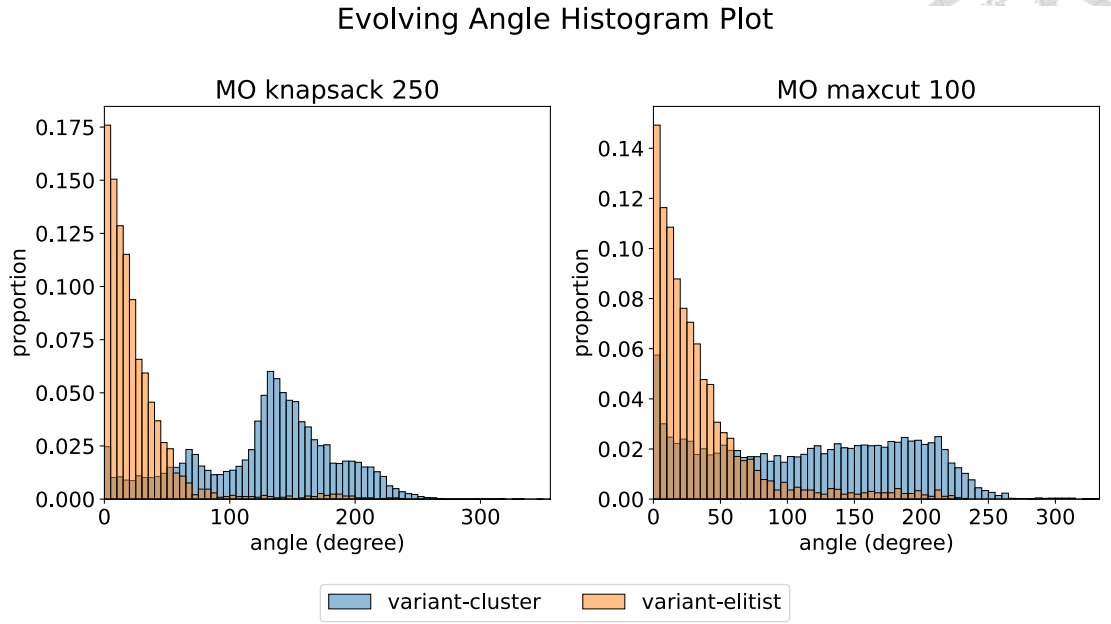


Figure 3.7: Comparative histogram plots illustrating the diversity in evolving angles of variant-cluster and variant-elitist for two selected NPC problems.

3.4 Evolving Regions

Further, in the scenario of bi-objective optimization, where two objectives are of interest, the contributions made by building blocks in multi-objective optimization may have varying impacts. These include positive effects on both objectives, beneficial effects on one objective with no influence on the other, or an enhancement of one objective to the detriment of the other. In order to more accurately characterize these three types of evolution, we divide the objective space, where the offspring is located, into three distinct regions as shown in Figure 3.8:

1. *Coherent* Region: This region encompasses offspring that demonstrate synergistic improvement across both objectives.

2. *Solitary* Region: This region includes offspring where one objective exhibits a dominant improvement, while the other objective experiences minimal change.

doi:10.6342/NTU202303309

3. *Incoherent* Region: This region contains offspring where one objective exhibits improvement, but the other objective displays a drawback.

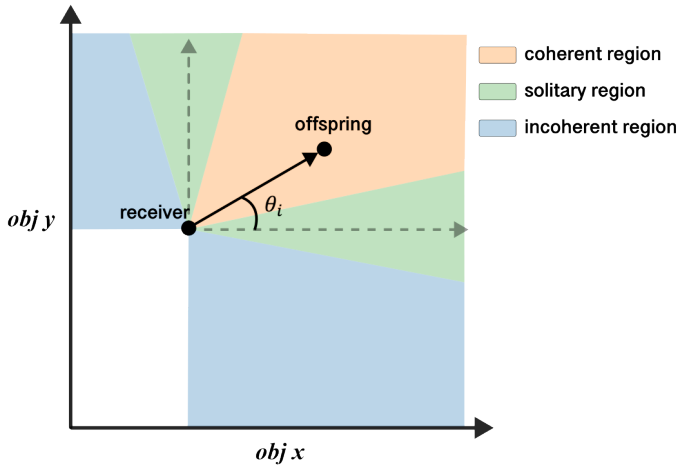


Figure 3.8: Graphical representation of three distinct evolving regions within the objective space. The angle of improvement, represented as θ_i, is shown, with the normalization of both objectives to the range $[0, 1]$, aiming to maximize both.

To adapt the angle of improvement to the three regions, we utilize empirical data from a variety of test problems. The angle range for the solitary region is assigned as $\frac{\pi}{4}$, with two subsections of the solitary region centered at 0 and $\frac{\pi}{2}$. The specific angle ranges for each region can be found in Table 3.2. Nevertheless, the values presented in Table 3.2 for the angle of improvement in the three regions may vary, subject to the distinct characteristics of the problem being studied and may demand further modifications.

Table 3.2: Angle of improvement, θ_i, for three evolving regions.

Region	Angle of Improvement (radians)
Coherent Region	$\frac{\pi}{8} \leq \theta_i \leq \frac{3\pi}{8}$
Solitary Region	$\left(-\frac{\pi}{8} < \theta_i < \frac{\pi}{8}\right) \cup \left(\frac{3\pi}{8} < \theta_i < \frac{5\pi}{8}\right)$
Incoherent Region	$\left(-\frac{\pi}{2} \leq \theta_i \leq -\frac{\pi}{8}\right) \cup \left(\frac{5\pi}{8} \leq \theta_i \leq \pi\right)$

From the perspective of advancing the Pareto front, the coherent region stands out as being highly aligned with the principles of selection based on Pareto-optimality. This region signifies offspring that exhibit improvements in both objective values, thus hold-

doi:10.6342/NTU202303309

ing a higher potential for discovering new non-dominated solutions. As highlighted in Table 3.9, the angle of improvement for the variant-elitist's successfully mixed offspring predominantly falls within the coherent region for two NPC problems. Conversely, the variant-cluster demonstrates a comparatively lesser proportion within the same region. It is important to note, however, that the variant-cluster bears a greater potential for assisting the receiver in improving towards the incoherent direction.

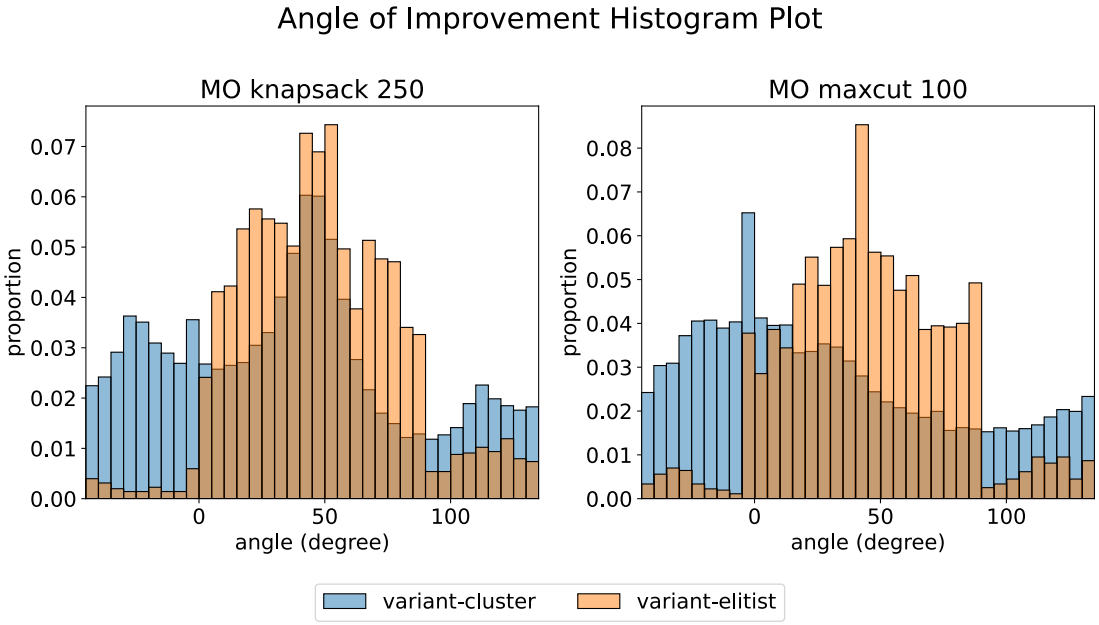


Figure 3.9: Comparative histogram plots illustrating the diversity in angle of improvement of variant-cluster and variant-elitist for two selected NPC problems.

3.5 Receiver-aware Donor Selection

Building upon the aforementioned empirical findings, we propose an extension to the existing settings to further emphasize the focus of elitist-guided mixing on evolution in the coherent direction so that it can better advance the Pareto front. Leveraging the observed empirical fact that the evolving angle tends to approach zero, we can establish a bound for donor selection in elitist-guided mixing. This bound ensures that the offspring generated through elitist-guided mixing have the highest probability of falling within the

doi:10.6342/NTU202303309

coherent region.

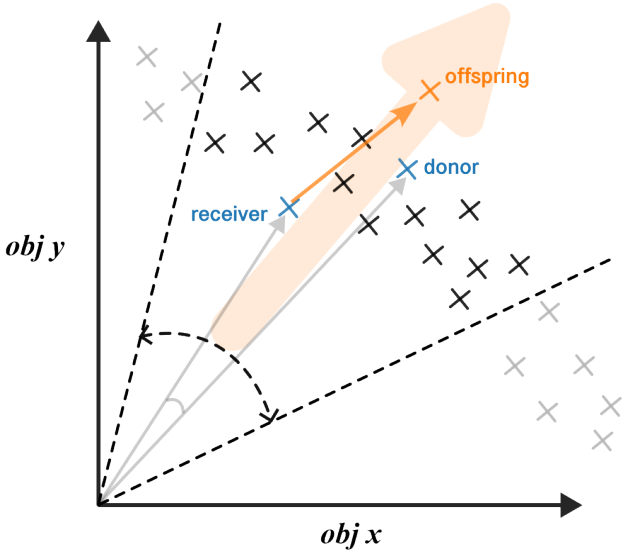


Figure 3.10: Visualization of conceptual bound to concentrate elitist-guided mixing predominantly in the coherent region.

To this end, we introduce the concept of the donor-receiver angle, denoted as θ_d. This angle is defined as the angle between the vector from the origin to the receiver and the vector from the origin to the donor, as depicted in Figure 3.11.

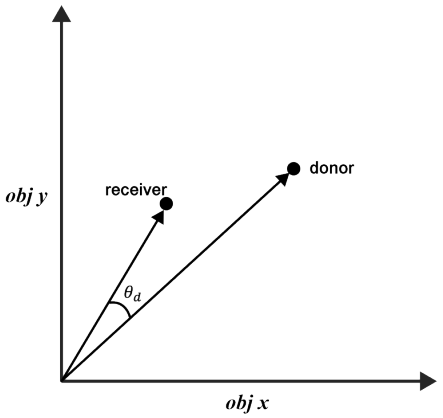


Figure 3.11: Visualization of donor-receiver angle, denoted as θ_d.

To facilitate our derivation, we introduce the following notation: Let's denote R as the receiver situated at coordinates (R_x, R_y), and D represents the donor located at coordinates $(R_x + \Delta x, R_y + \Delta y)$. The vector from R to the offspring O can be represented as $(\Delta x, \Delta y + \epsilon)$, with ϵ representing a small value attributed to the distribution of evolving

doi:10.6342/NTU202303309

angles within the context of elitist-guided mixing.

The angle of improvement for O, denoted as θ_i, can be mathematically expressed as the angle between the vector that signifies the direction of the objective x and the vector $\overrightarrow{RO}$, as follows:

$$\theta_i = \arctan\left(\frac{\Delta y + \epsilon}{\Delta x}\right). \tag{3.1}$$

To ensure that the resulting offspring exhibits objectives with coherent improvement, a constraint is imposed on the angle of improvement. This is expressed as:

$$\theta_{\texttt{low}} \leq \theta_i \leq \theta_{\texttt{high}}, \tag{3.2}$$

where $\theta_{\texttt{low}}$ is set to $\frac{\pi}{8}$ and $\theta_{\texttt{high}}$ is set to $\frac{3\pi}{8}$, as specified in Table 3.2.

Next, we define θ_R as the angle between the vector from the origin to R and the primary objective, objective x. By doing so, we can establish a constraint for the donor-receiver angle of R, denoted as θ_d, in the following manner:

$$\begin{aligned} &\theta_R - \arctan\left(\frac{R_y + \tan(\theta_{\texttt{high}})\Delta x + \epsilon}{R_x + \Delta x}\right) \leq \theta_d \\ &\leq \theta_R - \arctan\left(\frac{R_y + \tan(\theta_{\texttt{low}})\Delta x - \epsilon}{R_x + \Delta x}\right). \end{aligned} \tag{3.3}$$

Subsequently, we replace the current maximum value of the objective x, denoted as

doi:10.6342/NTU202303309

M_x, to $R_x + \Delta x$ to mitigate the variability of the bound. This substitution yields the final constraint on the donor-receiver angle:

$$\theta_R - \arctan\left(\frac{R_y + \tan(\theta_{\texttt{high}})(M_x - R_x) + \epsilon}{M_x}\right) \leq \theta_d \leq \theta_R - \arctan\left(\frac{R_y + \tan(\theta_{\texttt{low}})(M_x - R_x) - \epsilon}{M_x}\right). \tag{3.4}$$

Consequently, we suggest limiting the choice of donors to those that are situated within a particular angular range in relation to the receiver. This restriction is expected to result in offspring that are located in a coherent region of the objective space. To implement this, we integrate the bound into our algorithm, allowing for the selective choice of donors from the elitist archive. The workflow for this donor selection process is illustrated in Figure 3.12, and the corresponding algorithm is presented in Algorithm 5.

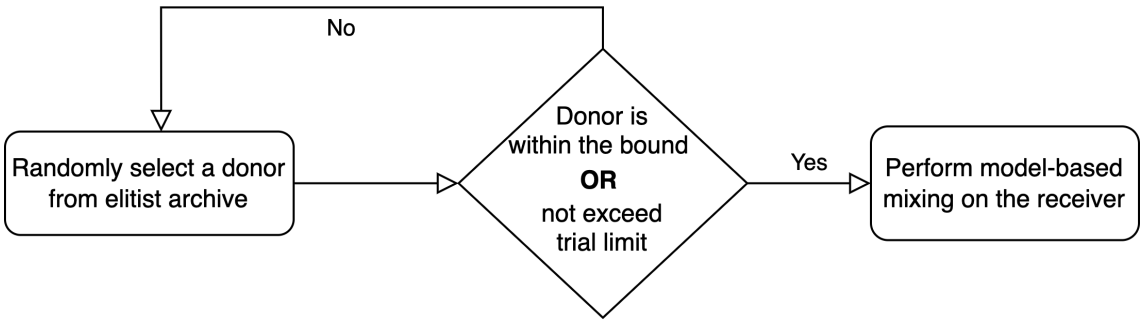


Figure 3.12: The flowchart of the receiver-aware donor selection mechanism.

The empirical outcome of integrating donor selection into elitist-guided mixing for the variant-elitist is depicted in Figure 3.13. The results demonstrate that in the MO knapsack problem, the donor selection mechanism improves the performance of MO-GOMEA. However, the selection mechanism weakens the enhancement and even leads to worse performance compared to MO-GOMEA in the MO maxcut problem. This observation

doi:10.6342/NTU202303309

Algorithm 5: ElitistGuidedMixingWithDonorSelection

input : R: receiver, E: elitist archive, L: linkage model,
output: R, E
$D \leftarrow$ choose a donor randomly from E with replacement
$\theta_D \leftarrow$ donor-receiver angle of D
$count \leftarrow 0$
while θ_D is outside the bound specified in Equation 4 and $count < \frac{|E|}{2}$ **do**
  $count \leftarrow count + 1$
  $D \leftarrow$ choose a donor randomly from E with replacement
$R, isImproved \leftarrow$ perform model-based mixing on R with D and L
return R, E

suggests that the need for evolution towards different regions may vary across different stage of evolution or problem contexts. Placing too much focus on the coherent region may result in the final Pareto front being too concentrated, limiting its spread, as shown in Figure 3.14. To address this, we propose an adaptive mechanism to dynamically adjust the proportion of the two types of mixing during the evolution process.

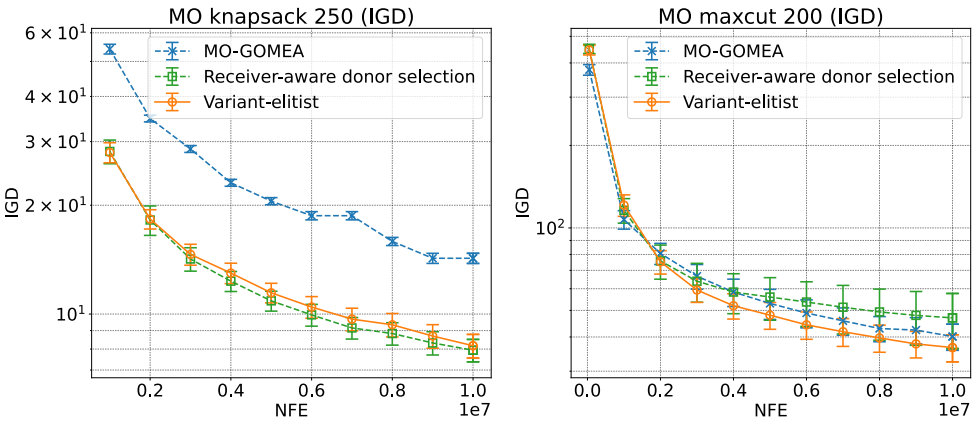


Figure 3.13: Convergence performance of MO-GOMEA, variant-elitist, and variant-elitist with donor selection measured by the IGD.

doi:10.6342/NTU202303309

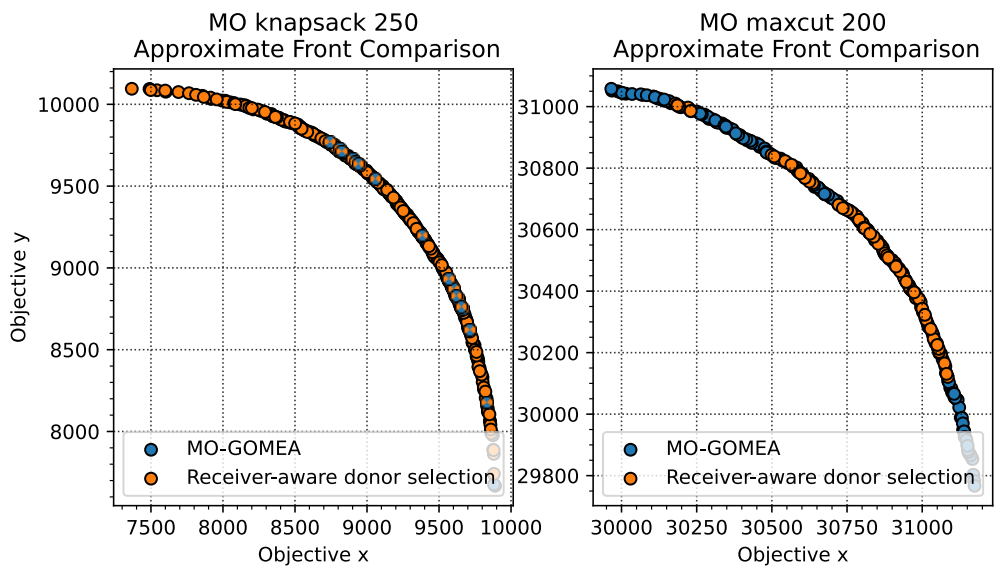


Figure 3.14: Visualization of how Pareto front will be concentrated after employing receiver-aware donor selection on MO knapsack 250 and MO maxcut 200.

3.6 Self-Adaptive Switching Based on Temporal Change of Objective Inter-relationships

We analyze the distribution of the three regions as the generations progress to gain insights into their dynamics. Figure 3.15 illustrates that regardless of whether it is the variant-cluster or variant-elitist, the proportion of coherent regions decreases in later generations. This decline indicates the increasing difficulty of uncovering innovations as the generations advance.

doi:10.6342/NTU202303309

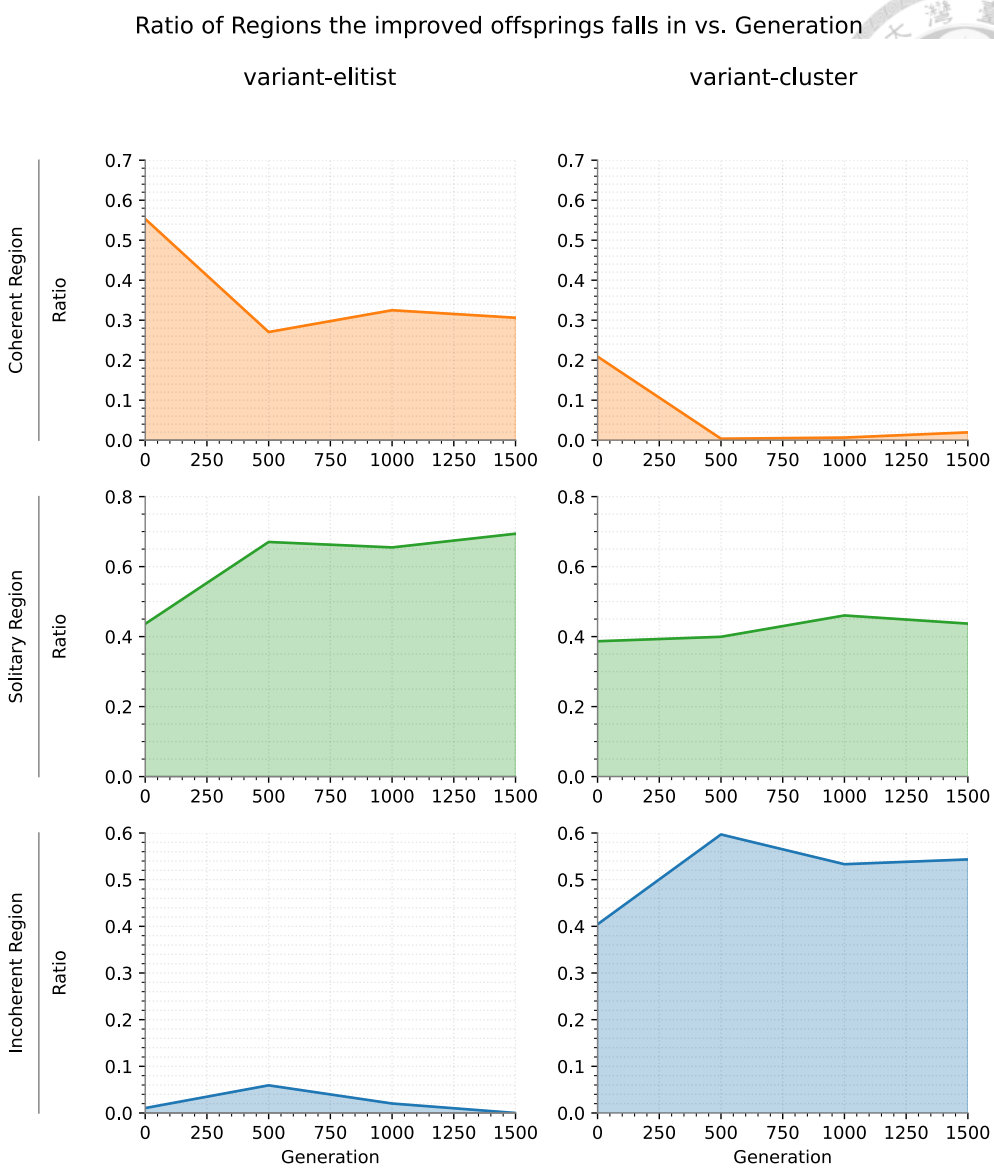


Figure 3.15: Visualization of the generation-wise distribution of improved offspring within three regions from both variant-cluster and variant-elitist for the MO maxcut problem.

Building on this empirical observation and previous research on innovation timing, we propose giving precedence to evolution in the direction of the coherent region during the early stages, before significant innovations emerge. In contrast, during the later stages characterized by the innovation period, the emphasis should be directed towards the solitary and incoherent regions, as shown in Figure 3.16.

doi:10.6342/NTU202303309

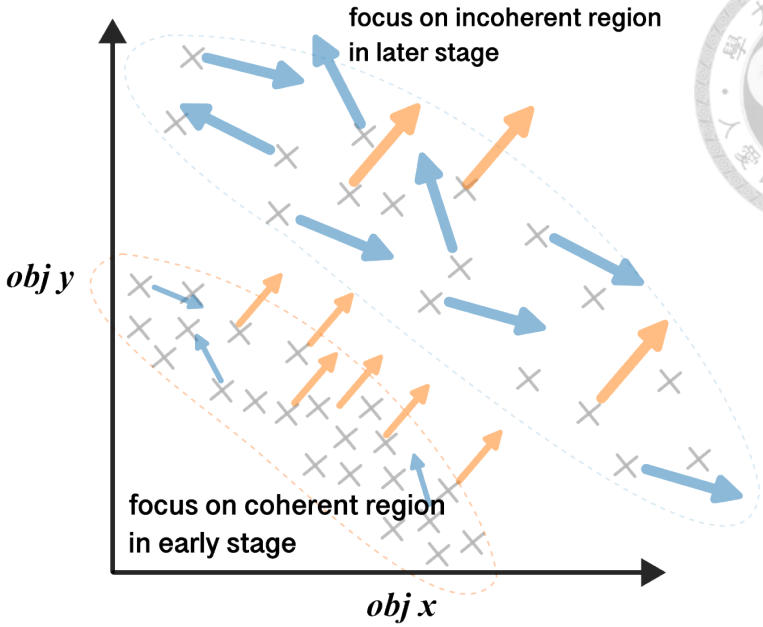


Figure 3.16: Conceptual visualization illustrating the focus on the coherent region during the early stages and the shift towards the incoherent region in the later stages of evolution.

To address this issue, we propose an adaptation mechanism that enables the algorithm to self-adapt between two mixing mechanisms. This mechanism introduces a self-adaptive index, denoted as p_e, along with a modifying parameter, γ. Based on the performance of each mixing mechanism in the previous generation, the algorithm adaptively adjusts the proportion of receivers adopting either elitist-guided mixing or cluster-guided mixing. The workflow of this adaptive mechanism is illustrated in Figure 3.17 and the algorithm is outlined in Algorithm 6. This self-adaptive nature allows the algorithm to flexibly respond to different stages of evolution and the varying contexts of problems.

doi:10.6342/NTU202303309

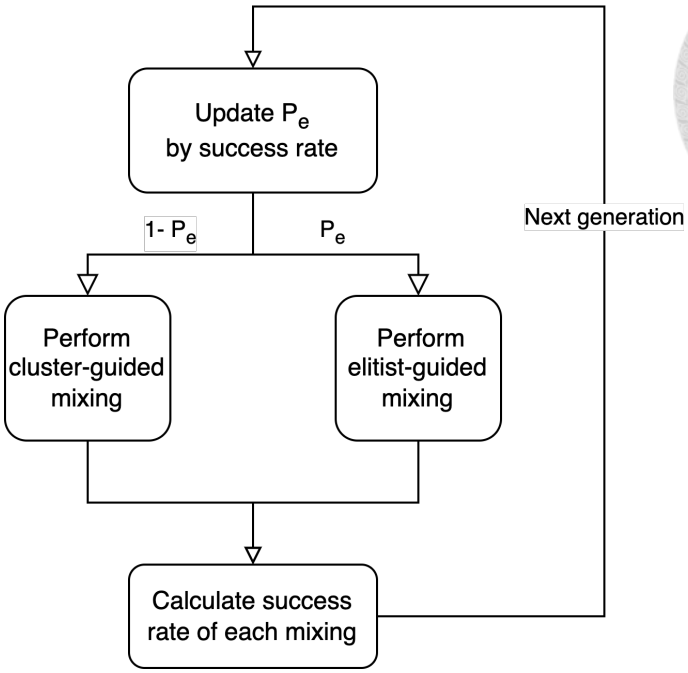


Figure 3.17: Flowchart illustrating the self-adaptive mechanism between cluster-guided and elitist-guided mixing.

Algorithm 6: Update p_e

r_c: success rate of cluster-guided mixing, r_e: success rate of elitist-guided mixing

input : γ: modifying parameter, p_e: proportion of elitist-guided mixing,
c_e: number of iterations performing elitist-guided mixing,
c_c: number of iterations performing cluster-guided mixing,
s_e: success count of elitist-guided mixing,
s_c: success count of cluster-guided mixing,

output: p_e

$r_c \leftarrow \frac{s_c}{c_c}$

$r_e \leftarrow \frac{s_e}{c_e}$

if $r_c > r_e$ **then**

  $p_e \leftarrow p_e - \gamma$

else if $r_c < r_e$ **then**

  $p_e \leftarrow p_e + \gamma$

return p_e

To prioritize the advancement of the Pareto front using elitist-guided mixing during the early stages, we initialize the self-adaptive index p_e to 0.9 and set the modifying parameter γ to 0.01. The empirical results, as shown in Figure 3.18, reveal that integrating the self-adaptive mechanism improves the performance, particularly in the case of the MO maxcut problem where donor selection initially resulted in poor performance. By exam-

doi:10.6342/NTU202303309

ining the achieved fronts of each algorithm, as shown in Figure 3.19, it becomes evident that the self-adaptive switching mechanism mitigates the issue of excessive concentration, preserves the beneficial gap-filling effect of cluster-guided mixing, and facilitates the approach with donor selection and self-adaptive mechanism to attain superior Pareto fronts.

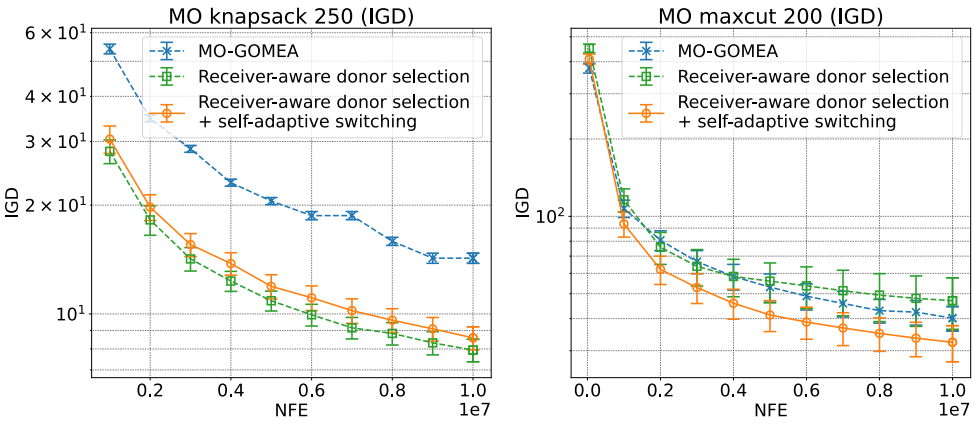


Figure 3.18: Convergence performance of MO-GOMEA, variant-elitist with donor selection, and variant-elitist with donor selection and self-adaptive mechanism measured by the inverted generational distance.

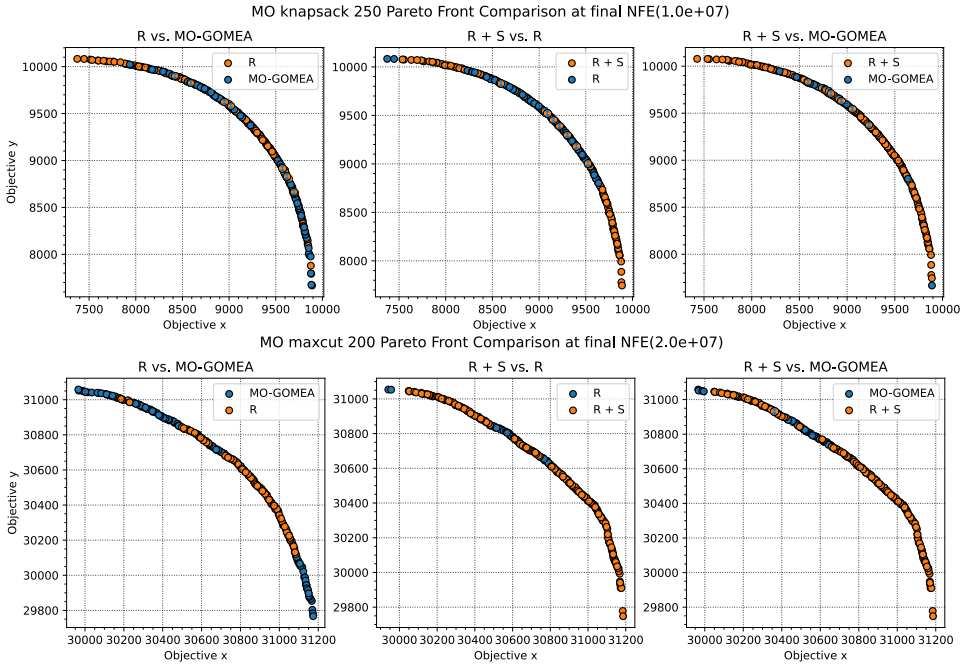


Figure 3.19: Comparison of Pareto front among MO-GOMEA, variant-elitist with donor selection (R), and the approach combining donor selection and self-adaptive switching (R+S) on MO knapsack 250 and MO maxcut 200 of MO-GOMEA.

doi:10.6342/NTU202303309

We also tracked the progression of the self-adaptive index p_e throughout the experiment. Confirming our expectations, p_e demonstrated a decline as generations progressed, validating the adaptivity of our proposed mechanism. Interestingly, while the specific trends varied depending on the problem context, the general descending trend remained consistent across all scenarios. This empirical finding aligns with our conjecture, substantiating the effectiveness of our strategy. The evolution of p_e over generations for different problem contexts can be seen in Figure 3.20.

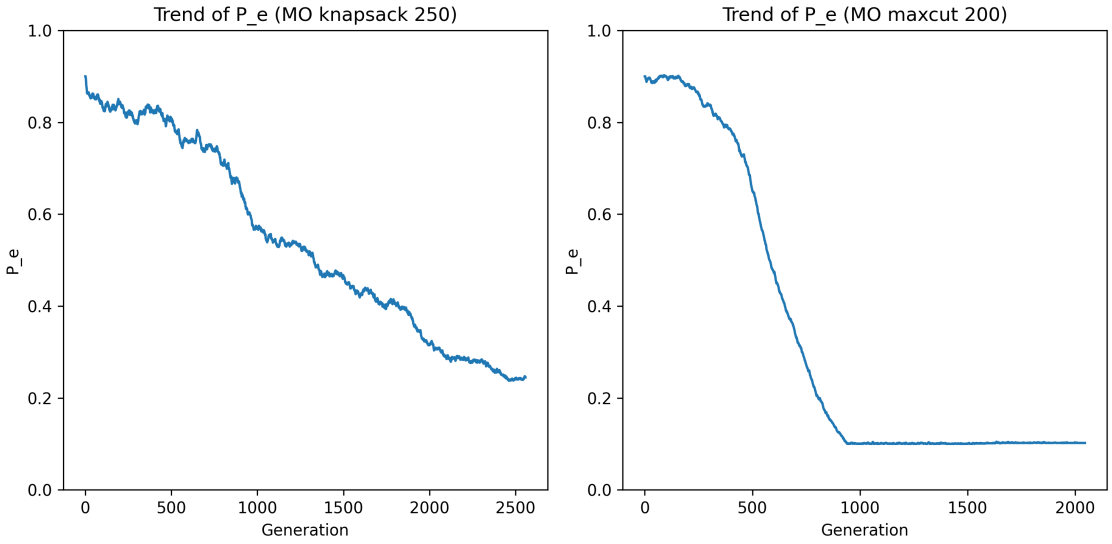


Figure 3.20: Trend of the self-adaptive index p_e across generations within various problem contexts, as observed in the first population of the IMS.

3.7 ADSM-MO-GOMEA

Integrating the donor selection mechanism and the self-adaptive switch, we propose a new mixing approach called Adaptive Donor Selection Mixing (ADSM). The algorithm is outlined in Algorithm 7. Incorporating ADSM into MO-GOMEA involves substituting the initial model-based recombination of the middle cluster with ADSM. Particularly, ADSM utilizes the multi-objective GOM for cluster-guided mixing and the FI mechanism for

doi:10.6342/NTU202303309

elitist-guided mixing. The resulting variant, named ADSM-MO-GOMEA, is described in Algorithm 8. Moreover, it is possible to integrate the IMS into this variant. Further empirical details on a diverse set of benchmark problems are presented in Chapter 4.

Algorithm 7: Adaptive Donor Selection Mixing

input : E: elitist archive, R: receiver, C: cluster of receiver,
p_e: proportion of elitist-guided mixing, L: linkage model,
output: R, E
$rand \leftarrow$ sample from a uniform distribution in the interval [0,1]
if $rand < p_e$ **then**
  R, $isImproved \leftarrow$
  `ElitistGuidedMixingWithDonorSelection`(R, D, L)
else
  $D \leftarrow$ randomly choose a donor from C
  R, $isImproved \leftarrow$ perform model-based mixing on R with D and L
if $isImproved$ **then**
  $E \leftarrow$ update E with R
return R, E

doi:10.6342/NTU202303309

Algorithm 8: ADSM-MO-GOMEA

$\mathcal{P}$: population, $\mathcal{P}_i$: the i-th chromosome in $\mathcal{P}$,
O_i : offspring generated by the i-th chromosome in $\mathcal{P}$, E: elitist archive,
p_e: proportion of elitist-guided-mixing,
s_c: success count of cluster-guided mixing,
c_c: number of iterations performing of cluster-guided mixing,
r_c: success rate of cluster-guided mixing,
s_e: success count of elitist-guided mixing,
c_e: number of iterations performing of elitist-guided mixing,
r_e: success rate of elitist-guided mixing,
input : p: population size, k: number of clusters, ℓ: problem size,
γ: modifying parameter for p_e
output: E
$\mathcal{P} \leftarrow$ initialize population with p and ℓ randomly
$E \leftarrow \varnothing$
$p_e,\ r_c,\ r_e \leftarrow 0$
while ¬`ShouldTerminate` **do**
  $c_e,\ c_c,\ s_e,\ s_c \leftarrow 0$
  $\{C_0, \cdots, C_{k-1}\} \leftarrow$ divide $\mathcal{P}$ into k clusters
  $\{L_0, \cdots, L_{k-1}\} \leftarrow$ learn linkage model for each cluster
  for $i \leftarrow 0$ **to** $p-1$ **do**
    $C_i \leftarrow$ get the cluster of $\mathcal{P}_i$
    if C_i is extreme cluster **then**
      $O_i,\ E \leftarrow$ perform single-objective GOM on $\mathcal{P}_i$ with C_i and L_i and update E
    else
      $O_i,\ E \leftarrow$ `AdaptiveDonorSelectionMixing`($\mathcal{P}_i$, C_i, E, L_i, p_e) (Algorithm 7)
      Update $c_e,\ c_c,\ s_e,\ s_c$
  $p_e, r_c,\ r_e \leftarrow$ `UpdatePe`($c_e,\ c_c,\ s_e,\ s_c$)
  $\mathcal{P} \leftarrow \{O_0,\ O_1, \cdots,\ O_{p-1}\}$
return E

doi:10.6342/NTU202303309

Chapter 4 Experiments and Results

In this chapter, we present the metrics utilized to evaluate the obtained front by the algorithm and the experimental setup. Additionally, we provide a comparison of the results and engage in discussions with the version of MO-GOMEA with IMS proposed in 2017 [19].

4.1 Experiment Settings and Performance Evaluation

The effectiveness of ADSM is assessed by comparing ADSM-MO-GOMEA with the standard MO-GOMEA algorithm. To ensure a fair comparison, we maintain consistency by using the same parameter settings that were optimized for MO-GOMEA. Both algorithms incorporate the IMS mechanism and a weak mutation strategy with a mutation probability of $p_m = \frac{1}{l}$ for the LOTZ problem [19]. For solutions violating constraints in the MO knapsack problem, identical repair mechanisms as MO-GOMEA are implemented [19, 41]. For the problem instances, we maintain the same problem sizes as in the case of MO-GOMEA, with sizes of 100, 250, 500, and 750 for the MO knapsack. Additionally, we consider problem sizes of 25, 50, and 100 for the MO maxcut. To further assess the scalability of ADSM-MO-GOMEA, we create larger instances by randomly generating sizes of 200 and 400 for the MO maxcut.

doi:10.6342/NTU202303309

The performance of the algorithm is evaluated differently for the scalable benchmark problems and the NPC problems. For the scalable benchmark problems, wherein the algorithms possess the ability to identify the entire Pareto front, the NFEs required to obtain the ground truth is utilized to assess the performance, with a lower value indicating better performance. For the NPC problems, two metrics, IGD and FO, are employed within a predetermined NFEs as listed in Table 4.1. Due to the difficulty in obtaining the exact ground truth for NPC problems,, an approximate ground truth is created by combining the fronts obtained from all runs of both algorithms. For each problem instance, each algorithm undergoes 60 independent trials, and the obtained metrics are utilized to calculate the mean and standard deviation for further analysis. It should be noted that for IGD metric, we are essentially comparing the relative magnitudes between two algorithms, as the objective values have not been normalized.

Table 4.1: Problem instance and number of function evaluation limit

Problem instance	**Number of function evaluation limit**
MO knapsack 100	1×10^7
MO knapsack 250	1×10^7
MO knapsack 500	1.5×10^7
MO knapsack 750	2×10^7
MO maxcut 25	5×10^5
MO maxcut 50	5×10^6
MO maxcut 100	1.5×10^7
MO maxcut 200	2×10^7
MO maxcut 400	2×10^7

4.2 Results and Discussions

In this section, we present the results from both algorithms and engage in a discussion regarding the final outcomes and the converging IGD metrics at different NFEs. The main focus of the discussion is to compare the influence of the ADSM mechanism and its

doi:10.6342/NTU202303309

performance under various problem contexts.

4.2.1 Comparison on Scalable Benchmarks

Figure 4.1 illustrates the mean NFEs required to reach the true Pareto front within scalable benchmark problems at various problem sizes for both the original MO-GOMEA and the proposed ADSM-MO-GOMEA algorithms. Notably, in the zeromax-onemax and trap5-inverse trap5 problems, the use of ADSM appears to augment the performance of MO-GOMEA without compromising its scalability or capacity to manage deceptive problems. Moreover, when integrated with weak mutation, ADSM-MO-GOMEA demonstrates improved efficiency in addressing problems with conditional dependencies, as evidenced by the results obtained for LOTZ. This observation suggests that the mutation strategy can be effectively combined with ADSM to further improve the performance of the algorithm.

4.2.2 Comparison on MO Knapsack

Table 4.2 compiles the mean IGD and FO results across four instances of the MO knapsack problem. The outcomes indicate that ADSM-MO-GOMEA consistently outperforms in terms of both IGD and FO, with a noticeably greater advantage in the FO aspect. As the problem size escalates, the benefits of employing ADSM become more evident. Figure 4.2 depicts the Inverted Generational Distance (IGD) performance over the course of the evolutionary process. This indicates that ADSM-MO-GOMEA starts off with superior IGD values from the very beginning. Moreover, this advantage not only remains consistent but also progressively amplifies with an increase in the number

doi:10.6342/NTU202303309

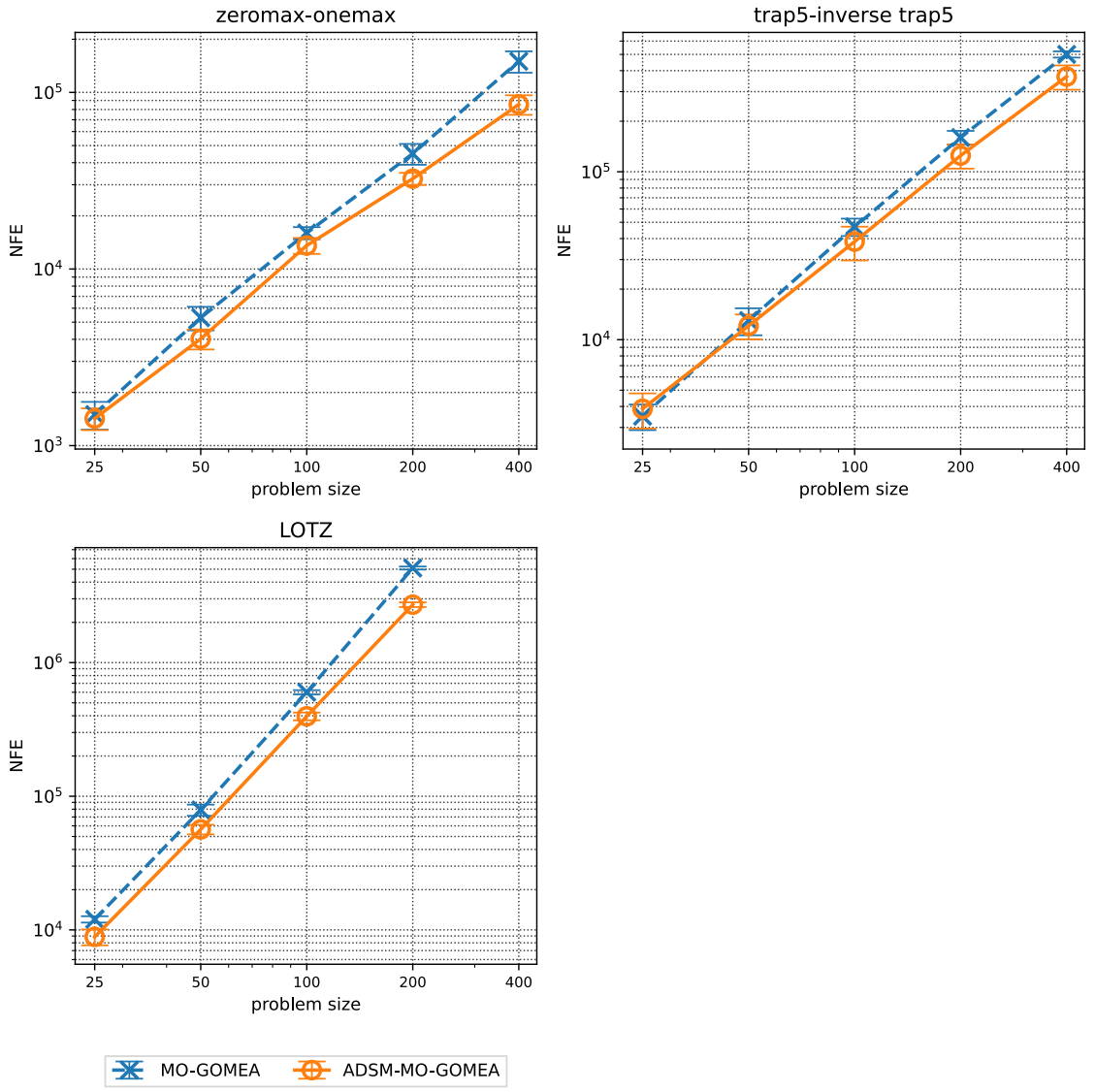


Figure 4.1: Comparison of NFEs required to reach the true Pareto front using both MO-GOMEA and ADSM-MO-GOMEA on scalable benchmark problems, with the bars illustrating the interval of certainty at the 95% level.

of function evaluations, particularly for larger problem instances. This suggests that the proposed ADSM-MO-GOMEA algorithm begins with a strong initial performance and demonstrates increased effectiveness over time in optimizing the IGD metric, particularly in more complex problem domains.

doi:10.6342/NTU202303309

Table 4.2: Comparison of IGD and FO metrics for MO knapsack obtained by ADSM-MO-GOMEA and MO-GOMEA. Statistical significance is determined using a t-test to assess the differences between the means of the compared groups, where a lower p-value indicates higher significance.

Metric	Instance	MO-GOMEA		ADSM-MO-GOMEA		p-value
		Mean	Std	Mean	Std	
IGD	100	1.72	0.27	**1.26**	0.21	2.3×10^{-18}
	250	14.27	0.90	**8.52**	1.32	8.2×10^{-54}
	500	28.79	4.35	**11.66**	2.11	4.8×10^{-53}
	750	62.51	4.55	**14.68**	2.28	5.6×10^{-99}
FO	100	108.43	1.80	**111.08**	2.01	8.9×10^{-12}
	250	266.97	16.37	**308.22**	13.23	1.1×10^{-29}
	500	376.37	14.09	**552.85**	23.60	4.1×10^{-81}
	750	509.83	15.83	**904.02**	42.22	1.1×10^{-95}

4.2.3 Comparison on MO Maxcut

Table 4.3 provides the mean IGD and FO results across four instances of the MO maxcut problem. MO-GOMEA exhibits a superior performance on smaller instances (25 and 50), yet comparable results are yielded for the instance with a problem size of 100. However, as the problem size expands to 200 and 400, ADSM-MO-GOMEA attains lower IGD and higher FO values, indicating its superior scalability. Figure 4.3 showcases the progression of IGD performance over time across four larger instances. The results indicate that the performance of ADSM-MO-GOMEA aligns with MO-GOMEA in the early stages and may seem less efficient when NFEs increase for smaller problem instances. As for larger problems, ADSM-MO-GOMEA outperforms MO-GOMEA from the outset. However, this advantage does not rise as substantially as in the MO knapsack. This could be ascribed to the more complex characteristics of the maxcut problem, such as its overlapping structure and symmetry property [5]. However, further exploration is required to confirm this.

doi:10.6342/NTU202303309

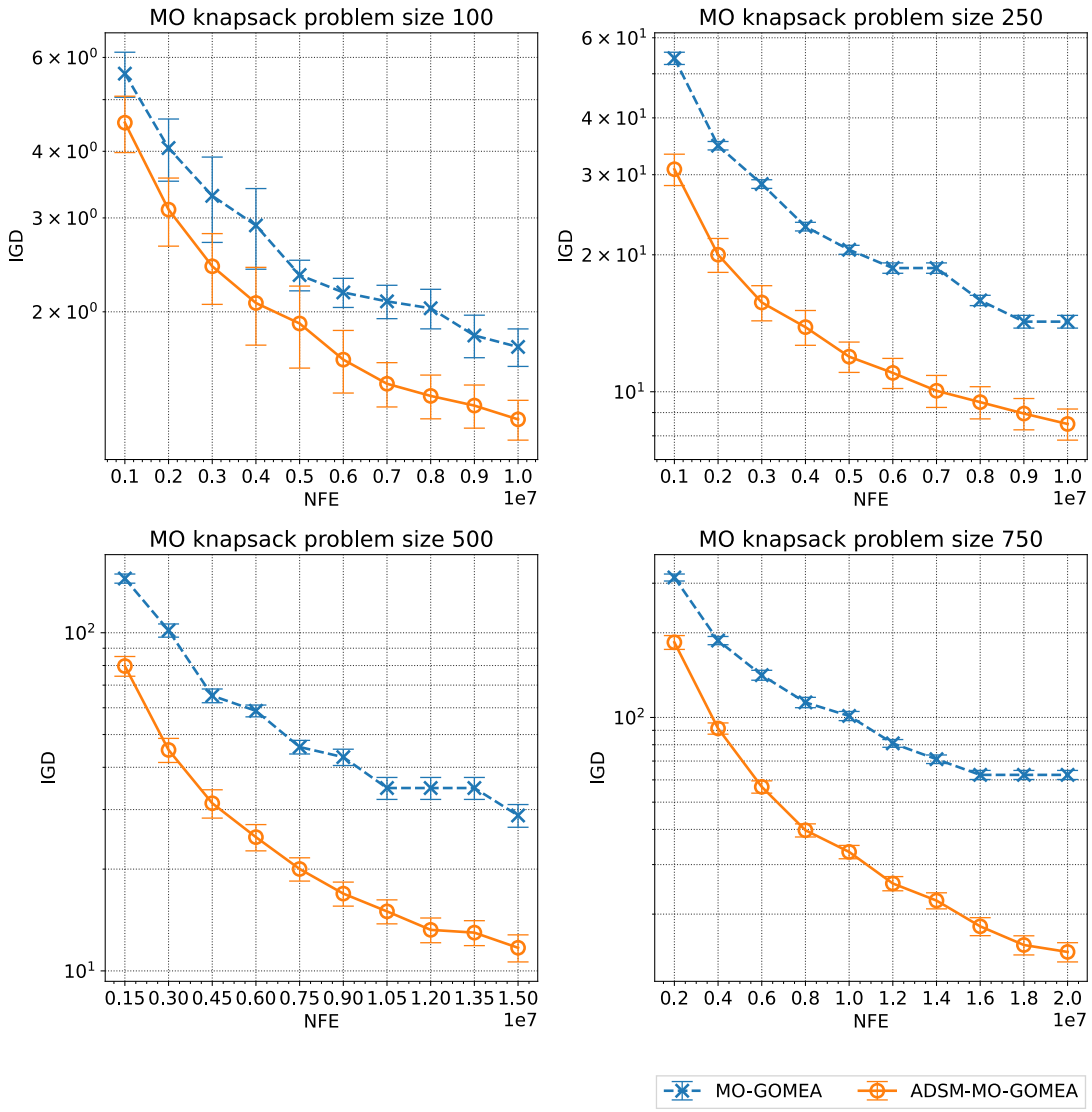


Figure 4.2: Performance of IGD convergence across four instances of MO knapsack, with the bars illustrating the interval of certainty at the 95% level.

doi:10.6342/NTU202303309

Table 4.3: Comparison of IGD and FO metrics for MO maxcut obtained by ADSM-MO-GOMEA and MO-GOMEA. Statistical significance is determined using a t-test to assess the differences between the means of the compared groups, where a lower p-value indicates higher significance.

Metric	**Instance**	**MO-GOMEA**		**ADSM-MO-GOMEA**		**p-value**
		Mean	**Std**	**Mean**	**Std**	
IGD	25	**0.00**	0.00	0.01	0.03	8.1×10^{-2}
	50	**0.13**	0.08	0.20	0.09	2.8×10^{-5}
	100	**2.45**	1.09	2.97	2.40	1.3×10^{-1}
	200	30.56	9.44	**28.17**	9.15	1.6×10^{-1}
	400	164.23	49.38	**135.17**	34.02	2.8×10^{-4}
FO	25	**17.00**	0.00	16.98	0.13	3.2×10^{-1}
	50	**41.40**	0.79	41.13	0.85	7.7×10^{-2}
	100	**119.17**	3.52	118.36	4.47	2.7×10^{-1}
	200	283.00	14.90	**293.63**	18.75	8.1×10^{-4}
	400	522.27	34.70	**644.80**	43.19	1.7×10^{-33}

doi:10.6342/NTU202303309

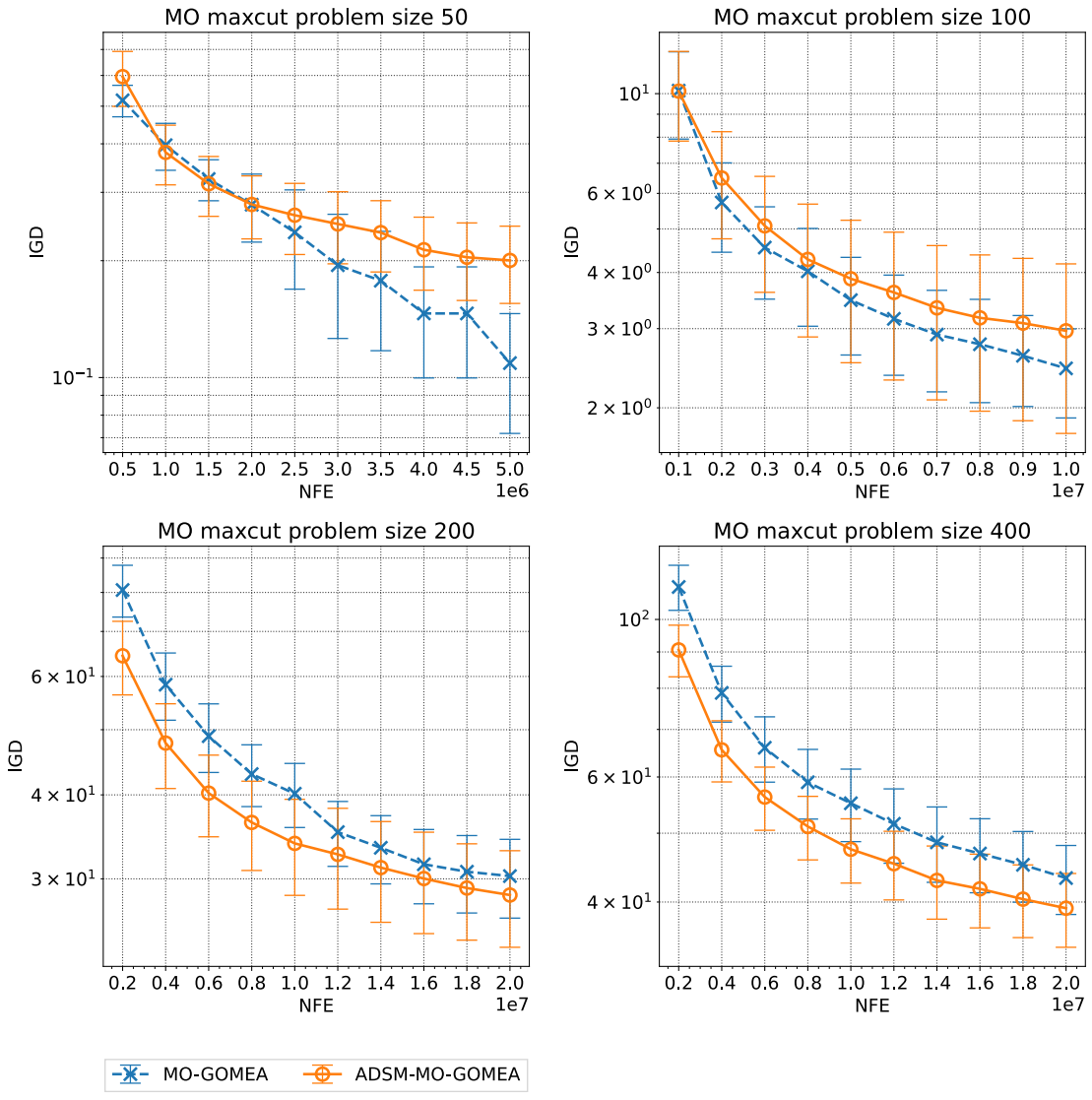


Figure 4.3: Performance of IGD convergence across four instances of MO maxcut, with the bars illustrating the interval of certainty at the 95% level.

doi:10.6342/NTU202303309

Chapter 5 Beyond Two Objectives

The traditional understanding of multi-objective problems typically involves up to five objectives, with the realm of many-objective optimization being considered when the number of objectives exceeds five [13]. Although the benchmark problems used in our study are designed with two objectives, there are instances where more than three objectives are involved. While ADSM-MO-GOMEA primarily focuses on bi-objective optimization, our research aims to explore the effects of objective inter-relationship on optimization performance in more complex scenarios.

5.1 Test Problems

This section extends the focus to advanced problem formulations, encompassing the extension of two prevalent NPC problems, MO knapsack and MO maxcut, to scenarios involving three and five objectives. Additionally, a significant issue from the realm of social networks, the multi-objective influence maximization problem (MIMP) [21, 22, 27, 30], is introduced into the test suite.

The influence maximization problem (IM) in the context of social networks has traditionally been treated as a single-objective optimization, aiming to maximize the influence spread of a selected set of seed nodes. However, However, in recent studies, the multi-

doi:10.6342/NTU202303309

objective version of the IM problem has been explored [22, 27]. For this research, we will utilize the formulation proposed in [3]. This MIMP involves three conflicting objectives to be simultaneously optimized:

1. **Maximizing the influence spread**: The main objective involves maximizing the influence spread. Traditionally, estimating the influence spread of a given seed set involves using Monte-Carlo simulations, a process that can be time-consuming. To increase efficiency, the expected influence score (EIS) is utilized, which provides a faster way to estimate influence spread in variable-length solutions, characteristic of MIMP.

$$\max\ EIS(S) = \left(1 + \frac{1}{|N_S^1 \setminus S|} \sum_{u \in N_S^1 \setminus S} p\tau_u^2\right) \times \sum_{i \in N_S^1 \setminus S} \left(1 - (1-p)^{\tau_i{}^1}\right) \quad (5.1)$$

2. **Minimizing the cost of seed nodes**: The second objective involves minimizing the cost associated with selecting seed nodes. Influential nodes often come with higher financial costs, and in this context, a node's cost is determined by its degree.

$$\min\ C(S) = \sum_{v \in S} c(v) \quad (5.2)$$

3. **Minimizing the number of seed nodes**: The third objective is geared towards minimizing the quantity of seed nodes. For better management and control, a reduced set of seed nodes is preferred.

$$\min\ k = |S| \quad (5.3)$$

doi:10.6342/NTU202303309

5.2 Modification of ADSM-MO-GOMEA

Though primarily designed for bi-objective optimization, ADSM can be extended to multi-objective problems through necessary modifications, resulting in a variant called ADSM*. This involves removing the donor selection segment from the elitist-guided mixing process, hence emphasizing the adaptation strategies of cluster-guided and elitist-guided mixing in ADSM*, as outlined in Algorithm 9.

Algorithm 9: ADSM*

input : E: elitist archive, R: receiver, C: cluster of receiver,
p_e: proportion of elitist-guided mixing, L: linkage model,
output: R, E
$rand \leftarrow$ sample from a uniform distribution in the interval [0,1]
if $rand < p_e$ **then**
  $D \leftarrow$ choose a donor randomly from E
  $R,\ isImproved \leftarrow$ perform model-based mixing on R with D and L
else
  $D \leftarrow$ choose a donor randomly from C
  $R,\ isImproved \leftarrow$ perform model-based mixing on R with D and L
if $isImproved$ **then**
  $E \leftarrow$ update E with R
return $R,\ E$

Analyzing the inter-relationship between objectives becomes increasingly complex as the quantity of objectives escalates. This scenario could give rise to $C_2^m * 3$ possible discussions, where m stands for the total number of objectives. Therefore, we simplify the initial representation of coherence levels by focusing solely on whether an objective has been improved, rather than showcasing the detailed intricacies of the three types of improvement. This alteration leads to the creation of the Coherent-n Region, with n indicating the number of improved objectives.

Additional experiments are carried out to scrutinize the properties of the two mixing methods, namely variant-elitist and variant-clustering. The assumption is that elitist

doi:10.6342/NTU202303309

mixing fosters coherent evolution, whereas cluster-guided mixing promotes incoherent evolution, mirroring the patterns observed in the bi-objective scheme. The empirical results, as demonstrated in Figure 5.1 and 5.2, validate this hypothesis. Notably, even as the objective count increases to three or five, a substantial improvement percentage of the variant-elitist remains associated with the most coherent region. Meanwhile, the variant-cluster tends to encourage a larger percentage in the less coherent region. This finding indicates that the variant elitist approach's capability to drive coherent improvement is not impeded by the increase in the number of objectives.

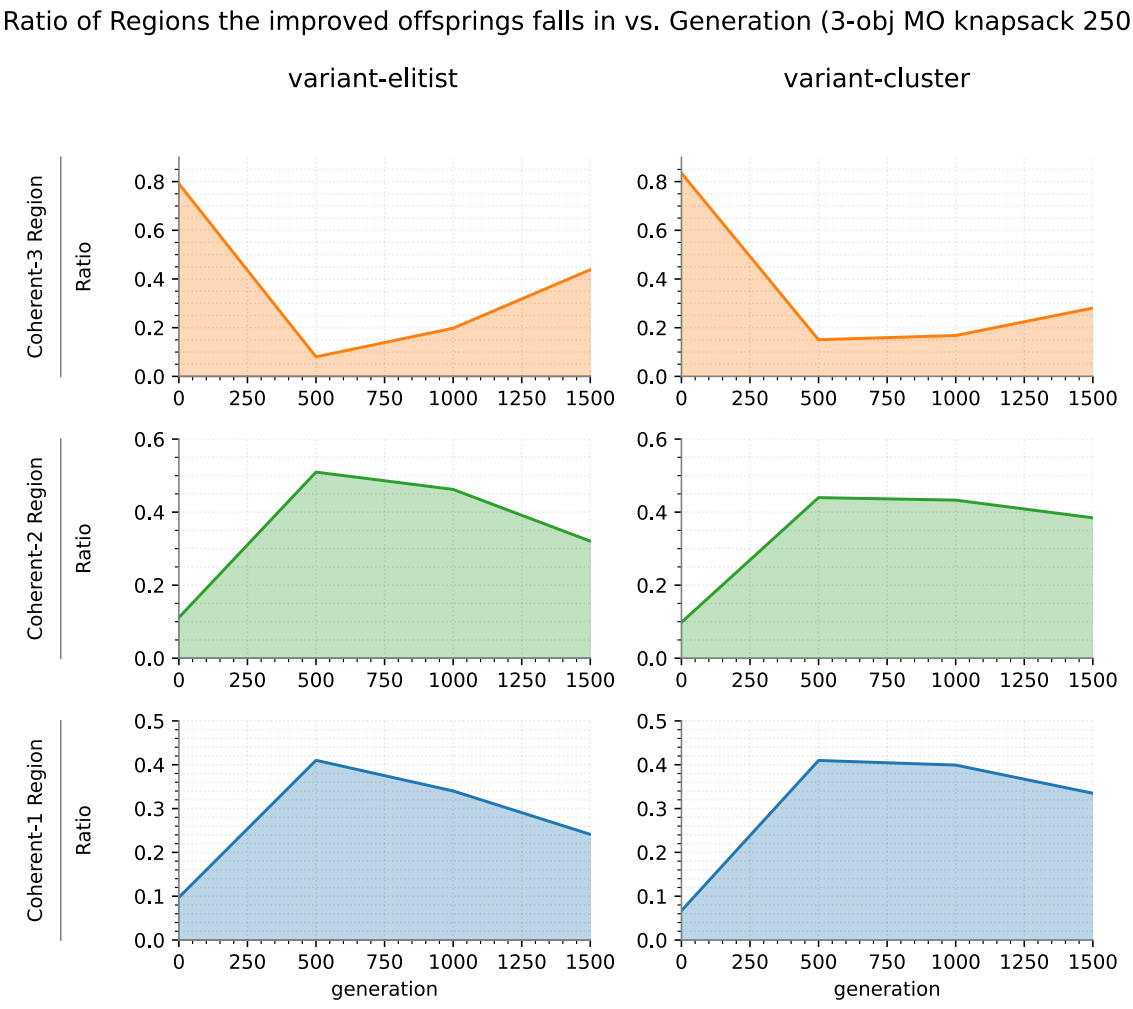


Figure 5.1: Distribution of coherent-n regions for variant-cluster and variant-elitist of MO knapsack 250 with three objectives.

doi:10.6342/NTU202303309

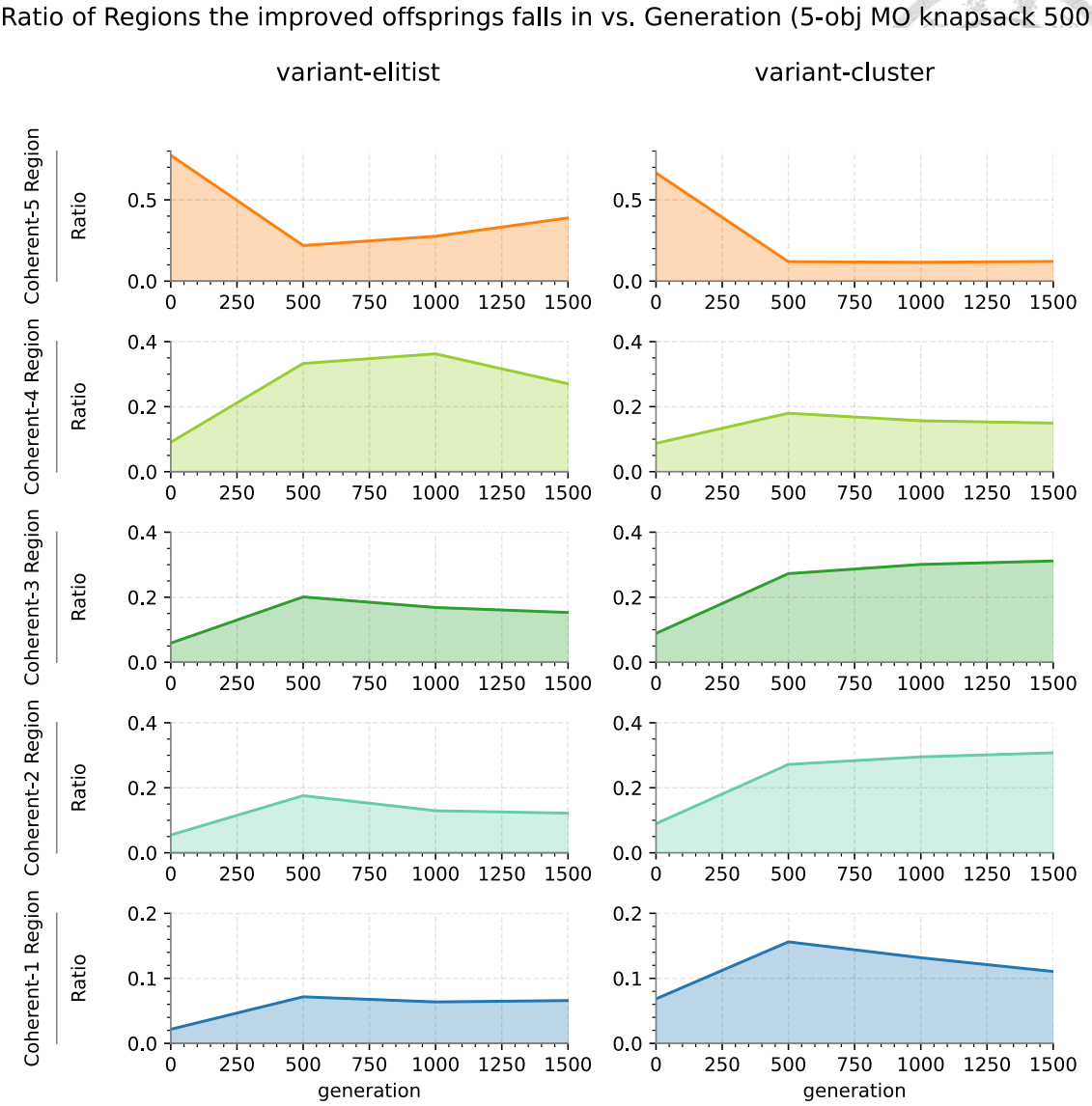


Figure 5.2: Distribution of coherent-n regions for variant-cluster and variant-elitist of MO knapsack 250 with five objectives.

With the empirical facts, we remain the conjecture of the in the early stage should conduct elitist-mixing and perform cluster-guided mixing in the later stage. So we remain the self-adaptation for the proportion of elitist-guided mixing in ADSM*-MO-GOMEA. The detailed steps of the ADSM*-MO-GOMEA algorithm are described below in Algorithm 10.

doi:10.6342/NTU202303309

Algorithm 10: ADSM*-MO-GOMEA

$\mathcal{P}$: population, $\mathcal{P}_i$: the i-th chromosome in $\mathcal{P}$,
O_i : offspring generated by the i-th chromosome in $\mathcal{P}$, E: elitist archive,
p_e: proportion of elitist-guided-mixing,
s_c: success count of cluster-guided mixing,
c_c: number of iterations performing of cluster-guided mixing,
r_c: success rate of cluster-guided mixing,
s_e: success count of elitist-guided mixing,
c_e: number of iterations performing of elitist-guided mixing,
r_e: success rate of elitist-guided mixing,
input : p: population size, k: number of clusters, ℓ: problem size,
γ: modifying parameter for p_e
output: E

```
P ← initialize population with p and ℓ randomly
E ← ∅
p_e, r_c, r_e ← 0
while ¬ShouldTerminate do
    c_e, c_c, s_e, s_c ← 0
    {C_0, ··· , C_{k−1}} ← divide P into k clusters
    {L_0, ··· , L_{k−1}} ← learn linkage model for each cluster
    for i ← 0 to p − 1 do
        C_i ← get the cluster of P_i
        if C_i is extrene cluster then
            O_i, E ← perform single-objective GOM on P_i with C_i and L_i and
              update E
        else
            O_i, E ← ADSM*(P_i, C_i, E, L_i, p_e) (Algorithm 9)
        Update c_e, c_c, s_e, s_c
    p_e, r_c, r_e ← UpdatePe(c_e, c_c, s_e, s_c)
    P ← {O_0, O_1, ··· , O_{p−1}}
return E
```

5.3 Experiment Settings

When conducting experiments involving NPC problems with three and five objectives, instances of various problem sizes were considered to maintain computational efficiency. We examine instances of the three-objective MO knapsack with sizes 100, 250, and 500, and the MO maxcut with sizes 50 and 100. For the five-objective problems, our analysis focuses solely on the MO knapsack instances with sizes 100 and 250. The expansion methodology for MO knapsack problem instances was adopted from SPEA [41],

doi:10.6342/NTU202303309

while MO maxcut problem instances were randomly generated, akin to the strategy used for larger bi-objective instances. The associated maximum NFEs for these instances is presented subsequently in Table 5.1.

Table 5.1: Problem instance and number of function evaluation limit

Problem instance	**Number of function evaluation limit**
Three-objective	
MO knapsack 100	1×10^7
MO knapsack 250	1×10^7
MO knapsack 500	1.5×10^7
MO maxcut 50	5×10^6
MO maxcut 100	1.5×10^7
Five-objective	
MO knapsack 100	1×10^7
MO knapsack 250	1×10^7

For MIMP, we have selected six network datasets from the network repository [29], as listed in Table 5.2. This selection was made based on the number of nodes within each instance, which aligns with the high-dimensional nature of the problem within the range of 100 to 300. The specific instances chosen and the corresponding limits on their function evaluations are delineated in the Table 5.2.

Table 5.2: The statistical information and number of function evaluation limit of the considered six networks, where $|V|$ is number of nodes, $|E|$ is number of edges and $\bar{d}$ is the average degree.

	Dataset	$\lvert V\rvert$	$\lvert E\rvert$	$\bar{d}$	**NFE limit**
d1	mammalia-dolphin-social	62	159	5	2×10^7
d2	mammalia-voles-plj-trapping-27	125	229	8	2×10^7
d3	reptilia-tortoise-network-bsv	136	554	8	2×10^7
d4	enron-only (email network)	143	623	8	2×10^7
d5	mammalia-dolphin-floridatravel	188	1000	10	2×10^7
d6	reptilia-tortoise-network-fi-2008	283	418	2	5×10^6

Furthermore, the parameter values for ADSM*-MO-GOMEA remained consistent with the bi-objective settings. Specifically, we employed an initial p_e value of 0.9, along with the modifying parameter γ of 0.01.

doi:10.6342/NTU202303309

5.4 Empirical Results

This section adopts the same analysis procedure as used in the bi-objective scheme. We first present the results for the three-objective NPC, followed by the five-objective NPC problems, and ultimately, the MIMP results. The results demonstrate that ADSM* can also boost the performance of MO-GOMEA in scenarios with more than two objectives, particularly when dealing with larger problem sizes. These results underline the promising advantages of exploiting the relationships among objectives in problems with more than two objectives.

5.4.1 Comparison on Three-objective NPC Problems

The outcomes of the experiment are presented in Table 5.3. It is evident that the ADSM*-MO-GOMEA demonstrates superior performance over the original MO-GOMEA. However, there are certain instances where MO-GOMEA demonstrates superior performance over ADSM*-MO-GOMEA, as highlighted by the FO metric. Nevertheless, the divergence, as demonstrated by the lower IGD value, unequivocally signifies the superiority of ADSM*-MO-GOMEA, where a lower IGD implies a more pronounced pushing out of the Pareto front.

Observations from the IGD convergence plots pertaining to the knapsack, as shown in Figure 5.3, display an analogous trend over time as observed in the bi-objective scenario, wherein the divergence augments concomitantly with the escalation of NFE. Concerning the MO maxcut results, as shown in Figure 5.4, they exhibit the enhancements initiated by ADSM* commencing from relatively smaller problem dimensions. This resonates

doi:10.6342/NTU202303309

Table 5.3: Comparison of IGD and FO metrics for three-objective MO knapsack and MO maxcut obtained by ADSM*-MO-GOMEA and MO-GOMEA. Statistical significance is determined using a t-test to assess the differences between the means of the compared groups, where a lower p-value indicates higher significance.

Metric	Instance	MO-GOMEA		ADSM*-MO-GOMEA		p-value
		Mean	Std	Mean	Std	
MO knapsack						
IGD	100	11.58	0.87	**6.38**	0.49	3.1×10^{-54}
	250	72.26	5.02	**27.80**	1.14	1.3×10^{-75}
	500	198.65	12.79	**161.89**	5.46	7.8×10^{-79}
FO	100	1597.43	52.41	**2305.02**	45.83	7.5×10^{-76}
	250	**2226.87**	174.00	2012.75	136.84	9.3×10^{-9}
	500	1871.73	227.73	**2140.02**	102.52	2.2×10^{-11}
MO maxcut						
IGD	50	1.35	0.58	**1.30**	0.48	7.3×10^{-1}
	100	15.01	1.34	**14.12**	2.04	5.1×10^{-2}
FO	50	**861.10**	21.76	873.63	14.78	1.15×10^{-2}
	100	**1153.77**	51.93	1125.27	77.48	9.96×10^{-2}

with the performance witnessed in the bi-objective context of MO maxcut. Despite the complexity induced by overlapping structures or symmetric characteristics which make Pareto front exploration more challenging, it is notable that when the space encompassing the Pareto fronts expands, ADSM* potentially facilitates a more efficient advancement and exploration of the Pareto front.

doi:10.6342/NTU202303309

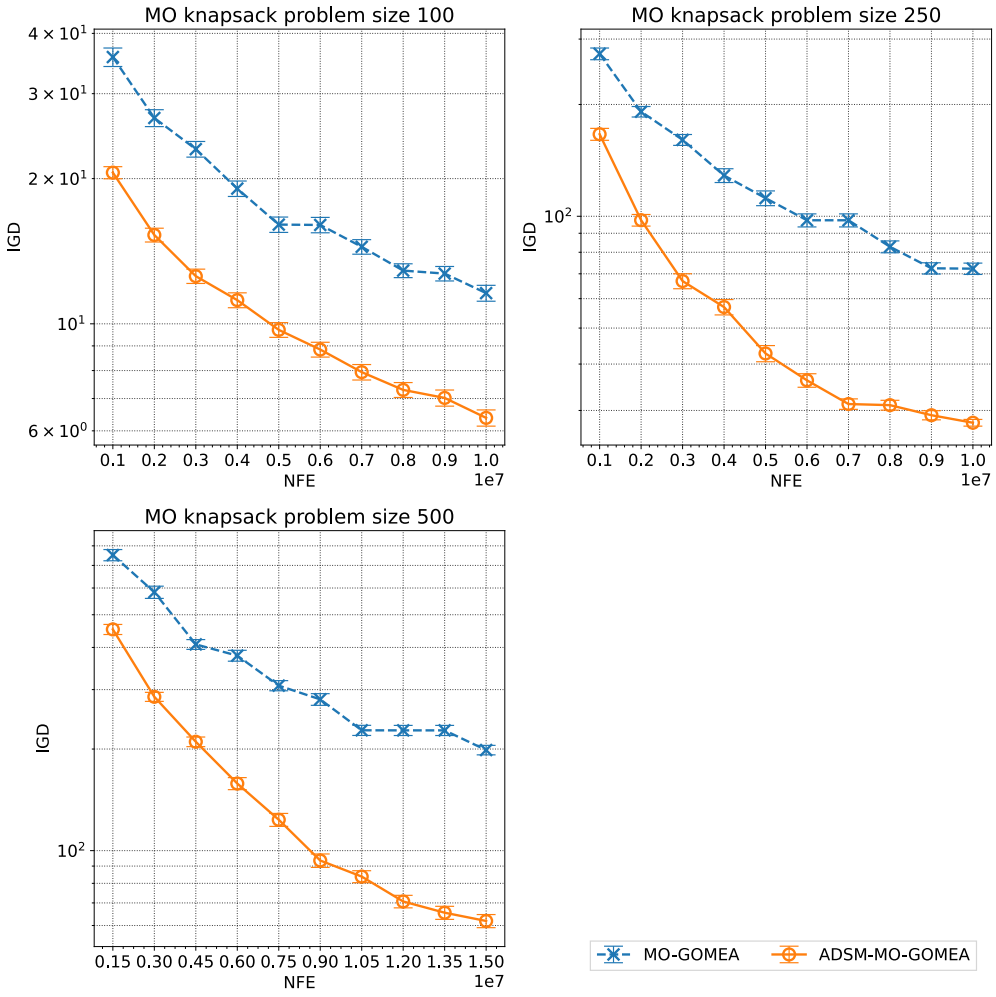


Figure 5.3: Performance of IGD convergence on three-objective MO knapsack, with the bars illustrating the interval of certainty at the 95% level.

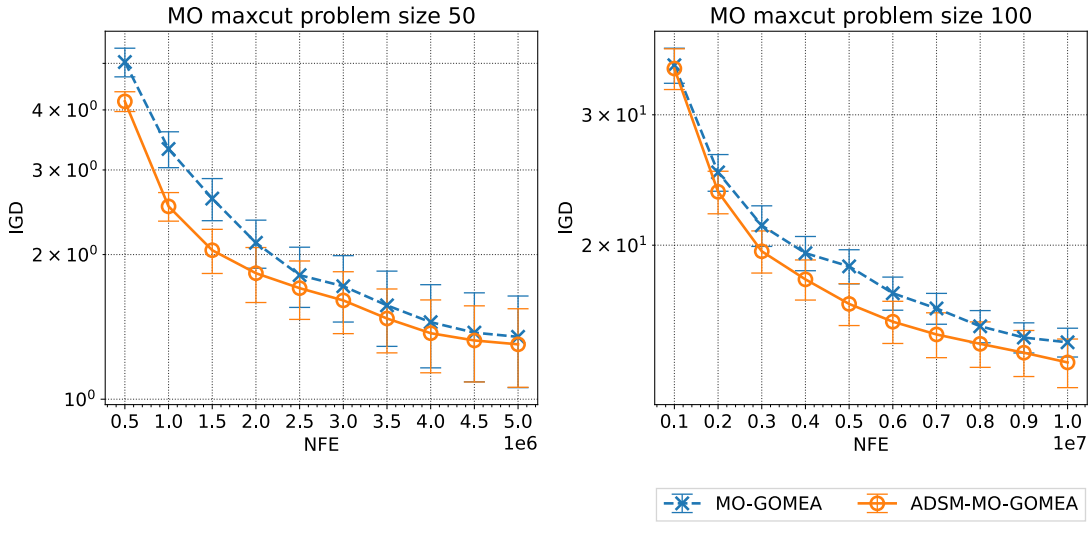


Figure 5.4: Performance of IGD convergence on three-objective MO maxcut, with the bars illustrating the interval of certainty at the 95% level.

doi:10.6342/NTU202303309

5.4.2 Comparison on Five-objective MO Knapsack

Regarding the five-objective scheme, its performance was evaluated on MO knapsack problems. The resulting data are depicted in Table 5.4, with corresponding Performance of IGD convergence illustrated in Figure 5.5. These outcomes suggest that the ADSM*-MO-GOMEA still maintains an advantage over the MO-GOMEA. Intriguingly, although ADSM*-MO-GOMEA exhibits superior performance over MO-GOMEA with fewer objectives, it generates comparable results to MO-GOMEA for the knapsack problem size of 100, incorporating five objectives.

Table 5.4: Comparison of IGD and FO metrics for five-objective MO knapsack obtained by ADSM*-MO-GOMEA and MO-GOMEA. Statistical significance is determined using a t-test to assess the differences between the means of the compared groups, where a lower p-value indicates higher significance.

Metric	Instance	MO-GOMEA		ADSM*-MO-GOMEA		p-value
		Mean	Std	Mean	Std	
IGD	100	**66.31**	1.94	77.27	2.22	9.3×10^{-1}
	250	293.44	12.82	**152.00**	5.62	1.9×10^{-48}
FO	100	**2277.57**	323.16	2189.11	291.77	2.9×10^{-1}
	250	2152.32	282.26	**2207.64**	206.13	4.0×10^{-1}

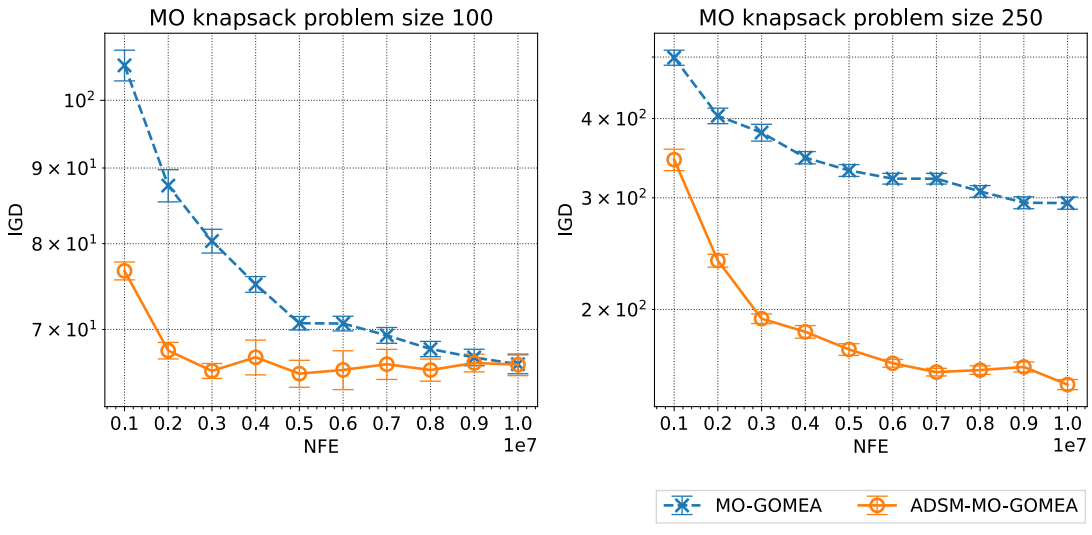


Figure 5.5: Performance of IGD convergence on five-objective MO knapsack, with the bars illustrating the interval of certainty at the 95% level.

doi:10.6342/NTU202303309

Further analysis was conducted on the approximate Pareto front, as demonstrated in Table 5.5. The ratio of the cumulative FO of each algorithm to the final combined FO, established as the ground truth, indicates that ADSM*-MO-GOMEA is capable of exploring a more substantial portion of the Pareto front, thereby revealing superior exploratory capability.

Table 5.5: Cumulative proportions in the ground truth of each algorithm

	MO-GOMEA	**ADSM*-MO-GOMEA**
Total Pareto front number	16105	
Cumulative FO	10839	11278
Proportion(cumulative/total)	48%	59%

5.4.3 Comparison on MIMP

Table 5.6 presents the outcomes across six problem instances, complemented by the corresponding IGD convergence plot in Figure 5.6. Notably, for smaller problem sizes, ADSM*-MO-GOMEA exhibits inferior performance relative to MO-GOMEA. For instance, with a problem size of 62, the divergence between ADSM*-MO-GOMEA and MO-GOMEA increases as NFE progresses. At a problem size of 125, both algorithms exhibit similar performance in the early stages, but ADSM*-MO-GOMEA degrades in the later stages.

Conversely, as the problem size escalates, the enhancements brought by ADSM to MO-GOMEA become apparent. For problem sizes of 136 and 188, the superior performance of ADSM*-MO-GOMEA is evident from the outset and persists as NFE progresses. With a problem size of 143, ADSM*-MO-GOMEA swiftly achieves a low IGD, ultimately aligning with MO-GOMEA. This convergence could be attributed to both algorithms converging towards either local or global optima.

doi:10.6342/NTU202303309

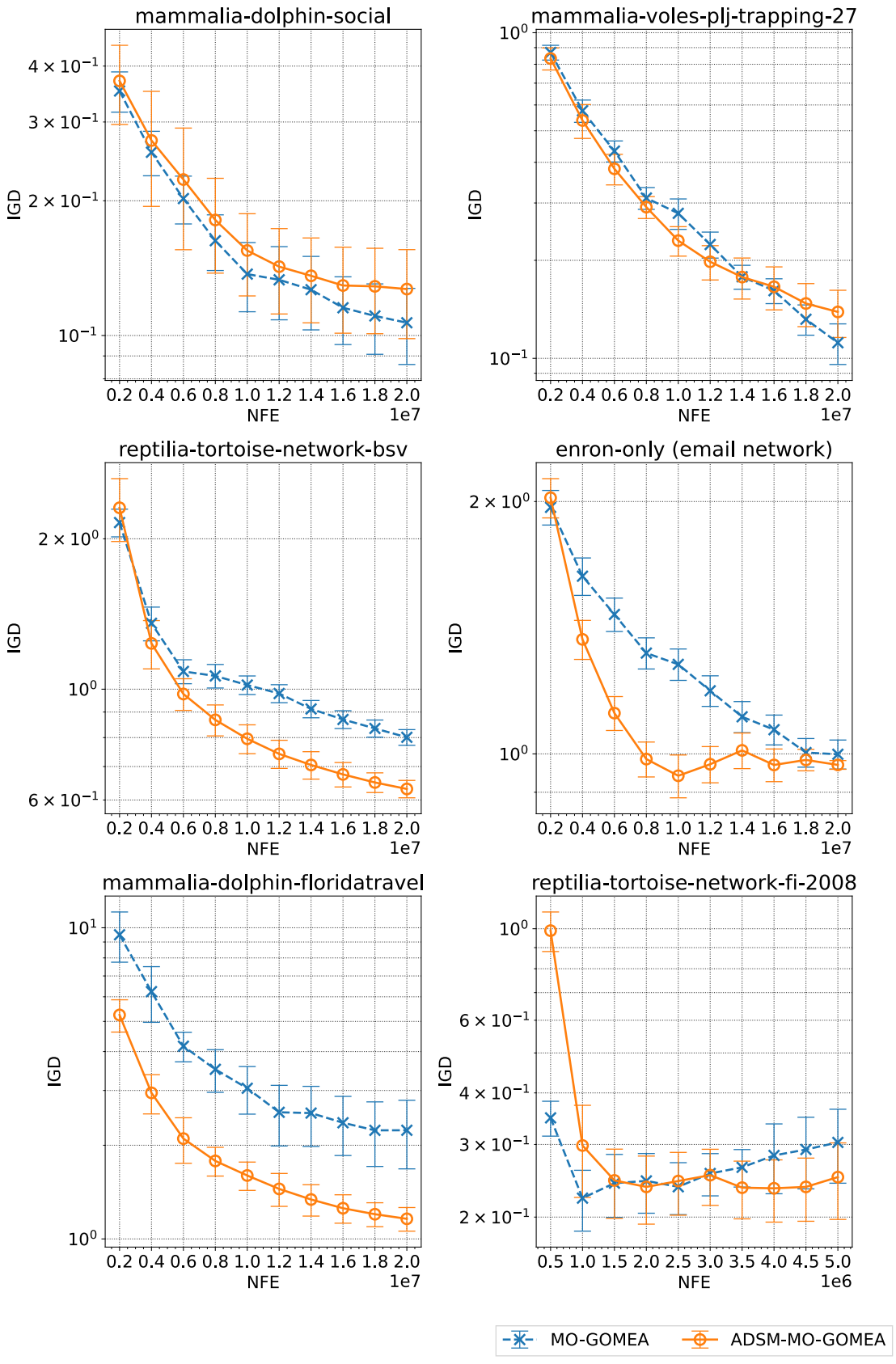


Figure 5.6: Performance of IGD convergence on MIMP, with the bars illustrating the interval of certainty at the 95% level.

doi:10.6342/NTU202303309

Table 5.6: Comparison of IGD and FO metrics for MIMP obtained by ADSM*-MO-GOMEA and MO-GOMEA. Statistical significance is determined using a t-test to assess the differences between the means of the compared groups, where a lower p-value indicates higher significance.

Metric	Instance	MO-GOMEA		ADSM*-MO-GOMEA		p-value
		Mean	Std	Mean	Std	
IGD	d1	**0.11**	0.04	0.13	0.06	2.7×10^{-1}
	d2	**0.11**	0.03	0.14	0.05	8.9×10^{-2}
	d3	0.80	0.06	**0.63**	0.05	2.1×10^{-9}
	d4	1.00	0.08	**0.97**	0.02	1.9×10^{-1}
	d5	2.23	1.10	**1.16**	0.20	1.3×10^{-3}
	d6	0.30	0.12	**0.25**	0.10	2.08×10^{-1}
FO	d1	**651.93**	13.57	649.27	20.25	6.75×10^{-1}
	d2	**2034.62**	41.80	2008.85	66.65	2.49×10^{-1}
	d3	1959.27	61.30	**2153.73**	57.67	1.05×10^{-9}
	d4	**2290.13**	80.38	1925.53	64.36	6.01×10^{-14}
	d5	1818.36	253.13	**1947.43**	84.06	8.18×10^{-2}
	d6	204.07	6.45	**206.27**	6.13	3.47×10^{-1}

doi:10.6342/NTU202303309

Chapter 6 Conclusion

In this thesis, the concept of building block contributions, drawn from single-objective optimization, has been applied to partition the objective space of bi-objective problems. This partitioning is based on whether the offspring displays coherent, solitary, or incoherent improvements in objective values. Through establishing a promising donor boundary for each receiver chromosome, we were able to guide the progression of the approximate front, enabling us to prioritize the generation of offspring towards region with coherent improvement. Furthermore, a self-adaptive parameter was implemented to dynamically balance the utilization of elitist-guided and cluster-guided mixing, facilitating an adaptive response to the different stages of evolution and problem context. This gave rise to the development of the ADSM method. Integrating ADSM within the framework of MO-GOMEA significantly amplified the efficacy of the optimization procedure, especially in the context of tackling problems with larger problem sizes.

While ADSM was initially designed for bi-objective optimization, preliminary experiments were conducted on leveraging the inter-relationship among objectives, particularly coherent and incoherent improvements, within problems characterized by three and five objectives. The outcomes indicated promising enhancements to the original algorithm. For future work, there is a need for a more in-depth investigation to accurately define the various regions where enhanced offspring fall, in relation to alterations in ob-

doi:10.6342/NTU202303309

jective values. Furthermore, with an increase in the number of objectives, the complexity of inter-objective relationships could escalate exponentially, going beyond mere pairwise interactions, necessitating further research.

In conclusion, this thesis explores the significance of building blocks in contributing to each objective and their inter-relationship across different objectives during the evolutionary process in multi-objective optimization. It highlights the significance of distinguishing coherent, solitary, and incoherent improvements, as well as the necessity to adapt the attention on these improvements according to the evolutionary stage and problem context. A new mixing mechanism to cater to this attribute is introduced in this thesis. The findings point towards the ubiquity of this attribute in numerous multi-objective optimization problems, thereby indicating the critical need for its understanding and analysis in future advancements of multi-objective optimization algorithms.

doi:10.6342/NTU202303309

References

[1] F. Altiparmak, M. Gen, L. Lin, and T. Paksoy. A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & industrial engineering*, 51(1):196–215, 2006.

[2] F. Barahona, M. Grötschel, M. Jünger, and G. Reinelt. An application of combinatorial optimization to statistical physics and circuit layout design. *Operations Research*, 36(3):493–513, 1988.

[3] T. K. Biswas, A. Abbasi, and R. K. Chakrabortty. An improved clustering based multi-objective evolutionary algorithm for influence maximization under variable-length solutions. *Knowledge-Based Systems*, 256:109856, 2022.

[4] P. A. Bosman. The anticipated mean shift and cluster registration in mixture-based EDAs for multi-objective optimization. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 351–358, 2010.

[5] P. A. Bosman and D. Thierens. Linkage neighbors, optimal mixing and forced improvements in genetic algorithms. In *Proceedings of the 14th annual conference on Genetic and evolutionary computation*, pages 585–592, 2012.

[6] P. A. Bosman and D. Thierens. More concise and robust linkage learning by filtering

doi:10.6342/NTU202303309

and combining linkage hierarchies. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation*, pages 359–366, 2013.

[7] P.-L. Chen, C.-J. Peng, C.-Y. Lu, and T.-L. Yu. Two-edge graphical linkage model for DSMGA-II. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 745–752, 2017.

[8] K. Deb and D. E. Goldberg. Sufficient conditions for deceptive and easy binary functions. *Annals of mathematics and Artificial Intelligence*, 10:385–408, 1994.

[9] K. Deb and H. Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE transactions on evolutionary computation*, 18(4):577–601, 2013.

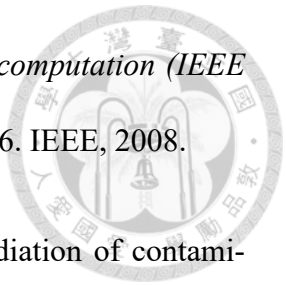
[10] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[11] G. R. Harik, F. G. Lobo, and K. Sastry. Linkage learning via probabilistic modeling in the extended compact genetic algorithm (ecga). In *Scalable optimization via probabilistic modeling*, pages 39–61. Springer, 2006.

[12] H. Ishibuchi, N. Akedo, and Y. Nojima. A study on the specification of a scalarizing function in MOEA/D for many-objective knapsack problems. In *Learning and Intelligent Optimization: 7th International Conference, LION 7, Catania, Italy, January 7-11, 2013, Revised Selected Papers 7*, pages 231–246. Springer, 2013.

[13] H. Ishibuchi, N. Tsukamoto, and Y. Nojima. Evolutionary many-objective optimiza-

doi:10.6342/NTU202303309



tion: A short review. In *2008 IEEE congress on evolutionary computation (IEEE world congress on computational intelligence)*, pages 2419–2426. IEEE, 2008.

[14] L. Jenkins. A bicriteria knapsack program for planning remediation of contaminated lightstation sites. *European Journal of Operational Research*, 140(2):427–433, 2002.

[15] N. Khan, D. E. Goldberg, and M. Pelikan. Multi-objective Bayesian optimization algorithm. In *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation*, pages 684–684, 2002.

[16] M. M. Kostreva, W. Ogryczak, and D. W. Tonkyn. Relocation problems arising in conservation biology. *Computers & Mathematics with Applications*, 37(4-5):135–150, 1999.

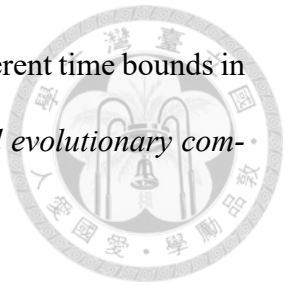
[17] N. H. Luong and P. A. Bosman. Elitist archiving for multi-objective evolutionary algorithms: To adapt or not to adapt. In *International Conference on Parallel Problem Solving from Nature*, pages 72–81. Springer, 2012.

[18] N. H. Luong, H. La Poutré, and P. A. Bosman. Multi-objective gene-pool optimal mixing evolutionary algorithms. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 357–364, 2014.

[19] N. H. Luong, H. La Poutré, and P. A. Bosman. Multi-objective gene-pool optimal mixing evolutionary algorithm with the interleaved multi-start scheme. *Swarm and Evolutionary Computation*, 40:238–254, 2018.

[20] R. T. Marler and J. S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26:369–395, 2004.

doi:10.6342/NTU202303309



[21] A. Mohammadi and M. Saraee. Finding influential users for different time bounds in social networks using multi-objective optimization. *Swarm and evolutionary computation*, 40:158–165, 2018.

[22] R. Olivares, F. Muñoz, and F. Riquelme. A multi-objective linear threshold influence spread model solved by swarm intelligence-based methods. *Knowledge-Based Systems*, 212:106623, 2021.

[23] M. Olsthoorn and A. Panichella. Multi-objective test case selection through linkage learning-based crossover. In *International Symposium on Search Based Software Engineering*, pages 87–102. Springer, 2021.

[24] M. Pelikan and D. E. Goldberg. Escaping hierarchical traps with competent genetic algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, pages 511–518, 2001.

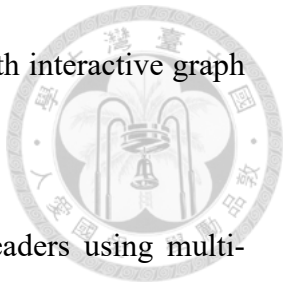
[25] M. Pelikan, D. E. Goldberg, and K. Sastry. Bayesian optimization algorithm, decision graphs, and Occam' s razor. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 519526, 2001.

[26] M. Pelikan, K. Sastry, and D. E. Goldberg. Multiobjective hboa, clustering, and scalability. In *Proceedings of the 7th annual conference on Genetic and evolutionary computation*, pages 663–670, 2005.

[27] J. F. Robles, M. Chica, and O. Cordon. Evolutionary multiobjective optimization to target social network influentials in viral marketing. *Expert systems with applications*, 147:113183, 2020.

[28] M. J. Rosenblatt and Z. Sinuany-Stern. Generating the discrete efficient frontier to the capital budgeting problem. *Operations Research*, 37(3):384–394, 1989.

doi:10.6342/NTU202303309



[29] R. A. Rossi and N. K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015.

[30] A. Sheikhahmadi and A. Zareie. Identifying influential spreaders using multi-objective artificial bee colony optimization. *Applied Soft Computing*, 94:106436, 2020.

[31] M. G. C. Tapia and C. A. Coello. Applications of multi-objective evolutionary algorithms in economics and finance: A survey. In *2007 IEEE congress on evolutionary computation*, pages 532–539. IEEE, 2007.

[32] D. Thierens. The linkage tree genetic algorithm. In *International Conference on Parallel Problem Solving from Nature*, pages 264–273. Springer, 2010.

[33] D. Thierens and P. A. Bosman. Optimal mixing evolutionary algorithms. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, pages 617–624, 2011.

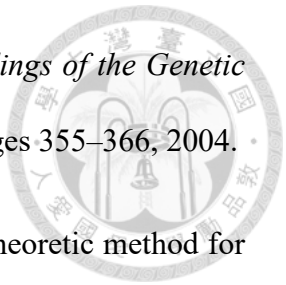
[34] D. A. Van Veldhuizen and G. B. Lamont. Multiobjective optimization with messy genetic algorithms. In *Proceedings of the 2000 ACM symposium on Applied computing-Volume 1*, pages 470–476, 2000.

[35] S. Verma, M. Pant, and V. Snasel. A comprehensive review on nsga-ii for multi-objective combinatorial optimization problems. *IEEE Access*, 9:57757–57791, 2021.

[36] R. A. Watson, G. S. Hornby, and J. B. Pollack. Modeling building-block interdependency. In *International Conference on Parallel Problem Solving from Nature*, pages 97–106. Springer, 1998.

[37] T.-L. Yu and D. E. Goldberg. Dependency structure matrix analysis: Offline utility

doi:10.6342/NTU202303309



of the dependency structure matrix genetic algorithm. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2004)*, pages 355–366, 2004.

[38] T.-L. Yu, A. A. Yassine, and D. E. Goldberg. An information theoretic method for developing modular architectures using genetic algorithms. *Research in Engineering Design*, 18:91–109, 2007.

[39] Q. Zhang and H. Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731, 2007.

[40] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK-report*, 103, 2001.

[41] E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4):257–271, 1999.

[42] J. B. Zydallis and G. B. Lamont. Explicit building-block multiobjective evolutionary algorithms for npc problems. In *The 2003 Congress on Evolutionary Computation, 2003. CEC'03.*, volume 4, pages 2685–2695. IEEE, 2003.

doi:10.6342/NTU202303309