An Improved Population Migration Algorithm for Solving Multi-Objective Optimization Problems

Qian ZHAO
College of Science, Inner Mongolia University of Technology, Hohhot, 010051, China

Xueying LIU
College of Science, Inner Mongolia University of Technology, Hohhot, 010051, China
E-mail: liuxy@imut.edu.cn

Shujun WEI
College of Computer Information Management, Inner Mongolia University of Finance and Economics, Hohhot, 010070, China

Received 30 November 2011

Accepted 19 June 2012

Abstract

The population migration algorithm is a very effective evolutionary algorithm for solving single-objective optimization problems, but very few applications are available for solving multi-objective optimization problems (MOPs). The current study proposes an improved population migration algorithm for solving MOPs based on the vector evaluated method and the dynamic weighted aggregation. The local search ability of the improved algorithm is greatly increased by using the population flow mode. The convergence of the improved algorithm is also proven. Performance metrics and experimental test results show that the improved algorithm is very feasible and effective for solving MOPs.

Keywords: population migration algorithm; multi-objective optimization; vector-evaluated method; dynamic weighted aggregation; population flow mode

1. Introduction

A multi-objective optimization problem (MOP) is a very common optimization problem, especially in science and engineering applications. In many real-world optimization problems, several conflicting objectives have to be optimized simultaneously. In business problems, for instance, a trade-off is usually made among the objectives, time, cost, and quality. Thus, the MOP is extremely significant.

In an MOP, a single optimal solution cannot be obtained when objectives conflict with one another because of the mutual constrain and conflict of each objective with the dominated variables[1]. Multi-objective optimization methods aim to find a set of equivalent solutions. These solutions are considered as equivalents when they are Pareto optimal[2], i.e., if no other solution could be found in the feasible region that performs better in at least one objective and equivalent or better in the rest. According to the criterion, an MOP has a set of Pareto optimal solutions[2]. The image of this set of solutions in the feasible objective space is called the Pareto front[2], which displays the objective trade-off characteristics for the problem.

The evolutionary algorithm (EA) seems to be especially suited for solving MOPs because of its abilities of simultaneously searching for multiple optimal solutions and performing better global searches in the search space. The genetic algorithm (GA) and the particle swarm optimization (PSO) are common EAs that can effectively solve MOPs. Schaffer proposed the first multi-objective GA, called the vector-evaluated genetic algorithm (VEGA)[3]. VEGA is easy to converge to the “beginning” and the “end” of the Pareto front, and

Published by Atlantis Press

it may not produce uniform distribution and abundant solutions in the Pareto front curve or surface. The defects of VEGA are also its advantages; some algorithms have difficulties in finding the "beginning" and the "end" of the Pareto front. After the development of VEGA, several multi-objective evolutionary algorithms (MOEAs) have been developed. The construction of a Pareto-rank, or domination-rank, which expresses the extent of domination of candidate solutions in a population, is a common practice in MOEAs. The selection in the solutions with identical Pareto-ranks is conducted based on a crowding measure. Moreover, the Pareto-rank promotes convergence to the Pareto front, and the crowding measure promotes diversity at the front. For example, the NPGA2[4], NSGA-||[5], SPEA2[6], and MOPMA[7] algorithms are some of the more recent representatives of MOEAs that are designed based on the principles above, but these are widely recognized as computationally expensive procedures.

The population migration algorithm (PMA) is a new global optimization EA proposed by Yonghua Zhou and Zongyuan Mao[8]. It has been successfully applied in some fields, but still meets difficulties in solving MOPs. The desired solution should not only converge to the Pareto front, but the diversity of the population should also be highly protected because of the diversity of the solutions of MOPs.

The current study proposes an improved population migration algorithm (IPMA) for solving MOPs based on the vector-evaluated method and dynamic weighted aggregation (DWA)[9,10]. First, the DWA is introduced to increase the diversity of the solution. The DWA is difficult to control at the "beginning" and "end" of the Pareto front, and thus, the vector-evaluated method is also introduced. The effectiveness and low computational cost of the new algorithm are shown in the following experiments.

2. Multi-objective optimization problem

An MOP is generally described as follows:

$$\min f(x) = (f_1(x), f_2(x), \cdots, f_n(x))$$
$$s.t.\begin{cases} g_j(x) \le 0, j = 1,2,\cdots,J \\ h_k(x) = 0, k = 1,2,\cdots H \end{cases}$$
$$x \in R^m. \tag{1}$$

where x is the decision vector that represents a solution, $f_i(1 \le i \le n)$ is the i^{th} objective function, and $g_j(1 \le j \le J)$ and $h_k(1 \le k \le H)$ are the inequality and equality constraint conditions, respectively.

The following basic concepts need to be introduced early on[11]:

Definition 1. *(Pareto dominance): a vector* $u = (u_1, u_2, \cdots, u_n)$ *is said to dominate* $v = (v_1, v_2, \cdots v_n)$ *if and only if* u *is partially less than* v*, i.e.,* $\forall i \in \{1,2,\cdots n\}, u_i \le v_i \wedge \exists j \in \{1,2,\cdots n\}, u_j < v_j$.

Definition 2. *(Pareto Optimal solution): in* $f: R^m \to R^n, x^*, x \in \Omega \subseteq R^m$ *, solution* x^* *is called the Pareto optimal solution if and only if no* $x \in \Omega$ *where* $f(x) = u = \{u_1, u_2, \cdots u_n\}$ *dominates (use Definition 1's scheme)* $f(x^*) = v = (v_1, v_2, \cdots, v_n\}$.

Definition 3. *(Pareto front): the set of all Pareto optimal decision vectors is called the Pareto optimal solution set of the problem, and the corresponding set of objective vectors is called the Pareto front.*

3. Improved population migration algorithm

3.1. *Population migration algorithm*

The PMA simulates population migration mechanisms. Migration mechanisms mainly include three aspects, namely, population flow, population migration, and population spread. On the one hand, people flow into their respective regions. On the other hand, a beneficial region attracts the population, and then the population migrates into the beneficial region. Finally, the population moves out of the beneficial region after accumulation and undergo spreading. Therefore, the algorithm has an ideal ability of global optimization. Zongben Xu provided improved algorithm steps of the PMA[12] that are more concise and convenient to describe.

3.2. *Approach of population flow*

The current study proposes a new way of population flow to enable the individual to excel in local search in the beneficial region. If $x^i(t)=(x_1^i(t),x_2^i(t),\cdots,x_m^i(t))$, $x_j^i(t)\in[a,b]$ and $y^i(t)=(y_1^i(t),y_2^i(t),\cdots,y_m^i(t))$ are the individual vectors before and after population flow, respectively. Therefore, its approach of population flow can be shown as follows:

$$y_j^i(t)=x_j^i(t)-\eta(t)+2\eta(t)r, i=1,2,\cdots,m., \qquad (2)$$

where $r=\sum_{i=1}^{N} rand()/N, rand()$ is a random number that distributes uniformly within (0,1). $\eta(t)$ is the radius of flow domains, and N is the population size. This way of population flow is not only helpful for local searching, but also ensures the convergence of the algorithm.

3.3. *Selecting the excellent individual*

The vector-evaluated method was inspired by the concept of VEGA[3], in which fractions of the next generations or subpopulations are selected from the previous generations according to each objective. The vector-evaluated method shows that if an MOP has n objective functions, then the problem will produce n subpopulations using VEGA. Moreover, each subpopulation size is N/n, where N is the population size. VEGA can find the "beginning" and "end" of the Pareto front. Therefore, the current study not only selects the Pareto optimal solution, but also takes part in providing solutions produced using the vector-evaluated method. This method improved the performance of the algorithm.

In $x(t)=(x^1(t),x^2(t),\cdots,x^N(t))$ as the individual vector before population flow, $y(t)$ is the individual vector after l times population flow, which includes Nl individuals, with l as the population flow scale and N as the population size.

The method of selecting excellent individuals is described as follows:

In $x(t)\cup y(t)$, N/n excellent individuals are selected and then put into $p_1(t)$ for each objective function. Simultaneously, the Pareto optimal solution is selected to put into $p_2(t)$.

Therefore, after the population flow, obtaining the excellent individual set is given by $pop(t)=p_1(t)\cup p_2(t)$.

3.4. *Dynamic weighted aggregation*

The conventional weighted aggregation (CWA)[9] is a simple weighted sum of the different objective functions converted into a single function. Even though CWA is the simplest approach for evolutionary multi-objective optimization, it has been severely criticized because it can only provide one Pareto solution in one optimization run. Thus, the current study introduces the DWA[9,10] to eliminate the said problem.

In the DWA, the constant weights are changed to time varying weights, $w_1(t)$ and $w_2(t)$, where t is the "time" measured in generation. $w_1(t)$ and $w_2(t)$ can be expressed as follows:

$$w_1(t)=|\sin(2\pi t/T)|, w_2(t)=1-w_1(t),$$

where T is the period, which is a user-defined parameter that controls how rapidly the weights cycle from 0 to 1 and back again.

Therefore, in using the DWA, the objective functions are summed into a weighted combination as follows:

$$F(x)=w_1(t)*f_1(x)+w_2(t)*f_2(x). \qquad (3)$$

3.5. *An improved population migration algorithm based on the vector-evaluated and dynamic weighted aggregation approaches*

The current study presents an IPMA based on the vector-evaluated and dynamic weighted aggregation techniques for solving MOPs. The steps of the IPMA are proposed based on the steps of PMA given by Zongben Xu[12].

The steps of the IPMA are as follows:

Step 1. In the initial step, the following should be keyed in: population size N, radius of initialized area $\delta(0)$, population pressure vigilance parameters $\alpha(0)$, and the size of floating population l. N individuals are then randomly generated, and $x(0)=(x^1(0),x^2(0),\cdots,x^N(0))$ should be noted. The living area of the initial population should be formed as follows:

$\Omega(0)=\bigcup_{i=1}^{N}B(x^i(t),\eta(t))$, where $B(x,r)$ indicates a sphere with x as the center and r as the radius, $i=1,2,\cdots,N$, and $t=0$.

Step 2. The following evolutionary steps are undertaken:

(2.1) As preparation, $\eta(t)=\delta(0)$ should be made.

(2.2) Population flow is calculated as follows:

(2.2.1) In every $B(x^i(t),\eta(t))$, l individuals must be generated using Equation (2), and then the generated Nl individuals should be used as component population $y(t)$ in $\Omega(t)$.

(2.2.2) In $x(t)\bigcup y(t)$, N/n excellent individuals should be selected and put into $p_1(t)$ for each objective function; simultaneously, the Pareto optimal solutions should be selected for application into $p_2(t)$ and $pop(t)=p_1(t)\bigcup p_2(t)$.

(2.3) For population migration, the following are considered:

(2.3.1) According to Equation (3), using the DWA construct function $F(x)$ for selecting N best individuals makes up the middle population as follows: $y_{best}(t)=(y_{best}^1(t),y_{best}^2(t),\cdots,y_{best}^N(t))$.

(2.3.2) The formed N beneficial region can be expressed as follows:

$\{B(y_{best}^i(t),\eta(t)),i=1,2,\cdots,N\}$.

(2.3.3) After proceeding with population flow in every beneficial region, namely, in every $B(y_{best}^i(t),\eta(t))$, l individuals should be obtained using Equation (2), and Nl individuals are generated for component population $z(t)$.

(2.3.4) In $z(t)\bigcup y_{best}(t)$, N/n excellent individuals are selected; these are put into and replace $p_1(t)$ for each objective function. Simultaneously, the Pareto optimal solutions are selected for application; these replace $p_2(t)$ and $pop(t)=p_1(t)\bigcup p_2(t)$. Then, individuals are obtained from $pop(t)$ and are deposited into $p(t)$.

(2.3.5) Shrinking the beneficial region involves the following:

$\eta(t)=(1-\Delta)\eta(t)$ should be set among $0<\Delta<1$.

(2.3.6) If $\eta(t)>\alpha(t)$, then step (2.3) may be followed; otherwise, (2.3.7) should be taken.

(2.3.7) The dominated solution in $p(t)$ may be removed, and then $p(t)$ is archived in q_{best}.

(2.3.8) $p_{best}(t)=q_{best}$ is set.

(2.4) In population spread, a new population should be defined, and then the system randomly produces N individuals. $x(t+1)$ should be noted.

$x(t+1)=(x^1(t+1),x^2(t+1),\cdots,x^N(t+1))$.

Step 3. In the termination step, if the iteration time t is shorter than the pre-assigned time, $t=t+1$ is set. Afterward, step 2 may be retaken; otherwise, the algorithm ends. $p_{best}(t)$ is the final Pareto optimal solution set for this algorithm.

4. Solving constrained multi-objective optimization problems

An MOP with constraint conditions is different from a non-constrained optimization problem. The feasible region of an MOP is less than the solution space; distinguishing feasible and infeasible solutions in the search space is necessary. Thus, according to the leading constraint principle presented by Deb[13], the current study introduces an infeasible degree (IFD)[14,15] and a threshold[14,15] of IFD to solve the problem above and to guide the search direction into the feasible optimum solution.

The IFD Φ of a candidate solution x is defined as follows:

$$\Phi(x)=\sum_{j=1}^{J}[\min\{0,g_j(x)\}]^2+\sum_{k=J+1}^{J+H}[h_k(x)]^2. \quad (4)$$

where $g_j(x),h_k(x)$ are inequality constraints (larger than or equal to the format) and equality constraints of the constrained optimization problem, respectively. J and H are the numbers of the inequality and equality constraints, respectively.

The IFD of the solution can be regarded as the distance between the solution and the feasible region. The IFD of the feasible solution is zero, but the IFD of the infeasible solution is greater than zero, and its violation becomes more severe as its IFD becomes larger.

According to the rules, the pressure of rejection on infeasible solutions should be increased as the iteration. The increase can be implemented using a threshold value Φ_{crit}, which is designed based on two parts, namely, the coefficient decrease with the iteration and the average IFD of the population. Φ_{crit} can be expressed as follows:

$$\Phi_{crit} = \frac{1}{T}(\sum_{i=1}^{N}\Phi(x_i))/N\,, \tag{5}$$

where temperature T is a coefficient that increases from T_{start} to T_{end} along the iterations of the algorithm to control the acceptable bound of the infeasible solutions. N is the size of the population.

Φ_{crit} is a threshold value that determines whether an infeasible solution is accepted or rejected. An infeasible solution is accepted if its IFD is less than or equal to Φ_{crit}, otherwise, it is rejected. The scales of the feasible solutions are sometimes smaller in the process of searching the population. Thus, some accepted infeasible solutions in the current population could be chosen to keep the size of the population fixed, which, in turn, could guide the population to converge to the feasible region gradually.

Therefore, as long as the IFD and the threshold of the IFD are used to choose excellent individuals in steps (2.2.2) and (2.3.4) of the IPMA, the IPMA can be used to solve constrained MOPs.

5. Convergence analysis of improved population migration algorithm

Theorem 1[16]**.** *Let a Genetic Algorithm fulfill the following conditions:*

(1) $\forall x', x \in I, x'$ *is reachable from* x *by means of mutation and recombination, and*

(2) the population sequence $P(0), P(1), \cdots P(t)$ *is monotone,* $i.e. \forall t$ *:*

$$\min\{\Phi(x(t+1)) \mid x(t+1) \in P(t+1)\}$$
$$\leq \min\{\Phi(x(t)) \mid x(t) \in P(t)\}$$

Then, $p\{\lim_{t\to\infty} x \in P(t)\} = 1$.

In setting F as the search space, which is searched by the IPMA, $p(t)$ is the t iterative population, $p_{best}(t)$ is the archiving set, and F^* is the Pareto optimal solution set in F.

Theorem 2. *In the case of unlimited iteration numbers, the IPMA can converge to the global Pareto optimal solution set at a probability of 1.*

Proof. According to theorem 1, the IPMA must be proven to fulfill the follow conditions:

(1) The IPMA can reference to an arbitrary individual in the search space F.

(2) In the population sequence $p_{best}(0), p_{best}(1), \cdots$, $p_{best}(t)$ is monotone.

First, the population flow of the population migration algorithm indicates that the algorithm can generate N individuals randomly and independently in the search space. Then, the divided areas are formed with the individual as the center at the same size. Random searches are conducted in each area. However, the IPMA proposes improved ways using population flow, but its idea of population flow remains the same. Given $\forall x', x \in F$, if $X(\varepsilon)$ is the divided area with x as the center, then $P(\{\lim_{t\to\infty} x' \in X(\varepsilon)\}) = 1$, which has been proven in literature 17. Therefore, along with infinite increase in iterations, the IPMA can refer to arbitrary individuals in the search space, and thus condition (i) can be satisfied.

Second, if $p_{best}(t+1) = p_{best}(t) \cup p(t+1)$, $p_{best}(t)$ is non-decreasing. Once solutions $x^* \in F^*$ are searched on $p(t)$, this solution will be kept in $p_{best}(t)$. However, if a solution $x^* \in F^*$ is a non-dominated solution, then it will always be kept in $p_{best}(t)$. Thus, the individual in F^* will be searched continually and will be kept in $p_{best}(t)$ to ensure that the $t+1$ iterative best fitness is equal to the t iterative best fitness at the least. It can obtain better fitness in $p_{best}(t+1) = p_{best}(t) \cup p(t+1)$ because it will produce the better non-dominated solution in $p(t+1)$. At the least, a result worse than the t iterative will not appear. Thus, the IPMA meets the monotonicity requirement, and condition (ii) is satisfied.

Therefore, the IPMA can converge to the global Pareto optimal solution set at a probability of 1.

6. Experiment results and performance metrics

6.1. *Performance metrics of the algorithm*

The Pareto optimal solution set can be obtained after executing the IPMA for MOPs. However, the performance of the IPMA is usually difficult to analyze and evaluate. Thus, the current study looks at this problem from three sides[18], namely, the convergence, the distribution, and the error ratios. Thus, three quantitative standards are used. The generational distance (GD) indicator evaluates the convergence of the algorithm, the spacing (SP) indicator evaluates the dispersion of the algorithm, and the error ratio (ER) evaluates the error rates of the algorithm.

PF_{known} is the Pareto front obtained using the IPMA. PF_{true} is the true Pareto front.

Definition 4. *GD. This metric may be effective in gauging the performance of the algorithm. GD is a value representing how "far"* PF_{known} *is from* PF_{true}. *GD is defined as follows:*

$$GD = \frac{\sqrt{\sum_{i=1}^{N} d_i^2}}{N}, \tag{6}$$

where N *is the number of vectors in* PF_{known}, *and* d_i *is the distance (in objective space) between each of these and the nearest number of* PF_{true}. *If GD=0, the non-dominated solution of using this algorithm belongs to the true Pareto optimal solution set.*

Definition 5. *SP. This metric measures the distribution of vectors throughout* PF_{known}. *SP is defined as follows:*

$$SP = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(\bar{d} - d_i)^2}, \tag{7}$$

where N *is the quantity of vectors at the Pareto front resolved using the algorithm,* $d_i = \min_j(\left| f_k^i(x) - f_k^j(x) \right|), i, j = 1,2,\cdots,N, j \neq i \cdot \bar{d}$ *is the average value of resolved* d_i, *as shown in* $\bar{d} = \frac{1}{N}\sum_{i=1}^{N} d_i$. *SP=0 indicates that the algorithm has an ideal distribution capacity.*

Definition 6. *ER. The algorithm reports a finite number of solutions. These solutions are or are not members of* PF_{true}. *They define this metric as follows:*

$$ER = \frac{\sum_{i=1}^{N} e_i}{N}, \tag{8}$$

where N *is the number of vectors in* PF_{known}. *If vector* i *is a member of* PF_{true}, *then ER=0, otherwise, ER=1. ER=0 indicates that every vector reported in* PF_{known} *using the proposed algorithm is actually in* PF_{true}.

6.2. *Test functions and test results*

Eight test functions were used and calculated using the MATLAB program to test the performance of the IPMA. Test function 1 is a classical function in MOP proposed by Schaffer. Test functions 2 and 3 are used in reference 2. Test functions 4, 5, and 6 are ZDT1, ZDT2, and ZDT3, respectively, which were proposed by Deb[18]. Test functions 7 and 8 are constrained multi-objective test functions. Test functions and experiment results are shown below:

- Test function 1 (MOP1):

$$\min f_1(x) = x^2,$$
$$\min f_2(x) = (x-2)^2,$$
$$x \in [-5,10].$$

The Pareto front of MOP1 is convex. Figure 1 shows the curve of the Pareto front obtained using the IPMA.

- Test function 2 (MOP2):

$$\min f_1(x) = \begin{cases} -x, x \le 1, \\ -2 + x, 1 < x \le 3, \\ 4 - x, 3 < x \le 4, \\ -4 + x, x > 4, \end{cases}$$
$$\min f_2(x) = (x-5)^2,$$
$$x \in [-5,10].$$

The Pareto front of MOP2 is discontinuous. Figure 2 shows the curve of the Pareto front obtained using the IPMA.

- Test function 3 (MOP3):

$$\min f_1(x,y) = (x^2 + y^2)^{1/8},$$
$$\min f_2(x,y) = ((x-0.5)^2 + (y-0.5)^2))^{1/4},$$
$$x, y \in [-5,10].$$

The Pareto front of MOP3 is concave. Figure 3 shows the curve of the Pareto front obtained using the IPMA.

- Test function 4 (MOP4):

$$\min f_1(x) = x_1,$$
$$\min f_2(x) = g(x) * (1 - \sqrt{\frac{f_1(x)}{g(x)}}),$$
$$g = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i, x_i \in [0,1], i = 1,2,\cdots\cdots n, n = 30.$$

MOP4 is a high-dimensional test function, and its Pareto front is both convex and continuous. Figure 4 shows the curve of the Pareto front obtained using the IPMA.

- Test function 5 (MOP5):

$$\min f_1(x) = x_1,$$

$$\min f_2(x) = g(x) * (1 - (\frac{f_1(x)}{g(x)})^2),$$

$$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i, x_i \in [0,1], i = 1,2,\cdots\cdots n, n = 30.$$

MOP5 is a high-dimensional test function, and its Pareto front is concave. Figure 5 shows the curve of the Pareto front obtained using the IPMA.

- Test function 6 (MOP6):

$$\min f_1(x) = x_1,$$

$$\min f_2 = g(x) * (1 - \sqrt{\frac{f_1(x)}{g(x)}} - \frac{f_1(x)}{g(x)}\sin(10\pi f_1(x))),$$

$$g(x) = 1 + \frac{9}{n-1}\sum_{i=2}^{n} x_i, x_i \in [0,1], i = 1,2,\cdots\cdots n, n = 10.$$

MOP6 is a high-dimensional test function, and its Pareto front is discontinuous. Figure 6 shows the curve of the Pareto front obtained using the IPMA.

- Test function 7 (MOP7):

$$\min f_1(x) = x^2,$$

$$\min f_2(x) = (x-2)^2,$$

Subject to $x^2 - 2.5x + 1.5 \geq 0,$

$$-10 \leq x \leq 10.$$

The objective functions MOP7 are similar to those of MOP1. The Pareto front of MOP7 is discontinuous because of the increase in constrained conditions. Figure 7 shows the curve of the Pareto front obtained using the IPMA.

- Test function 8 (MOP8):

$$\min f_1(x) = 4x_1^2 + 4x_2^2,$$

$$\min f_2(x) = (x_1 - 5)^2 + (x_2 - 5)^2,$$

Subject to

$$(x_1 - 5)^2 + x_2^2 \leq 25,$$

$$(x_1 - 8)^2 + (x_2 + 3)^2 \geq 7.7,$$

$$0 \leq x_1 \leq 5, 0 \leq x_2 \leq 3.$$

The Pareto front of MOP8 is convex. Figure 8 shows the curve of the Pareto front obtained using the IPMA.

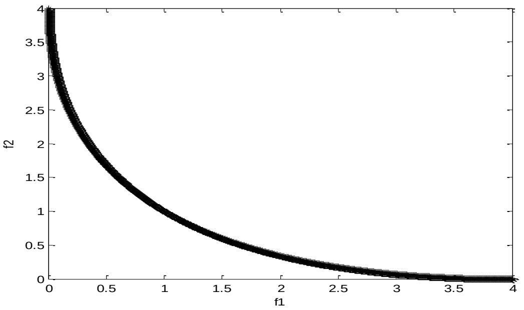

Fig.1 Pareto front of MOP1

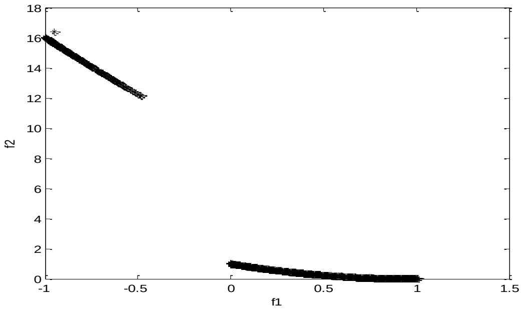

Fig.2 Pareto front of MOP2

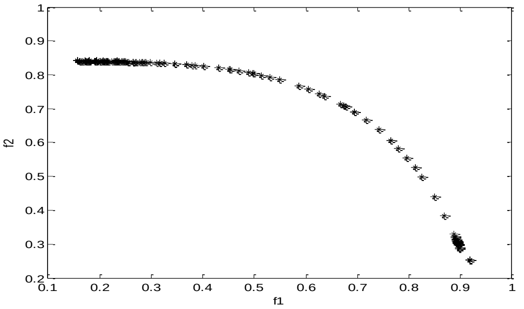

Fig.3 Pareto front of MOP3

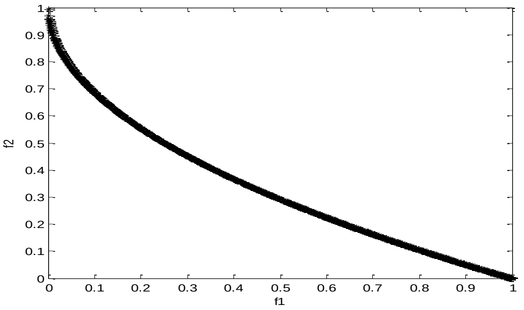

Fig.4. Pareto front of MOP4

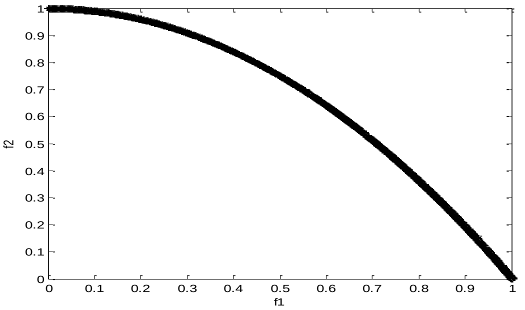

Fig.5. Pareto front of MOP5

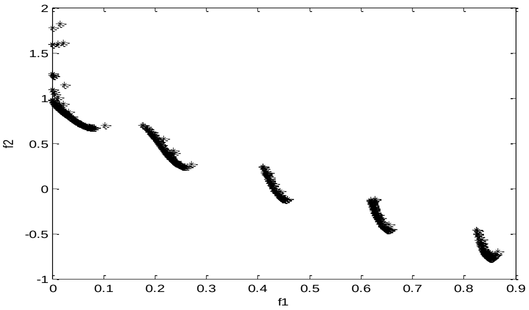

Fig.6. Pareto front of MOP6

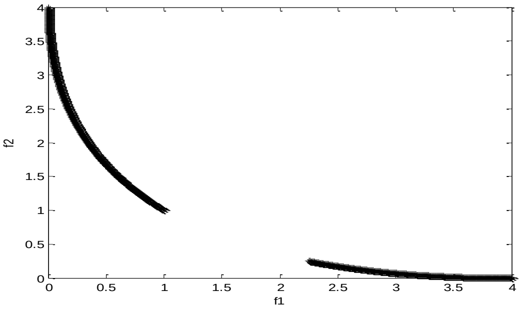

Fig.7. Pareto front of MOP7

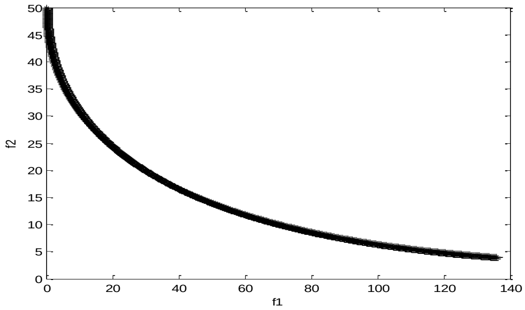

Fig.8. Pareto front of MOP8

In addition, the current study analyzes the performance of the Pareto optimal solution set of test functions produced using the IPMA in the three aspects, namely, GD, SP, and ER. The results are shown in Table 1.

Table 1. Performance metrics

	GD	SP	ER
MOP1	2.5019E-017	0.0044	0
MOP2	6.6055E-004	0.0094	0.0019
MOP3	3.9402E-005	0.0073	0.0100
MOP4	0.0014	6.3129E-004	0.0059
MOP5	0.0011	9.7212E-004	0.0088
MOP6	0.0033	0.0037	0.0321
MOP7	4.6030E-006	0.0068	3.8100E-004
MOP8	0.0198	0.0179	0.0548

As seen in the experiment results in Figs. 1-8, the Pareto front curve can be accurately obtained for the eight test functions using the IPMA. Compared with reference 2, the IPMA can solve the high dimension and discontinuous unconstrained optimization problem. Moreover, the results of the constrained MOP are better than those of reference 15. As seen in the experimental data obtained using the performance metrics of the IPMA, the values of GD and SP are lesser and are also better than the performance metrics of the NPGA2[4], NSGA||[5], SPEA2[6], and MOPMA[7] algorithms. These results show that the convergence of the IPMA and the dispersion of the Pareto optimal solution set are very strong. The curve of the Pareto front is properly distributed, and thus, the IPMA is proven feasible and effective in solving MOPs.

7. Conclusion

The current study proposed an IPMA for solving MOPs. Compared with the standard PMA, the IPMA not only adopts the vector-evaluated method, but also introduces the DWA. Moreover, the IPMA is an improved approach in population flow. The convergence of the IPMA was proven. Based on the visual graphics and performance metrics of the Pareto optimal solution set in the experimental results, the IPMA was shown to be feasible and effective for solving MOPs.

Acknowledgement

This work was supported by the Scientific Research Project of the Higher Education Institutions of Inner Mongolia Autonomous Region No. NJZY12070.

References

1. C.A. Coello Coello, D.A. Van Veldhuizen and G.B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems* (Kluwer, New York, 2002).
2. L. B. Zhang and C. G. Zhou (eds.), Solutions of Multi-Objective Optimization Based on Particle Swarm Optimization, *J. Computer Research and Development.* 41(7) (2004) 1286-1291.
3. J. D. Schaffer, Multiple objective optimization with vector evaluated genetic algorithm, in *Proc. 1st Int. Conf. Genetic Algorithms and Their Application,* eds. J. J. Grefenstette (Hillsdale, 1985), pp. 93-100.
4. M. Erickson and A. Mayer (eds), The Niched Pareto genetic algorithm 2 applied to the design of groundwater remediation system, in Proc. 1st Int. Conf. Evolutionary Multi-Criterion Optimization, EMO2001, eds. E. Zitzler and K. Deb (Springer-Verlag, Berlin, 2001), pp. 682-695.
5. K. Deb and A. Pratap (eds), A fast and elitist multi-objective genetic algorithm:NASA-II, in *IEEE Transactions on Evolutionary Computation.* 6(2) (2002) 182-197.
6. E. Zitzler and M. Laumanns, SPEA2: Improving the strength Pareto evolutionary algorithm, in *Evolutionary Methods for Design, Optimization and Control with Application to Industrial Problems,* eds. K. Giannakoglou and D. T. Tsahlis (Springer-Verlag, Berlin, 2002). pp. 95-100.
7. Q. F. Luo and Y. Y. Zhou, A Fast-stable Optimization Algorithm for Multi-objective Population Migration, in *2010 International Conference of Information Science and Management Engineering* (xi'an, China, 2010). pp. 237-241.
8. Y. H. Zhou and Z. Y. Mao, A New Search Algorithm for Global Optimization: Population Migration Algorithm (I), *J. South China University of Technology (Natural Science Edition)* 31(3) (2003) 1-5.
9. Y. Jin and T. Okabe (eds), Dynamic Weighted Aggregation for Evolutionary Multi-Objective Optimization: Why Does It Work and How?, in *Evolutionary Computation Conference*, eds. L. Spector (Morgan Kaufmann, San Francisco, 2001). pp. 1042-1049.
10. K. E. Parsopoulous and M. N. Vrahatis, Particle Swarm Optimization Method in Multi-objective Problems, in *Proc of the ACM Symp on Applied Computing 2002 (SAC 2002)* (ACM Press, New York, 2002). pp. 603-607.
11. J. Huang and Y.B. Liu, MOEAQ: AQoS-Aware Multicast Routing Algorithm for MANET, *Expert Systems with Applications* 37(2)(2010) 1391-1399.
12. Z. B. Xu, *Computation Intelligence (Volume I)* (Beijing High Education Press, Beijing, 2004).
13. K. Deb and A. Pratap, Constrained Test Problems for Multi-objective Evolutionary Optimization, in *KanGAL report.* (Indian Institute Technology, Kanpur, 2002).
14. K. Deb, An efficient constraint handling method for genetic algorithm, *J. Computer Methods in Applied Mechanics and Engineering* 186 (2000) 311-338.
15. Y. X. Wang and L. C. Liu (eds), Constrained multi-objective optimization evolutionary algorithm, *J. Tsinghua Univ (Sci&Tech)* 45(1) (2005) 103-106.
16. T. Bäck, *Evolutionary algorithm in the theory and practice* (Oxford university press, New York, 1996).
17. Y. H. Zhou and Z. Y. Mao, A New Search Algorithm for Global Optimization: Population Migration Algorithm (II), *J. South China University of Technology (Natural Science Edition)* 31(4) (2003) 41-43.
18. D. A. Veldhuizen and G.B. Lamont, Multi-objective evolutionary algorithm test suits, in *Proceedings of the 1999 ACM Symposium on Applied Computing Conference.* (Florida, 1999), pp. 113-119.