Egyptian Informatics Journal 25 (2024) 100422

Contents lists available at [ScienceDirect](http://www.ScienceDirect.com/)

Egyptian Informatics Journal

journal homepage: [www.sciencedirect.com](http://www.sciencedirect.com/)

A niche-based evolutionary algorithm with dual cooperative archive for  solving constrained multi-objective optimization problems

Fengyu Guo [a](#_bookmark0), Hecheng Li [b](#_bookmark1)*,*[∗](#_bookmark2)

a *School of Computer Science and Technology, Qinghai Normal University, Xining, 810016, China*

b *School of Mathematics and Statistics, Qinghai Normal University, Xining, 810016, China*

A R T I C L E I N F O A B S T R A C T

*Keywords:*

Constrained multi-objective optimization Dual stage algorithm

Infeasible individual utilization Niche-based selection method

Constrained multi-objective optimization problems (CMOPs) are commonly encountered in engineering practice. The key to effectively solving these problems lies in achieving a timely balance between convergence, diversity, and feasibility during iterations. Furthermore, the appropriate utilization of infeasible solutions is crucial for identifying potential feasible regions. In order to accomplish this comprehensive objective, we propose a novel dual-stage constrained multi-objective evolutionary algorithm (CMOEA) called NACMOEA in this paper. It can be characterized by the following features: 1) Introducing a novel niche-based individual selection and infeasible solution utilization strategy to enhance convergence, diversity, and feasibility. 2) Presenting a cooperative search strategy assisted by dual archives to approximate the constrained Pareto front (CPF) from both feasible and infeasible perspectives, thereby improving the eﬃciency of obtaining the complete CPF. 3) Designing a new stage switch method based on non-dominant coverage rate to ensure proper completion of search stage switching. Extensive experiments demonstrate that NACMOEA exhibits competitive comprehensive performance when compared with other advanced CMOEAs.

## Introduction

CMOPs are widespread in scientific and engineering practice, such as front rail design [[6](#_bookmark41)], pressure vessel design [[12](#_bookmark52)], etc. To solve such problems normally requires CMOEAs to optimize multiple conflicting objectives with constraint restrictions. A general mathematical expres- sion of CMOPs can be formulated as follows:

minimize *𝐹* (**𝐱**)= (*𝑓*1(**𝐱**)*,* … *, 𝑓𝑚*(**𝐱**))*𝑇*

subject to *𝑔* (**𝐱**) ≤ 0*, 𝑗* = 1*,* … *, 𝑝*

the overall constraint violation is defined as:

∑*𝑞*

Φ(**𝐱**)= *𝐶𝑉𝑗* (**𝐱**) (3)

*𝑗*=1

is 0; otherwise, it is considered infeasible. To compare the performance a solution is called a feasible solution when its total constraint violation

Given feasible solutions **𝐱** and **𝐲**, **𝐱** is said to Pareto dominate **𝐲** (denoted of solutions for CMOPs, we introduce the concept of Pareto dominance. as **𝐱** *≺* **𝐲**) if *𝑓𝑘*(**𝐱**) ≤ *𝑓𝑘*(**𝐲**) for all *𝑘* ∈ (1*, ..., 𝑚*) and *𝑓𝑘*(**𝐱**) *< 𝑓𝑘*(**𝐲**) for at

*𝑗*

*ℎ𝑗* (**𝐱**)= 0*, 𝑗* = *𝑝* + 1*,* … *, 𝑞*

**𝐱** ∈Ω

(1)

least one *𝑘*. A solution is considered Pareto optimal when no other fea-

sible solution dominates it. The set of all Pareto optimal solutions in the

search space is known as the Pareto optimal set (PS). The representation of PS in the objective space forms the Pareto optimal front (PF).

where **𝐱** is a decision variable vector in the search space Ω, *𝐹* (**𝐱**) is an objective vector with *𝑚* objective function values. *𝑔𝑗* (**𝐱**)(*ℎ𝑗* (**𝐱**)) are the

*𝑗*th inequality (equality constraint). The constraint violation of **𝐱** for the

*𝑗*th constraint can be calculated as follows:

{ { }

When traditional optimization algorithms are adopted to handle CMOPs, they face computational challenges that the algorithms have to identify the feasible region and keep diversity of individuals for the pur- pose of convergence. In fact, in most of CMOPs, constraints can always

cause the search space infeasible, especially, some tight constraints can

*𝐶𝑉* (**𝐱**)=

max {0*, 𝑔𝑗* (**𝐱**)

*, 𝑗* = 1*,* … *, 𝑝,*

(2)

make the feasible region overly narrow and even isolated. As a result,

*𝑗* max |

0*, ℎ𝑗*

(**𝐱**)

|

|

}

| *, 𝑗*

|

=

|

*𝑝* + 1*,* …

*, 𝑞*

the algorithm has diﬃculty striking a balance between feasibility and

\* Corresponding author.

*E-mail addresses:* [fengyu\_guo1990@163.com](mailto:fengyu_guo1990@163.com) (F. Guo), [lihecheng@qhnu.edu.cn](mailto:lihecheng@qhnu.edu.cn) (H. Li).

<https://doi.org/10.1016/j.eij.2023.100422>

Received 30 September 2023; Received in revised form 9 November 2023; Accepted 20 November 2023

Available online 28 November 2023

1110-8665/© 2023 THE AUTHORS. Published by Elsevier BV on behalf of Faculty of Computers and Artificial Intelligence, Cairo University. This is an open access article under the CC BY-NC-ND license ([http://creativecommons.org/licenses/by-nc-nd/4.0/).](http://creativecommons.org/licenses/by-nc-nd/4.0/))

convergence, easily getting stuck in a local optimum and failing to lo- cate the full CPF.

A well-established technique for global optimization with great ro- bustness and broad applicability is the multi-objective evolutionary algorithm (MOEA). MOEAs offer a collection of non-dominated approx- imately optimal solutions for multiple conflicting goals. Hence, they can handle complicated situations that are challenging for classical op- timization procedures. In all popular categories of MOEAs, there are three representatives. Dominance-based MOEAs are categorized as the first category, such as NSGA-II [[5](#_bookmark39)] and SPEA2 [[41](#_bookmark64)]. In the second cate- gory of MOEAs, an objective decomposition scheme is adopted by using Chebyshev’s formula, such as MOEA/D [[37](#_bookmark56)] and MOEA/DD [[13](#_bookmark54)]. And the third is indicator-based MOEAs such as IBEA [[40](#_bookmark62)] and HypE [[3](#_bookmark36)]. These MOEAs can effectively handle unconstrained multi-objective op- timization problems, but they have some restrictions when applied to CMOPs, some or even all of the optimal solutions found by these al- gorithms lie in infeasible regions that are invalid for CMOPs. In order to overcome the shortcomings, the research community has developed several different constraint handling techniques (CHTs) to assist MOEAs of this kind to search for optimal feasible solutions. CHTs can be viewed as a selection strategy for handling constraints, which is used to select some infeasible solutions with low constraint violation. These infeasi- ble solutions enable MOEAs to have the ability of crossing infeasible regions and eventually converge to the CPF.

The research on CMOEAs has attracted wide attention in the aca- demic community and has made significant progress in recent years. However, there are still some issues in this field that need to be ad- dressed. The first issue worth considering is how to better balance the algorithm’s convergence, diversity, and feasibility. Due to the influence of constraints, achieving a balance in these characteristics is challenging for CMOEAs. Consequently, the algorithm is prone to getting stuck in a local feasible or optimal region and failing to discover all feasible re- gions, which hinders its convergence towards the CPF or only allows it to find a portion of it. The second problem is how to effectively utilize information from infeasible solutions. Specifically, identifying infeasi- ble solutions located near undiscovered feasible regions can enhance the algorithm’s exploration ability. Additionally, utilizing high-convergence infeasible solutions located near the feasible region can expedite the convergence of feasible individuals.

To address the aforementioned issue, this paper proposes a niche- based two-stage evolutionary algorithm called NACMOEA. This algo- rithm is supported by dual collaborative archives and incorporates a novel strategy for utilizing infeasible solutions. The key characteristics of the proposed approach are outlined below.

1. The search process of the proposed NACMOEA is divided into two stages, during which a new dual archive collaboration strategy is developed.
2. A niche-based individual selection and infeasible individual uti- lization strategy is proposed to better balance the convergence, diversity, and feasibility of NACMOEA.
3. A novel non-dominant coverage rate metric is designed for switch- ing the search stage.
4. The proposed algorithm exhibits competitive performance com- pared with other advanced algorithms on a large number of com- plex CMOP test problems.

The remainder of the article is organized as follows. In Section [2](#_bookmark3), we briefly review the recent literature of CMOEAs and introduce the com- mon CHTs. The motivation for this paper is subsequently elicited. The specific mechanisms of the proposed algorithm are described in detail in Section [3](#_bookmark5), followed by a description of the experimental studies in Section [4](#_bookmark18). Finally, the conclusions and future work are presented in Section [5](#_bookmark26).

## Background

This section provides a review of existing CMOEAs, with a focus on analyzing CHTs in CMOEAs, comparing the advantages and weaknesses of several common CHTs, and presenting the motivation for this paper.

* 1. *Literature review*

Significant progress has been made in the academic community’s research on CMOP, with various approaches proposed. These include those based on multiple populations, utilization of infeasible solutions, multi-stage strategies, and the balancing of objectives and constraints. The subsequent section presents an overview of representative literature concerning these methodologies.

* + 1. *CMOEAs based on multi-population/archive*

The CHT based on a dual-archive set was first presented by C- TAEA [[14](#_bookmark57)]. In this approach, the two archive sets have complementary roles to ensure population diversity while promoting population con- vergence and feasibility. In CCMO [[30](#_bookmark42)] framework, a dual-population co-evolutionary strategy based on a weak cooperative relationship is provided. In this strategy, two sub-populations evolve independently and share their offspring in their respective environmental selection stages. The dual-population-based evolutionary algorithm proposed in c-DPEA [[24](#_bookmark27)] uses an adaptive penalty function to assist evolution using information from infeasible solutions and creates a new ob- jective function to balance population convergence and diversity. In BiCo [[15](#_bookmark59)], a bidirectional co-evolutionary strategy is introduced to ap- proximate the CPF from both the feasible and infeasible sides. The CMOCSO [[22](#_bookmark28)] framework solves CMOPs through a competition and cooperation mechanism between two swarm optimizers, effectively improving convergence. For constrained many-objective optimization problems, the MOEA/D-DAE [[39](#_bookmark60)] framework offers a detection and es- cape mechanism that uses the overall constraint violation rate of change to determine whether the population is trapped in a local optimum. In MFO [[10](#_bookmark48)], the constraints of the original CMOP are relaxed to create a simpler auxiliary problem; transferring experience gained from solving this auxiliary problem can be helpful in solving the original problem.

* + 1. *CMOEAs using infeasible solutions*

In C-AnEA [[33](#_bookmark49)], an angle-based constraint dominance rule and den- sity estimation method are provided, which give the infeasible solutions with excellent convergence a chance to survive. This helps the popula- tion cross the infeasible regions. In ShiP [[19](#_bookmark65)], a shift-based penalty is imposed on infeasible solutions, retaining those that do not fall into the region dominated by Locally Feasible Nadir after the shift. Inspired by multi-task evolution, the MTCMO [[26](#_bookmark31)] framework uses a highly related dynamic auxiliary task for the complex original problem and designs an improved constraint relaxation method to leverage high-quality infeasi- ble solutions.

* + 1. *CMOEA based on a multi-stage strategy*

while the *𝜖*-constrained technique-based search is defined as the pull In PPS [[8](#_bookmark44)], the unconstrained search is defined as the push stage,

stage. The PPTA [[27](#_bookmark34)] framework presents a push-pull search strategy based on the assistance of a dual archive, which effectively improves the performance of PPS. The DTAEA [[34](#_bookmark50)] framework suggests a two-phase approach, where the first phase conducts a dual population search based on weak co-evolution and the second phase conducts a feasibility- oriented single population search that drives the population to converge to the final CPF. In DD-CMOEA [[25](#_bookmark32)], a dual-stage and dual-population strategy is employed to enhance the search capability of the algorithm by implementing different divisions of labor and collaboration strate- gies for each population in different stages. In FNDS [[36](#_bookmark55)], a population partitioning method based on feasible non-dominated solution sets is proposed, which dynamically divides the entire population into three

mutually exclusive subsets and performs different tasks for these sub- sets to guide them towards approaching CPF.

* + 1. *CMOEA based on objective and constraint balancing strategies*

In CMOEA-MS [[31](#_bookmark45)], a new strategy for evaluating objective func- tions is provided to achieve a better balance between objectives and constraints by assigning different priorities to them. The TOR [[20](#_bookmark68)] framework rewrites the original fitness function as a sum of two rank- ings based on CDP and Pareto dominance, controlling the weights through feasible rate to achieve a balance between objectives and con- straints. The CMME [[23](#_bookmark29)] framework effectively addresses the issue of decreasing selection pressure on populations as the number of ob- jectives increases by employing two ranking strategies. DPSEA [[32](#_bookmark46)]

dynamically adjusts the trade-off between objectives and constraints

violation. The *𝜖*-constrained technique overcomes the shortcomings of

CDP to a certain extent, enhancing the exploration ability of CMOEA,

the parameter *𝜖* plays a key role in the search process of CMOEA, too and is widely used in MOEA/D-DAE [[39](#_bookmark60)] and I-DBEA [[2](#_bookmark35)]. Noting that large *𝜖* can cause infeasible solutions far from the feasible region to be selected, and too small value of *𝜖* possibly causes similar problems as

CDP.

*2.2.3. Two-rank fitness method*

A new fitness is originally created in TOR [[20](#_bookmark68)], where two ranks

*𝑅𝑝*(**𝐱**) and *𝑅𝑐* (**𝐱**) of solution **𝐱** are calculated based on Pareto dominance and CDP, respectively. Then, a new fitness *𝐹* (*̂***𝐱**) is calculated by taking

the weighted sum of these two ranks.

through population size.

* 1. *Common CHTs*

min

*𝐹̂*(**𝐱**)= *𝛼𝑅𝑐* (**𝐱**)+ (1 − *𝛼*)*𝑅𝑝*(**𝐱**)

*𝛼* = 0*.*5+ 0*.*5*𝑃𝑓*

(5)

The MOEA research community has recently attempted to eﬃ- ciently solve CMOPs by combining different MOEAs with multiple types of CHTs. CHTs, which are a crucial technique for handling con- straints, play a particularly critical role in improving the performance of CMOEAs. Some representative CHTs are introduced as follows.

* + 1. *Constraint domination principle (CDP)*

CDP is the most representative CHT employed in CMOEAs, which was originally proposed by Deb et al. in NSGA-II [[5](#_bookmark39)]. It classifies the population into three categories and assigns a higher selection priority to solutions with high feasibility and excellent convergence. The ben-

apply. Assuming that **𝐱** and **𝐲** are two solutions, we say that **𝐱** con- efit of CDP is that its driving concept is straightforward and simple to strained dominates **𝐲** (denoted by **𝐱** *≺𝐶𝐷𝑃* **𝐲**) when one of the following

conditions is satisfied.

* + - * Φ(**𝐱**) = 0*,* Φ(**𝐲**) ≠ 0;
      * Φ(**𝐱**) = 0*,* Φ(**𝐲**) = 0*, 𝑎𝑛𝑑* **𝐱** *≺* **𝐲**;
      * Φ(**𝐱**) ≠ 0*,* Φ(**𝐲**) ≠ 0*, 𝑎𝑛𝑑* Φ(**𝐱**) *<* Φ(**𝐲**).

CDP is widely used in CMOEAs, such as NSGA-II [[5](#_bookmark39)] and C-NSGA-III [[9](#_bookmark47)]. However, due to its preference for feasible solutions, the population can easily get trapped in the local optimum region, thereby ignoring the benefits of infeasible solutions.

* + 1. *𝜖-constrained technique*

The *𝜖*-constrained technique is another typical CHT that stems from

factor *𝜖*. For an infeasible solution **𝐱**, if its overall constraint violations a partial improvement of CDP, which introduces a constraint relaxation Φ(**𝐱**) are less than *𝜖*, the solution can still survive, and then these infea-

For two solutions **𝐱** and **𝐲**, we say that **𝐱** *𝜖*-constrained dominates **𝐲** sible solutions are thereby incorporated into the evolutionary process. (denoted by **𝐱** *≺𝜖* **𝐲**) when one of the following conditions is satisfied.

* + - * Φ(**𝐱**) *< 𝜖,* Φ(**𝐲**) *< 𝜖, 𝑎𝑛𝑑* **𝐱** *≺* **𝐲**;
      * Φ(**𝐱**) = Φ(**𝐲**) = *𝜖, 𝑎𝑛𝑑* **𝐱** *≺* **𝐲**;
      * Φ(**𝐱**) *> 𝜖,* Φ(**𝐲**) *> 𝜖, 𝑎𝑛𝑑* Φ(**𝐱**) *<* Φ(**𝐲**).

An improved strategy to adaptively control *𝜖* is designed in [[8](#_bookmark44)], which is formulated as:

{

(1 − *𝜏*)*𝜀*(*𝑘* − 1)*,* if *𝑟𝑘 < 𝛼*

The weight is adaptively controlled by the feasible rate *𝑃𝑓* . However,

when the proportion of feasible solutions is very low, this approach

functions similarly to CDP. Consequently, the algorithm can easily be- come trapped in a local optimum.

The academic community has also recommended some additional CHTs besides the previously mentioned ones. The C3M [[28](#_bookmark37)] framework determines the priority of constraints by analyzing the relationship be- tween multiple constraints, and deals with constraints according to their priority. The NRC [[16](#_bookmark61)] framework proposed a multi-objective method based CHT, which uses three procedures of non-dominated sorting, re- versed non-dominated sorting, and constrained crowding distance sort- ing to move the population toward to CPF. In MOEA/D-DPF [[21](#_bookmark30)], a dynamic penalty function approach is developed to solve CMOPs that is integrated into the MOEA/D framework in which the parameters gradually change with generation number. In TiGE-2 [[38](#_bookmark58)], with the aim of achieving the balance of performances, three distinct indicators are developed while considering the convergence, diversity, and fea- sibility. These indicators are then co-optimized using multi-objective techniques.

* 1. *Motivation*

From the literature survey conducted above, it is evident that cur- rent CMOEAs only focus on either the algorithm’s balance scheme or the utilization of infeasible solutions when dealing with CMOPs. How- ever, they fail to effectively combine these two strategies. Furthermore, existing CMOEAs lack the capability to explore extensively within the infeasible region and obtain a comprehensive CPF. Infeasible solutions play a vital role in discovering unexplored feasible regions and enhanc- ing algorithm convergence; however, their potential advantage remains underutilized. Based on the above discussions, we propose employing the following strategies to address the limitations of existing CMOEAs:

1. A dual archive procedure, consisting of a convergence archive (CA) and a feasibility archive (FA), is established for assignment division and collaboration purposes. The CA retains the UPF discovered by the algorithm, which is utilized to create angle-based niches. These niches are then employed to partition the objective space and de- termine the spatial relationship between individuals. As for FA, it functions as the output population responsible for storing elite so- lutions near the CPF. It employs a niche-based individual updating strategy to strike a balance among convergence, feasibility, and di-

versity objectives.

*𝜀*(*𝑘*)=

*𝜀*(0)

(1− *𝑘* )*𝑐𝑝*

*𝑇𝑐*

*,* if *𝑟𝑘* ≥ *𝛼*

(4)

1. The precise utilization of infeasible solutions facilitates the explo- ration of unknown feasible regions and enhances the convergence

where *𝑘* represents the *𝑘*th generation, *𝑟𝑘* is the feasible ratio, *𝜏* and *𝛼*

respectively. *𝑐𝑝*, usually assigned 2, is used to control the speed of re- are two control parameters which are usually assigned 0.05 and 0.95, ducing *𝜖* value in the case of *𝑟𝑘* ≥ *𝛼*, and *𝜖*(0) is the maximal constraint

of feasible solutions. We strategically pair infeasible solutions with optimal feasibility and feasible solutions with optimal convergence within the same niche, followed by executing a crossover opera- tion for these matched individuals. The resulting offspring exhibit

promising potential to simultaneously improve both solution feasi- bility and convergence.

The algorithm effectively achieves a balance between convergence, diversity, and feasibility while eﬃciently utilizing infeasible solutions. The subsequent section presents a comprehensive account of the pro- posed NACMOEA algorithm.

**Algorithm 1:** Framework of NACMOEA.

**Input:** population size: *𝑁* ; maximum generation: *𝐺𝑚* ; non-dominant coverage threshold: *𝐶𝑡* ; generation gap: *𝐺𝑝* ;

**Output:** *𝐹𝐴*

**1** *𝑡* ← 1, *𝑠𝑡𝑎𝑔𝑒* ← 1;

**2** *𝑃* ← Random initialize {**𝐱**1*,* **𝐱**2*, ...,* **𝐱***𝑁* } ;

**3** *𝐶𝐴, 𝐹𝐴* ← *𝑃* ;

**4 while** *𝑡* ≤ *𝐺𝑚* **do**

**5 if** *𝑡 > 𝐺𝑝* **then**

1. **** *𝐶* ← nondominant coverage of *𝑃* by Equation ([6](#_bookmark6)) ;
2. **if** *𝐶* ≥ *𝐶𝑡* &*𝑠𝑡𝑎𝑔𝑒* == 1 **then**
3. **** *𝑠𝑡𝑎𝑔𝑒* ← 2 ;
4. **if** *𝑠𝑡𝑎𝑔𝑒==1* **then**
5. *𝑂𝑝* ← Select mating parents from *𝑃* and generate *𝑁* offspring ;
6. *𝑃* ← EnvironmentalSelection([*𝑃 , 𝑂𝑝* ]*, 𝑁, 𝑠𝑡𝑎𝑔𝑒*) ;
7. *𝐶𝐴* ← UpdateCA([*𝐶𝐴, 𝑃* ], *𝑁* ) ;
8. *𝐹𝐴* ← UpdateFA([*𝐹𝐴, 𝑃* ], *𝑁* , *𝑠𝑡𝑎𝑔𝑒*) ;
9. *𝑡* ← *𝑡* +1 ;

**15 if** *𝑠𝑡𝑎𝑔𝑒==2* **then**

1. *𝑂𝑝* ← Select mating parents from *𝑃* and generate *𝑁* offspring ;
2. *𝑂𝑓𝑎* ← Select mating parents from *𝐹𝐴* and generate *𝑁* ∕2 offspring ;
3. *𝑂𝑟* ← using InfeasibleUtilization(*𝐶𝐴, 𝐹𝐴, 𝑃, 𝑁* ) Select mating parents from *𝑃* and *𝐹𝐴* and generate *𝑁* ∕2 offspring ;
4. *𝑂𝑡* ← *𝑂𝑝* ∪ *𝑂𝑓𝑎* ∪ *𝑂𝑟* ;
5. *𝑃* ← EnvironmentalSelection([*𝑃 , 𝑂𝑝* ], *𝑁* , *𝑠𝑡𝑎𝑔𝑒*) ;
6. *𝐹𝐴* ← UpdateFA(*𝐶𝐴*, [*𝐹𝐴, 𝑂𝑡* ], *𝑁* , *𝑠𝑡𝑎𝑔𝑒*) ;
7. *𝑡* ← *𝑡* +2 ;
8. **return** *𝐹𝐴* ;

## Our approach

This section provides a detailed introduction to the framework and various algorithm components of NACMOEA.

* 1. *Framework of NACMOEA*

the generation counter *𝑡* and search stage are set to 1. The popula- The framework of NACMOEA is shown in Algorithm [1](#_bookmark4). Initially, tion *𝑃* is randomly initialized with *𝑁* individuals. Subsequently, *𝐶𝐴* and *𝐹𝐴* are initialized by *𝑃* . The algorithm starts from stage one,

and the subsequent steps are iterated until the termination condition is met. Initially, to determine if the algorithm reaches a stable state, we

compute the non-dominant coverage rate *𝐶* of the current population

using Equation ([6](#_bookmark6)). If this metric surpasses a predefined threshold *𝐶𝑡* ,

which is described later. The *𝜖*-constrained technique-based environ-

to form the next generation *𝑃* . With the aim of discovering the com- mental selection is implemented on the combined offspring population

for updating *𝐹𝐴*. In each iteration of stage two, since a total of 2*𝑁* in- plete CPF, a niche-based environmental selection strategy is employed

dividuals are generated and evaluated, resulting in an increase of 2 for

*𝑡* in each iteration. At the end of iteration, *𝐹𝐴* is outputted as the final

CPF found by NACMOEA.

Note that our proposed framework employs a unique two-stage search strategy, distinct from existing approaches. In the convergence- oriented search stage, constraints are not entirely ignored but rather assigned a small weight. Consequently, feasible solutions with good convergence have the potential to be retained, facilitating the explo- ration of the optimal feasible region for the problem. Similarly, in the feasibility-oriented search stage, convergence is not completely ne- glected but also given less emphasis. At this juncture, infeasible so- lutions with good convergence and low constraint violation may be preserved. Since these solutions are located near the feasible region and closer to the ideal point, they have the potential to improve the convergence of feasible solutions.

* 1. *Decision of search stage*

Switching between search stages is crucial because the proposed NACMOEA employs different search strategies during each stage. As the main population approaches the UPF and reaches a steady state, NACMOEA switches its focus to finding the CPF. Previous algorithms used metrics based on representative points like ideal and nadir points to reflect evolutionary state, but this strategy has a disadvantage in that nadir points are unstable and can change with each generation, leading to inaccurate estimations of evolutionary state. In this paper, we intro- duce a non-dominant coverage rate-based metric that overcomes these shortcomings.

The main idea of our proposed metric is that, according to the definition of Pareto dominance, when the population evolves into a steady state, the majority of individuals will become non-dominated and evenly distributed on the PF. Therefore, by comparing the domi-

nance relationship between the current generation (*𝑡*) population and

its previous generation (*𝑡* − *𝐺𝑝* ), if a larger proportion of individuals in

*𝑡* are non-dominated compared to those in *𝑡* − *𝐺𝑝* , it indicates that the

*𝑡* generation population has already converged to the PF and achieved

a steady state. Conversely, if there still exist some dominance relation-

ships between individuals in both populations, it suggests that further search should be continued as the evolution of the population has not yet stagnated at this stage. Our proposed metric has better stability com- pared to the metric based on representative points, but it also incurs a relatively higher computational cost due to the introduction of non- dominated sorting. The metric is shown as follows:

|{*𝑎* ∈ *𝑃 𝑡* ∣ ∀*𝑏* ∈ *𝑃 𝑡*−*𝐺𝑝 , 𝑎 ⊁ 𝑏*}|

|

|

the algorithm proceeds to stage two. In stage one, NACMOEA under- takes a convergence-oriented search to derive the UPF. A tournament

*𝐶* = | |

|*𝑃 𝑡*|

(6)

selection is performed on *𝑃* to select elite parents, and the simulated bi-

on selected parents to generate *𝑁* offspring *𝑂𝑝* . The next generation nary crossover operator and polynomial mutation operator are applied

ronmental selection on combined *𝑃* and *𝑂𝑝* . The *𝐶𝐴* and *𝐹𝐴* employ population is then generated by performing convergence-oriented envi-

the convergence-oriented and feasibility-oriented update strategy, re- spectively, to collect solutions that are close to the UPF and solutions near the optimal feasible region. In stage two, NACMOEA aims to ap-

selecting parents from *𝑃* , *𝑁* Offspring *𝑂𝑝* are generated. Since individ- proximate the complete CPF from both feasible and infeasible sides. By uals in *𝐹𝐴* are located inside or near the optimal feasible region, their generated offspring can further explore the feasible region, so *𝑁* ∕2 off- spring *𝑂𝐹𝐴* are generated by the elite parents from *𝐹𝐴*. Another *𝑁* ∕2

offspring are generated using the infeasible solution utilization strategy,

where *𝑃 𝑡* is the current *𝑡*th generation population, *𝑃 𝑡*−*𝐺𝑝* is the (*𝑡* −

*𝐺𝑝*)th historical generation population. When this metric is higher than the predefined threshold *𝐶𝑡* , it indicates that more than *𝐶𝑡*% individuals

ical populations, implying that the *𝑡*th generation population has been in the current population are non-dominated with respect to the histor- improved slightly when compared to the (*𝑡* − *𝐺𝑝*)th generation. Hence,

a stage switch should be executed.

* 1. *Environmental selection*

generation population for the main population *𝑃* . The fitness function In this paper, environmental selection is employed to select the next

used in this paper is defined as fellow, which is the same as that in SPEA2 [[41](#_bookmark64)].

∑

*𝐹𝑖𝑡*(**𝐱**)= ∣ *𝑅*

# ∣+ 1

(7)

**Algorithm 3:** UpdateCA.

**𝐲**

**𝐲**∈*𝑆***𝐱**

*𝑑𝑖𝑠𝑡*(**𝐱***,* **𝐱**′)+2

**Input:** archive: *𝐶𝐴*; current population: *𝑃* ; population size: *𝑁*

where *𝑅***𝐱** is the set of solutions dominated by the solution **𝐱** and *𝑆***𝐱**

**Output:** updated archive: *𝐶𝐴*

is the set of solutions dominating the solution **𝐱**. **𝐱**′ denotes the

√

⌊ 2*𝑛*⌋

**1** *𝑃𝑡* ← *𝐶𝐴* ∪ *𝑃* ;

**2** *𝐹𝑖𝑡𝑡* ← Fitness of *𝑃𝑡* by Equation ([7](#_bookmark7)) based on Pareto dominance ;

nearest neighbor to **𝐱**. The first part of Equation ([7](#_bookmark7)) denotes the total number of individuals dominated by the individual dominating **𝐱** in the

solution set, and the second√part is the inverse of the Euclidean distance

between **𝐱** and its nearest *𝑛*⌋ neighbor **𝐱**′. A smaller *𝐹 𝑖𝑡*(**𝐱**) indicates

⌊

2

**3** *𝑆𝑐* ← index of solutions which *𝐹𝑖𝑡𝑡 <* 1 ;

**4 if** ∣ *𝑆𝑐* ∣*< 𝑁* **then**

**5** *𝑅* ← sort *𝐹𝑖𝑡𝑡* ;

**6** *𝑆𝑐* ← *𝑅*(1 ∶ *𝑁* ) ;

the better quality of **𝐱**, and when *𝐹𝑖𝑡*(**𝐱**) *<* 1, it means **𝐱** is nondominant. According to the above discussion, *𝐹𝑖𝑡*(**𝐱**) is determined by the dom-

**7 if** ∣ *𝑆𝑐* ∣*> 𝑁* **then**

**8** *𝑆𝑐* ← use truncation method to select *𝑁* solutions from *𝑆𝑐* ;

**9** *𝐶𝐴* ← *𝑃* (*𝑆* ) ;

inance relationship between **𝐱** and other individuals in the population,

*𝑡 𝑐*

so the *𝐹𝑖𝑡*(**𝐱**) changes with different dominance rules. Since the search goal of *𝑃* changes with search stage, the fitness function needs to be re-

formulated to realize the switch of search goal. Inspired by TOR [[20](#_bookmark68)], the switch of search goal can be achieved by assigning different weight to the rank-based fitness. Specifically, the dominance relationship of population is determined by Pareto dominance and CDP, respectively, and then the fitness is calculated by Equation ([7](#_bookmark7)), respectively. Sorting

these two fitness values yields the rank-based fitness *𝑅𝑝*(**𝐱**) and *𝑅𝑐* (**𝐱**),

where *𝑅𝑝*(**𝐱**) represents the rank of **𝐱** in population based on Pareto dominance. Similarly, *𝑅𝑐* (**𝐱**) represents the rank of **𝐱** based on CDP. Fi- nally, by assigning different weight to *𝑅𝑝*(**𝐱**) and *𝑅𝑐* (**𝐱**), the final fitness

*𝐹𝑖𝑡𝑟*(**𝐱**) is given below:

*𝐹𝑖𝑡* (**𝐱**)= *𝛼* ∗ *𝑅* (**𝐱**)+ (1 − *𝛼*)∗ *𝑅* (**𝐱**) (8)

**10 return** *𝐶𝐴*;

solutions with the *𝐹𝑖𝑡 <* 1 is equal to *𝑁* , these solutions are directly selected for updating *𝐶𝐴*.

* 1. *Update strategy of 𝐹𝐴*

**Algorithm 4:** UpdateFA: stage 1.

**Input:** archive: *𝐹𝐴*; current population: *𝑃* ; population size: *𝑁* ;

**Output:** updated archive: *𝐹𝐴*;

**1** *𝑃𝑡* ← *𝐹𝐴* ∪ *𝑃* ;

* + 1. *𝐹𝑖𝑡𝑡* ← Calculate the Fitness of *𝑃𝑡* by Equation ([7](#_bookmark7)) based on CDP ;
    2. *𝑆𝑐* ← index of solutions which *𝐹𝑖𝑡𝑡 <* 1 ;
    3. **if** ∣ *𝑆𝑐* ∣*< 𝑁* **then**

*𝑟 𝑝 𝑐*

**5** *𝑅* ← sort *𝐹𝑖𝑡𝑡* ;

As shown in Equation ([8](#_bookmark9)), *𝐹𝑖𝑡𝑟*(**𝐱**) is calculated by the sum of *𝑅𝑐* (**𝐱**) and *𝑅𝑝*(**𝐱**) with different weights, respectively. In stage one, *𝛼* is set to 0.9 and *𝐹𝑖𝑡𝑟*(**𝐱**) mainly consists of *𝑅𝑝*(**𝐱**). The algorithm executes convergence-oriented search with a weight of 0.1 for *𝑅𝑐* (**𝐱**), while fea-

sible solutions with good convergence are still likely to be retained, these feasible solutions can help for finding the optimal feasible re-

gions. In stage two, *𝛼* is set to 0.1 and *𝐹𝑖𝑡𝑟*(**𝐱**) mainly consists of *𝑅𝑐* (**𝐱**).

for *𝑅𝑝*(**𝐱**), during which infeasible solutions with low constraint viola- The algorithm executes feasibility-oriented search with a weight of 0.1

tion and good convergence may be selected, which can help improve the convergence of the population. The pseudo code of environmental selection is shown in Algorithm [2](#_bookmark10).

**Algorithm 2:** EnvironmentalSelection.

**Input:** temp population: *𝑃𝑡* ; pop size: *𝑁* ; search stage: *𝑠𝑡𝑎𝑔𝑒*

**Output:** next generation population: *𝑃*

1. *𝐹𝑖𝑡𝑝* ← Calculate the Fitness of *𝑃𝑡* by Equation ([7](#_bookmark7)) based on Pareto dominance ;
2. *𝐹𝑖𝑡𝑐* ← Calculate the Fitness of *𝑃𝑡* by Equation ([7](#_bookmark7)) based on CDP ;
3. *𝑅𝑝* ← sort *𝐹𝑖𝑡𝑝* ;

**6** *𝑆𝑐* ← *𝑃𝑡* (*𝑅*(1 ∶ *𝑁* )) ;

**7 if** ∣ *𝑆𝑐* ∣*> 𝑁* **then**

1. **** *𝑆𝑐* ← use truncation method select *𝑁* solutions from *𝑆𝑐* ;
2. *𝐹𝐴* ← *𝑃𝑡* (*𝑆𝑐* ) ;
3. **return** FA ;

In the stage one, *𝐹𝐴* collects highly feasible solutions obtained during the evolutionary process. Subsequently, in the stage two, *𝐹𝐴*

conducts a localized search around these solutions discovered in the initial stage, progressively approaching the CPF from the feasible side.

The update strategy of *𝐹𝐴* in stage one is similar to that of Algorithm [3](#_bookmark8)

with the only difference being step 2 (line 2), where the fitness of *𝑃𝑡* is

calculated using CDP instead of Pareto dominance.

In stage two, *𝐹𝐴* employs a niche-based update strategy designed

to generate an angle-based niche and select individuals within it to

preserve adequate diversity and suﬃcient selection pressure. Only one solution is chosen in each niche. To establish the angle-based niche, we first need to compute the angle between solutions situated on the UPF, which is expressed as follows:

1. *𝑅𝑐* ← sort *𝐹𝑖𝑡𝑐* ;
2. **if** *𝑠𝑡𝑎𝑔𝑒==1* **then**

**6** *𝛼*=0.9 ;

1. **else**

(

*𝐴𝑛𝑔𝑙𝑒*(**𝐱***,* **𝐲**)= *𝑎𝑟𝑐𝑐𝑜𝑠*

*𝐹* (**𝐱**)∙ *𝐹* (**𝐲**) )

‖*𝐹* (**𝐱**)‖2 ⋅ ‖*𝐹* (**𝐲**)‖2

(9)

1. **** *𝛼*=0.1 ;
2. *𝐹𝑖𝑡𝑟* ← calculate the rank-based Fitness by Equation ([8](#_bookmark9)) ;
3. *𝑅* ← sort *𝐹𝑖𝑡𝑟* ;
4. *𝑃* ← select solutions of *𝑅*(1 ∶ *𝑁* ) from *𝑃𝑡* ;
5. **return** *𝑃* ;

Equation ([9](#_bookmark11)) calculates the angle between two solutions **𝐱** and **𝐲** in the objective space, where *𝐹* (**𝐱**) ∙ *𝐹* (**𝐲**) represents the inner product of solutions **𝐱** and **𝐲**. The niche size determines the number and location of

solutions falling into it, which has a significant impact on the selection

pressure and diversity. In order to provide suﬃcient selection pressure

for evolution and ensure diversity, the niche size *𝐴𝑛* is adaptively set

* 1. *Update strategy of 𝐶𝐴*

to the maximum angle between individuals in *𝐶𝐴*, which is shown as

follows:

In stage one, the UPF found by the main population is preserved via

*𝐶𝐴*, while in stage two, *𝐶𝐴* is not updated, and is used to generate

*𝐴* ={ *𝑚𝑎𝑥*

**𝐱***,***𝐲**∈*𝐶𝐴,***𝐱**≠**𝐲**

*𝑛*

*𝐴𝑛𝑔𝑙𝑒*(**𝐱***,* **𝐲**)} (10)

niches. The update strategy for *𝐶𝐴* is shown in Algorithm [3](#_bookmark8). Initially,

the original *𝐶𝐴* is merged with the current population *𝑃* to build a tem- porary population *𝑃𝑡* and its fitness *𝐹𝑖𝑡𝑡* is calculated. When the number of solutions with the *𝐹𝑖𝑡𝑡 <* 1 is less than *𝑁* , the first *𝑁* solutions are se-

when the angle between a candidate solution **𝐲** and a given solution **𝐱** on UPF is smaller than the *𝐴𝑛* , **𝐲** is considered to be located within the niche of **𝐱**. *𝜃***𝐜𝐩**(*𝑡*) represents the set of solutions *𝑃* located within the niche of *𝑡*th solution in *𝐶𝐴*, i.e., **𝐱***𝑐* . The equation as shown below:

lected in descending order according to their fitness. When the number { }

*𝑡*

of solutions with the *𝐹𝑖𝑡𝑡 <* 1 is greater than *𝑁* , the ∣ *𝑆𝑐* ∣ −*𝑁* solutions

are truncated according to the crowding degree. When the number of

*𝜃𝑐𝑝*(*𝑡*)=

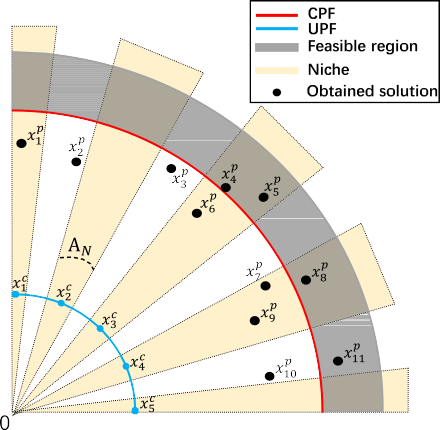
**𝐲** ∈ *𝑃* ∣ *𝐴𝑛𝑔𝑙𝑒*(**𝐱***𝑐 ,* **𝐲**) *< 𝐴𝑛*

**𝐱***𝑐* ∈*𝐶𝐴 𝑡*

*𝑡*

(11)

plores the feasible region in this approximation direction. The selected individuals in this iteration are illustrated in Fig. [1](#_bookmark15)(b). The pseudo code



for updating strategy of *𝐹𝐴* in stage two is presented in Algorithm [5](#_bookmark14).

**Algorithm 5:** UpdateFA: stage 2.

**Input:** archive: *𝐶𝐴*, *𝐹𝐴*; offspring: *𝑂*; population size: *𝑁* ; search stage: *𝑠𝑡𝑎𝑔𝑒*

**Output:** updated archive: *𝐹𝐴*;

**1** *𝐴𝑐𝑎* ← *𝐴𝑛𝑔𝑙𝑒* (**𝐱***,* **𝐲**) ;

**𝐱***,***𝐲**∈*𝐶𝐴,***𝐱**≠**𝐲**

1. *𝐴𝑛* ← the maximum element of *𝐴𝑐𝑎* by Equation ([10](#_bookmark12));
2. *𝑃* ← *𝐹𝐴* ∪ *𝑂𝑡* ;
3. *𝐴𝑐𝑝* ← *𝐴𝑛𝑔𝑙𝑒* (**𝐱***,* **𝐲**) ;

**𝐱**∈*𝐶𝐴,***𝐲**∈*𝑃*

1. *𝜃𝑐𝑝* ← solutions in *𝑃* located within the niche of *𝐶𝐴* by Equation ([11](#_bookmark13)) ;
2. **foreach** *𝑖* = 1*,* 2*,* … *, 𝑁* **do**

**7 if** ∣ *𝜃𝑐𝑝*(*𝑖*) ∣= 1 **then**

**8 ** *𝐹𝐴*(*𝑖*) ← *𝜃𝑐𝑝*(*𝑖*) ;

**9 if** ∣ *𝜃𝑐𝑝*(*𝑖*) ∣*>* 1 **then**

**10** *𝐹𝐴*(*𝑖*) ← *𝑎𝑟𝑔𝑚𝑖𝑛 𝐹𝑖𝑡𝑐* (**𝐱**) ;

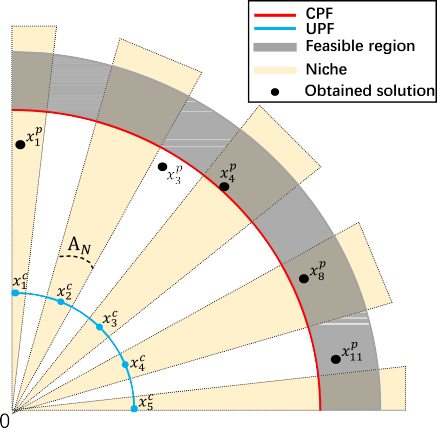
**𝐱**∈*𝜃𝑐𝑝* (*𝑖*)

**11 if** ∣ *𝜃𝑐𝑝*(*𝑖*) ∣= 0 **then**

**12 𝐜𝟏***,* **𝐜𝟐** ← *𝑎𝑟𝑔𝑚𝑖𝑛 𝐴𝑛𝑔𝑙𝑒*(**𝐱***𝑖* ∈ *𝐶𝐴* ∣ **𝐱***𝑖,* **𝐜**) ;

**𝐜**∈*𝑃*

**13** *𝐹𝐴*(*𝑖*) ← *𝑎𝑟𝑔𝑚𝑖𝑛 𝐹𝑖𝑡𝑐* (**𝐱**)



**𝐱**∈{**𝐜𝟏***,***𝐜𝟐**}

**14 return** *𝐹𝐴* ;

* 1. *Infeasible solution utilization*



**Fig. 1.** Illustration of the update strategy of FA in stage 2.

To provide further clarification on the update strategy of *𝐹𝐴* in stage two, we present the main concept in Fig. [1](#_bookmark15). As depicted in

*𝑐*

In order to explore more viable regions and enhance the conver- gence of existing feasible solutions, a niche-based strategy for utilizing

population *𝑃* undergoes *𝜖*-constrained environment selection, gradu- infeasible solutions is adopted in stage two. In this strategy, the main ally approaching CPF from the infeasible side. Meanwhile, *𝐹𝐴* updates

itself based on feasibility priority, approaching CPF from the feasible side. Niches are employed to determine the positional relationship be- tween feasible and infeasible solutions within the same approximation direction. This allows for matching a feasible solution with optimal con- vergence to an infeasible solution with the lowest CV within the same niche. Subsequently, an offspring generation operator is applied to these matched solutions, generating offspring that have potential for explor- ing unknown feasible regions and improving the convergence of feasible solutions.

Fig. [2](#_bookmark17). As shown in Fig. [2](#_bookmark17)(a), since *𝐹𝐴* is a feasibility-oriented archive, The mechanism of the infeasible solution utilization is illustrated in

solutions within *𝐹𝐴*, denoted as *𝑥𝑓* , are mostly located within feasible

Fig. [1](#_bookmark15)(a), **𝐱***𝑡* represents the UPF identified by *𝐶𝐴* during stage one,

in *𝐶𝐴* is computed using Equation ([9](#_bookmark11)), with the maximum value being which is utilized for niche creation. The angle between two solutions

region. Conversely, solutions *𝑖* main population *𝑃* , denoted as *𝑥𝑝*

are located in infeasible region, approaching CPF from infeasible side.

within *𝑖*

Similar to the update strategy of *𝐹𝐴* in stage two, the solutions **𝐱***𝑐* in

assigned as the niche size *𝐴𝑛* . Subsequently, based on Equation ([11](#_bookmark13)),

**𝐱***𝑝* denotes candidate solutions that are allocated into distinct niches.

*𝐶𝐴*

*𝑖*

are used to create angle-based niches, with a maximum angle be-

*𝑖 𝑐 𝑝*

tween **𝐱***𝑐* set as niche size *𝐴𝑛* . Afterwards, **𝐱***𝑝* have been assigned to

For instance, the angle between **𝐱**1 and **𝐱**1 is smaller than *𝐴𝑛* , indi- *𝑖 𝑖 𝑐*

cating that **𝐱***𝑝* is positioned within the niche of **𝐱***𝑐* . Consequently, we

distinct niches based on their angular relationships with respect to **𝐱***𝑖* .

1 1 We denote *𝜃𝑐𝑝*(*𝑡*) as the collection of **𝐱***𝑝* within the niche of **𝐱***𝑐* , where

define *𝜃𝑐𝑝*(1) as a set containing only **𝐱***𝑝*. Similarly, for *𝜃𝑐𝑝*(2), it remains

*𝑐 𝑖 𝑝 𝑝 𝑡*

1 *𝑝*

*𝑝 𝑝*

for instance, if the angle between **𝐱**3 and both **𝐱**4 and **𝐱**5 is smaller than

empty. However, for *𝜃𝑐𝑝*(3), it comprises of the elements: {**𝐱**4 *,* **𝐱**5 *,* **𝐱**6 }

*𝐴𝑛* , then we have *𝜃𝑐𝑝*(3) = {**𝐱***𝑝,* **𝐱***𝑝*}. Similarly, we define *𝜃𝑐𝑓* (*𝑡*) as the

and so forth. The selection is conducted on each niche to update *𝐹𝐴*,

considering the number of individuals located within the same niche,

there are three cases: In case one, where only one solution is located

within the niche (e.g., *𝜃𝑐𝑝*(1) = {**𝐱***𝑝*}. Regardless of its feasibility, it is

4 5

collection of vectors from *𝐹𝐴* that fall within the niche of vector **𝐱***𝑡* ,

(*𝑐*)

Given our objective to select suitable paired parents from both *𝑃* and thereby characterizing the set of feasible solutions for a given context.

chosen for updating *𝐹𝐴*

1

**𝐱***𝑝*

*𝐹𝐴* for offspring generation, there are two cases in which a parent can

. This is because 1 retains diversity informa-

tion as an isolated individual, which may contribute to enhancing the

ist within a single niche (e.g., *𝜃𝑐𝑝*(3) = {**𝐱***𝑝,* **𝐱***𝑝,* **𝐱***𝑝*}), a feasibility priority integrity of the CPF. In case two, when multiple candidate solutions ex-

4 5 6

for updating *𝐹𝐴*. Therefore, *𝑥𝑝* is selected. For case three, there exist no selection is employed. The best individual within the niche is selected solutions within the niche, such as when *𝜃𝑐𝑝*(2) = ∅. As the *𝐹𝐴* serves as

4

a feasibility-orientated archive, we choose two individuals with angles

be selected from *𝐹𝐴*. For case one, multiple solutions exist within the niche of *𝑥𝑐* . Given that *𝐹𝐴* prioritizes feasibility, enhancing conver-

gence becomes crucial as a majority of its individuals are feasible. The parent is selected based on the minimal sum of objective vectors since it can roughly indicate the convergence for feasible solutions in the same

*𝑡*

niche. For instance, consider the niche of *𝑥𝑐* , where *𝜃𝑐𝑓* (4) = {*𝑥𝑓 , 𝑥𝑓* }.

4 5 6

As the sum of objective vectors for *𝑥𝑓* is lower than that of *𝑥𝑓* , *𝐹𝐴* se-

closest to **𝐱**

5 6

, denoted as **𝐱***𝑝* and **𝐱***𝑝*. Among them, we select the one *𝑓*

*𝑐*2

2 *𝑝*3

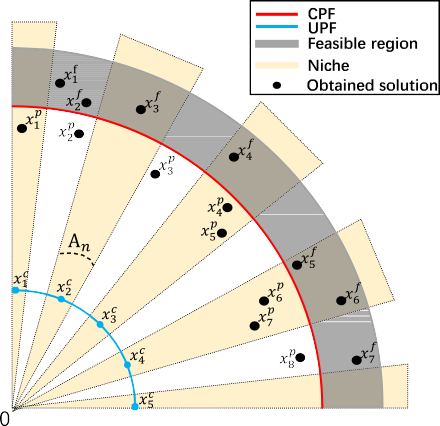
lects *𝑥*5 as the parent in this niche. In case two, no solution from *𝐹𝐴*

with the best feasibility, namely **𝐱**3 , to ensure that the algorithm ex-

*𝑡*

exists within the niche of *𝑥𝑐* , and thus, the individual in *𝐹𝐴* with the

**Algorithm 6:** InfeasibleUtilization.



**Input:** archive: *𝐶𝐴*; archive: *𝐹𝐴*; main population: *𝑃* ; population size: *𝑁*

**Output:** paired solutions: *𝑁𝑜𝐹 , 𝑁𝑜𝑃*

**1** *𝐴𝑐𝑎* ← *𝐴𝑛𝑔𝑙𝑒* (**𝐱***,* **𝐲**) ;

**𝐱***,***𝐲**∈*𝐶𝐴,***𝐱**≠**𝐲**

* + 1. *𝐴𝑛* ← the *𝑁* th element of *𝐴𝑐𝑎* by Equation ([10](#_bookmark12));
    2. *𝐴𝑐𝑝* ← *𝐴𝑛𝑔𝑙𝑒* (**𝐱***,* **𝐲**) ;

**𝐱**∈*𝐶𝐴,***𝐲**∈*𝑃*

* + 1. *𝐴𝑐𝑓* ← *𝐴𝑛𝑔𝑙𝑒* (**𝐱***,* **𝐲**) ;

**𝐱**∈*𝐶𝐴,***𝐲**∈*𝐹𝐴*

* + 1. *𝜃𝑐𝑝* ← solutions of *𝑃* located within the niche of *𝐶𝐴* by Equation ([11](#_bookmark13)) ;
    2. *𝜃𝑐𝑓* ← solutions of *𝐹𝐴* located within the niche of *𝐶𝐴* by Equation ([11](#_bookmark13)) ;
    3. **foreach** *𝑖* = 1*,* 2*,* … *, 𝑁* **do**

**8 if** ∣ *𝜃𝑐𝑓* (*𝑖*) ∣≥ 1 **then**

**9** *𝑁𝑜𝐹* (*𝑖*) ← *𝑎𝑟𝑔𝑚𝑖𝑛*

**𝐱**∈*𝜃𝑐𝑓* (*𝑖*)

**10 if** ∣ *𝜃𝑐𝑓* (*𝑖*) ∣= 0 **then**

∑*𝑚*

*𝑖*=1

*𝑓𝑖* (**𝐱**) ;

**11** *𝑁𝑜𝐹* (*𝑖*) ← *𝑎𝑟𝑔𝑚𝑖𝑛 𝐴𝑛𝑔𝑙𝑒*(**𝐱***𝑐 ,* **𝐱**) ;

**𝐱**∈*𝐹𝐴*

**12 if** ∣ *𝜃𝑐𝑝*(*𝑖*) ∣≥ 1 **then**

**13** *𝑁𝑜𝑃* (*𝑖*) ← *𝑎𝑟𝑔𝑚𝑖𝑛*

**𝐱**∈*𝜃𝑐𝑝* (*𝑖*)

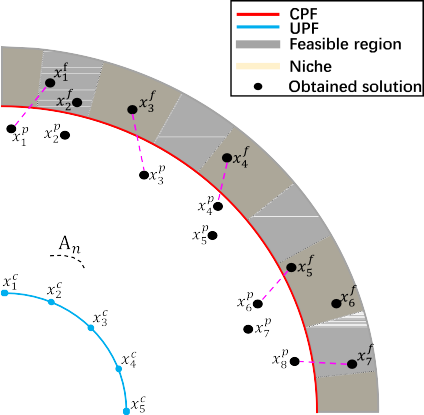
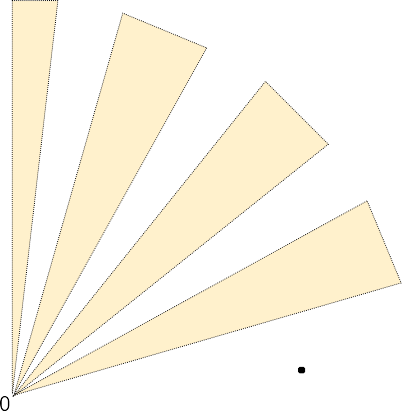
**14 if** ∣ *𝜃𝑐𝑝*(*𝑖*) ∣= 0 **then**

*𝑖*

Φ(**𝐱**) ;

**15** *𝑁𝑜𝑃* (*𝑖*) ← *𝑎𝑟𝑔𝑚𝑖𝑛 𝐴𝑛𝑔𝑙𝑒*(**𝐱***𝑐 ,* **𝐱**) ;

**𝐱**∈*𝑃 𝑖*



**16 return** *𝑁𝑜𝐹 , 𝑁𝑜𝑃*

jective space. To strike a balance between algorithm performance and exploration of more feasible regions, we propose a method of niche-based environment selection and utilization of infeasible solu- tions. Specifically, the functionality of niche-based environment se- lection and infeasible solution utilization can be summarized as fol- lows:



**Fig. 2.** Illustration of the infeasible solution utilization strategy.

smallest angle to *𝑥𝑐* is chosen. This operation is designed to ensure the exploration of CPF in the approximate direction of *𝑥𝑐* . Taking the niche of *𝑥𝑐* as an illustrative example, it can be observed that there exists no solution of *𝐹𝐴* within the niche. Consequently, the individual closest to this niche, denoted as *𝑥𝑓* , is selected.

1

1

*𝑖*

*𝑡*

The parent selection strategy for individuals in *𝑃* follows a similar

als of *𝑃* exist within the same niche. Since *𝑃* evolves from the UPF concept, with the only distinction being that when multiple individu-

and approaches the CPF from an infeasible side, it primarily comprises infeasible individuals with excellent convergence. Consequently, it is preferable to choose an individual with lower constraint violation as

instance, considering the niche of **𝐱***𝑐* , where *𝜃𝑐𝑝*(4) = {*𝑥𝑝, 𝑥𝑝*} and *𝑥𝑝* a mating parent to generate offspring with improved feasibility. For exhibits a lower constraint violation to *𝑥𝑝*, we *𝑥𝑝*. matched parents from *𝐹𝐴* and *𝑃* are illustrated in Fig. [2](#_bookmark17)(b). Please refer

compared 7 select 6 The

4 6 7 6

to Algorithm [6](#_bookmark16) for the corresponding pseudo code.

* 1. *Further discussion*

In this article, the niche technique is introduced to provide valu- able insights into population evolution. By employing angle-based niches, we can identify small regions in the objective space that share similar directions, thereby facilitating the partitioning of the population based on these regions. Individuals assigned to the same niche can be considered as exploring similar directions in the ob-

1. In the stage 2 update strategy of *𝐹𝐴*, a niche-based environmental

selection strategy is employed to choose candidate feasible individ-

uals. This approach aims at selecting the most feasible individual within a niche where candidate individuals congregate, thereby enhancing the overall feasibility of the population. In instances where niches are sparsely populated with candidates, this strategy ensures exploration in an approximate direction by selecting the individual closest to that trajectory, thus augmenting population diversity.

1. In the utilization strategy of infeasible solutions, angle-based niches are employed to provide directional information. Feasible and in- feasible solutions that lie approximately in the same direction are paired together for generating offspring individuals. These off- spring individuals possess the potential to explore the CPF along this approximate direction, thereby enhancing both convergence and diversity of feasible solutions and facilitating the attainment of a complete CPF.

As mentioned above, angle-based niche can preserve diversity in- formation and provide reasonable selection pressure for the envi- ronmental selection, which is crucial for balancing the convergence, diversity, and feasibility of the population. In addition, it can also provide positional information for utilizing infeasible solutions and guide the population to explore unknown CPF, which is partic- ularly important for improving the overall performance of NAC- MOEA.

## Experimental setup

In this section, we initially provide an overview of the experimen- t’s various components, including benchmark problems, performance metrics, parameter settings, and advanced CMOEAs for comparison. Subsequently, we employ the proposed algorithm to address distinct CMOPs and contrast the statistical results with other advanced CMOEAs to assess the performance of our proposed method.

* 1. *Benchmark suits and CMOEAs for comparison*

Two well-known benchmark suits MW [[18](#_bookmark66)] and LIR-CMOP [[7](#_bookmark43)] were chosen as test problem for the empirical study. MW notable for its close approximation to the real CMOPs and extensive coverage of dif- ferent problem types, while LIR-CMOPs stand out due to their large infeasible region and significant diﬃculty in solving. To evaluate the performance of NACOMEA, two typical CMOEAs, C-NSGAII-CDP [[5](#_bookmark39)] and C-MOEA/D [[9](#_bookmark47)], and five advanced CMOEAs, CMOEA-MS [[31](#_bookmark45)], C- TAEA [[14](#_bookmark57)], ToP [[17](#_bookmark63)], DC-NSGAIII [[11](#_bookmark51)], TiGE-2 [[38](#_bookmark58)] were selected for

peer comparison. All experiments in this paper were conducted using PlatEMO [[35](#_bookmark53)].

* 1. *Performance metrics*

In this section, two comprehensive performance evaluation met- rics, namely inverted generational distance (IGD) [[4](#_bookmark38)] and hypervolume (HV) [[42](#_bookmark67)], are introduced to compare the performance of the proposed algorithm with that of existing advanced algorithms.

1. IGD mainly evaluates the convergence and diversity performance of the algorithm by calculating the average of the minimum dis-

solution set obtained by the algorithm. Suppose *𝑃𝐹* is the real tance sum between each point on the real Pareto front and the Pareto front of the problem and *𝑃𝐹*∗ is the Pareto front found

by the algorithm, the IGD of proposed algorithm is calculated as follows:

of function evaluations *𝑀𝑎𝑥𝐹 𝐸* is set to 200000 for LIR-CMOP bench- in all CMOEAs is set to 10. For a fair comparison. The maximal number

mark suite, and 80000 for MW benchmark suite. All CMOEAs apply these settings for a fair comparison, the other parameter settings of all CMOEAs in comparison are the same as suggested in their correspond- ing original articles.

The parameters for the *𝜖*-constrained technique in NACMOEA are as

follows. *𝜏* is 0.05; *𝛼* is 0.95; *𝑇𝑐* is (0*.*9 ∗ *𝑀𝑎𝑥𝐹 𝐸*)∕*𝑁* and *𝑐𝑝* is 2.

generation gap *𝐺𝑝* is 20; the coverage rate threshold *𝐶𝑡* is 0.95. The parameters for compute the search stage are set as follows. The

The aforementioned parameter configurations remained constant throughout the experimental procedure.

Each algorithm runs 30 times independently on each test function, and the mean and standard deviation of IGD and HV values are pre- sented in corresponding tables. In particular, the best results are high- lighted in boldface. To have a statistically sound conclusion, we use the

Wilcoxon’s rank sum test at a significant level of 5% to validate the

significance of the better performance achieved by the proposed NAC-

MOEA with respect to the other compared algorithms.

*4.4. Analysis of experimental results*

In this section, the performance of NACMOEA is compared with other eight elite CMOEAs mentioned earlier on LIR-CMOP and MW benchmark CMOPs.

1. *Comparisons on LIR-CMOP suite:* There are a total of 14 CMOPs

∑

**𝐱**∈*𝑃𝐹*

*𝑑𝑒𝑑*

(**𝐱***, 𝑃𝐹* ∗)

within the LIR-CMOP benchmark suite. The experimental results based

on the IGD are presented in Table [1](#_bookmark19). The analysis reveals that NAC-

*𝐼𝐺𝐷* =

where *𝑑*

*𝑒𝑑*

∣ *𝑃𝐹* ∣ (12)

(**𝐱***, 𝑃𝐹*∗) represents the minimum Euclidean distance be-

∗

MOEA demonstrates the most superior performance on 9 test problems. In comparison, CMOEA-MS and DCNSGAIII exhibit exceptional perfor- mance on 3 and 2 test problems, respectively. However, C-MOEA/D,

tween **𝐱** and *𝑃𝐹* . The smaller IGD value indicates the better per-

formance of the algorithm.

1. HV assesses the comprehensive performance of the algorithm by determining the hypervolume value of the space enclosed by the non-dominated solution set and the reference point. The HV of a solution set can be calculated as follows:

∣*𝑆*∣

*𝐻𝑉* = *𝜆*( ∪ *𝑣𝑖*) (13)

*𝑖*=1

where *𝜆* represents the Lebesgue measure, *𝑣𝑖* denotes the hyper-

lution, and *𝑆* is the non-dominated solution set. The larger HV volume formed by the reference point and the non-dominated so-

indicates the better comprehensive performance of the algorithm.

It is necessary to adequately sample the true PF of test problem for calculating IGD. According to the approach of [[29](#_bookmark40)], about 10000 sample

of the output population is set to (1*, ...,* 1), and the point (1*.*1 ∗ *𝑧𝑛𝑎𝑑* ) is points are sampled on the real PF. To calculate HV, the reference point

used to normalize the objective values.

* 1. *Parameter setting*

DE operator is used to generate offspring in PPS [[8](#_bookmark44)], while genetic operator is used in all other compared CMOEAs.

Simulated binary crossover (SBX) [[1](#_bookmark33)] and polynomial mutation (PM) [[5](#_bookmark39)] are used as the genetic operator for CMOEAs that use genetic operator as the evolutionary operator, with the following parameter set- tings:

1. Crossover probability *𝑝𝑐* =1 and distribution index *𝜂𝑐* = 20.
2. Mutation probability *𝑝𝑚* = 1∕*𝑛* and distribution index *𝜂𝑚* = 20.

rameters *𝐶𝑅* and *𝐹* in DE operator are set to 1 and 0.5, respectively. For CMOEAs which adopt DE as the evolutionary operator, the pa- The population size *𝑁* is set to 100. The number of decision variable *𝐷*

C-NSGAII-CDP, C-TAEA, ToP, and TiGE2 failed to achieve the best results on any of the test problems. To be explicit, the proposed NAC- MOEA demonstrates best performance on LIR-CMOP1-LIR-CMOP6, LIR- CMOP8, and LIR-CMOP10-LIR-CMOP11, indicating that NACMOEA is effective in addressing scenarios where UPF and CPF are separated and the feasible region is narrow (LIR-CMOP1-LIR-CMOP4). In instance that the feasible region is obstructed by a large area of infeasible region (LIR-CMOP5-LIR-CMOP6, LIR-CMOP8), NACMOEA is also capable of

crossing through a multitude of infeasible regions and reaching the CPF. Furthermore, NACMOEA can cope with cases where CPF and UPF partially overlap and the feasible region is narrow and isolated (LIR-CMOP10-LIR-CMOP11). Table [2](#_bookmark20) displays the HV result of eight compared CMOEAs on LIR-CMOP1-LIR-CMOP14. As illustrated, NAC- MOEA exhibits its superior performance on eight CMOPs and competi- tive against the other two CMOPs.

Fig. [3](#_bookmark21) illustrates the populations with the median IGD obtained from the eight CMOEAs on LIR-CMOP11, whose feasible region is isolated and narrow. It can be discerned that some solutions of C-MOEA/D, C-TAEA, and TiGE-2 fail to converge to the CPF, demonstrating insuﬃ- cient convergence capability. While the populations of C-NSGAII-CDP, CMOEA-MS, and ToP fail to maintain good diversity, which results in the failure to obtain a complete CPF. The populations of DCNSGAIII and NACMOEA exhibit an optimal balance between convergence, diversity, and feasibility, with all solutions within these populations traversing the infeasible region, thus achieving the complete CPF.

1. *MW benchmark suite comparisons:* The MW benchmark suite con- tains a total of 14 CMOPs. Table [3](#_bookmark23) presents the experimental results of NACMOEA and other advanced CMOEAs on the MW based on IGD. The proposed NACMOEA demonstrates superior performance in terms of MW and attains the most favorable outcomes across 6 test problems. Meanwhile, C-TAEA excels in 5 test problem solutions. Both CMOEA-MS and DCNSGAIII exhibit equivalent eﬃciency on a single test problem. However, C-NSGAII-CDP, ToP, and TiGE-2 fail to achieve optimal re- sults in any test problems.

*F. Guo and H. Li*

*Egyptian Informatics Journal 25 (2024) 100422*

9

**Table 1**

Statistical results of IGD obtained by C-MOEA/D, C-NSGAII-CDP, C-TAEA, CMOEA-MS, DCNSGAIII, TOP, TiGE2 and the proposed NACMOEA on the LIR-CMOP benchmark suite. Best result in each row is highlighted.

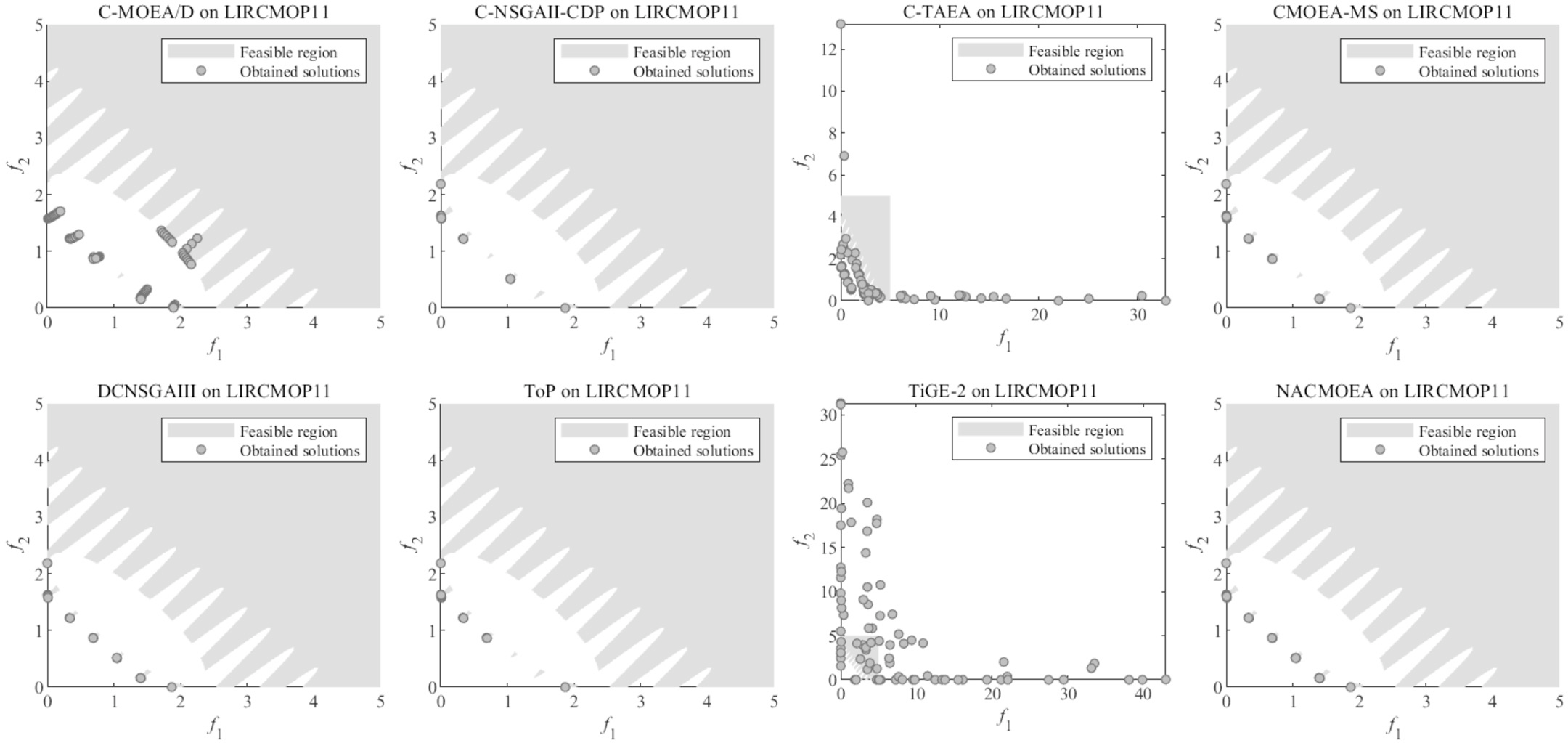
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Problem | M | C-MOEA/D | C-NSGAII-CDP | C-TAEA | CMOEA-MS | DCNSGAIII | ToP | TiGE-2 | NACMOEA |
|  |  |  |  |  |  |  |  |  |  |
| LIRCMOP1 | 2 | 1.6569e-1 (5.04e-2) - | 2.3454e-1 (6.23e-2) - | 2.2888e-1 (1.34e-1) - | 3.9355e-1 (1.46e-1) - | 3.7499e-2 (1.67e-2) - | 1.2399e-1 (8.86e-2) - | 9.5208e-2 (4.30e-2) - | 1.8245e-2 (4.95e-3) |
| LIRCMOP2 | 2 | 1.2404e-1 (3.42e-2) - | 1.6739e-1 (5.47e-2) - | 8.3119e-2 (3.46e-2) - | 3.2164e-1 (1.08e-1) - | 4.3175e-2 (1.57e-2) - | 1.6413e-1 (9.70e-2) - | 1.0882e-1 (2.59e-2) - | 2.2810e-2 (5.00e-3) |
| LIRCMOP3 | 2 | 1.7927e-1 (6.05e-2) - | 2.3718e-1 (8.16e-2) - | 2.9440e-1 (1.91e-1) - | 3.6344e-1 (1.21e-1) - | 4.8155e-2 (3.11e-2) - | 3.6246e-1 (8.21e-2) - | 8.7310e-2 (2.50e-2) - | 2.6055e-2 (1.39e-2) |
| LIRCMOP4 | 2 | 1.7869e-1 (5.62e-2) - | 2.3887e-1 (7.25e-2) - | 1.7807e-1 (1.09e-1) - | 2.8070e-1 (8.74e-2) - | 3.8080e-2 (1.61e-2) - | 3.1630e-1 (5.20e-2) - | 1.0289e-1 (2.52e-2) - | 2.7916e-2 (1.04e-2) |
| LIRCMOP5 | 2 | 7.1368e-1 (5.33e-1) - | 6.5485e-1 (5.22e-1) - | 9.2411e-2 (2.14e-2) - | 1.9767e-2 (2.38e-2) ≈ | 2.6591e-2 (1.58e-2) - | 1.4394e-1 (3.55e-1) ≈ | 3.2770e-1 (1.06e-1) - | 1.4251e-2 (1.34e-2) |
| LIRCMOP6 | 2 | 8.2848e-1 (6.04e-1) - | 5.6214e-1 (5.39e-1) - | 1.3693e-1 (1.29e-1) - | 1.4576e-2 (3.75e-2) ≈ | 6.3814e-2 (2.46e-1) - | 4.3586e-2 (8.31e-2) ≈ | 4.4798e-1 (2.01e-1) - | 1.1531e-2 (1.44e-2) |
| LIRCMOP7 | 2 | 7.0524e-2 (2.93e-2) - | 2.2115e-2 (2.82e-2) - | 2.0159e-2 (7.50e-3) - | 7.0537e-3 (6.80e-4) + | 1.2531e-2 (2.73e-3) - | 8.6212e-3 (3.08e-4) + | 1.5998e-1 (5.01e-2) - | 9.1088e-3 (2.73e-3) |
| LIRCMOP8 | 2 | 8.1395e-2 (4.07e-2) - | 2.6587e-2 (3.53e-2) - | 2.7893e-2 (4.12e-2) - | 1.4851e-2 (2.98e-2) - | 1.3797e-2 (4.24e-3) - | 2.0973e-2 (6.73e-2) - | 3.1627e-1 (1.34e-1) - | 8.8445e-3 (4.59e-3) |
| LIRCMOP9 | 2 | 4.0501e-1 (8.67e-2) - | 4.3878e-1 (1.33e-1) - | 6.8170e-2 (3.36e-2) ≈ | 2.5178e-1 (1.35e-1) - | 1.3509e-2 (2.24e-2) ≈ | 3.3808e-1 (9.57e-2) - | 7.1003e-1 (2.51e-1) - | 7.4281e-2 (6.86e-2) |
| LIRCMOP10 | 2 | 1.9883e-1 (6.37e-2) - | 3.0191e-1 (1.03e-1) - | 7.8340e-2 (6.72e-2) - | 8.0013e-2 (5.69e-2) - | 9.1694e-3 (1.49e-2) - | 5.5883e-3 (3.20e-4) - | 4.8089e-1 (6.05e-2) - | 4.8836e-3 (1.44e-4) |
| LIRCMOP11 | 2 | 2.1262e-1 (1.15e-1) - | 1.9104e-1 (1.49e-1) - | 1.3390e-1 (3.97e-2) - | 8.0190e-2 (4.66e-2) - | 5.3132e-3 (1.30e-2) - | 9.8869e-2 (6.27e-2) - | 5.0013e-1 (8.55e-2) - | 2.4242e-3 (5.43e-5) |
| LIRCMOP12 | 2 | 1.5644e-1 (6.35e-2) - | 1.2374e-1 (5.54e-2) - | 1.9280e-2 (6.95e-3) - | 6.0949e-2 (5.06e-2) - | 3.6273e-3 (1.04e-3) + | 1.8157e-2 (3.95e-2) - | 5.0892e-1 (3.10e-1) - | 1.0385e-2 (2.51e-2) |
| LIRCMOP13 | 3 | 9.2899e-2 (1.77e-7) + | 1.5868e-1 (2.20e-1) - | 1.0855e-1 (2.50e-3) - | 9.2582e-2 (9.38e-4) + | 9.2899e-2 (7.55e-7) + | 1.2742e-1 (3.96e-3) - | 3.5011e-1 (6.96e-2) - | 9.5820e-2 (1.28e-3) |
| LIRCMOP14 | 3 | 9.5314e-2 (1.14e-6) + | 1.2325e-1 (5.39e-3) - | 1.1158e-1 (1.02e-3) - | 9.4933e-2 (1.14e-3) + | 9.5555e-2 (1.18e-4) + | 1.1987e-1 (2.93e-3) - | 5.8779e-1 (3.35e-1) - | 9.7403e-2 (1.06e-3) |
| +/-/≈ |  | 2/12/0 | 0/14/0 | 0/13/1 | 3/9/2 | 3/10/1 | 1/11/2 | 0/14/0 |  |

**Table 2**

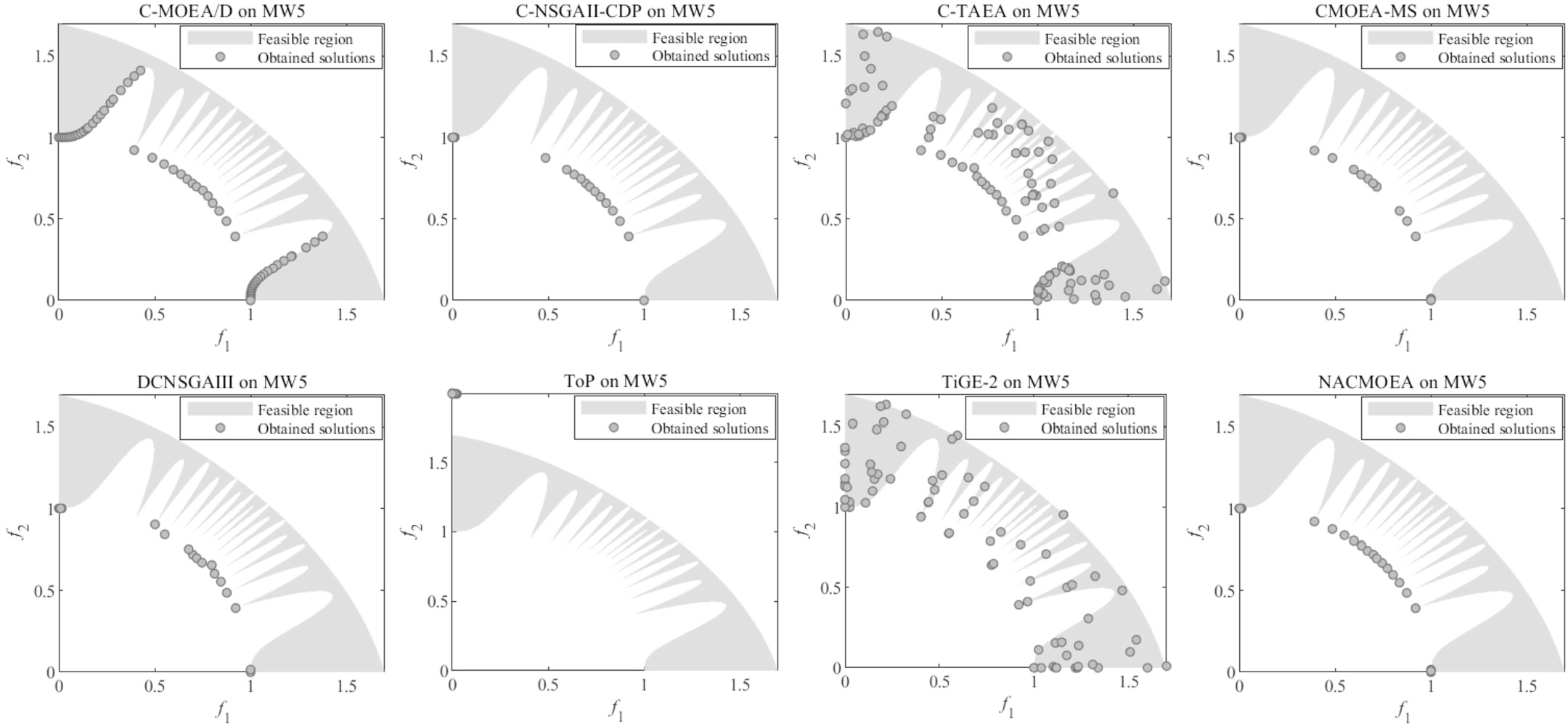
Statistical results of HV obtained by C-MOEA/D, C-NSGAII-CDP, C-TAEA, CMOEA-MS, DCNSGAIII, TOP, TiGE2 and the proposed NACMOEA on the LIR-CMOP benchmark suite. Best result in each row is highlighted.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Problem | M | C-MOEA/D | C-TAEA | CMOEA-MS | DCNSGAIII | ToP | TiGE-2 | C-NSGAII-CDP | NACMOEA |
|  |  |  |  |  |  |  |  |  |  |
| LIRCMOP1 | 2 | 1.6276e-1 (1.69e-2) - | 1.3963e-1 (3.58e-2) - | 1.1314e-1 (2.83e-2) - | 2.2456e-1 (6.91e-3) - | 1.8180e-1 (3.84e-2) - | 1.9850e-1 (1.22e-2) - | 1.3970e-1 (2.02e-2) - | 2.3362e-1 (1.81e-3) |
| LIRCMOP2 | 2 | 2.8900e-1 (1.86e-2) - | 3.1564e-1 (2.19e-2) - | 2.0975e-1 (4.51e-2) - | 3.4092e-1 (5.59e-3) - | 2.8395e-1 (4.51e-2) - | 3.1786e-1 (1.18e-2) - | 2.6745e-1 (2.84e-2) - | 3.5198e-1 (2.05e-3) |
| LIRCMOP3 | 2 | 1.3679e-1 (1.99e-2) - | 1.1618e-1 (3.97e-2) - | 9.7910e-2 (2.27e-2) - | 1.9494e-1 (8.75e-3) - | 9.3626e-2 (1.74e-2) - | 1.7208e-1 (1.03e-2) - | 1.2359e-1 (2.12e-2) - | 2.0253e-1 (3.66e-3) |
| LIRCMOP4 | 2 | 2.3706e-1 (2.48e-2) - | 2.2814e-1 (4.06e-2) - | 1.9664e-1 (3.95e-2) - | 3.0198e-1 (5.30e-3) - | 1.8434e-1 (2.69e-2) - | 2.7634e-1 (1.07e-2) - | 2.1412e-1 (3.01e-2) - | 3.0621e-1 (4.64e-3) |
| LIRCMOP5 | 2 | 1.0112e-1 (1.11e-1) - | 2.5760e-1 (1.16e-2) - | 2.8525e-1 (8.51e-3) ≈ | 2.8004e-1 (7.44e-3) - | 2.5323e-1 (9.49e-2) - | 1.7679e-1 (3.47e-2) - | 1.1451e-1 (1.26e-1) - | 2.8507e-1 (9.27e-3) |
| LIRCMOP6 | 2 | 6.9360e-2 (8.13e-2) - | 1.4987e-1 (3.89e-2) - | 1.9423e-1 (8.45e-3) ≈ | 1.8518e-1 (3.61e-2) - | 1.8121e-1 (3.31e-2) - | 8.4661e-2 (2.31e-2) - | 9.7847e-2 (7.55e-2) - | 1.9443e-1 (4.98e-3) |
| LIRCMOP7 | 2 | 2.6636e-1 (1.01e-2) - | 2.8780e-1 (4.02e-3) - | 2.9472e-1 (5.83e-4) + | 2.9097e-1 (1.54e-3) - | 2.9389e-1 (1.66e-4) ≈ | 2.3614e-1 (1.29e-2) - | 2.8739e-1 (1.17e-2) - | 2.9322e-1 (1.68e-3) |
| LIRCMOP8 | 2 | 2.6496e-1 (1.15e-2) - | 2.8719e-1 (1.03e-2) - | 2.9161e-1 (1.19e-2) - | 2.9058e-1 (1.94e-3) - | 2.9146e-1 (1.28e-2) - | 1.8754e-1 (3.98e-2) - | 2.8844e-1 (1.01e-2) - | 2.9342e-1 (3.15e-3) |
| LIRCMOP9 | 2 | 4.2860e-1 (4.70e-2) - | 5.2607e-1 (1.45e-2) - | 4.7208e-1 (5.78e-2) - | 5.6258e-1 (5.92e-3) ≈ | 4.6686e-1 (2.87e-2) - | 3.0050e-1 (9.71e-2) - | 4.0757e-1 (8.28e-2) - | 5.4903e-1 (2.28e-2) |
| LIRCMOP10 | 2 | 5.9514e-1 (3.47e-2) - | 6.6790e-1 (2.78e-2) - | 6.7741e-1 (2.60e-2) - | 7.0415e-1 (5.60e-3) - | 7.0737e-1 (1.67e-4) - | 4.2773e-1 (2.44e-2) - | 5.5529e-1 (6.03e-2) - | 7.0813e-1 (8.07e-5) |
| LIRCMOP11 | 2 | 6.2555e-1 (8.43e-2) - | 6.4309e-1 (1.33e-2) - | 6.5930e-1 (2.67e-2) - | 6.9289e-1 (4.59e-3) - | 6.3190e-1 (4.18e-2) - | 4.1228e-1 (3.89e-2) - | 5.7907e-1 (1.10e-1) - | 6.9405e-1 (5.58e-6) |
| LIRCMOP12 | 2 | 5.8330e-1 (2.66e-2) - | 6.1068e-1 (2.95e-3) - | 5.9785e-1 (2.25e-2) - | 6.2001e-1 (4.64e-4) + | 6.1335e-1 (1.87e-2) - | 4.1179e-1 (1.16e-1) - | 5.6721e-1 (2.72e-2) - | 6.1745e-1 (1.03e-2) |
| LIRCMOP13 | 3 | 5.5962e-1 (8.49e-7) + | 5.4685e-1 (1.50e-3) - | 5.5613e-1 (1.79e-3) + | 5.5962e-1 (2.18e-6) + | 5.1327e-1 (3.93e-3) - | 4.5642e-1 (2.88e-2) - | 5.1311e-1 (9.70e-2) - | 5.5036e-1 (1.55e-3) |
| LIRCMOP14 | 3 | 5.5957e-1 (1.24e-5) + | 5.4635e-1 (8.91e-4) - | 5.5500e-1 (1.04e-3) + | 5.5856e-1 (4.15e-4) + | 5.2808e-1 (3.80e-3) - | 3.6538e-1 (1.16e-1) - | 5.3038e-1 (4.50e-3) - | 5.5165e-1 (1.49e-3) |
| +/-/≈ |  | 2/12/0 | 0/14/0 | 3/9/2 | 3/10/1 | 0/13/1 | 0/14/0 | 0/14/0 |  |

*F. Guo and H. Li Egyptian Informatics Journal 25 (2024) 100422*



**Fig. 3.** Populations with median IGD obtained by C-MOEA/D, C-NSGAII-CDP, C-TAEA, CMOEA-MS, DCNSGAIII, MOEA/D-DAE, TOP, TiGE2 and the proposed NACMOEA on the LIR-CMOP11.



**Fig. 4.** Populations with median IGD obtained by C-MOEA/D, C-NSGAII-CDP, C-TAEA, CMOEA-MS, DCNSGAIII, MOEA/D-DAE, TOP, TiGE2 and the proposed NACMOEA on the MW5.

The rationale behind this phenomenon lies in the adoption of a feasibility-oriented search strategy by C-NSGAII-CDP and ToP, which excessively prioritizes constraint satisfaction at the expense of main- taining diversity during the search process. The convergence speed of TiGE-2 is relatively slow, with most individuals failing to converge near the CPF within an equivalent number of function evaluations. The feasible region of the MW test set is primarily characterized by dis- connected or narrow traits, which makes it easy for CMOEA to get trapped into local feasible regions. NACMOEA exhibits superior per- formance on MW1, MW5, MW7, MW9, and MW11-MW12. Specifically, both MW1 and MW11 present a disconnected geometry of CPF. The fea- sible regions of MW5 and MW7 are connected; however, their CPFs are

situated along narrow boundaries of the feasible region, posing a signif- icant challenge to obtain. On the other hand, the CPF and UPF of both MW9 and MW12 are entirely separate. By effectively handling such complex CMOPs through the utilization of infeasible solutions, NAC- MOEA demonstrates its eﬃcacy. Table [4](#_bookmark24) showcases the hypervolume (HV) results obtained by NACMOEA as well as eight other CMOEAs for comparison purposes on the aforementioned CMOPs within the context of this study. Overall, our proposed NACMOEA outperforms all oth- ers on six CMOPs while remaining competitive on the remaining four CMOPs.

Fig. [4](#_bookmark22) illustrates the experimental results of the eight CMOEAs on the MW5 problem, where the CPF of MW5 is discrete and located at the end

10

*F. Guo and H. Li*

*Egyptian Informatics Journal 25 (2024) 100422*

11

**Table 3**

Statistical results of IGD obtained by C-MOEA/D, C-NSGAII-CDP, C-TAEA, CMOEA-MS, DCNSGAIII, MOEA/D-DAE, TOP, TiGE2 and the proposed NACMOEA on the MW benchmark suite. Best result in each row is highlighted.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Problem | M | C-MOEA/D | C-NSGAII-CDP | C-TAEA | CMOEA-MS | DCNSGAIII | ToP | TiGE-2 | NACMOEA |
| MW1 | 2 | 3.8561e-3 (1.75e-3) - | 3.6776e-2 (4.47e-2) - | 2.0128e-3 (7.42e-5) - | 1.8543e-3 (9.90e-4) ≈ | 2.3465e-3 (6.75e-5) - | 2.8692e-1 (1.90e-1) - | 2.4913e-2 (3.19e-2) - | 1.6795e-3 (2.09e-5) |
| MW2 | 2 | 1.2379e-2 (9.82e-3) ≈ | 1.6165e-2 (9.60e-3) ≈ | 8.9175e-3 (7.87e-3) + | 1.5435e-2 (7.99e-3) ≈ | 8.6716e-3 (7.04e-3) + | 9.2834e-2 (7.33e-2) - | 7.9361e-2 (1.78e-1) - | 1.4736e-2 (9.21e-3) |
| MW3 | 2 | 5.2770e-3 (4.16e-4) - | 6.6536e-3 (4.99e-3) - | 4.9632e-3 (2.17e-4) ≈ | 5.3178e-3 (3.12e-4) - | 6.2302e-3 (7.46e-4) - | 3.5980e-1 (4.10e-1) - | 2.2675e-2 (3.60e-3) - | 5.0381e-3 (1.74e-4) |
| MW4 | 3 | 4.1180e-2 (1.87e-4) + | 5.5181e-2 (2.98e-3) - | 4.6595e-2 (3.00e-4) - | 4.1859e-2 (3.89e-4) + | 4.1722e-2 (7.26e-4) + | 2.7012e-1 (1.58e-1) - | 9.2694e-2 (3.83e-2) - | 4.2366e-2 (4.69e-4) |
| MW5 | 2 | 1.8168e-2 (7.31e-2) - | 6.2432e-2 (1.86e-1) - | 1.4708e-2 (2.70e-3) - | 2.3237e-2 (2.50e-2) - | 2.2877e-2 (8.89e-3) - | 6.8731e-1 (2.35e-1) - | 5.6428e-2 (1.36e-2) - | 1.1153e-3 (7.44e-4) |
| MW6 | 2 | 1.2610e-2 (8.93e-3) ≈ | 1.6776e-2 (9.91e-3) - | 5.9709e-3 (4.09e-3) ≈ | 1.1695e-2 (7.85e-3) ≈ | 1.0195e-2 (8.90e-3) ≈ | 3.4238e-1 (2.50e-1) - | 3.9546e-2 (1.24e-2) - | 1.2809e-2 (1.09e-2) |
| MW7 | 2 | 4.6394e-3 (1.01e-4) - | 1.9810e-2 (8.16e-2) - | 7.1631e-3 (5.87e-4) - | 2.3438e-2 (2.33e-2) ≈ | 5.8954e-3 (7.36e-4) - | 1.1128e-1 (1.94e-1) - | 3.7463e-2 (1.91e-2) - | 4.3507e-3 (2.77e-4) |
| MW8 | 3 | 5.0244e-2 (2.45e-3) - | 5.5203e-2 (4.57e-3) - | 5.3037e-2 (1.63e-3) - | 4.4060e-2 (2.08e-3) ≈ | 4.8387e-2 (1.44e-3) - | 3.3577e-1 (3.37e-1) - | 6.8464e-1 (1.44e-1) - | 4.4152e-2 (1.35e-3) |
| MW9 | 2 | 7.3367e-3 (1.62e-3) - | 8.8554e-3 (2.50e-3) - | 8.4538e-3 (6.24e-4) - | 6.7675e-2 (1.42e-1) - | 1.1275e-2 (2.76e-3) - | 3.7694e-1 (3.21e-1) - | 3.2327e-2 (4.76e-3) - | 4.5016e-3 (2.28e-4) |
| MW10 | 2 | 3.8720e-2 (4.60e-2) - | 5.8309e-2 (4.41e-2) - | 7.9136e-3 (7.04e-3) ≈ | 2.4414e-2 (2.24e-2) ≈ | 1.7853e-2 (1.51e-2) ≈ | 1.5442e-1 (7.43e-2) - | 4.2238e-2 (3.22e-2) - | 2.7905e-2 (3.67e-2) |
| MW11 | 2 | 5.6499e-2 (1.74e-1) - | 1.9457e-1 (3.08e-1) - | 1.5161e-2 (2.53e-3) - | 6.3523e-3 (7.84e-4) - | 8.2962e-3 (1.19e-3) - | 5.5260e-1 (2.76e-1) - | 3.0114e-2 (4.59e-3) - | 5.9915e-3 (1.40e-4) |
| MW12 | 2 | 4.8073e-3 (7.73e-5) - | 5.6128e-3 (2.31e-4) - | 7.6779e-3 (5.78e-4) - | 5.1780e-3 (2.48e-4) - | 6.3666e-3 (4.10e-4) - | 6.6322e-1 (2.53e-1) - | 2.8330e-2 (5.06e-3) - | 4.7568e-3 (1.03e-4) |
| MW13 | 2 | 4.6458e-2 (3.07e-2) ≈ | 7.7992e-2 (7.63e-2) - | 2.2057e-2 (1.51e-2) ≈ | 5.8588e-2 (3.96e-2) - | 4.5000e-2 (2.61e-2) ≈ | 4.3198e-1 (4.18e-1) - | 4.3113e-1 (2.67e-1) - | 4.3385e-2 (3.11e-2) |
| MW14 | 3 | 2.1173e-1 (1.01e-3) - | 1.2433e-1 (6.04e-3) - | 1.1007e-1 (3.07e-3) + | 1.2893e-1 (1.87e-2) - | 1.4194e-1 (3.76e-2) - | 2.3854e-1 (1.86e-1) - | 1.6169e-1 (1.08e-2) - | 1.1840e-1 (5.38e-2) |
| +/-/≈ |  | 1/10/3 | 0/13/1 | 2/8/4 | 1/7/6 | 2/9/3 | 0/14/0 | 0/14/0 |  |

**Table 4**

Statistical results of HV obtained by C-MOEA/D, C-NSGAII-CDP, C-TAEA, CMOEA-MS, DCNSGAIII, MOEA/D-DAE, TOP, TiGE2 and the proposed NACMOEA on the MW benchmark suite. Best result in each row is highlighted.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Problem | M | C-MOEA/D | C-NSGAII-CDP | C-TAEA | CMOEA-MS | DCNSGAIII | ToP | TiGE-2 | NACMOEA |
| MW1 | 2 | 4.8742e-1 (3.47e-3) - | 4.4922e-1 (3.95e-2) - | 4.8899e-1 (2.13e-4) - | 4.8971e-1 (2.00e-3) - | 4.8893e-1 (1.76e-4) - | 2.2173e-1 (1.51e-1) - | 4.6145e-1 (2.88e-2) - | 4.9005e-1 (2.34e-5) |
| MW2 | 2 | 5.6709e-1 (1.69e-2) ≈ | 5.6059e-1 (1.57e-2) ≈ | 5.7319e-1 (1.38e-2) + | 5.6170e-1 (1.35e-2) ≈ | 5.7346e-1 (1.25e-2) + | 4.5845e-1 (8.90e-2) - | 5.1129e-1 (1.22e-1) - | 5.6294e-1 (1.51e-2) |
| MW3 | 2 | 5.4477e-1 (3.47e-4) + | 5.4247e-1 (5.36e-3) - | 5.4507e-1 (2.17e-4) + | 5.4467e-1 (2.73e-4) + | 5.4507e-1 (2.24e-4) + | 3.1253e-1 (2.40e-1) - | 5.3102e-1 (1.83e-3) - | 5.4446e-1 (2.34e-4) |
| MW4 | 3 | 8.4161e-1 (1.56e-4) + | 8.2397e-1 (3.81e-3) - | 8.3824e-1 (2.31e-4) - | 8.3938e-1 (4.42e-4) ≈ | 8.4126e-1 (6.14e-4) + | 5.4336e-1 (1.85e-1) - | 7.8394e-1 (3.62e-2) - | 8.3955e-1 (5.97e-4) |
| MW5 | 2 | 3.1504e-1 (3.51e-2) - | 3.0074e-1 (5.78e-2) - | 3.1522e-1 (1.87e-3) - | 3.1250e-1 (1.22e-2) - | 3.1077e-1 (5.70e-3) - | 8.2000e-2 (4.72e-2) - | 2.8979e-1 (8.46e-3) - | 3.2424e-1 (2.32e-4) |
| MW6 | 2 | 3.1344e-1 (1.31e-2) ≈ | 3.0763e-1 (1.42e-2) - | 3.2095e-1 (7.16e-3) ≈ | 3.1472e-1 (1.16e-2) ≈ | 3.1658e-1 (1.34e-2) ≈ | 1.7015e-1 (9.05e-2) - | 2.8412e-1 (2.05e-2) - | 3.1324e-1 (1.56e-2) |
| MW7 | 2 | 4.1137e-1 (9.93e-5) - | 4.0655e-1 (3.05e-2) - | 4.0941e-1 (6.60e-4) - | 4.0953e-1 (3.98e-3) - | 4.1199e-1 (3.14e-4) - | 3.6098e-1 (7.83e-2) - | 3.9279e-1 (4.56e-3) - | 4.1288e-1 (2.34e-4) |
| MW8 | 3 | 5.3763e-1 (1.32e-2) ≈ | 5.2031e-1 (1.52e-2) - | 5.3173e-1 (8.37e-3) - | 5.4168e-1 (8.34e-3) ≈ | 5.3743e-1 (9.49e-3) - | 2.9876e-1 (1.56e-1) - | 1.5211e-1 (5.35e-2) - | 5.4210e-1 (8.38e-3) |
| MW9 | 2 | 3.8980e-1 (2.85e-3) - | 3.8880e-1 (3.52e-3) - | 3.9244e-1 (1.78e-3) - | 3.4469e-1 (9.85e-2) - | 3.8545e-1 (4.64e-3) - | 1.7460e-1 (1.57e-1) - | 3.6109e-1 (6.97e-3) - | 3.9857e-1 (2.20e-3) |
| MW10 | 2 | 4.2067e-1 (3.41e-2) - | 4.0382e-1 (3.00e-2) - | 4.4831e-1 (9.52e-3) ≈ | 4.3085e-1 (2.01e-2) ≈ | 4.3715e-1 (1.59e-2) ≈ | 3.4289e-1 (4.29e-2) - | 4.1520e-1 (2.70e-2) - | 4.2971e-1 (2.85e-2) |
| MW11 | 2 | 4.3483e-1 (4.42e-2) - | 3.9943e-1 (7.80e-2) - | 4.4242e-1 (1.10e-3) - | 4.4738e-1 (5.15e-4) - | 4.4381e-1 (3.08e-3) - | 3.0570e-1 (6.67e-2) - | 4.3502e-1 (2.29e-3) - | 4.4808e-1 (1.13e-4) |
| MW12 | 2 | 6.0517e-1 (2.32e-4) + | 6.0400e-1 (2.35e-4) - | 6.0096e-1 (6.83e-4) - | 6.0411e-1 (2.84e-4) - | 6.0246e-1 (6.43e-4) - | 1.1174e-1 (1.62e-1) - | 5.7832e-1 (6.21e-3) - | 6.0488e-1 (2.40e-4) |
| MW13 | 2 | 4.5835e-1 (1.67e-2) ≈ | 4.4014e-1 (4.42e-2) - | 4.7095e-1 (8.55e-3) ≈ | 4.5381e-1 (2.07e-2) ≈ | 4.6310e-1 (1.02e-2) ≈ | 2.9427e-1 (1.22e-1) - | 3.5085e-1 (5.91e-2) - | 4.5928e-1 (1.48e-2) |
| MW14 | 3 | 4.4554e-1 (1.34e-3) - | 4.5468e-1 (3.51e-3) - | 4.6841e-1 (2.20e-3) ≈ | 4.6787e-1 (3.69e-3) + | 4.6353e-1 (1.11e-2) - | 4.0499e-1 (8.51e-2) - | 4.5033e-1 (6.08e-3) - | 4.6712e-1 (9.71e-3) |
| +/-/≈ |  | 3/7/4 | 0/13/1 | 2/8/4 | 2/6/6 | 3/8/3 | 0/14/0 | 0/14/0 |  |

**Table 5**

Statistical results of IGD obtained by NACMOEA-ARC, NACMOEA-IF and the original NACMOEA on the MW benchmark suite. Best result in each row is highlighted.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Problem | M | NACMOEA-ARC | NACMOEA-IF | NACMOEA |
| MW1 | 2 | 9.8597e-3 (2.78e-3) - | 1.6650e-3 (1.98e-5) + | 1.6778e-3 (1.84e-5) |
| MW2 | 2 | 2.4404e-2 (7.17e-3) - | 1.4694e-2 (8.38e-3) ≈ | 1.3798e-2 (9.88e-3) |
| MW3 | 2 | 1.6669e-2 (2.82e-3) - | 5.2215e-3 (2.72e-4) - | 5.0323e-3 (2.02e-4) |
| MW4 | 3 | 8.4703e-2 (8.18e-3) - | 4.2451e-2 (5.13e-4) ≈ | 4.2284e-2 (5.81e-4) |
| MW5 | 2 | 2.8989e-3 (6.43e-3) ≈ | 3.5553e-3 (2.95e-3) - | 6.8643e-4 (5.47e-4) |
| MW6 | 2 | 2.5836e-2 (1.28e-2) - | 1.4665e-2 (1.56e-2) ≈ | 1.4953e-2 (1.42e-2) |
| MW7 | 2 | 1.2514e-2 (2.09e-3) - | 5.9121e-3 (8.64e-3) ≈ | 4.2336e-3 (2.37e-4) |
| MW8 | 3 | 1.0818e-1 (1.83e-2) - | 4.4219e-2 (2.35e-3) ≈ | 4.4389e-2 (2.22e-3) |
| MW9 | 2 | 1.5321e-2 (2.43e-3) - | 4.5408e-3 (2.68e-4) - | 4.4631e-3 (3.75e-4) |
| MW10 | 2 | 3.1349e-2 (2.74e-2) - | 1.2871e-2 (1.42e-2) ≈ | 2.0970e-2 (2.26e-2) |
| MW11 | 2 | 1.6987e-2 (3.11e-3) - | 5.9809e-3 (1.06e-4) ≈ | 5.9678e-3 (1.14e-4) |
| MW12 | 2 | 1.5053e-2 (3.16e-3) - | 4.7991e-3 (1.10e-4) - | 4.7379e-3 (1.05e-4) |
| MW13 | 2 | 8.0508e-2 (4.28e-2) - | 3.1691e-2 (2.86e-2) ≈ | 4.2194e-2 (2.91e-2) |
| MW14 | 3 | 1.8823e-1 (3.07e-2) - | 1.0449e-1 (2.16e-2) ≈ | 1.0535e-1 (2.10e-2) |

+/-/≈ 0/13/1 1/4/9

of the narrow tunnel-like feasible region, which requires a high diver- sity of the population. The convergence of C-MOEA/D, C-TAEA, TiGE-2, and ToP is insuﬃcient, with a significant number of solutions failing to converge near the CPF. The absence of diversity in C-NSGAII-CDP and DCNSGAIII leads to a partial loss of CPF. Our proposed NACMOEA demonstrates exceptional performance in both convergence and diver- sity, effectively handling complex feasible region cases.

1. *Eﬀectiveness of each component in NACMOEA:* To evaluate the eﬃcacy of components within the NACMOEA, ablation studies are con- ducted in this section. The original NACMOEA and its two variant versions are compared on the MW benchmark suite. The first vari- ant, NACMOEA-ARC, excludes the dual archive cooperation mechanism present in the original NACMOEA and employs only the main popula- tion for evolution. It is important to note that as this variant does not utilize the archive for infeasible solutions, the associated module is also omitted. The second variant, NACMOEA-IF, retains the dual archive co- operation mechanism but eliminates the infeasible solutions utilization mechanism.

The experimental outcomes of NACMOEA and its variants based on IGD are presented in Table [5](#_bookmark25). A comparison reveals that NACMOEA out- performs the first variant, NACMOEA-ARC, on the majority of test prob- lems. This is attributed to the fact that NACMOEA-ARC has eliminated the dual archive cooperation mechanism and infeasible solution uti- lization strategy, resulting in a considerable performance degradation. The underlying architecture of NACMOEA-ARC is a straightforward two-stage algorithm. The second variant, NACMOEA-IF, demonstrates a comparable performance to NACMOEA on six test problems. This is attributed to the partial or complete overlap between the CPF and UPF of these test problems. For instance, there is a partial overlap between CPF and UPF for MW1, MW6, and MW8. The CPF and UPF for MW10 and MW13 are highly similar, and the CPF of MW14 completely over- laps with the UPF. Given the negligible or non-existent gap between UPF and CPF in the objective space, the proposed infeasible solution utilization strategy is ineffective, resulting in NACMOEA-IF exhibiting approximately the same performance as NACMOEA on these types of test problems. However, the performance of NACMOEA-IF is inferior to NACMOEA for test problems with a distinct separation between CPF and UPF, such as MW3, MW9, and MW12, thereby demonstrating the eﬃcacy of the proposed infeasible solution utilization strategy.

## Conclusions

In this paper, we propose a novel two-stage CMOEA called NAC- MOEA to address the challenging CMOPs with intricate feasible regions. The algorithm’s core concept involves dividing the search process into two stages: convergence-oriented search and feasibility-oriented search.

At different stages of the search, dual archives perform distinct roles and collaborate effectively. The transition between these stages is de- termined by evaluating the non-dominant coverage rate metric, which accurately reflects the dominance relationship of the current genera- tion population compared to its history counterparts. In stage one, our proposed approach can rapidly converge to UPF since the influence of constraints is relatively weak. At this stage, the obtained UPF is saved

by *𝐶𝐴*, while *𝐹𝐴* collects the elite individuals that are clustered around

the feasible side, *𝐹𝐴* employs a niche-based feasibility priority select the optimal feasible region. In stage two, to approach the CPF from strategy, whereas the population utilizes an *𝜖*-constrained technique

cial niche information is provided by *𝐶𝐴*. Furthermore, a niche-based to approximate the CPF from the infeasible side. In this process, cru-

strategy for utilizing infeasible solutions is established to enhance con- vergence speed towards CPF from both feasible and infeasible sides.

In the experimental section, we compare the proposed algorithm with seven other advanced CMOEAs on a total of 28 test problems from the LIR-CMOP and MW test sets. These test problems encompass vari- ous forms of feasible regions, effectively validating the performance of CMOEA across different types of CMOPs. The statistical results demon- strate that NACMOEA outperforms the compared algorithms in terms of overall performance. To further validate the effectiveness of com- ponents in NACMOEA, we compare two variants with the original NACMOEA using MW suite. The experimental findings reveal that in- corporating a dual archive cooperation mechanism and an infeasible solution utilization strategy significantly enhance the performance of NACMOEA.

Through an analysis of the limitations of existing CMOEAs, this pa- per proposes a multi-stage framework based on the niche technique. This study substantiates the significance of a balanced strategy encom- passing convergence, feasibility, and diversity. Furthermore, it unveils the potential of infeasible solutions in exploring unknown feasible re- gions and enhancing algorithm performance. While the proposed algo- rithm exhibits promising results in experiments, there is scope for ex- tending its application to tackle more challenging CMOPs. For instance, improvements can be made to environment selection and archive up- date strategies while augmenting selection pressure to address many- objective constrained optimization problems. Additionally, embedding multi-modal optimization techniques could prove beneficial for solving constrained multi-modal multi-objective optimization problems.

## CRediT authorship contribution statement

**Fengyu Guo:** Conceptualization, Methodology, Software, Writing – original draft. **Hecheng Li:** Supervision, Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 61966030).

## References

1. [Agrawal RB, Deb K, Agrawal RB. Simulated binary crossover for continuous search](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibA1E5B9588E76752D029FCA7A4BCE902Es1) [space. Complex Syst 1994;9:115–48.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibA1E5B9588E76752D029FCA7A4BCE902Es1)
2. [Asafuddoula M, Ray T, Sarker R. A decomposition-based evolutionary algorithm for](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibD13F968FC78F2CD3FE41301165BA09AAs1) [many objective optimization. IEEE Trans Evol Comput 2015;19:445–60.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibD13F968FC78F2CD3FE41301165BA09AAs1)
3. Bader J, Zitzler E. HypE: an algorithm for fast hypervolume-based many- objective optimization. Evol Comput 2011;19:45–76. [https://doi.org/10.1162/](https://doi.org/10.1162/EVCO_a_00009) [EVCO\_a\_00009](https://doi.org/10.1162/EVCO_a_00009).
4. [Bosman PAN, Thierens D. The balance between proximity and diversity in multiob-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib382FF9B65A0BF6FDADDB5413D55663EFs1) [jective evolutionary algorithms. IEEE Trans Evol Comput 2003;7(2):174–88.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib382FF9B65A0BF6FDADDB5413D55663EFs1)
5. [Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib5824438B3ADF7C3D781B6836604CC609s1) [algorithm: NSGA-II. IEEE Trans Evol Comput 2002;6:182–97.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib5824438B3ADF7C3D781B6836604CC609s1)
6. [Fan L, Tatsuo Y, Tao X, Lin Y, Liu H. A novel hybrid algorithm for solving mul-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibB942B6D5FDBF98F35DA87768FA424B98s1) [tiobjective optimization problems with engineering applications. Math Probl Eng](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibB942B6D5FDBF98F35DA87768FA424B98s1) [2018;2018:1–15.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibB942B6D5FDBF98F35DA87768FA424B98s1)
7. [Fan Z, Li W, Cai X, Huang H, Fang Y, You Y, et al. An improved epsilon constraint-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib06E1D13E24454045581E1FC139CA757Fs1) [handling method in MOEA/D for CMOPs with large infeasible regions. Soft Comput](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib06E1D13E24454045581E1FC139CA757Fs1) [2019.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib06E1D13E24454045581E1FC139CA757Fs1)
8. Fan Z, Li W, Cai X, Li H, Wei C, Zhang Q, et al. Push and pull search for solving constrained multi-objective optimization problems. Swarm Evol Com- put 2019;44:665–79. <https://doi.org/10.1016/j.swevo.2018.08.017>. [https://www.](https://www.sciencedirect.com/science/article/pii/S2210650218300233) [sciencedirect.com/science/article/pii/S2210650218300233](https://www.sciencedirect.com/science/article/pii/S2210650218300233).
9. [Jain H, Deb K. An evolutionary many-objective optimization algorithm using](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib16422967D87B27B6771EC62F8354AA3Fs1) [reference-point based nondominated sorting approach, part II: handling constraints](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib16422967D87B27B6771EC62F8354AA3Fs1) [and extending to an adaptive approach. IEEE Trans Evol Comput 2014;18:602–22.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib16422967D87B27B6771EC62F8354AA3Fs1)
10. Jiao R, Xue B, Zhang M. A multiform optimization framework for constrained mul- tiobjective optimization. IEEE Trans Cybern 2022:1–13. [https://doi.org/10.1109/](https://doi.org/10.1109/TCYB.2022.3178132) [TCYB.2022.3178132](https://doi.org/10.1109/TCYB.2022.3178132).
11. Jiao R, Zeng S, Li C, Yang S, Ong YS. Handling constrained many-objective optimiza- tion problems via problem transformation. IEEE Trans Cybern 2021;51:4834–47. <https://doi.org/10.1109/TCYB.2020.3031642>.
12. [Kannan BK, Kramer SN. An augmented Lagrange multiplier based method for mixed](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib5ED7963AF522F2605155CE8146659F7As1) [integer discrete continuous optimization and its applications to mechanical design.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib5ED7963AF522F2605155CE8146659F7As1) [Trans ASME J Mech Des 1994;116:405–11.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib5ED7963AF522F2605155CE8146659F7As1)
13. [Li Ke, Deb Kalyanmoy. An evolutionary many-objective optimization algorithm](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib4F714F56967C14D92CFF2CD287AF980Es1) [based on dominance and decomposition. IEEE Trans Evol Comput 2014.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib4F714F56967C14D92CFF2CD287AF980Es1)
14. Li K, Chen R, Fu G, Yao X. Two-archive evolutionary algorithm for constrained multiobjective optimization. IEEE Trans Evol Comput 2019;23:303–15. [https://doi.](https://doi.org/10.1109/TEVC.2018.2855411) [org/10.1109/TEVC.2018.2855411](https://doi.org/10.1109/TEVC.2018.2855411).
15. [Liu Z, Wang B, Tang K. Handling constrained multiobjective optimization problems](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibD46125BBA6EBAB10381E31E9E4789A2Bs1) [via bidirectional coevolution. IEEE Trans Cybern 2021.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibD46125BBA6EBAB10381E31E9E4789A2Bs1)
16. Liu ZZ, Qin Y, Song W, Zhang J, Li K. Multiobjective-based constraint-handling tech- nique for evolutionary constrained multiobjective optimization: a new perspective. IEEE Trans Evol Comput 2022:1. <https://doi.org/10.1109/TEVC.2022.3194729>.
17. [Liu ZZ, Wang Y. Handling constrained multiobjective optimization problems with](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib754E0DF4BDEE74EF585E6941EAB67473s1) [constraints in both the decision and objective spaces. IEEE Trans Evol Comput](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib754E0DF4BDEE74EF585E6941EAB67473s1) [2019:1.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib754E0DF4BDEE74EF585E6941EAB67473s1)
18. [Ma Z, Wang Y. Evolutionary constrained multiobjective optimization: test suite con-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibF9F315DE90492C8259307985379C2A4Es1) [struction and performance comparisons. IEEE Trans Evol Comput 2019:1.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibF9F315DE90492C8259307985379C2A4Es1)
19. Ma Z, Wang Y. Shift-based penalty for evolutionary constrained multiobjective op- timization and its application. IEEE Trans Cybern 2023;53:18–30. [https://doi.org/](https://doi.org/10.1109/TCYB.2021.3069814) [10.1109/TCYB.2021.3069814](https://doi.org/10.1109/TCYB.2021.3069814).
20. Ma Z, Wang Y, Song W. A new fitness function with two rankings for evolu- tionary constrained multiobjective optimization. IEEE Trans Syst Man Cybern Syst 2021;51:5005–16. <https://doi.org/10.1109/TSMC.2019.2943973>.
21. [Maldonado HM, Zapotecas-Martínez S. A dynamic penalty function within MOEA/D](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibBF813B3B8E253929360C9461E6668A6Ds1) [for constrained multi-objective optimization problems. In: IEEE congress on evolu-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibBF813B3B8E253929360C9461E6668A6Ds1) [tionary computation; 2021.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibBF813B3B8E253929360C9461E6668A6Ds1)
22. Ming F, Gong W, Li D, Wang L, Gao L. A competitive and cooperative swarm optimizer for constrained multi-objective optimization problems. IEEE Trans Evol Comput 2022:1. <https://doi.org/10.1109/TEVC.2022.3199775>.
23. Ming F, Gong W, Wang L, Gao L. A constrained many-objective optimization evolu- tionary algorithm with enhanced mating and environmental selections. IEEE Trans Cybern 2022:1–13. <https://doi.org/10.1109/TCYB.2022.3151793>.
24. Ming M, Trivedi A, Wang R, Srinivasan D, Zhang T. A dual-population-based evo- lutionary algorithm for constrained multiobjective optimization. IEEE Trans Evol Comput 2021;25:739–53. <https://doi.org/10.1109/TEVC.2021.3066301>.
25. Ming M, Wang R, Ishibuchi H, Zhang T. A novel dual-stage dual-population evo- lutionary algorithm for constrained multiobjective optimization. IEEE Trans Evol Comput 2022;26:1129–43. <https://doi.org/10.1109/TEVC.2021.3131124>.
26. Qiao K, Yu K, Qu B, Liang J, Song H, Yue C, et al. Dynamic auxiliary task-based evolutionary multitasking for constrained multi-objective optimization. IEEE Trans Evol Comput 2022:1. <https://doi.org/10.1109/TEVC.2022.3175065>.
27. Qin C, Ming F, Gong W, Gu Q. Constrained multi-objective optimization via two archives assisted push–pull evolutionary algorithm. Swarm Evol Com- put 2022;75:101178. <https://doi.org/10.1016/j.swevo.2022.101178>. [https://www.](https://www.sciencedirect.com/science/article/pii/S2210650222001456) [sciencedirect.com/science/article/pii/S2210650222001456](https://www.sciencedirect.com/science/article/pii/S2210650222001456).
28. Sun R, Zou J, Liu Y, Yang S, Zheng J. A multi-stage algorithm for solving multi- objective optimization problems with multi-constraints. IEEE Trans Evol Comput 2022:1. <https://doi.org/10.1109/TEVC.2022.3224600>.
29. [Tian Y, Cheng R, Zhang X, Su Y, Jin Y. A strengthened dominance relation consider-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib9C38EBB7384DADA637BE8899EFA68E6Fs1) [ing convergence and diversity for evolutionary many-objective optimization. IEEE](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib9C38EBB7384DADA637BE8899EFA68E6Fs1) [Trans Evol Comput 2018.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib9C38EBB7384DADA637BE8899EFA68E6Fs1)
30. Tian Y, Zhang T, Xiao J, Zhang X, Jin Y. A coevolutionary framework for constrained multiobjective optimization problems. IEEE Trans Evol Comput 2021;25:102–16. <https://doi.org/10.1109/TEVC.2020.3004012>.
31. Tian Y, Zhang Y, Su Y, Zhang X, Tan KC, Jin Y. Balancing objective optimization and constraint satisfaction in constrained evolutionary multiobjective optimization. IEEE Trans Cybern 2022;52:9559–72. <https://doi.org/10.1109/TCYB.2020.3021138>.
32. Wang BC, Shui ZY, Feng Y, Ma Z. Evolutionary algorithm with dynamic population size for constrained multiobjective optimization. Swarm Evol Com- put 2022;73:101104. <https://doi.org/10.1016/j.swevo.2022.101104>. [https://www.](https://www.sciencedirect.com/science/article/pii/S2210650222000748) [sciencedirect.com/science/article/pii/S2210650222000748](https://www.sciencedirect.com/science/article/pii/S2210650222000748).
33. Wang C, Xu R. An angle based evolutionary algorithm with infeasibil- ity information for constrained many-objective optimization. Appl Soft Com- put 2020;86:105911. <https://doi.org/10.1016/j.asoc.2019.105911>. [https://www.](https://www.sciencedirect.com/science/article/pii/S1568494619306921) [sciencedirect.com/science/article/pii/S1568494619306921](https://www.sciencedirect.com/science/article/pii/S1568494619306921).
34. Wang Y, Liu Y, Zou J, Zheng J, Yang S. A novel two-phase evolutionary algo- rithm for solving constrained multi-objective optimization problems. Swarm Evol Comput 2022;75:101166. <https://doi.org/10.1016/j.swevo.2022.101166>. [https://](https://www.sciencedirect.com/science/article/pii/S2210650222001341) [www.sciencedirect.com/science/article/pii/S2210650222001341](https://www.sciencedirect.com/science/article/pii/S2210650222001341).
35. [Ye T, Ran C, Zhang X, Jin Y. PlatEMO: a Matlab platform for evolutionary multi-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib659048009450B6D71EDB8CE7F493892Fs1) [objective optimization. IEEE Comput Intell Mag 2017;12:73–87.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib659048009450B6D71EDB8CE7F493892Fs1)
36. Zhang K, Xu Z, Yen GG, Zhang L. Two-stage multi-objective evolution strategy for constrained multi-objective optimization. IEEE Trans Evol Comput 2022:1. [https://](https://doi.org/10.1109/TEVC.2022.3202723) [doi.org/10.1109/TEVC.2022.3202723](https://doi.org/10.1109/TEVC.2022.3202723).
37. [Zhang Q, Hui L. MOEA/D: a multiobjective evolutionary algorithm based on decom-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib92D84787154481502FCB7CAB702BBC1Fs1) [position. IEEE Trans Evol Comput 2008;11:712–31.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib92D84787154481502FCB7CAB702BBC1Fs1)
38. [Zhou Y, Min Z, Wang J, Zhang Z, Zhang J. Tri-goal evolution framework for con-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibEA5D7EE97693A56FEDAE0C221F010310s1) [strained many-objective optimization. IEEE Trans Syst Man Cybern Syst 2018.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bibEA5D7EE97693A56FEDAE0C221F010310s1)
39. Zhu Q, Zhang Q, Lin Q. A constrained multiobjective evolutionary algorithm with detect-and-escape strategy. IEEE Trans Evol Comput 2020;24:938–47. [https://doi.](https://doi.org/10.1109/TEVC.2020.2981949) [org/10.1109/TEVC.2020.2981949](https://doi.org/10.1109/TEVC.2020.2981949).
40. [Zitzler E, Künzli S. Indicator-based selection in multiobjective search. In: 8th inter-](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib511ED8D0A59A79E364DF8C40A71C32E8s1) [national conference on parallel problem solving from nature; 2004.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib511ED8D0A59A79E364DF8C40A71C32E8s1)
41. [Zitzler E, Laumanns M, Thiele L. SPEA2: improving the strength Pareto evolutionary](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib6800CA5201A3FAFCA7B7B8971E0D1B58s1) [algorithm. Technical Report Gloriastrasse. 2001.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib6800CA5201A3FAFCA7B7B8971E0D1B58s1)
42. [Zitzler E, Thiele L. Multiobjective evolutionary algorithms: a comparative case study](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib8B6EFA65661E98B20754CA6BBDDE88ABs1) [and the strength Pareto approach. IEEE Trans Evol Comput 1999;3:257–71.](http://refhub.elsevier.com/S1110-8665(23)00078-6/bib8B6EFA65661E98B20754CA6BBDDE88ABs1)