

Article

A Projection-Based Evolutionary Algorithm for Multi-Objective and Many-Objective Optimization

Funan Peng^{1,2,3}, Li Lv^{1,2}, Weiru Chen³ and Jun Wang^{3,*}

¹ Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China; pengfunan@syuct.edu.cn (F.P.); lli@sict.ac.cn (L.L.)

² University of Chinese Academy of Sciences, Beijing 101408, China

³ College of Computer Science and Technology, Shenyang University of Chemical Technology, Shenyang 110142, China; chenweiru@syuct.edu.cn

* Correspondence: wj_software@hotmail.com

Abstract: Many-objective optimization problems (MaOPs) are challenging optimization problems in scientific research. Research has tended to focus on algorithms rather than algorithm frameworks. In this paper, we introduce a projection-based evolutionary algorithm, MOEA/PII. Applying the idea of dimension reduction and decomposition, it divides the objective space into projection plane and free dimension(s). The balance between convergence and diversity is maintained using a Bi-Elite queue. The MOEA/PII is not only an algorithm, but also an algorithm framework. We can choose a decomposition-based or dominance-based algorithm to be the free dimension algorithm. When it is an algorithm framework, it exhibits a better performance. We compare the performance of the algorithm and the algorithm with the MOEA/PII framework. The performance is evaluated by benchmark test instances DTLZ1-7 and WFG1-9 on 3, 5, 8, 10, and 15 objectives using IGD-metric and HV-metric. In addition, we investigated its superior performance on the wireless sensor networks deployment problem using C-metric. Moreover, determining objective domain for the objects of the wireless sensor networks deployment problem reduces the time and makes the solution set more responsive to user needs.

Keywords: many-objective optimization; decomposition; evolutionary algorithm; projection plane; free dimension; objective domain



Citation: Peng, F.; Lv, L.; Chen, W.; Wang, J. A Projection-Based Evolutionary Algorithm for Multi-Objective and Many-Objective Optimization. *Processes* **2023**, *11*, 1564. <https://doi.org/10.3390/pr11051564>

Academic Editor: Roberto Pisano

Received: 18 April 2023

Revised: 15 May 2023

Accepted: 18 May 2023

Published: 19 May 2023



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

1. Introduction

Multi-objective optimization problems (MOPs) [1] arise frequently in various engineering and scientific applications, where multiple conflicting objectives need to be optimized simultaneously. In many real-world problems, there can be two or three objectives. However, in many applications, the number of objectives is much higher, leading to many-objective optimization problems (MaOPs) [2].

The many-objective optimization problem (MaOPs) is a common problem in the fields of engineering and scientific computing [3]. Efforts focused on optimizing the multi-objective problem have important practical application value [4].

The difference between solving multi-objective optimization problems and solving single-objective optimization problems lies in the fact that there is more than one objective to optimize, and there is a constraint relationship between these objectives. In general, they are conflicting, that is, the performance improvement of one objective will cause the degradation of other objectives. It is more difficult to solve many-objective optimization problems than to solve multi-objective optimization problems.

Moreover, real-world problems, such as those involving wireless sensor network deployment [5], water distribution system design [6], electric vehicle charging station problems [7], bulldozer blade in soil cutting [8], structural health monitoring [9], gesture

recognition problems [10], etc., have been modeled using multi-objectives that demand efficient MOEAs. Designing an efficient algorithm to find a set of finite representative solutions in the objective space and balance the convergence and diversity is a huge challenge.

1.1. Motivation

Solving MaOPs is a challenging task due to the exponential increase in the number of Pareto-optimal solutions and the difficulty of balancing convergence and diversity in the objective space. Solving such problems is a common concern in academic and engineering circles.

There are many difficulties in the field of many-objective optimization. The most prominent problems are as follows: first, with the increasing number of objectives, the number of Pareto-optimal solutions grows exponentially. However, the selection pressure of the algorithms is often insufficient, and it is impossible to efficiently screen out the potential representative solution set from a large number of Pareto-optimal solution sets. Second, along with the increasing number of objectives, the objective space increases exponentially. More specifically, it is not straightforward for the customers to identify the most appropriate solutions from the many-objective optimal solution based on these optimization techniques.

Therefore, there is a need to develop new algorithms that can address these issues and further improve the quality of solutions.

In this paper, we propose a projection-based multi- and many-objective optimization evolution algorithm. It applies dimension reduction to divide the objective space into the projection plane and free dimension(s), and it maintains the diversity by carving the projection grids uniformly. Individuals evolve on the projection plane in each projection grid at the same time, using the Bi-Elite strategy. On the free dimension(s), we choose the state-of-the-art evolutionary algorithms to evolve, which maintains the convergence ability. In this process, the balance between convergence and diversity is maintained.

1.2. Contributions

This paper focuses on the challenge of efficiently finding a set of solutions in the objective space for MaOPs.

The main contributions of this paper are summarized as follows:

- (1) We propose an efficient many-objective evolutionary algorithm framework, MOEA/PII. Applying the idea of dimension reduction and decomposition, MOEA/PII divides the objective space into the projection plane and free dimension(s). Firstly, the objective dimensions chosen by the user are selected as the projection plane and evolution is carried out on it. Then, on the free dimensions that consist of the remaining objective dimensions, free dimension algorithms are applied for further evolution. In this way, the objective space is effectively divided, and the idea of dimensionality reduction is applied to evolution;
- (2) MOEA/PII carves the projection plane into projection grids, and the evolution in each grid is carried out. In each projection grid, the number of individuals to be selected decreases, and accordingly, the selection pressure is alleviated;
- (3) On the projection plane, the customer can determine the objective domain, i.e., the objective value within a certain range, which helps them to identify the most appropriate solutions more efficiently;
- (4) In the proposed Bi-Elite strategy, each projection has a ConEliteList, saving individuals projected in the grid. There is also a ProEliteList for the projection plane, saving individuals outside the determined domain. They are used for sorting, using the free dimension algorithm;
- (5) On the free dimensions, we can apply any other MOEAs. The convergence of the solutions in each grid is determined by the free dimension algorithm. The distribution of the projection grid is uniform, which ensures the overall distribution of the solutions. MOEA/PII addresses the challenges of MaOPs by balancing the convergence and diversity of solutions;
- (6) In solving real-world problem, the study of the wireless sensor network deployment problem shows that MOEA/PII can efficiently perform optimized deployment. By

determining the objective domain, the time is reduced and the solution set becomes more responsive to user needs.

The remainder of this paper is organized as follows. Section 2 introduces the four types of many-objective optimization algorithms and describes the projection-based method and presents some definitions for it. Section 3 presents the proposed MOEA/PII framework. MOEA/PII is described in detail. In Section 4, computational experiments and a real-world problem are presented to study the performance of the proposed MOEA/PII framework. Finally, conclusions are given in Section 5.

2. Preliminaries

In general, evolution many-objective algorithms are grouped into four classes.

The first class is the Pareto-dominance-based algorithm. It refers to a Pareto-optimal set that represents the trade-off among objectives [11]. Such algorithms are effective in solving the multi-objective optimization problem. However, they encounter challenges in solving MaOPs due to the high computational complexity, the low efficiency of the algorithm, and their difficult in solving practical optimization problems [12].

The above problems are solved by a new diversity evaluation mechanism based on Pareto dominance, which is introduced to enhance the selection pressure of the algorithm in solving MaOPs. For example, the algorithms with a generalized Pareto optimality criterion [13] can produce more selection pressure within certain ranges. A grid-domination [14] mechanism is used to balance the convergence and diversity. Some algorithms use fuzzy logic to improve the probability that a solution dominates others [15], such as fuzzy domination [16]. The algorithms proposed in [17,18] are based on ε -dominance relationship. Some algorithms judge the dominance relationship by θ -dominance, such as, θ -DEA [19].

The second class is composed of decomposition-based algorithms, which choose a scalar method to decompose MaOPs into many scalar optimization sub-problems. In 2007, Zhang et al. proposed a decomposition-based multi-objective evolutionary algorithm (MOEA/D) [20], which has become a popular algorithm framework in recent years. In MOEA/D, the distance relationship between weight vectors is defined as the neighbor relationship between sub-problems. According to the characteristics of MaOPs, experts proposed many improvement methods. The algorithms in [21,22] exhibit high performance in many-objective optimization problems with irregular Pareto fronts. Asafuddoula, M. et al. proposed a decomposition-based evolutionary algorithm, I-DBEA [23].

The next class is composed of indicator-based many-objective evolutionary algorithms, which guide the search direction, sort the individuals in the evolutionary population, and select the new population in the evolution process using indicators. For example, Sun, Y.A. et al. [24] proposed the IGD (inverted generational distance) indicator-based evolutionary algorithm for solving MaOPs. According to the ability to reflect convergence and diversity at the same time, the IGD indicator is employed as the selector with the proposed proximity distance assignments. Liu, Chao et al. proposed a new hyper-volume-based differential evolution algorithm (MODEhv) [25] for MaOPs in 2017. The HV (hypervolume) indicator, as the solution selector, is efficient in solving MaOPs [26]. The R2 indicator is a performance indicator with a high correlation with HV. The two-stage R2 indicator-based evolutionary algorithm (TS-R2EA) [27] for many-objective optimization was proposed by Li Fei et al. The algorithm achieved good results to balance the characteristics of convergence and diversity.

The last group is composed of a hybrid method. They make full use of the advantages of the two algorithms. For example, refs. [28–30] combine the evolution strategies of different methods. To handle MaOPs, Xiang Y et al. hybridize the decomposition-based algorithm and the artificial bee colony (ABC) algorithm [31]. Zhao H T et al. proposed an algorithm to accelerate the convergence speed by hybridizing the decomposition-based many-objective algorithm and the ant colony optimization (ACO) algorithm [32]. FDEA-I (fraction dominance evolutionary algorithms I) and FDEA-II (fraction dominance evolutionary algorithms II) [33] use the fractional dominance relationship to select the best solution, and use the sub-space selection mechanism of the improved objective space decomposition strategy to maintain diversity at a later stage.

Each algorithm's classification and approach is shown in Table 1.

Table 1. Each algorithm's classification and approach.

Classify	Algorithm	Year	Approaches
Dominance-based algorithm	Generalized Pareto optimality criterion [13].	2016	Generalizes Pareto optimality both symmetrically and asymmetrically by expanding the dominance area of solutions to enhance the scalability of existing Pareto-based algorithms.
	Grid domination [14].	2013	Grid dominance and grid difference; uses three grid-based criteria, grid ranking, grid crowding distance, and grid coordinate point distance, to determine the fitness.
	Fuzzy domination [16].	2014	The concept of fuzzy logic is adopted to define a fuzzy Pareto domination relation.
	Based on ϵ -dominance relationship [17].	2019	Using ϵ -dominance, the objective space is divided into boxes, diversity is ensured by them, and a single solution is established in each box.
	ϵ -domination-based Two_Arch2 algorithm [18].	2022	It uses the diversity archive (DA) and convergence archive (CA); ϵ -domination is sorted in the DA; the boundary protection strategy is based on indicator sorting in the CA.
	θ -DEA [19].	2016	Based on using the θ -dominance relation to sort in the environmental selection phase, solutions are allocated into different clusters represented by well-distributed reference points.
Decomposition-based algorithms	MOEA/D [20].	2008	Decomposes a multi-objective optimization problem into a number of scalar optimization sub-problems, which optimize by only using information from neighbor sub-problems.
	The algorithm [21].	2022	Based on decomposition and hierarchical clustering selection, uniformly distributed reference vectors and adaptive reference vectors are used to divide the population and rank the dominant individuals, respectively.
	OD-RVEA [22].	2018	An adaptive reference vector adjustment strategy to solve irregular PF problems.
	I-DBEA [23].	2015	A means of reference point generation, which uses associated solutions to reference directions.
Indicator-based algorithm	IGD indicator-based [24].	2019	The IGD indicator is employed in each generation to select the solutions with favorable convergence and diversity.
	MODEhv [25].	2017	The hyper-volume indicator is incorporated for the selection of solutions to be varied and kept in an archive.
	TS-R2EA [27].	2018	R2 indicator-based achievement scalarizing function for the primary selection; the reference vector guides the objective space partition approach and the secondary selection strategy.
Hybrid method	KnRVEA [30].	2019	Based on knee points (convergence) and reference vector adaptation strategies; a knee adaptation strategy (diversity).
	Decomposition-based artificial bee colony algorithm [31].	2019	Decomposes a number of sub-problems (maintains diversity), which are simultaneously optimized by a modified ABC algorithm (rapid convergence).
	Decomposition-based ant colony [32].	2020	Decomposition-based many-objective ant colony optimization; an adaptive reference point mechanism.
	FDEA-II [33].	2021	Fractional dominance relation (strong convergence); the objective space decomposition approach is improved with a sub-space selection mechanism (to maintain the population diversity), a fractional dominance relation, and an improved objective space decomposition strategy.

In this paper, we introduced a new projection-plane-based evolutionary algorithm for many-objective optimization. It is different from the four types of algorithms mentioned above. It achieves dimensionality reduction by dividing all the objective dimensions into a projection plane [34] and free dimension, and determines the objective domain of a certain (some) objective dimension(s) according to user requirements to improve the solution speed. The projection plane is carved into projection grids; parallel evolution among each projection grid in the objective domain is carried out at the same time, so as to alleviate the selection pressure, maintain the diversity, and improve efficiency. The state-of-the-art evolutionary algorithms used on free dimensions maintain the convergence of the evolution. Compared to MOEA/P [34], MOEA/P is an algorithm that solves multi-objective problems by evolving on each ordering projection grid one by one. It sorts the solutions by calculating the value of the projection box, while MOEA/PII sorts solutions by using a free dimension algorithm.

2.1. Problem Definition

Definition 1. *Many-objective optimization problems (MaOPs): taking the minimization problem as an example, a many-objective optimization problem can be defined as follows in (1):*

$$\begin{aligned} \text{Minimize } & (f_1(X), f_2(X), f_3(X) \cdots \cdots f_M(X), X \in \Omega) \\ & g_j(X) \geq 0, j = 1, \cdots \cdots, J \\ & h_k(X) = 0, k = 1, \cdots \cdots, K \\ & X \in \Omega \end{aligned} \quad (1)$$

where $f_1(X), f_2(X), f_3(X) \cdots \cdots f_M(X)$ are the M objective functions; Ω is the decision space; $g_j(X), h_k(X)$ are constraint functions; J is the number of inequalities, and K is the number of equalities.

2.2. Definitions on MOEA/P Framework

Definition 2. *Projection Plane: According to the objective decision requirements, some major objective dimensions are selected to form the projection plane, and the objective dimensions on the projection plane are called projection dimensions. The projection plane P is a pure subset of the multi-objective function set F , namely $P \subset F$.*

Definition 3. *Free Dimension: The other dimensions of the objective space, except for the projection plane, are called free dimensions. The set composed of all free dimensions is called the free dimension set D , $D = F - P$.*

Definition 4. *Projection Grid: The projection plane is carved into a set of projection grids by segments on each projection dimension. The barycenter point of the projection grid is used as the projection grid vector, denoted V^g . In this paper, the projection grid vectors are simply called projection grid. V^g consists of the values of the dimensions that make up the projection plane.*

Definition 5. *Non-Pareto-Projection-Dominated Solution: Non-Pareto projection-dominated solution x^p is a solution that is not Pareto-projection-dominated by any other solutions; correspondingly, solution set ND^P is the set of all non-Pareto-projection-dominated solutions.*

Definition 6. *Pareto Projection Front: The objective set corresponding to non-Pareto-projection-dominated solution set ND^P is called the Pareto projection front, PPF.*

Definition 7. *Objective Distance ϵ : In the process of finding a solution, the purpose of setting the objective distance is to ensure that the distribution of the space will not be uneven due to the local proximity between the objective individuals. It means that the value of at least one objective dimension between any two objective individuals is not less than ϵ . Otherwise, it will be regarded as the same objective individual.*

Definition 8. *Free Dimension Algorithm:* The algorithm used on the free dimension is a free dimension algorithm. For example, the multi-objective optimization evolutionary algorithms, such as MOEA/DD, NSGAIII, and so on.

2.3. Definitions on New MOEA/PII Framework

Definition 9. *Objective Domain:* The objective domain is a scope of user-determined objective value, on a certain (some) objective dimension(s).

The objective dimension(s), which defined the objective domain, is always used as the projection plane. Alternatively, a certain (some) objective dimension(s) is selected to define the objective domain on the determined projection plane.

Definition 10. *Projection Elite:* If the projection point of the individual on the projection plane is outside the objective domain, we calculate the Tchebysheff distance between the boundary of the objective domain and the projection points. We choose the nearest top N individuals as projection elites and put them into the projection elites queue ProEliteList.

Definition 11. *Convergence Elite:* If the projection point of the individual on the projection plane is in the objective domain, we choose the top N individuals of each neighbor grid, which are sorted by the free dimension algorithm. They are regarded as the convergence elites of each neighbor grid and are placed into the convergence elites queue ConEliteList.

Definition 12. *Bi-Elite Queue:* In the Bi-Elite queue, there are convergence elites, followed by projection elites.

3. MOEA/PII Framework

The MOEA/PII framework is a multi-objective and many-objective evolutionary algorithm framework based on projection plane.

3.1. MOEA/PII Framework

Many-objective problems are divided into two parts. Some dimensions are regarded as a projection plane. The other dimensions are regarded as free dimensions. The projection plane is carved into projection grids, on which the population evolves. On the free dimensions, we can choose the traditional MOEAs and their state-of-the-art counterparts, such as MOEA/D, NSGAI (non-dominated sorting genetic algorithm-II), NSGAIII (non-dominated sorting genetic algorithm-III), and MOEA/DD, which are applied as free dimension algorithms.

The projection plane is uniformly divided into projection grids, according to the number of segments, customized by the users. If the objective domain is defined, the segments are divided on the objective domain. In the evolution process of a projection grid, the distribution of solutions is adjusted by the objective distance ϵ , which guarantees diversity in the grid. Meanwhile, the uniformity of projection grid distribution ensures the diversity in the whole projection plane space.

The determined objective domain on the projection plane is shown in Figure 1. The working process of MOEA/PII is illustrated in Figure 2. In the evolution process, the offspring reproduced by crossover and mutation operations and each individual p in the offspring are located in a projection grid G_i or out of the objective domain. If p is in G_i , it will be stored in ConEliteList(i) by the free dimension algorithm. If p falls outside the objective domain, it will be stored in ProEliteList and sorted according to its distance from the domain. The next generation population, which is composed of the solutions in ConEliteLists and the solutions in ProEliteList, is added when the population quantity is not sufficient.

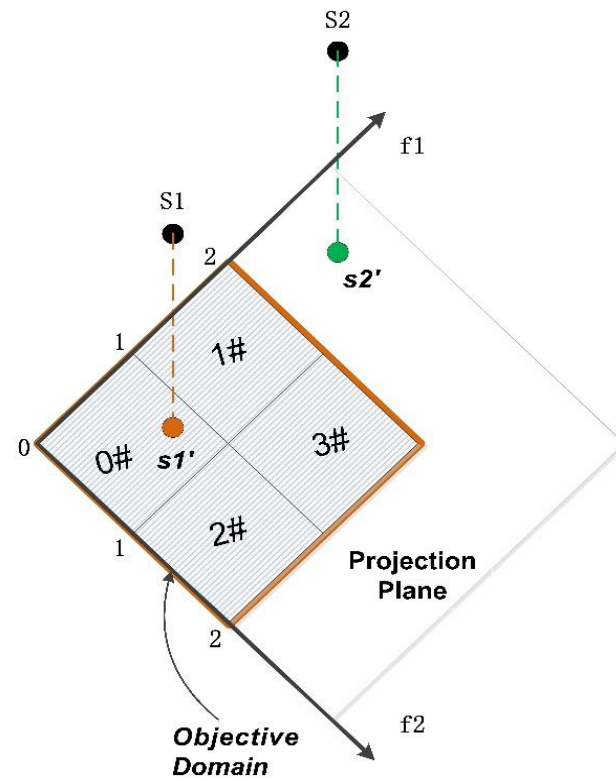


Figure 1. Projection plane, projection grid, projection point, and objective domain.

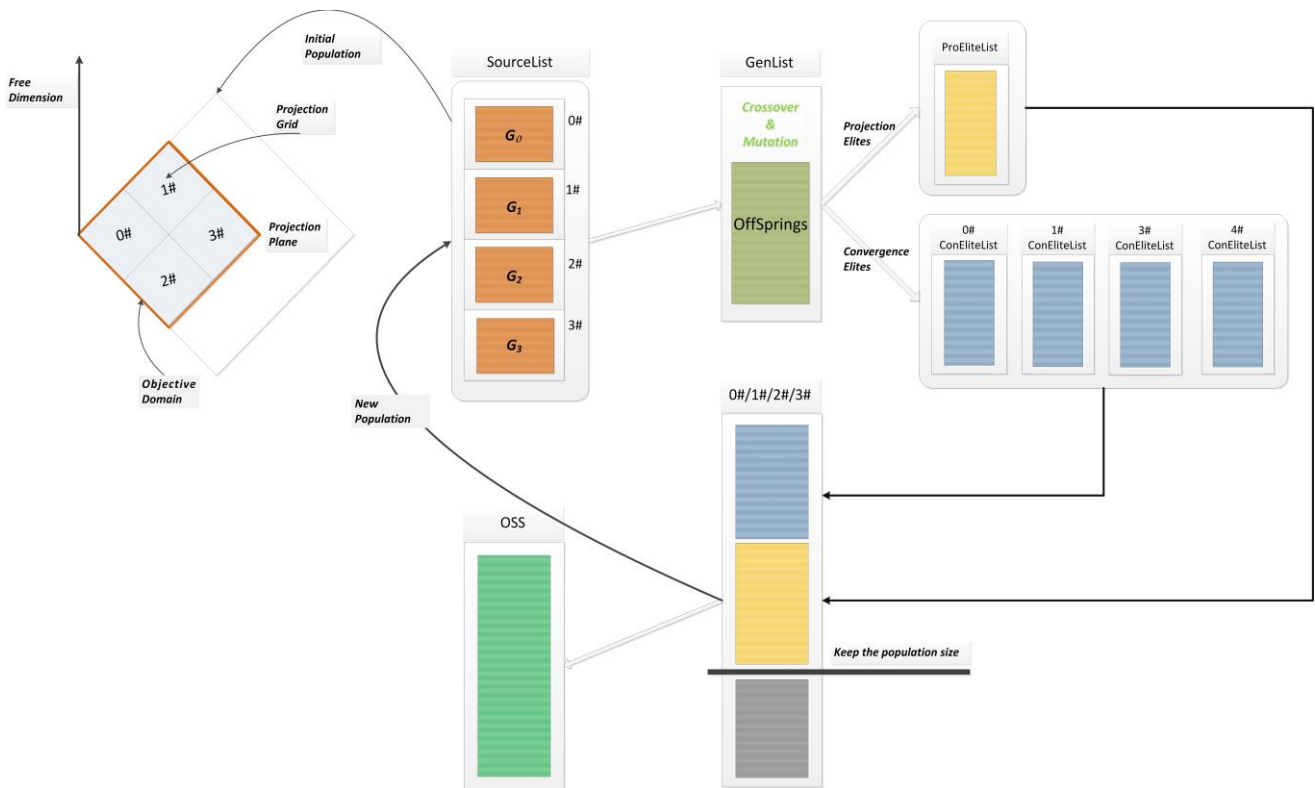


Figure 2. MOEA/PII working process.

Assuming a 4-objective problem, where f_1 , f_2 , f_3 , and f_4 are the objectives, users have selected f_1 and f_2 as the projection plane, and determined the objective domain according to their requirements, i.e., the values of f_1 and f_2 are both within the range of

[0, 2]. Each projection dimension is divided into two segments, resulting in four projection cells, as shown in Figure 1. The free dimension algorithm employs MOEA/DD. The population N is initialized in each projection grid, with individuals saved in the SourceList. After performing crossover and mutation operations, the generated offspring are saved in the GenList.

Assuming $S1$ and $S2$ are two individuals in the GenList, after calculating the objective values for each dimension, we obtain $S1$ (0.8, 0.8, 3,4), $S2$ (2.5, 0.5, 2,6). $s1'$ (0.8, 0.8) and $s2'$ (2.5, 0.5) are the projection points of $S1$ and $S2$ on the projection plane, respectively. $s1'$ (0.8, 0.8) falls into the projection grid G_0 and is saved in the ConEliteList(0); $s2'$ (2.5, 0.5) falls outside the determined objective domain and is saved in the ProEliteList. When the individuals are saved into the corresponding elite list, the free dimension algorithm is used to perform dominance sorting.

After completing the evolution of all individuals in the GenList, the ConEliteList(i) (where i is from 0 to 3) and ProEliteList are formed. The ConEliteList(i) is sequentially put into a temporary solution set according to the dominance results. If it is not sufficient to reach the population size, individuals from the ProEliteList are supplemented until the population size is reached. The individuals in this temporary solution set are saved in the OSS and are used as the initial population for the next iteration evolution. The solution set generated in each iteration is stored in the OSS according to the dominance relationship.

The convergence of MOEA/PII is determined by the free dimension algorithm. The distribution is guaranteed by the uniform partition of the projection grid. By determining the objective domain, individuals that do not meet user requirements cannot participate in evolution, making evolution more effective. Applications in the real world, such as WSN deployment problems, are detailed in Section 4.

The MOEA/PII framework working process is expressed in the algorithm. The meanings of the notations used in the algorithm are shown in Table 2.

Table 2. The meanings of notations in algorithms.

Notations	Meanings
DS	objective decision space
NumOfGrids	the number of projection grids
E	evolution generation
N	initial population size
P	individual
G_i	the i th projection grid
SourceList	all the individuals of the grids
GenList	offspring by mutation and crossover operation
ProEliteList	individual that fall outside the objective domain will be stored in ProEliteList
ConEliteList(i)	if individual is in G_i , it will be stored in ConEliteList(i) by the free dimension algorithm
OSS	objective solution set

The Algorithms 1 and 2 works as follows:

Algorithm 1: MOEA/PII

Input:

- (1) Many-Objective Problem (MaOP);
 - (2) End condition;
 - (3) DS: Objective decision space;
 - (4) NumOfGrids: the number of projection grid;
 - (5) E: Evolution generation;
 - (6) N: Initial Population Size;
 - (7) Free Dimension Algorithm;
-

Algorithm 1: Cont.

Output: Objective Solution Set (OSS);**Algorithm:****Step 1:** Set one (more) objective dimension(s) as the projection plane according to **DS**; According to the NumofGrids to be carved, divide each normalized objective dimension into segments.**Step 2:** Initialize the populationInitialize population, the population size is N ; Construct the chromosome and the initialize individual gene sequence ensuring that all individuals meet the MOP constraints; for each individual, perform the initial calculation to obtain the corresponding objective function value;Put all individuals into **SourceList**;**Step 3:** Population evolutionFor $e = 1$ to E :

{

Step 3.1: Set Bi-Elite queues for the i th projection grid G_i , **ConEliteList**(i) and **ProEliteList**;**Step 3.2:** All individuals in **SourceList** are crossed and mutated with their neighbor each other. The new generation offspring is put into **GenList**;**Step 3.3:** According to the objective value of each individual in **GenList**, update the the Bi-Elite queues for each projection grid; (Algorithm 2)**Step 3.4:** Put the Bi-Elite queue back into **SourceList**, with the sequence of convergence elites first and then projection elites. At the same time, filter and sort them by the fitness value and objective distance ϵ .

}

Step 4: The solutions in **SourceList** are the optimal solution set(OSS).

Algorithm 2: Bi-Elite

Input:

- (1) GenList
- (2) NumOfNeighbor: the number of Neighbor;
- (3) NeighborPool: the neighbor vector;
- (4) Free Dimension Algorithm;

Output:Bi-Elite Queue of each projection grid: **ConEliteList**(i) and **ProEliteList**;**Algorithm:**For each p in **GenList**

{

Step 1: Calculate the position of p so as to obtain an index of a projection grid G_i , where p falls in the projection grid, or is its distance from the domain if it falls outside of the objective domain.**Step 2:** Fill the Bi-Elite Queue**Step 2.1:** If p is on G_i , store it in **ConEliteList**(i) using the free dimension algorithm.**Step 2.2:** If p falls outside of the objective domain, it will be stored in **ProEliteList** and sorted according to its distance from the domain.

}

3.2. The Advantage of Bi-Elite Strategy

N: population size

M: the number of objectives

Sorting $//(N \times M)$ times

Apply the MOEA/PII algorithm framework with Bi-Elite strategy

P: the number of the projection plane dimensions

F: the number of free dimensions

```

M = P + F
If ConEliteList(S) ← In(ProjectionGrid, N)
ProEliteList(R) ← Out(ObjectiveDomain, N)
And N = S + R
Then For each S in ConEliteList on F free dimensions
Sorting //(S × F) times
For each R in ProEliteList on P projection plane
Sorting //(R × P)times

```

We can see that $N \times M = (S + R) \times M = S \times M + R \times M \ggg S \times F + R \times P$, so the Bi-Elite strategy compressed the evolutionary magnitude.

3.3. Computational Complexity Analysis

1. The complexity of the crossover and mutation operation.

For each generation, with the crossover probability of 1.0 and the mutation probability of 1.0 (1/n), we can obtain four individuals for each individual in the population. The size of the initial population is N, and the offspring population is 4N. Each individual should be compared with the others in the population. The sorting complexity is $O(4N)$.

2. The complexity of the Bi-Elite sorting.

We choose the binary search as the sorting approach to the Bi-Elite queues. The number of solutions in ConEliteList is S, and the number of free dimension is F. The complexity of sorting in the ConEliteList is $O(N \cdot \log S \cdot F)$. The number of solutions in the ProEliteList is R, and the number of projection plane dimension is P. The complexity of sorting in the ProEliteList is $O(N \cdot \log R \cdot P)$. The total complexity of sorting with the Bi-Elite strategy is $O(N \cdot \log S \cdot F + N \cdot \log R \cdot P)$, where $M = F + P$ and $N = S + R$. Correspondingly, the complexity of binary search without a Bi-Elite strategy is $O(N \cdot \log N \cdot M)$.

We can see that the Bi-Elite strategy compressed the evolutionary magnitude, $O(N \cdot \log S \cdot F + N \cdot \log R \cdot P) \lll O(N \cdot \log N \cdot M)$.

4. Experimental Studies and Discussions

4.1. Benchmark Test Instances

In our experiments, each algorithm was ran independently 30 times for each test instance. We introduce the general parameter settings for the experiments. The benchmark test instances of DTLZ1-7. It is widely used in the comparison of multi-objective evolutionary algorithms. DTLZ [35] was designed by Deb, K., Thiele, L., Laumanns, M., and Zitzler in 2002. The DTLZ test instances have different characteristics, and this creates many difficulties for the algorithm to converge to the Pareto front. DTLZ1 and DTLZ3 include many difficulties for the algorithm to converge to the Pareto front. DTLZ2 and DTLZ4 are used to test the ability of the algorithm to deal with different shape problems. DTLZ3 is a highly multimodal problem. DTLZ4 is a test problem whose density of the points on the true PF is strongly biased. This test problem is designed to verify whether an MOEA is able to maintain a proper distribution of the candidate solutions. DTLZ5 and DTLZ6 are the degenerated test problems, whose PFs are irregular. The DTLZ7 test problem has disconnected and degenerate PFs.

In addition, the benchmark test instances of WFG1-9 [36] are widely used in the comparison of multi-objective evolutionary algorithms. WFG was proposed by Walking Fish Group. WFG1 is designed with a flat bias and a mixed structure of the PF. WFG2 is a test problem that has a disconnected PF. WFG3 is a difficult problem where the PF is degenerate and the decision variables are non-separable. WFG4 to WFG9 are designed with different difficulties in the decision space. WFG4 is a multi-modal problem, WFG5 is a landscape deception problem and WFG6, WFG7, WFG8, and WFG9 are non-separable; their true PFs are the same convex structure. All of them can be extended to any number of objectives and decision vectors.

4.2. Performance Indicator

1. IGD-metric.

Inverted generational distance (IGD) [37] was used as the performance metric in our experimental analysis. IGD measures the average distance from a set of solutions P^* in the true PF to the approximation solution set S .

It can be formulated as follows in (2):

$$\text{IGD}(S, P^*) = \frac{\sum_{x \in P^*} \text{dist}(x, S)}{|P^*|} \quad (2)$$

where P^* is a solution set uniformly sampled in the true PF. $\text{dist}(x, S)$ is the Euclidean distance between the solution S and its nearest point $x \in P^*$, and $|P^*|$ is the cardinality of P^* . The smaller the value of IGD, the better its convergence and diversity, and the closer it is to the true PF. It indicates that S is the sub-set of P^* when $\text{IGD}(S, P^*) = 0$. $\text{IGD}(S, P^*)$ can measure both the diversity and convergence of S in a sense.

2. HV-metric.

Hyper-volume (HV) [38] is the volume of the objective space dominated by the approximate solution set S with the bounded of $r^* \cdot r^* = (r_1^*, r_2^*, \dots, r_m^*)$ is dominated by all the solutions in the approximation solution set S , where r^* is the reference point in the objective space.

It can be formulated as follows in (3):

$$\text{HV}(S) = \text{VOL} \left(\bigcup_{x \in S} [f_1(x), r_1^*] \times \dots \times [f_m(x), r_m^*] \right) \quad (3)$$

where $f_1(x), \dots, f_m(x)$ is the objective function value of x . The larger the value of HV , the closer it is to the true PF. HV is affected by the number of objectives and the selection of reference points.

3. C-metric.

C-metric [39] is an indicator to evaluate the convergence performance of the algorithm using two PF approximate solution sets, and its calculation is based on the Pareto-dominance relationship. Suppose A and B comprise the approximate solution set, then the C-metric represents the proportion shown in (4), in which the individuals in B are dominated by at least one individual in A .

$$\text{C}(A, B) = \frac{|\{u \in B | \exists v \in A : v \prec u\}|}{|B|} \quad (4)$$

C-metric $(A, B) = 1$ means that all individuals in B are dominated by individuals in A , that is, the convergence of A is better.

4.3. Experiments on Pareto Front

In order to investigate the performance of algorithms with MOEA/PII framework, the experiments were carried out on the benchmark test cases. Throughout this paper, we use an abbreviation of N3P2 to denote the algorithm NSGAIII with the MOEA/PII framework, and DDP2 to denote the algorithm MOEA/DD with the MOEA/PII framework.

In addition, all of the algorithms compared in this study have several parameters. The parameters of the crossover operator and mutation operator are shown in Table 3.

Table 3. Parameter settings.

Parameters	Settings
SegmentsCrossOver	$P_c = 1.0$
Random Mutation	$P_m = 1/n$

In an effort to compare the performance of algorithms, we set the population size and generations for the original algorithms (1000, 1000). When applying the MOEA/PII framework, the population initializes in each projection grid, so the number of initial population size will increase by a certain multiple. The population size is calculated by the number of projection grids (G), $1000/G$. Accordingly, the generation size for the algorithms with MOEA/PII is calculated by the number of projection dimensions (D), $1000/(D + 1)$. For example, if the number of the projection dimension is 3 and there are 2 segments on each dimension, the generation size is 250 and population size is 125. The parameters of generations and population size are shown in Table 4.

Table 4. General parameter settings. Number of generations for different algorithms (generations and population size).

ALGORITHM	M = 3	M = 5	M = 8	M = 10	M = 15
MOEA/DD	(1000, 1000)	(1000, 1000)	(1000, 1000)	(1000, 1000)	(1000, 1000)
NSGAIII	(1000, 1000)	(1000, 1000)	(1000, 1000)	(1000, 1000)	(1000, 1000)
DDP2	(500, 500)	(500, 500)	(300, 250)	(200, 250)	(200, 150)
N3P2	(500, 500)	(500, 500)	(300, 250)	(200, 250)	(200, 150)

DTLZ1 includes many difficulties, so it is a challenge for the algorithm to converge to the Pareto front. DTLZ4 has different shape problems, so that the density of the points on the true PF is strongly biased. DTLZ5 is a degenerated test problem whose PFs are irregular. WFG1, WFG2, and WFG3 are a mixed structure, disconnected, and degenerate, respectively. They have a certain degree of representativeness.

It is worth noting that for the benchmark problem DTLZ and WFG, DDP2 and N3P2 tend to obtain more diverse Pareto approximate solutions in the PF with 3-objective, as illustrated in Figures 3 and 4.

In order to observe the performance of DDP2 and N3P2 on MaOPs visually, the parallel coordinates of the non-dominated front obtained by the MOEA/DD and DDP2 algorithms on DTLZ1-4 DTLZ1, DTLZ4, and DTLZ5 with 15-objective, and those obtained by the NSGAIII and N3P2 algorithms on WFG1-4 with 10-objective are depicted in Figures 5 and 6, respectively. As can be seen, DDP2 and N3P2 were able to obtain a solution set with a better diversity.

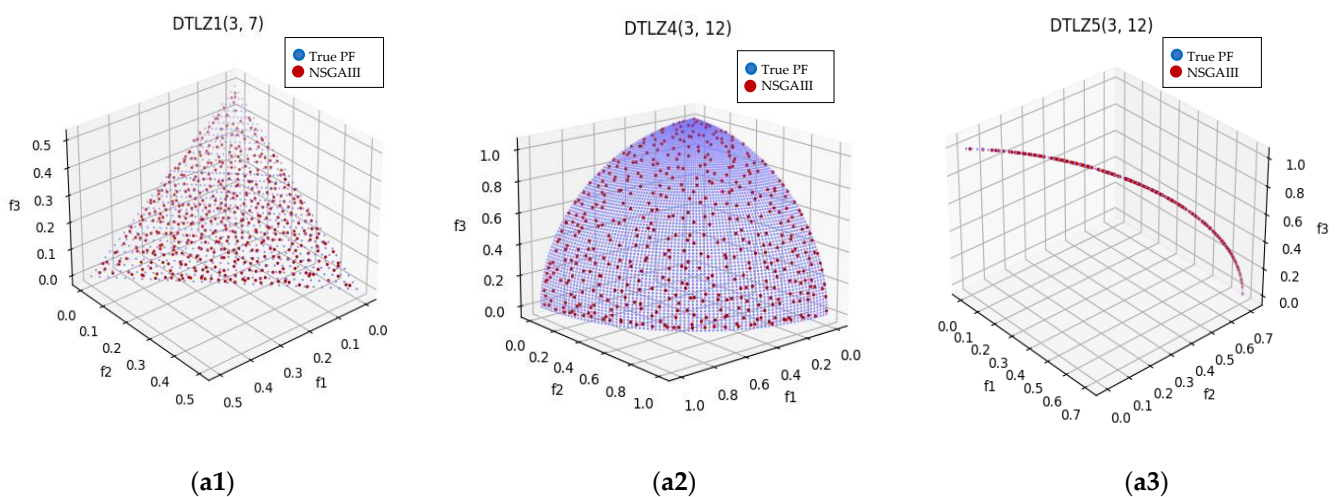


Figure 3. Cont.

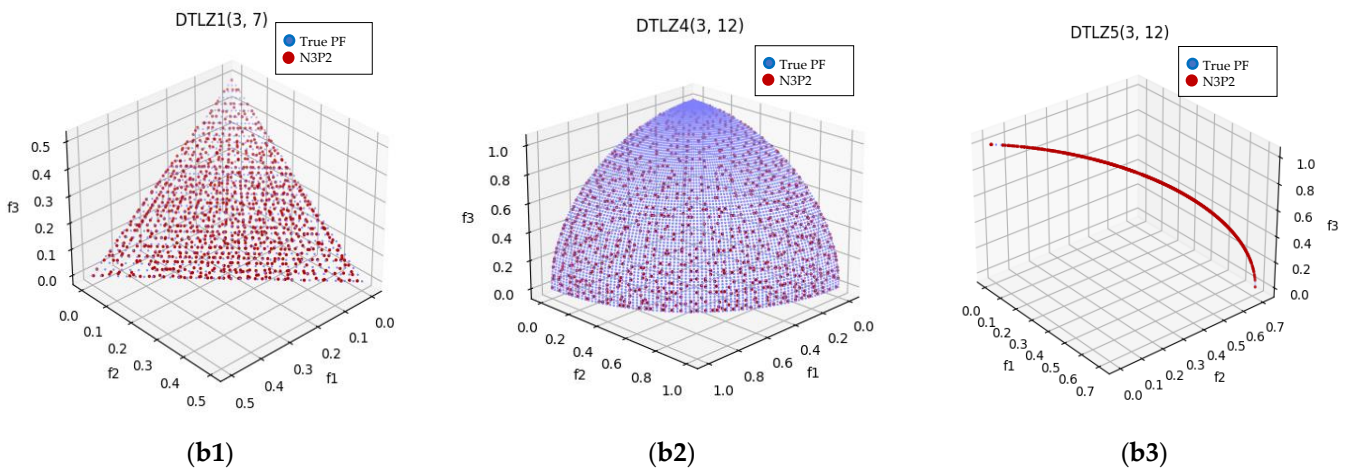


Figure 3. Plot of the approximate PFs with NSGAIII and N3P2 for benchmark test instance DTLZ1, DTLZ4, and DTLZ5 with 3-objective. From (a1–a3) with NSGAIII, and from (b1–b3) with N3P2.

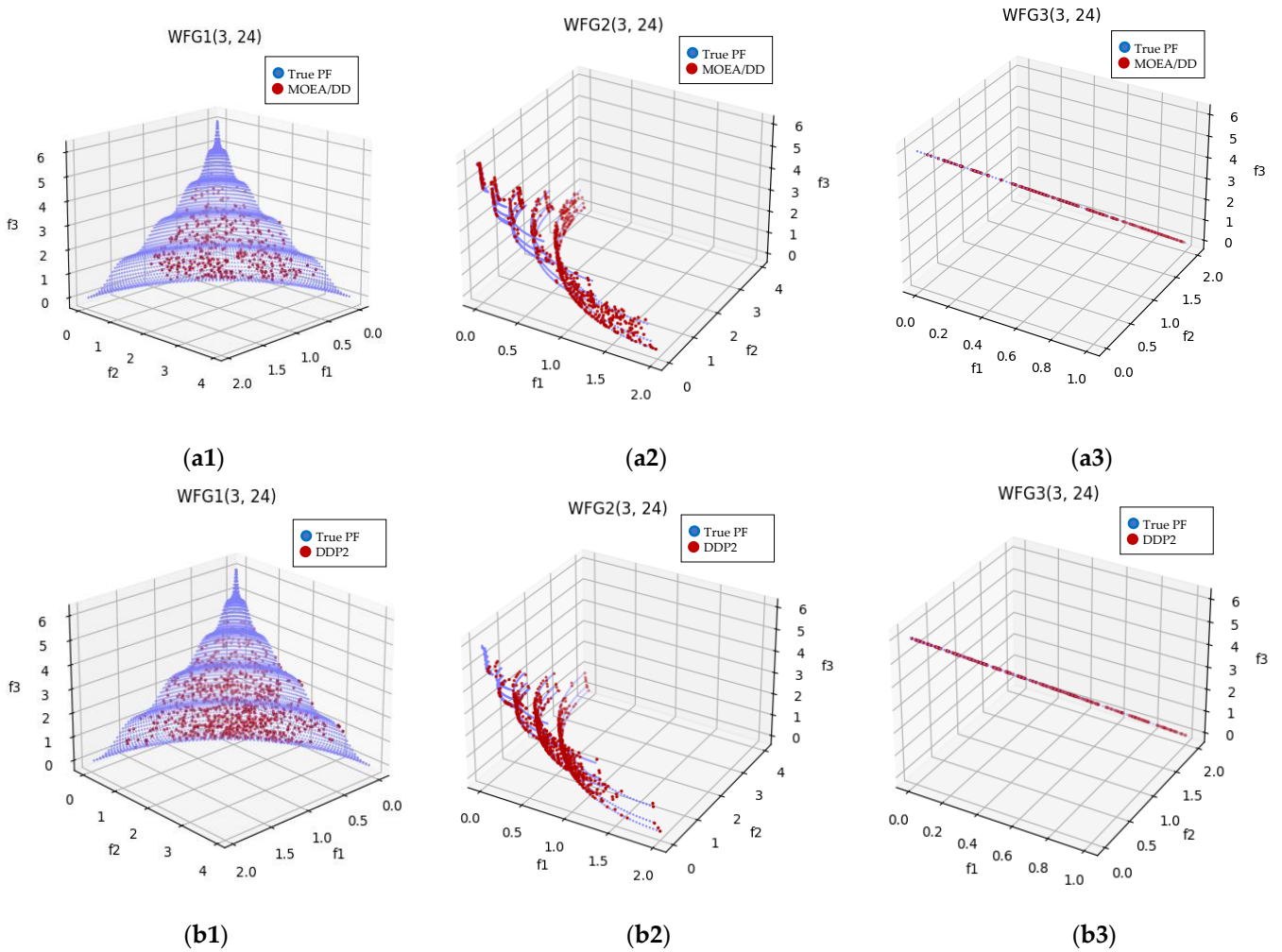


Figure 4. Plot of the approximate PFs with MOEA/DD and DDP2 for benchmark test instance WFG1–3 with 3-objective. From (a1–a3) for WFG1–3 with MOEA/DD, from (b1–b3) with DDP2.

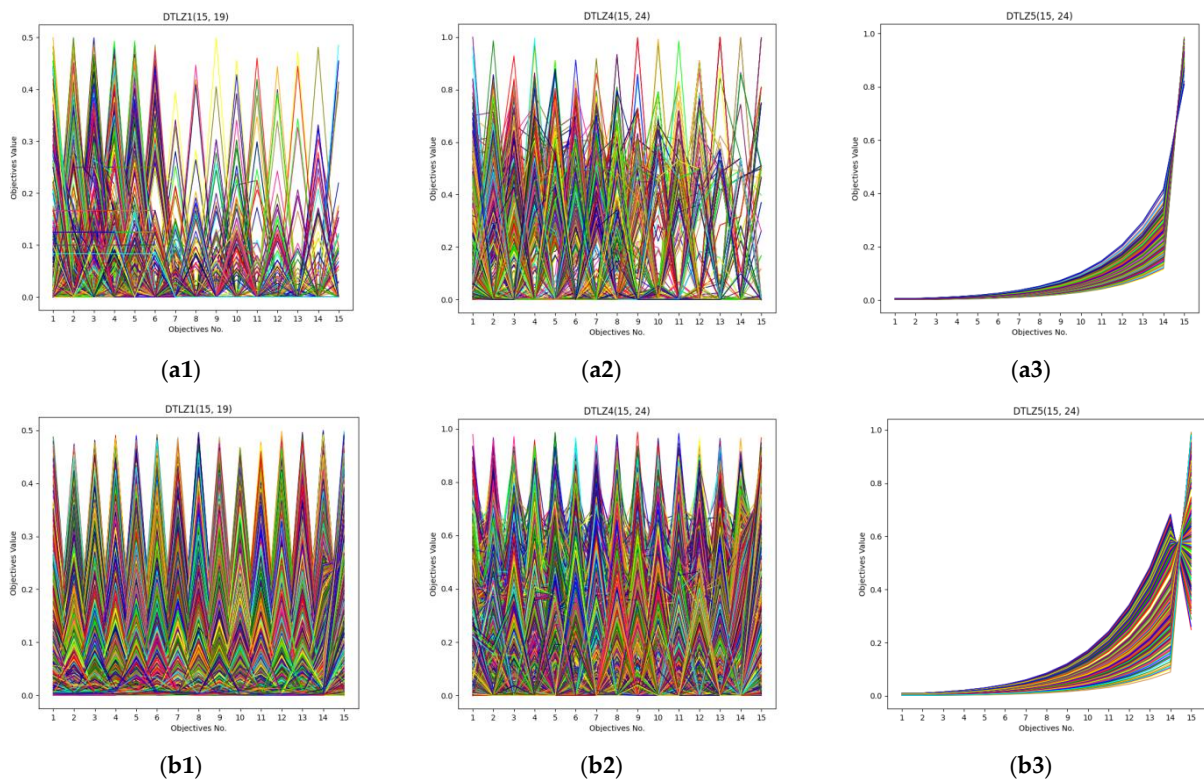


Figure 5. Parallel coordinates of the non-dominated solutions obtained by MOEA/DD and DDP2 for benchmark test instances DTLZ1, DTLZ4, and DTLZ5 with 15-objective. From (a1–a3) with MOEA/DD, and from (b1–b3) with DDP2 (legends: each colored line represents a solution).

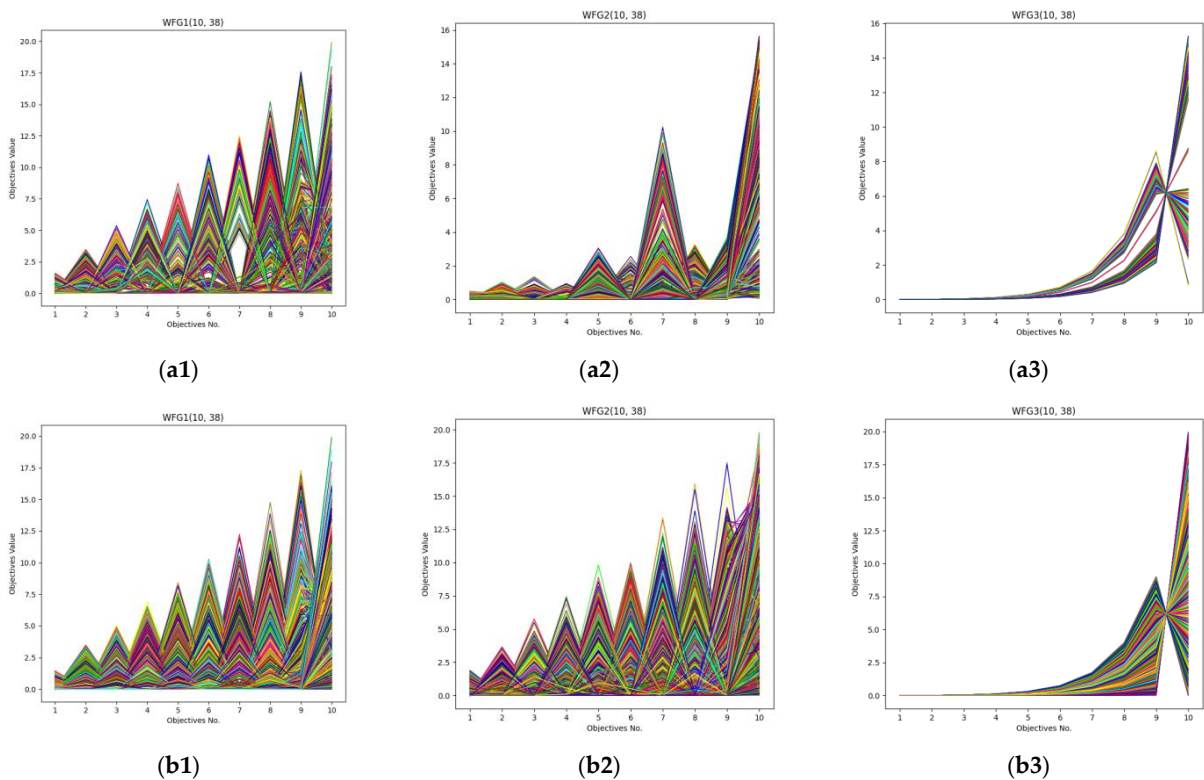


Figure 6. Parallel coordinates of the non-dominated solutions obtained by NSGAIII and N3P2 for benchmark test instance WFG1-3 with 10-objective. From (a1–a3) with NSGAIII, and from (b1–b3) with N3P2 (legends: each colored line represents a solution).

4.4. Performance Analysis

The results of the IGD-metric values obtained by the MOEA/DD and DDP2 algorithms on the DTLZ test instances are summarized in Table 5, in which the best results are highlighted. The results of the IGD-metric values obtained by the NSGAIII and N3P2 algorithms on the WFG test instances are summarized in Table 6, in which the best results are highlighted. The results of the HV-metric values obtained by the NSGAIII and N3P2 algorithms on the DTLZ test instances are summarized in Table 7, in which the best results are highlighted.

Table 5. The mean and deviation value of the IGD-metric of MOEA/DD and DDP2 from 30 independent runs for each DTLZ test instance. Best performance is highlighted in bold face in gray background.

Test Instances	Algorithm	M = 3	M = 5	M = 8	M = 10	M = 15
DTLZ1	MOEA/DD	0.010 (0.001)	0.110 (0.002)	0.179 (0.001)	0.180 (0.002)	0.181 (0.002)
	DDP2	0.010 (0.001)	0.075 (0.002)	0.173 (0.001)	0.173 (0.001)	0.176 (0.001)
DTLZ2	MOEA/DD	0.026 (0.000)	0.198 (0.001)	0.454 (0.001)	0.524 (0.001)	0.542 (0.001)
	DDP2	0.028 (0.000)	0.169 (0.002)	0.415 (0.001)	0.507 (0.002)	0.542 (0.001)
DTLZ3	MOEA/DD	0.030 (0.000)	0.260 (0.002)	0.435 (0.001)	0.522 (0.001)	0.553 (0.001)
	DDP2	0.029 (0.000)	0.196 (0.001)	0.371 (0.001)	0.417 (0.003)	0.543 (0.001)
DTLZ4	MOEA/DD	0.034 (0.001)	0.229 (0.001)	0.451 (0.001)	0.448 (0.001)	0.456 (0.002)
	DDP2	0.031 (0.001)	0.197 (0.001)	0.363 (0.001)	0.401 (0.001)	0.449 (0.002)
DTLZ5	MOEA/DD	0.053 (0.000)	0.113 (0.000)	0.394 (0.001)	0.472 (0.001)	0.501 (0.001)
	DDP2	0.009 (0.001)	0.069 (0.001)	0.307 (0.001)	0.421 (0.001)	0.479 (0.001)
DTLZ6	MOEA/DD	0.078 (0.001)	0.156 (0.001)	0.377 (0.001)	0.468 (0.001)	0.558 (0.001)
	DDP2	0.016 (0.001)	0.079 (0.001)	0.316 (0.001)	0.468 (0.001)	0.534 (0.001)
DTLZ7	MOEA/DD	0.065 (0.001)	0.192 (0.001)	0.435 (0.001)	0.477 (0.001)	0.565 (0.001)
	DDP2	0.062 (0.001)	0.167 (0.001)	0.402 (0.001)	0.461 (0.001)	0.526 (0.001)

Table 6. The mean and deviation value of the IGD-metric of NSGAIII and N3P2 from 30 independent runs for each WFG test instance. Best performance is highlighted in bold face in gray background.

Test Instances	Algorithm	M = 3	M = 5	M = 8	M = 10	M = 15
WFG 1	NSGAIII	0.098 (0.001)	0.591 (0.001)	4.406 (0.000)	7.081 (0.000)	11.896 (0.001)
	N3P2	0.106 (0.000)	0.575 (0.000)	1.504 (0.001)	1.861 (0.001)	4.789 (0.001)
WFG 2	NSGAIII	0.124 (0.001)	0.559 (0.000)	7.169 (0.001)	10.920 (0.000)	18.124 (0.001)
	N3P2	0.133 (0.001)	0.533 (0.001)	2.583 (0.000)	3.498 (0.000)	10.534 (0.000)
WFG 3	NSGAIII	0.073 (0.000)	0.350 (0.000)	1.301 (0.000)	1.785 (0.000)	2.635 (0.001)
	N3P2	0.020 (0.001)	0.111 (0.000)	0.686 (0.001)	1.250 (0.000)	2.348 (0.001)
WFG 4	NSGAIII	0.133 (0.000)	0.690 (0.000)	2.441 (0.000)	4.572 (0.000)	6.337 (0.001)
	N3P2	0.093 (0.001)	0.532 (0.000)	1.901 (0.001)	3.442 (0.001)	5.106 (0.001)
WFG 5	NSGAIII	0.151 (0.000)	0.752 (0.000)	2.971 (0.000)	6.833 (0.000)	8.926 (0.001)
	N3P2	0.112 (0.001)	0.526 (0.000)	2.088 (0.001)	3.549 (0.000)	6.479 (0.001)
WFG 6	NSGAIII	0.230 (0.000)	1.256 (0.000)	3.257 (0.000)	6.164 (0.000)	8.764 (0.001)
	N3P2	0.097 (0.000)	0.558 (0.001)	2.068 (0.001)	3.882 (0.001)	4.586 (0.001)
WFG 7	NSGAIII	0.381 (0.000)	1.081 (0.000)	3.261 (0.000)	5.129 (0.000)	7.873 (0.001)
	N3P2	0.790 (0.001)	0.531 (0.000)	2.012 (0.001)	3.609 (0.001)	5.985 (0.001)
WFG 8	NSGAIII	0.162 (0.000)	1.026 (0.000)	3.419 (0.000)	5.656 (0.000)	8.164 (0.001)
	N3P2	0.094 (0.000)	0.526 (0.001)	2.103 (0.001)	3.864 (0.001)	6.325 (0.001)
WFG 9	NSGAIII	0.187 (0.000)	0.812 (0.000)	3.248 (0.000)	5.506 (0.000)	8.487 (0.001)
	N3P2	0.102 (0.001)	0.546 (0.001)	2.071 (0.001)	3.624 (0.000)	6.109 (0.001)

Table 7. The mean and deviation value of the HV-metric of NSGAIII and N3P2 from 30 independent runs for each DTLZ test instance. Best performance is highlighted in bold face in gray background.

Test Instances	Algorithm	M = 3	M = 5	M = 8	M = 10	M = 15
DTLZ1	NSGAIII	0.265 (0.016)	0.249 (0.023)	0.232 (0.021)	0.227 (0.021)	0.220 (0.026)
	N3P2	0.271 (0.021)	0.261 (0.013)	0.238 (0.021)	0.235 (0.019)	0.225 (0.025)
DTLZ2	NSGAIII	0.226 (0.032)	0.183 (0.016)	0.112 (0.023)	0.283 (0.023)	0.118 (0.033)
	N3P2	0.226 (0.022)	0.205 (0.023)	0.152 (0.012)	0.247 (0.027)	0.180 (0.026)
DTLZ3	NSGAIII	0.229 (0.031)	0.224 (0.023)	0.180 (0.011)	0.162 (0.022)	0.133 (0.016)
	N3P2	0.241 (0.024)	0.232 (0.018)	0.189 (0.014)	0.181 (0.031)	0.174 (0.033)
DTLZ4	NSGAIII	0.278 (0.023)	0.318 (0.023)	0.243 (0.033)	0.203 (0.031)	0.164 (0.021)
	N3P2	0.316 (0.024)	0.332 (0.022)	0.266 (0.025)	0.253 (0.034)	0.190 (0.032)
DTLZ5	NSGAIII	0.202 (0.023)	0.213 (0.018)	0.287 (0.023)	0.183 (0.026)	0.161 (0.023)
	N3P2	0.216 (0.022)	0.223 (0.017)	0.299 (0.012)	0.195 (0.022)	0.179 (0.019)
DTLZ6	NSGAIII	0.158 (0.018)	0.144 (0.012)	0.176 (0.034)	0.148 (0.021)	0.131 (0.023)
	N3P2	0.166 (0.022)	0.166 (0.016)	0.190 (0.025)	0.153 (0.022)	0.147 (0.012)
DTLZ7	NSGAIII	0.109 (0.019)	0.084 (0.019)	0.123 (0.020)	0.141 (0.023)	0.097 (0.022)
	N3P2	0.137 (0.022)	0.096 (0.013)	0.136 (0.011)	0.146 (0.022)	0.105 (0.022)

In order to provide a visual representation, Figure 7a–c show the statistical trend in the data from Tables 5–7 respectively. Two groups of data from each table were chosen for a clear and intuitive illustration.

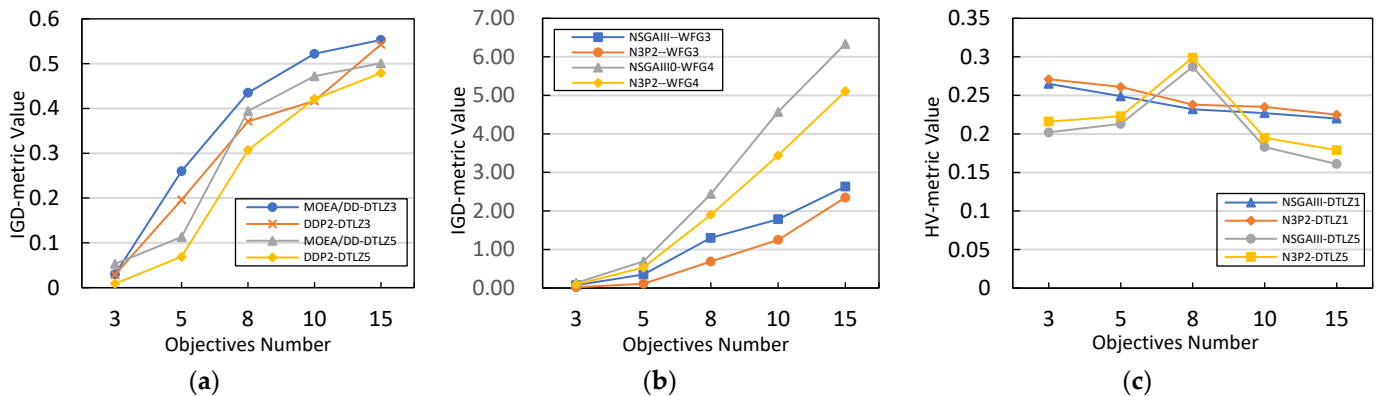


Figure 7. (a) The mean IGD-metric value of MOEA/DD and DDP2 on DTLZ from Table 5; (b) The mean IGD-metric value of NSGAIII and N3P2 on WFG from Table 6; (c) The mean HV-metric value of NSGAIII and N3P2 on DTLZ from Table 7.

It can be seen that the algorithms (MOEA/DD and NSGAIII) with MOEA/PII framework exhibited the best performance on the benchmark test instances DTLZ1–7 and WFG1–9, with 3-, 5-, 8- 10-, and 15-objective.

4.5. Experiments on Projection Grids

This experiment was focused on projection grids with the MOEA/PII framework. After choosing the projection dimensions, we set the segments on them to carve the projection plane. Each projection dimension was segmented, and all segments of each dimension were combined to form the projection grids of the projection plane.

We ran the algorithms on the 5-objective DTLZ1 test instance and recorded the results of the IGD-metric values in Tables 8 and 9.

Table 8. The mean and deviation value of the IGD-metric of MOEA/DD and NSGAIII for 30 independent runs on 5-objective DTLZ1 test instance.

	Algorithm	IGD
DTLZ1	MOEA/DD	0.110 (0.002)
	NSGAIII	0.031 (0.000)

Table 9. The mean and deviation value of the IGD-metric of DDP2 and N3P2 for 30 independent runs on 5-objective DTLZ1 test instance with different numbers of projection grids.

	Projection Grids	DDP2	N3P2
DTLZ1	2D_2S/D_4G	0.075 (0.001)	0.025 (0.001)
	3D_2S/D_8G	0.073 (0.001)	0.027 (0.000)
	2D_3S/D_9G	0.067 (0.001)	0.023 (0.001)
	2D_4S/D_16G	0.063 (0.000)	0.02 (0.001)
	4D_2S/D_16G	0.081 (0.001)	0.03 (0.001)
	3D_3S/D_27G	0.062 (0.000)	0.017 (0.001)
	3D_4S/D_64G	0.053 (0.000)	0.014 (0.000)
	4D_3S/D_81G	0.053 (0.000)	0.014 (0.000)
	4D_4S/D_256G	0.053 (0.000)	0.014 (0.000)

The column of projection grids in Table 9 and the X-axis coordinate scale in Figure 8 are annotated by the number of projection dimensions, segments per dimension, and projection grids. For example, a projection plane is two dimensional (2D), and each dimension is divided into two segments (2S/D). The number of combination is four, which is the number of projection grids (4G). It is recorded as 2D_2S/D_4G.

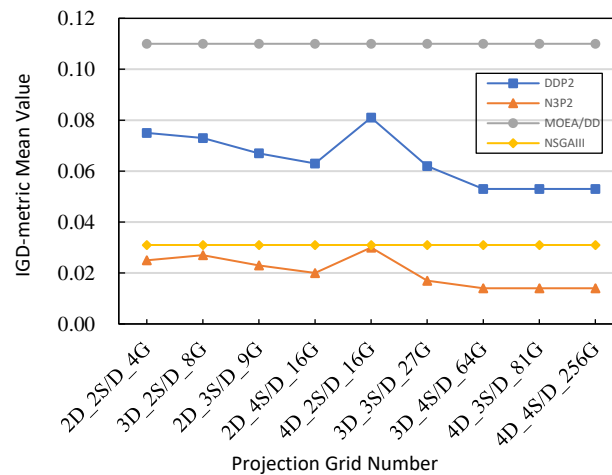


Figure 8. The mean value of the IGD-metric obtained by MOEA/DD, NSGAIII, DDP2, and N3P2 on the 5-objective DTLZ1 benchmark test instance with different numbers of projection grids.

Figure 8 shows the mean value of the IGD-metric obtained by MOEA/DD, NSGAIII, DDP2, and N3P2 on the 5-objective DTLZ1 benchmark test instance with different numbers of projection grids. We can see that for the same number of projection dimensions, the more the projection grids, and the better the IGD-metric value.

There was also another scenario of 2D_4S/D_16G and 4D_2S/D_16G, which was different. We know that the IGD-metric value expresses the convergence and diversity of the algorithm at the same time. Because 4D_2S/D_16G has two segments for each dimension, the diversity is worse than for four segments for each dimension (2D_4S/D_16G). Therefore, the IGD-metric value of 2D_4S/D_16G was better than that of 4D_2S/D_16G. In addition, for the same number of projection grids, we should choose the fewest projection dimensions and more segments.

4.6. Experiments on Wireless Sensor Networks (WSNs) Deployment Problem

Since the performance of the proposed algorithm was competitive on unconstrained test problems, we investigated its performance on wireless sensor networks deployment problem [40]. When we deploy sensor nodes, we need to consider a series of problems, such as the coverage of the detection space, the connectivity among sensor nodes, the cost, the network life cycle, and so on. However, there are mutual constraints and imbalanced goals among these problems, so we usually regard them as multi-objective optimization problems.

Suppose that a 100-square-meter plane with a width of 10 m and a length of 10 m is evenly divided into grids, each denoted C_{ij} . This is the detection area detected by the WSN composed of a set of sensor nodes.

The WSN is composed of a set of sensors $S = s_1, s_2 \dots s_n$ and a set of sink nodes $SK = sk_1, sk_2 \dots sk_m$. The cost of the wireless sensor network includes two parts: the sensor node cost and sink node cost. The cost can be calculated using (5).

$$f_1 = C_s * \| S \| + C_{sk} * \| SK \| \quad (5)$$

where C_s and C_{sk} represent the purchase costs of the sensor node and sink node, respectively. S and SK are the quantity of sets S and SK , respectively.

Each sensor node has a sensing area and can sense a target within its sensing radius R . The Euclidean distance between C_{ij} and the sensor node S_i is denoted $d(C_{ij}, S_i)$. The case in which the target C_{ij} is covered is represented by (6).

$$cov(C_{ij}, S_i) = \begin{cases} 1 & d(C_{ij}, S_i) \leq R \\ 0 & otherwise \end{cases} \quad (6)$$

The coverage fraction can be calculated by (7).

$$f_2 = \frac{\sum_i \sum_j cov(C_{ij}, S_i)}{100} \times 100\% \quad (7)$$

Information transmission in WSNs should ensure security, efficiency, and reliability, and connectivity can be evaluated based on the received signal strength calculated at the receiving node. In the process of radio propagation, assuming the signal transmission point is b_{ij} , the signal strength at this point is P_{TX} , and the receiving point is n_{ij} . The received signal strength at this point is $RSSI(i, j)$. The relationship between the transmitted signal strength and the received signal strength is expressed in (8).

$$RSSI(i, j) = P_{TX} - PL(b_{ij}, n_{ij}) \quad (8)$$

The connection is calculated by the $RSSI(i, j)$, which is greater than the minimum signal strength P_{tsh} required for communication or not. This is shown in (9).

$$con(i, j) = \begin{cases} 1 & RSSI(i, j) \geq P_{tsh} \\ 0 & otherwise \end{cases} \quad (9)$$

The connectivity fraction can be calculated using (10).

$$f_3 = \frac{\sum_i \sum_j con(i, j)}{100} \times 100\% \quad (10)$$

The deployment cost is relative to the number of sensor nodes; the coverage fraction satisfies the maximum coverage of the detection area and the connectivity. f_1 , f_2 , and f_3 are the objectives to be optimized. We want to obtain the highest coverage and connectivity for the lowest cost (with the fewest sensor nodes). These objectives are in conflict, which makes finding the optimal solution difficult, creating a multi-objective problem.

4.6.1. Compare the C-Metric between Algorithms and with the MOEA/PII Framework

We compare the coverage performance of the MOEA/DD algorithm and DDP2 algorithm, and NSGAIII algorithm and N3P2 algorithm using the C-metric indicator. The results are shown in Tables 10 and 11, with a generation iteration of 500. We can see in Table 10 that 23.2% of the solutions in the solution set obtained by MOEA/DD were dominated by those obtained by DDP2. Instead, only 1.5% of the solutions in the solution set obtained by DDP2 were dominated by MOEA/DD. This means that the solutions in the solution set obtained by DDP2 were better than those obtained by the MOEA/DD algorithm. In addition, in Table 11, we can see that 18.8% of the solutions in the solution set obtained by NSGAIII were dominated by those obtained by N3P2. Instead, only 10.6% of the solutions in the solution set obtained by N3P2 were dominated by NSGAIII. This indicates that the solutions in the solution set obtained by N3P2 were better than those obtained by the NSGAIII algorithm.

Table 10. The comparison of C-Metric between MOEA/DD and DDP2.

Algorithm	1 MOEA/DD	2 DDP2
1 MOEA/DD	—	C(1, 2) = 0.015 (0.002)
2 DDP2	C(2, 1) = 0.232 (0.004)	—

Table 11. The comparison of C-Metric between NSGAIII and N3P2.

Algorithm	1 NSGAIII	2 N3P2
1 NSGAIII	—	C(1, 2) = 0.106 (0.003)
2 N3P2	C(2, 1) = 0.188 (0.002)	—

4.6.2. Experiments on Determining the Objective Domain

The optimal WSN deployment solution uses the fewest sensor nodes to obtain the highest connectivity and coverage fraction. This is a multi-objective optimization problem. Multi-objective optimization evolutionary algorithms can solve these. However, the algorithms obtain a set of non-dominated solutions, and some of them are not suitable for customers. This makes it difficult for the customer to choose a suitable solution. If they prefer a high coverage fraction, they further filter the solutions.

We chose the N3P2 algorithm, set the population size to 60 and the generation iteration to 100, and ran the experiment 30 times independently to obtain the average value for comparison.

In the same deployment scenario, we set the cost and connectivity (f_1 and f_3) as the free dimensions, and the coverage fraction (f_2) as the projection plane. In one experiment, there was no determination on any dimensions. In the other experiment, the objective domain on the projection plane of the coverage fraction (f_2) was determined, and the objective domain was from 50% to 100%.

We can see in Table 12 that when no dimensions were determined, the algorithm ran for a long time but obtained less satisfactory solutions than what was required by the customer. Determining the objective domain of the objects of the wireless sensor networks deployment problem made the solution set run faster and made it more responsive to user needs.

Table 12. The comparison of the average time and number of solutions that meet user requirements by determining the objective domain or not with N3P2 algorithm.

Determined the Objective Domain	Average Time (s)	Average Number of Solutions that Meet User Requirements
No determined.	12.397	13.1
Coverage fraction scope [50–100%]	9.779	22.5

5. Conclusions

This paper proposes a MOEA/PII algorithm framework to solve many-objective optimization problems. MOEA/PII is an algorithm framework based on projection plane. Different from the other MOEAs for MaOPs, the MOEA/PII is a framework that can be combined with the original algorithms. Some objective dimensions are chosen to be the projection plane, which evolve via the projection grids. The left objective dimensions are the free dimensions that evolve via the originally selected algorithm. In addition, the non-dominated solutions are sorted and temporarily stored in the convergence elite and projection elite lists. The population of the next generation is composed of the solutions in ConEliteLists, and the solutions in ProEliteList are added when the population quantity is not sufficient.

In an attempt to alleviate the problem of selection pressure, the MOEA/PII carves the projection plane into projection grids, and the evolution process is carried out in each grid. In each projection grid, the number of individuals to be selected decreases, and accordingly, the selection pressure is alleviated.

The mechanism of Bi-Elite ensures that there are solutions in each grid. In the MOEA/PII framework, the distribution of the projection grid is uniform, which ensures the overall distribution of the solutions. The convergence of the solutions in each grid is determined by the free dimension algorithm.

In addition, we assessed the performance of the MOEA/PII framework from five aspects. First, the solutions set produced by MOEA/PII on 3-objective test instances was closer to Pareto fronts and was well-distributed on the PFs. For 15-objective test instances, the solutions were shown in the parallel coordinate. Secondly, we compared the IGD-metric and HV-metric values of the original algorithms and the original algorithms with the MOEA/PII framework running on benchmark test instances, i.e., DTLZ1-7 and WFG1-9, with 3, 5, 8, 10, and 15 objectives. The findings of this study support the idea that the MOEA/PII framework is efficient. Thirdly, the experiments demonstrated ways in which a projection grid could be chosen. The evidence from this study suggests that more segments lead to more projection grids, which improves the IGD-metric value. There was one exception: when we chose the fewest projection dimensions with more segments at the same number of grids. Fourthly, we investigated our framework's performance on the wireless sensor networks deployment problem. The solutions in the solution set obtained by the algorithms with MOEA/PII were better than those obtained by the original algorithms. Finally, when the user determined the objective domain, the solution set was more responsive to user needs, and the time was reduced.

The experiments were carried on two state-of-the-art MOEAs (MOEA/DD and NS-GAIII) on DTLZ1-7, WFG1-9 benchmark test instances, and a wireless sensor networks deployment problem, which demonstrated superior performance with MOEA/PII. In summary, these findings highlight a role for the MOEA/PII framework. Our study provides a novel framework for solving MaOPs.

Author Contributions: Conceptualization, F.P., L.L. and W.C.; methodology, F.P. and W.C.; software, F.P. and W.C.; validation, L.L. and J.W.; investigation, L.L. and J.W.; writing—original draft preparation, F.P.; writing—review and editing, F.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Regional Key Project of the Science and Technology Service Network Plan (STS Plan) of the Chinese Academy of Sciences (Grant No. KFJ-ST-S-QYZD-2021-19-003).

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy.

Acknowledgments: The authors sincerely appreciate the support from the Scientific research project of Liaoning Province Education Department (LJKMZ20220783) and (LJKMZ20220781); the national foreign expert project plan (G2022006008L); China University Innovation Fund (2021LD06009); the Liaoning Province’s “million talents project” (LRS [2019] No. 45); the Liaoning Province Nature Fund Project (No. 2022-MS-291); the Scientific research project of Liaoning Province Department of Education (LJ2020024, 2022); the Basic Research Project of Education Department of Liaoning Province (No. LJ2020022); the Opening Project of Beijing Key Laboratory of Sensors of Beijing Information Science and Technology University (2020CGKF005).

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

References

- Zhou, A.M.; Qu, B.Y.; Li, H.; Zhao, S.Z.; Suganthan, P.N.; Zhang, Q.F. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm Evol. Comput.* **2011**, *1*, 32–49. [\[CrossRef\]](#)
- Behmanesh, R.; Rahimi, I.; Gandomi, A.H. Evolutionary Many-Objective Algorithms for Combinatorial Optimization Problems: A Comparative Study. *Arch. Comput. Methods Eng.* **2021**, *28*, 673–688. [\[CrossRef\]](#)
- Pereira, J.L.J.; Oliver, G.A.; Francisco, M.B.; Cunha, S.S.; Gomes, G.F. A Review of Multi-objective Optimization: Methods and Algorithms in Mechanical Engineering Problems. *Arch. Comput. Methods Eng.* **2022**, *29*, 2285–2308. [\[CrossRef\]](#)
- Osamy, W.; Khedr, A.M.; Salim, A.; Al Ali, A.I.; El-Sawy, A.A. Coverage, Deployment and Localization Challenges in Wireless Sensor Networks Based on Artificial Intelligence Techniques: A Review. *IEEE Access* **2022**, *10*, 30232–30257. [\[CrossRef\]](#)
- Benatia, M.A.; Sahnoun, M.; Baudry, D.; Louis, A.; El-Hami, A.; Mazari, B. Multi-Objective WSN Deployment Using Genetic Algorithms Under Cost, Coverage, and Connectivity Constraints. *Wirel. Pers. Commun.* **2017**, *94*, 2739–2768. [\[CrossRef\]](#)
- Bi, W.W.; Chen, M.J.; Shen, S.W.; Huang, Z.Y.; Chen, J. A Many-Objective Analysis Framework for Large Real-World Water Distribution System Design Problems. *Water* **2022**, *14*, 13. [\[CrossRef\]](#)
- Liu, J.; Huang, J.; Hu, J.Z. Multi-objective optimisation method of electric vehicle charging station based on non-dominated sorting genetic algorithm. *Int. J. Glob. Energy Issue* **2022**, *44*, 413–426. [\[CrossRef\]](#)
- Barakat, N.; Sharma, D. Evolutionary multi-objective optimization for bulldozer and its blade in soil cutting. *Int. J. Manag. Sci. Eng. Manag.* **2019**, *14*, 102–112. [\[CrossRef\]](#)
- Colombo, L.; Todd, M.D.; Sbarufatti, C.; Giglio, M. On statistical Multi-Objective optimization of sensor networks and optimal detector derivation for structural health monitoring. *Mech. Syst. Signal Process.* **2022**, *167*, 20. [\[CrossRef\]](#)
- Otis, M.J.D.; Vandewynckel, J. A Many-Objective Simultaneous Feature Selection and Discretization for LCS-Based Gesture Recognition. *Appl. Sci.* **2021**, *11*, 9787. [\[CrossRef\]](#)
- Petchrompo, S.; Coit, D.W.; Brintrup, A.; Wannakrairot, A.; Parlikad, A.K. A review of Pareto pruning methods for multi-objective optimization. *Comput. Ind. Eng.* **2022**, *167*, 22. [\[CrossRef\]](#)
- He, Z.A.; Yen, G.G. Many-Objective Evolutionary Algorithm: Objective Space Reduction and Diversity Improvement. *IEEE Trans. Evol. Comput.* **2016**, *20*, 145–160. [\[CrossRef\]](#)
- Zhu, C.W.; Xu, L.H.; Goodman, E.D. Generalization of Pareto-Optimality for Many-Objective Evolutionary Optimization. *IEEE Trans. Evol. Comput.* **2016**, *20*, 299–315. [\[CrossRef\]](#)
- Yang, S.X.; Li, M.Q.; Liu, X.H.; Zheng, J.H. A Grid-Based Evolutionary Algorithm for Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2013**, *17*, 721–736. [\[CrossRef\]](#)
- Karimi, N.; Feylizadeh, M.R.; Govindan, K.; Bagherpour, M. Fuzzy multi-objective programming: A systematic literature review. *Expert Syst. Appl.* **2022**, *196*, 19. [\[CrossRef\]](#)
- He, Z.N.; Yen, G.G.; Zhang, J. Fuzzy-Based Pareto Optimality for Many-Objective Evolutionary Algorithms. *IEEE Trans. Evol. Comput.* **2014**, *18*, 269–285. [\[CrossRef\]](#)
- Menchaca-Mendez, A.; Montero, E.; Antonio, L.M.; Zapotecas-Martinez, S.; Coello, C.A.; Riff, M.C. A Co-Evolutionary Scheme for Multi-Objective Evolutionary Algorithms Based on epsilon-Dominance. *IEEE Access* **2019**, *7*, 18267–18283. [\[CrossRef\]](#)
- Wu, T.W.; An, S.G.; Han, J.Q.; Shentu, N.Y. An epsilon-domination based two-archive 2 algorithm for many-objective optimization. *J. Syst. Eng. Electron.* **2022**, *33*, 156–169. [\[CrossRef\]](#)
- Yuan, Y.; Xu, H.; Wang, B.; Yao, X. A New Dominance Relation-Based Evolutionary Algorithm for Many-Objective Optimization. *IEEE Trans. Evol. Comput.* **2016**, *20*, 16–37. [\[CrossRef\]](#)
- Zhang, Q.; Hui, L. MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Trans. Evol. Comput.* **2008**, *11*, 712–731. [\[CrossRef\]](#)
- Sun, Y.H.; Xiao, K.L.; Wang, S.Q.; Lv, Q.Y. An evolutionary many-objective algorithm based on decomposition and hierarchical clustering selection. *Appl. Intell.* **2022**, *52*, 8464–8509. [\[CrossRef\]](#)
- Zhao, H.T.; Zhang, C.S.; Zhang, B.; Duan, P.B.; Yang, Y. Decomposition-based sub-problem optimal solution updating direction-guided evolutionary many-objective algorithm. *Inf. Sci.* **2018**, *448*, 91–111. [\[CrossRef\]](#)
- Asafuddoula, M.; Ray, T.; Sarker, R. A Decomposition-Based Evolutionary Algorithm for Many Objective Optimization. *IEEE Trans. Evol. Comput.* **2015**, *19*, 445–460. [\[CrossRef\]](#)

24. Sun, Y.A.; Yen, G.G.; Yi, Z. IGD Indicator-Based Evolutionary Algorithm for Many-Objective Optimization Problems. *IEEE Trans. Evol. Comput.* **2019**, *23*, 173–187. [[CrossRef](#)]
25. Liu, C.; Zhao, Q.; Yan, B.; Gao, Y. A new hypervolume-based differential evolution algorithm for many-objective optimization. *Rairo-Oper. Res.* **2017**, *51*, 1301–1315. [[CrossRef](#)]
26. Shang, K.; Ishibuchi, H.; He, L.J.; Pang, L.M. A Survey on the Hypervolume Indicator in Evolutionary Multiobjective Optimization. *IEEE Trans. Evol. Comput.* **2021**, *25*, 1–20. [[CrossRef](#)]
27. Li, F.; Cheng, R.; Liu, J.C.; Jin, Y.C. A two-stage R2 indicator based evolutionary algorithm for many-objective optimization. *Appl. Soft. Comput.* **2018**, *67*, 245–260. [[CrossRef](#)]
28. Dhiman, G.; Soni, M.; Pandey, H.M.; Slowik, A.; Kaur, H. A novel hybrid hypervolume indicator and reference vector adaptation strategies based evolutionary algorithm for many-objective optimization. *Eng. Comput.* **2021**, *37*, 3017–3035. [[CrossRef](#)]
29. de Oliveira, M.C.; Delgado, M.R.; Britto, A. A hybrid greedy indicator- and Pareto-based many-objective evolutionary algorithm. *Appl. Intell.* **2021**, *51*, 4330–4352. [[CrossRef](#)]
30. Dhiman, G.; Kumar, V. KnRVEA: A hybrid evolutionary algorithm based on knee points and reference vector adaptation strategies for many-objective optimization. *Appl. Intell.* **2019**, *49*, 2434–2460. [[CrossRef](#)]
31. Xiang, Y.; Zhou, Y.R.; Tang, L.P.; Chen, Z.F. A Decomposition-Based Many-Objective Artificial Bee Colony Algorithm. *IEEE Trans. Cybern.* **2019**, *49*, 287–300. [[CrossRef](#)] [[PubMed](#)]
32. Zhao, H.T.; Zhang, C.S.; Zhang, B. A decomposition-based many-objective ant colony optimization algorithm with adaptive reference points. *Inf. Sci.* **2020**, *540*, 435–448. [[CrossRef](#)]
33. Qiu, W.B.; Zhu, J.H.; Wu, G.H.; Fan, M.F.; Suganthan, P.N. Evolutionary many-Objective algorithm based on fractional dominance relation and improved objective space decomposition strategy. *Swarm Evol. Comput.* **2021**, *60*, 16. [[CrossRef](#)]
34. Lu, X.; Yang, S.; Peng, F.; Chen, W. An Evolutionary Algorithm for Multi-objective Optimization Problem Based on Projection Plane: MOEA/P. In Proceedings of the 2021 The 5th International Conference on Algorithms, Computing and Systems, Xi'an, China, 24–26 September 2021.
35. Deb, K.; Thiele, L.; Laumanns, M.; Zitzler, E. Scalable multi-objective optimization test problems. In Proceedings of the Congress on Evolutionary Computation, Honolulu, HI, USA, 12–17 May 2002.
36. Huband, S.; Hingston, P.; Barone, L.; While, L. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Trans. Evol. Comput.* **2006**, *10*, 477–506. [[CrossRef](#)]
37. Zhang, Q.; Zhou, A.; Zhao, S.; Suganthan, P.N.; Tiwari, S. Multiobjective optimization Test Instances for the CEC 2009 Special Session and Competition. *Mech. Eng.* **2008**, 1–29. Available online: https://www.researchgate.net/publication/265432807_Multiobjective_optimization_Test_Instances_for_the_CEC_2009_Special_Session_and_Competition (accessed on 17 April 2023).
38. Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C.M.; Fonseca, V. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evol. Comput.* **2003**, *7*, 117–132. [[CrossRef](#)]
39. Zitzler, E.; Thiele, L. Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* **1999**, *3*, 257–271. [[CrossRef](#)]
40. Lv, L.; Peng, F.; Chen, W.; Wang, J. Node Deployment of Wireless Sensor Networks Based on MOEA/P Algorithm. In Proceedings of the 2021 13th International Conference on Communication Software and Networks (ICCSN), Chongqing, China, 4–7 June 2021.

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