Algorithm Based on the Grey Wolves Attack Technique Method for Generating Pareto Optimal Front

Wendinda Bamogo, Alexandre Som, Kounhinir Somé*

Abstract—**This paper proposes an algorithm for solving multiobjective optimization problems using the attack technique of the Grey Wolf. It is a metaheuristic method called a Multiobjective Optimizer based on Grey Wolf Attack Technique (MOGWAT). In fact, it is inspired by the modified Hybrid Grey Wolf Optimizer and Genetic Algorithm (HmGWOGA), which is a single objective optimization algorithm specially designed for positive objective functions. The MOGWAT method combines the multiple objective functions of the initial problem into a single objective function, and then penalizes constraint functions to get an unconstrained single-objective optimization. The use of an effective single-objective optimizer allows reaching the optimal solutions. These solutions are also the Pareto optimal solutions of the initial problem according to some parameters. Through some theorems, we have established the theoretical foundation and performance of our method. Furthermore, in order to highlight the numerical performance of the method, we have tackled three groups of problems: 16 test problems from the Zitzler-Deb-Thiele benchmarks, 2 instances from the CEC 2009 benchmarks, and 2 real-world problems from literature. Our numerical results have been compared to the ones obtained with the NSGA-II method. This comparison was made using some computed performance parameters. The outcomes of the comparison have enabled us to prove the effectiveness and efficiency of our new approach in terms of speed and convergence.**

Index Terms—**Multiobjective optimization; Metaheuristic methods; Pareto Optimality; Grey Wolf optimizer**

I. Introduction

THE multiobjective optimization concept is extensively employed for the modeling and resolution of real-life problems. To solve a real-life problem using mathematical tools, two important steps must be taken: the mathematical formulation and the finding of an adapted method. Therefore, it is essential to master the resolution of multiobjective optimization problems. It should be noted that, there are no universal methods for these kinds of problems in literature. In these mathematical programs, several conflicting objectives are considered simultaneously, and this situation imposes that there is no optimal solution. The resolution of these problems leads to a set of solutions called Pareto optimal solutions [32]. In practice, the existing methods are predominanly methods for approximating solutions [15], [16], [25], and are evaluated in terms of performance on computational complexity, convergence, and distribution. Nowadays, it is almost impossible to find a method in the literature that can solve all multiobjective optimization problems efficiently. This is the reason why many researchers are always working on this topic.

The majority of existing methods for the solving of multiobjective optimization attempt to convert the initial problem into a single objective optimization problem through an aggregation function. The literature contains many aggregation functions [29], but in this work, we have chosen the ϵ-constraint approach. It is one of the best transformations that preserves the Pareto optimality of solutions. It consists of selecting only one objective function to optimize and converting the others into constraints. After that, the Lagrangian penalty function has been used to obtain an unconstrained single objective function. A good optimizer method is required to achieve optimal solutions at this time.

Many works in the literature propose methods for solving single objective optimization problems. We are especially interested in works that focus on the Grey Wolf Optimizer (GWO). In 2019, Sawadogo et al. [17] developed a modified Hybrid Grey Wolf optimizer and genetic algorithm (HmGWOGA) for global optimization of positive functions; Fu et al. [20] focused on dynamically dimension Search Grey Wolf Optimizer Based on Positional Interaction Information; Wen et al. [18] aimed to develop on an efficient and robust Grey Wolf Optimizer algorithm for large-scale numerical optimization; Muhammed et al. [19] produced some results on Grey Wolf Optimizer-Based Tuning of a hybrid LQR-PID controller for foot trajectory control of a quadruped robot. In 2020, Shubham et al. [14] presented their works on an enhanced leadership-inspired Grey Wolf Optimizer for the global optimization problem; In 2021, Farshad et al. [10] have focused on an enhanced Grey Wolf Optimizer with a velocity-aided global aided global search mechanism; Wei et al. [9] are about path planning of UAV based on Improved adaptive Grey Wolf Optimization algorithm; Amir et al. [13] provided improving algorithms of the Grey Wolf Optimizer to solve global optimization problem; In 2022, Xinyang et al. [6] are focused on dimensional learning strategy-based Grey Wold optimizer for solving the global optimization problem; Safora et al. [5] on a condition-based Grey Wolf Optimizer algorithm for the global optimization problems; Eslan et al. [7] are

Manuscript received May 2, 2023; revised January 4, 2024.

W. Bamogo is a PHD student of the Laboratoire de Mathematiques, Informatique et Applications of Université Norbert ZONGO, Burkina Faso (email: menabamogo@gmail.com).

A. Som is PHD graduate of the Laboratoire d'Analyse Numerique Informatique et de Biomathématique of Université Joseph KI-ZERBO, Burkina Faso (email: irenotesom89@gmail.com).

* K. Somé is a professor of the Unité de Formation et de Recherche en Sciences et Technologies of Université Norbert ZONGO, BP 376 Koudougou, Burkina Faso (corresponding author to provide phone: 00226 76547203; email: sokous11@hgmail.com).

focused on hybrid Grey Wolf Optimization-based Gaussian process regression model for simulating deterioration behavior of highway tunnel components; Zeynab et al. [4] are focused on a new enhanced hybrid Grey Wolf Optimizer combined with elephant herding optimization algorithm for engineering optimization.

In all previous works, GWO was used to solve single objective optimization problems. According to Sawadogo et al. [17], the HmGWOGO method was built for positive functions without constraints, and some excellent optimal solutions were found. The challenge was to extend it in order to solve optimization problems with several objectives.

This work proposes a metaheuristic method based on Grey Wolf attack technique for catching prey to solve a multiobjective optimization problem. We call it Multiobjective Optimizer based on Grey Wolf Attack Technique (MOGWAT). MOGWAT is an extension of the HmGWOGA algorithm, which was originally designed to solve single-objective optimization problems. It arises from a combination of the ϵ-constraint approach and the HmGWOGA algorithm. In order to demonstrate the optimality of obtained solutions and the good complexity of our algorithm, we have proposed three theorems and some numerical results. We have computed the Pareto optimal solutions of twenty test problems taken from the literature [16], [31], [32], [34], [35]. This allows us to determine the computational time of our method. Moreover, we have computed some performance parameters about convergence and distribution for the obtained solutions. Based on these results, we have conducted a comparative study with the NSGA-II method. According to this comparison, our proposed method can be presented as the best choice for solving multiobjective optimization problems when decision makers need a fast and efficient convergence method.

This paper is organized around sections. After Section I denoted to the introdactory paragraph, Section II, will describe materials and methods. Section III, will provide details on the MOGWAT method. Section IV-C, will present results and discussion of this work. The conclusion of the study will be drown in Section V.

II. Materials and methods

A. Multiobjective optimization concepts

Let us consider the multiobjective optimization problem in the following formulation:

$$\begin{cases} \min\big(f_1(x), f_2(x), \cdots, f_p(x)\big); p \geq 2 \\ g_j(x) \leq 0,\ j=\overline{1,m}; \\ x \in \mathbb{R}^n; \end{cases} \tag{P}$$

where $f = \big(f_1, f_2, \cdots, f_p\big)$ is the vector which components are objective functions that are subjected to the constraint function $g = \big(g_1, g_2, \cdots, g_m\big)$. Let us set that $\chi = \{x \in \mathbb{R}^n : g(x) \leq 0\}$ and $\mathcal{Y} = f(\chi)$ respectively the decision space and objective space of the problem (P).

Definition 1. *An objective vector $f(x^*) \in \mathcal{Y}$ is said non dominated point if there is not another objective vector $f(x)$ such as $f_i(x) \leq f_i(x^*)$ for all $i = \overline{1,p}$ and $f_k(x) < f_k(x^*)$ for at least one index k.*

In this case, if $x^* \in \chi$, it is called Pareto optimal solution of the problem (P) *and the set of non dominated points of problem* (P) *is called Pareto front.*

Definition 2. *The Pareto front is a direct representation of Pareto optimal solutions by objective functions.*

Throughout the rest of this paper, we are going to denote by χ_E the Pareto optimal solutions set of the problem (P) and $\mathcal{Y}_E$ the set of the non dominated points.

For the solutions of the problem (P), there are many methods using an aggregation function to transform the problem into a single-objective optimization problem. Here, we have used the ϵ-constraint approach. It gives way rewording the initial problem (P) as follows:

$$\begin{cases} \min f_k(x); \\ f_i(x) \leq \epsilon_i,\ i \neq k,\ i=\overline{1,p}; \\ g_j(x) \leq 0,\ j=\overline{1,m} \\ x \in \mathbb{R}^n; \end{cases} \tag{P$_\epsilon$}$$

where $\epsilon \in \mathbb{R}^{p-1}$. Let us set $\Omega = \{x \in \mathbb{R} : f_i(x) \leq \epsilon_i, i = \overline{1,p}, i \neq k$ and $g_j(x) \leq 0, j = \overline{1,m}\}$.

Let us consider the problem (P) with $p = 1$. That implies that we must minimize a single-objective function. In the literature review, most methods for solving this kind of problems proceed by transforming value into an unconstrained optimization problem by using a penalty function [1], [11], [12], [22]. One of the most commonly used is the Lagrangian penalty function which combines the objective function and all constraint functions as follows:

$$L_f(x,\eta) = f(x) + \sum_{j=1}^{m} \eta_j g_j(x)$$

where $\eta_j \geq 0,\ j = \{1, 2, \cdots, m\}$ are Lagrangian penalty parameters.

Various methods have been proposed today for obtaining the optimal solutions to a single-objective function like L_f. In the case of non-linearity, where the exact solution is hard to find, we have to use the one that gives a good approximation.

B. HmGWOGA method

In recent works, a modified hybrid Grey Wolf Optimizer and genetic algorithm (HmGWOGA) [17] was proposed for the optimal solutions of positive functions. This algorithm is a combination of Grey Wolf Optimizer (GWO) [3], [18], [19], [23] and Genetic Algorithm (GA) [2], [8], [20], [31], [32]. Its principle is to use genetic operators to obtain a population with high-performance before applying the steps of hunting of grey wolves. The main steps of the Grey Wolves hunting technique are to pursue, hunt, approach prey, encircle and harass prey until it stops and finally attacks.

Note that, the family of grey wolves is organized into four levels, of which the first level is positioned by the appointed leader (α), who is assisted by the wolf (β) at the second level. On the third level is the wolf (δ) and on the fourth level, we have the rest of wolves called (ω). In hunting, this hierarchy is respected, which makes wolves (α) the best hunting solution, followed by wolf (β) and so on. The wolves initiate the pursuit, encircle the prey, and torment it until they immobilize it. At this moment, they can attack the prey. Mathematically, encirclement is modeled as follows [17], [19], [23]:

$$\begin{cases} \overrightarrow{D}(i) = |\overrightarrow{C}.\overrightarrow{X}_p(i) - \overrightarrow{X}(i)| \\ \overrightarrow{X}(i+1) = \overrightarrow{X}_p(i) - \overrightarrow{A}.\overrightarrow{D}(i) \end{cases} \tag{1}$$

where i denote the number of current iteration, $\overrightarrow{A} = 2a\vec{r}_1 - a$, $\overrightarrow{C} = 2\vec{r}_2$, a is a coefficient which decreases relative to iterations. It is defined by[17], [19], [23]:

$$a = 2\left(1 - \frac{i^d}{MaxInter^d}\right) \tag{2}$$

where i is the current iteration, d the space dimension, MaxInter is the maximal number of iterations. $\overrightarrow{X}_p$ is the vector given the position of the prey, $\overrightarrow{X}$ is the vector given the position of green wolves, $\vec{r}_1$ and $\vec{r}_2$ are random vectors belong in $[0,1]$.

When $|\overrightarrow{A}| < 1$, then the wolf (α) converges toward the prey to attack it as presented in the Fig. 1 (a) and when $|\overrightarrow{A}| > 1$ the wolf (α) is looking for a prey as shown in the Fig. 1 (b) [13].

The position of wolves $(\alpha), (\beta)$ and (δ) are individually adjusted according to prey and those of wolves (ω) follows the principle of hierarchy. The mathematical modeling of positions of these three wolves is [13], [17], [18], [19]:

$$\begin{cases} \overrightarrow{D}_\alpha(i) = |\overrightarrow{C}_1.\overrightarrow{X}_\alpha(i) - \overrightarrow{X}(i)|, \\ \overrightarrow{D}_\beta(i) = |\overrightarrow{C}_2.\overrightarrow{X}_\beta(i) - \overrightarrow{X}(i)|, \\ \overrightarrow{D}_\delta(i) = |\overrightarrow{C}_3.\overrightarrow{X}_\delta(i) - \overrightarrow{X}(i)|, \end{cases} \tag{3}$$

where $\overrightarrow{C}_1, \overrightarrow{C}_2$ and $\overrightarrow{C}_3$ are random vectors, $X_\alpha, X_\beta, X_\delta$ are respectively the positions of $(\alpha), (\beta)$ and (δ). The new best position of wolves, which is the optimal solution, is:

$$X(i+1) = 0.7 \times X_1(i) + 0.2 \times X_2(i) + 0.1 \times X_3(i) \tag{4}$$

where:

$$\begin{cases} \overrightarrow{X}_1(i) = \overrightarrow{X}_\alpha(i) - \overrightarrow{A}_1.\overrightarrow{D}_\alpha(i), \\ \overrightarrow{X}_2(i) = \overrightarrow{X}_\beta(i) - \overrightarrow{A}_2.\overrightarrow{D}_\beta(i), \\ \overrightarrow{X}_3(i) = \overrightarrow{X}_\delta(i) - \overrightarrow{A}_3.\overrightarrow{D}_\delta(i). \end{cases} \tag{5}$$

In practice, the HmGWOGA method has given better results (it is faster and more convergent) than the initial GWO method on single objective optimization problems [17].

C. Performance metrics

In the multiobjective optimization field, we have metrics that allow us to evaluate the performance of a given method. Some of these metrics evaluate the performance of a method and others are used to compare directly two methods at a time. For the first group we have used the generational distance metrics (γ) [16], [29], [31], [32], spread metrics (Δ) [16], [29], [31], [32] and spacing metric (S) [34], [35]. For the second metrics group, we have used the contribution-metric ($Cont$) and C-metric for a direct comparison of MOGWAT and NSGA-II, this is a reference method, on some multiobjective optimization test problems.

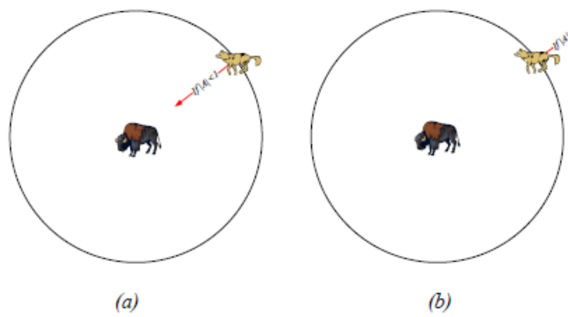


Fig. 1. Search process by grey wolf for find a prey.

Generational distance (γ)

γ is a metric to evaluate the convergence of the method toward the Pareto optimal solutions. It consists is evaluating of the distance between obtained solutions and analytic solutions. It is denoted by γ and its compute formula is :

$$\gamma = \frac{1}{K}\left(\sum_{i=1}^{K} d_i^l\right)^{\frac{1}{l}} \tag{6}$$

where K is the number of obtained solutions; l is an integer such that $1 \le l \le +\infty$; d_i, $i = \overline{1,K}$ the Euclidean distance between obtained solution and the analytic solutions; the value of γ is always between zero and one. When it is closed to zero, the convergence of the used method is good.

Spread (Δ)

Δ measures the degree of uniform distribution achieved by the solutions obtained. A method has good uniform distribution of solutions if the value of this metric is close to zero. It is noted by Δ and is defined by:

$$\Delta = \frac{\sum_{i=1}^{m} d_i^e + \sum_{i=1}^{|Q|} |d_i - \overline{d}|}{\sum_{i=1}^{m} d_i^e + |Q|\overline{d}} \tag{7}$$

where d_i are Euclidean distances between neighboring solutions with $\overline{d}$ their average value; d_i^e is the distance between the extreme solution of analytic Pareto front with obtained Pareto front.

Spacing S

The spacing metric S is also used to measure the uniformity of the distribution of the Pareto optimal solutions. For a bi-objective problem, it is defined as follows:

$$S = \left[\frac{1}{n-1} \times \sum_{i=1}^{n} \left(d_i - \overline{d}\right)^2\right]^{\frac{1}{2}} \tag{8}$$

where$n$ is the number of the obtained solutions, $\forall\ i,j \in \{1,...,n\}$ and $i \neq j$,

$$d_i = \min_j \left(|f_1^i(x) - f_1^j(x)| + |f_2^i(x) - f_2^j(x)| \right),$$

and $\overline{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$. When the value of spacing S is next to 0, the distribution of obtained solution on the Pareto front is good. Indeed, if the Pareto front is discontinuous, the value of the parameter S will be very large.

Contribution ($Cont$)
The metric contribution is proposed by Zitzler and allows quick evaluation of the improvement brought by a method over to another. We can use it to compare two sets of solutions obtained using two different methods. It is given by:

$$Cont(A,B) = \frac{\frac{|P_X|}{2} + |W_A| + |N_A|}{|P_X| + |W_A| + |N_A| + |W_B| + |N_B|} \quad (9)$$

where A and B denote methods; P_X denote the set of identical solutions given by the two methods; W_A and N_A denote respectively the sets of solutions of A that dominate at least one solution of B and the set of solution of A that are not comparable to those of B. W_B, N_B are defined in same logic[29], [30], [33].

C-metric
The C metric is also a proposition of Zitzler to calculate the ratio of solutions from a method that dominate another method. It is defined by:

$$C(A,B) = \frac{|\{b \in B / \exists a \in A : a \preceq b\}|}{|B|} \quad (10)$$

Where $a \preceq b$ means that the solution a is preferred to solution b.

D. Test problems

For us to evaluate the performance of our method, we have chosen some test problems to be solved[16], [32], [34], [35] from the literature review.

1) Test problems of Zitzler-Deb-Thiele:
We have dealt with eighteen test problems shared out into two groups. The Table I gives the test problems of which we know the analytic front and Table II gives the test problems of which we do not know the analytic front.

2) Test problems from CEC 2009:
We have selected two test problems from the CEC 2009 test problems to evaluate our method in relation to NSGA-II.

3) Test problems from Engineering area:
Two problems have been dealt with : Four-bar truss design problem and Cantilever beam design problem.

Example 1. *The four-bar truss design problem is a well-known problem in the structural optimization field[34], [35], in which structural volume (f_1) and displacement (f_2) of a 4-bar truss should be minimized. As can be seen in the following equations, there are four design variables ($x_1 - x_4$) related to cross sectional area of members* $1, 2, 3,$ *and* 4.

$$\begin{cases} \min f_1(x) = 200 \times (2 \times x_1) + \sqrt{2 \times x_2} + \sqrt{x_3 + x_4} \\ \min f_2(x) = 0,01 \times (\frac{2}{x_1} + \frac{2\sqrt{2}}{x_2} - (\frac{2\sqrt{2}}{x_3} + \frac{2}{x_1})) \\ 1 \leq x_1 \leq 3 \\ 1,4142 \leq x_2 \leq 3 \\ 1,4142 \leq x_3 \leq 3 \\ 1 \leq x_4 \leq 3 \end{cases} \quad (11)$$

Example 2. *The cantilever beam design problem is another*

TABLE I
PROBLEMS WHICH ANALYTIC PARETO FRONT ARE KNOWN.

$$PL1 \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = \dfrac{1+x_2}{x_1} \\ 0.1 \leq x_1 \leq 1 \\ 0 \leq x_2 \leq 5 \end{cases}$$

$$PL2 \begin{cases} \max f_1(x) = 1.1 - x_1 \\ \max f_2(x) = 60 - \dfrac{1+x_2}{x_1}, \\ 0,1 \leq x_1 \leq 1 \\ 0 \leq x_2 \leq 5 \end{cases}$$

$$PL3 \begin{cases} \min f_1(x) = x^2 \\ \min f_2(x) = (x-2)^2 \\ -5 \leq x \leq 5 \end{cases}$$

$$PL4 \begin{cases} \min f_1(x) = 1 - \exp\left(-\sum_{i=1}^{n} (x_i - \frac{1}{\sqrt{n}})^2 \right) \\ \min f_2(x) = 1 - \exp\left(-\sum_{i=1}^{n} (x_i + \frac{1}{\sqrt{n}})^2 \right) \\ -4 \leq x_i \leq 4,\ i = \overline{1,10} \end{cases}$$

$$PL5 \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = g(x) \times \left(1 - \sqrt{\dfrac{f_1(x)}{g(x)}}\right) \\ g(x) = 1 + \dfrac{9}{n-1} \times \sum_{i=2}^{n} x_i \\ 0 \leq x_i \leq 1,\ i = \overline{1,30} \end{cases}$$

$$PL6 \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = g(x) \times \left(\sqrt{1 - \dfrac{f_1(x)}{g(x)}}\right) \\ g(x) = 1 + \dfrac{9}{n-1} \times \sum_{i=2}^{n} x_i \\ 0 \leq x_i \leq 1,\ i = \overline{1,30} \end{cases}$$

$$PL7 \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = g(x) \times \left(1 - \left(\dfrac{f_1(x)}{g(x)}\right)^2\right) \\ g(x) = 1 + \dfrac{9}{n-1} \times \sum_{i=2}^{n} x_i \\ 0 \leq x_i \leq 1,\ i = \overline{1,30} \end{cases}$$

$$PL8 \begin{cases} \min f_1(x) = x_1 \\ \min f_2(x) = g(x) \times h(x) \\ g(x) = 1 + \dfrac{9}{n-1} \times \sum_{i=2}^{n} x_i \\ h(x) = 1 - \sqrt{\dfrac{f_1(x)}{g(x)}} - \dfrac{f_1(x)}{g(x)} \sin\left(10\pi f_1(x)\right) \\ 0 \leq x_i \leq 1,\ i = \overline{1,30} \end{cases}$$

TABLE II
PROBLEMS WHICH ANALYTIC PARETO FRONT ARE NOT KNOWN

$$PL9\begin{cases}\min f_1(x)=x_1^2\\ \min f_2(x)=\dfrac{1+x_2^2}{x_1^2}\\ \sqrt{0,1}\le x_1\le 1\\ 0\le x_2\le\sqrt{5}\end{cases}$$

$$PL10\begin{cases}\min f_1(x)=x_1\\ \min f_2(x)=1+x_2^2-x_1-0.2\sin\left(\pi x_1\right)\\ 0\le x_1\le 1\\ -2\le x_2\le 2\end{cases}$$

$$PL11\begin{cases}\min f_1(x)=1-\exp(-4x_1\sin^4(5\pi x_1))\\ \min f_2(x)=g(x)\times\left(1-(\frac{f(x)}{g(x)})^4\right)\\ g(x)=1+9x_2^2\\ 0\le x_1\le 1\\ -1\le x_2\le 1\end{cases}$$

$$PL12\begin{cases}\min f_1(x)=4x_1^2+4x_2^2\\ \min f_2(x)=(x_1-5)^2+(x_2-5)^2\\ (x_1-5)^2+x_2\le 25\\ (x_1-8)^2+(x_2+3)^2\ge 7.7\\ x_1,x_2\ge 0\end{cases}$$

$$PL13\begin{cases}\min f_1(x)=\displaystyle\sum_{i=1}^{2}\left(-10\exp\left(-0.2\sqrt{x_i^2+x_{i+1}^2}\right)\right)\\ \min f_2(x)=\displaystyle\sum_{i=1}^{3}\left(|x_i|^{0.8}+5\sin(x_i^3)\right)\\ -5\le x_1,x_2,x_3\le 5\end{cases}$$

$$PL14\begin{cases}\min f_1(x)=x_1\\ \min f_2(x)=1-\sqrt{\dfrac{f_1(x)}{g(x)}}\\ g(x)=1+10(n-1)+\displaystyle\sum_{i=2}^{n}(x_i^2-10\cos\left(4\pi x_i\right))\\ 0\le x_1\le 1\\ -5\le x_i\le 5,\ i=\overline{2,10}\end{cases}$$

$$PL15\begin{cases}\min f_1(x)=x_1\\ \min f_2(x)=x_2\\ x_1^2+x_2^2-1-0,1\cos\left(16\arctan(\frac{x_1}{x_2})\right)\ge 0\\ (x_1-0.5)^2+(x_2-0.5)^2\le 0.5\\ 0\le x_1,x_2\le\pi\end{cases}$$

$$PL16\begin{cases}\min f_1(x)=0.5(x_1^2+x_2^2)+\sin\left(x_1^2+x_2^2\right)\\ \min f_2(x)=\dfrac{(3x_1-2x_2+4)^2}{8}+\dfrac{(x_1-x_2+1)^2}{27}+15\\ \min f_1(x)=\dfrac{1}{x_1^2+x_2^2+1}-1.1*\exp\left[-(x_1^2+x_2^2)\right]\\ -3\le x_1,x_2\le 3\end{cases}$$

well known problem in the field of concrete engineering [34], [35], in which weight (f_1) and end deflection (f_2) of a cantilever beam should be minimized. There are two design variables: diameter (x_1) and length (x_2).

$$\begin{cases}\min f_1(x)=0,25\times\rho\times\pi\times x_2\times x_1^2\\ \min f_2(x)=\dfrac{64\times P\times x_2^3}{3\times E\times\pi\times x_1^4}\\ -S+(32\times P\times x_2)/\pi\times x_1^3\le 0\\ -\delta+(64\times P\times x_2^3)/(3\times E\times\pi\times x_1^4)\\ 0,01\le x_1\le 0,05\\ 0,20\le x_2\le 1\end{cases}\tag{12}$$

with $P=1, E=207000000, S=300000, \delta=0,005,$

TABLE III
CEC 2009 TEST PROBLEMS

$$uf1\begin{cases}\min f_1(x)=x_1+\dfrac{2}{|J_1|}\displaystyle\sum_{j\in J_1}\left[x_j-\sin(6\pi x_1+\frac{j\pi}{n})\right]^2\\ \min f_2(x)=1-\sqrt{x_1}+\dfrac{2}{|J_2|}\displaystyle\sum_{j\in J_2}\left[x_j-\sin(6\pi x_1\right.\\ \qquad\left.+\dfrac{j\pi}{n})\right]^2\\ J_1=\{j/j\text{ is odd and }2\le j\le n\}\\ J_2=\{j/j\text{ is even and }2\le j\le n\}\\ 0\le x_1\le 1\\ -1\le x_j\le 1; j=\overline{2:n}, n=30\end{cases}$$

$$uf2\begin{cases}\min f_1(x)=x_1+\dfrac{2}{|J_1|}\displaystyle\sum_{j\in J_1}y_j^2\\ \min f_2(x)=1-\sqrt{x_1}+\dfrac{2}{|J_2|}\displaystyle\sum_{j\in J_2}y_j^2\\ J_1=\{j/j\text{ is odd and }2\le j\le n\}\\ J_2=\{j/j\text{ is even and }2\le j\le n\}\\ y_j=x_j-[0,3x_1^2\cos(24\pi x_1+\dfrac{4j\pi}{n})+0,6x_1]\times\\ \qquad\cos(6\pi x_1+\dfrac{j\pi}{n})\\ 0\le x_1\le 1\\ -1\le x_j\le 1; j=\overline{2:n}, n=30\end{cases}$$

$\rho=7800.$

III. MOGWAT METHOD

A. Description

The principle of MOGWAT method is to transform any multiobjective optimization problem into an unconstrained single objective optimization problem before its complete resolution. The main steps are described below.

Step 1: aggregation. It consists in using the ϵ-constraint approach to transform multiple objective functions into single objective functions. Indeed, we choose one of the objective functions to minimize, while the others are transformed into constraint functions. This allows us to reword the initial problem (P) as follows:

$$\begin{cases}\min f_k(x);\\ f_i(x)\le\epsilon_i,\ i\ne k,\ i=\overline{1,p};\\ g_j(x)\le 0,\ j=\overline{1,m}\\ x\in\mathbb{R}^n;\end{cases}\tag{P$_\epsilon$}$$

where $\epsilon\in\mathbb{R}^{p-1}$. Let us set $\Omega=\{x\in\mathbb{R}: f_i(x)\le\epsilon_i, i=\overline{1,p}, i\ne k$ and $g_j(x)\le 0, j=\overline{1,m}\}$. The component $\epsilon_i,\ i\ne k,\ i=\overline{1,p}$ is irrelevant for (P_ϵ), but the convention to include it will be convenient in the following theorems.

Step 2: penalization. It aims to transform the problem (P_ϵ) into an unconstrained problem by using a Lagrangian penalty function [16], [25], [27]. With which, the new formulation of (P_ϵ) as follows:

$$\begin{cases}Global.\min L(f,\epsilon,\eta)\\ x\in\mathbb{R}^n;\end{cases}\tag{P$_\epsilon^\eta$}$$

where $L(f,\epsilon,\eta)=f_k(x)+\eta\displaystyle\sum_{j=1,j\ne k}^{m+p}\Big(f_j(x)-\epsilon_j+g_j(x)+|f_j(x)-\epsilon_j+g_j(x)|\Big)$; and η a large real number.

Step 3: resolution. It aims to propose solutions to the initial problem by solving the last formulation

(P_ϵ^η). With the using of HmGWOGA algorithm, we can write :

$$\chi_E^\epsilon = HmGWOGA\{L(f,\epsilon,\eta)\}.$$

B. Algorithm

Then, the algorithm of MOGWAT method can be summarize as follows.

Algorithm for MOGWAT

Data : $\min_{x\in\chi} f(x)$;

Step 1

- Choose one objective f_k;
- Estimate ϵ_j by solving $\min_{x\in\chi} f_j(x)$, $\max_{x\in\chi} f_j(x)$;
- and build problem (P_ϵ) : $\min_{x\in\Omega} f_k(x)$;

Step 2 : Fix η value and build $L(f_k,\epsilon,\eta)$

Step 3

- Choose 100 different values of ϵ
- for each value of ϵ Find x_E^ϵ :

$$L(f_k(x_E^\epsilon),\epsilon,\eta) = \text{Global.} \min_{x\in\mathbb{R}^n} L(f_k(x),\epsilon,\eta)$$

- Compute $f(x_E^\epsilon)$

IV. Results and discussion

A. Theoretical performance results

The theoretical performance study of the MOGWAT method has been done on the complexity of its algorithm and the optimality of obtained solutions.

The following two theorems guarantees the optimality of the solutions obtained by the using of MOGWAT method.

Theorem 1. *Let $\epsilon_i \in [\min_{x\in\chi} f_i(x), \max_{x\in\chi} f_i(x)]$, $i=\overline{1,p}$, $i \neq K$ be some fixed parameters. Let χ_E^ϵ be a global optimal solutions set of (P_ϵ). Then $\chi_E \setminus \chi_E^\epsilon = \emptyset$.*

Proof of Theorem 1: Let $x^* \in \chi$ such as $x^* \in \chi_E \setminus \chi_E^\epsilon$ then, $x^* \in \chi_E$ and $x^* \notin \chi_E^\epsilon$. As $x^* \notin \chi_E^\epsilon$, then there exists at least one $x \in \chi_E^\epsilon$ such as $f_k(x) < f_k(x^*)$ and for all $i \in \{1,2,\cdots,p\}$ with $i \neq k$, we have $f_i(x) \leq \epsilon_i$. Let us set $\epsilon_i = f_i(x^*)$ with $i \in \{1,2,\cdots,p\}$ and $i \neq k$, we obtain $f_k(x) < f_k(x^*)$ and $f_i(x) \leq f_i(x^*)$. That is a contradiction with the assumption in which $x^* \in \chi_E$. Hence the result of the theorem. ■

The following theorem is a sufficient condition proving that the formulation (P_ϵ^η) conserves the optimality of its solutions for the initial problem:

Theorem 2. *Let $\epsilon_i \in [\min_{x\in\chi} f_i(x), \max_{x\in\chi} f_i(x)]$, $i=\overline{1,p}$, $i \neq k$ be some fixed parameters and η a fixed large real number. Let x^* be the global optimal solution of (P_ϵ^η). Then, x^* is also the global optimal solution of problem (P_ϵ).*

Proof of Theorem 2: Let x^* be a global optimal solution of the problem (P_ϵ^η). Then, for all $x \in \mathbb{R}^n$, $L(f(x^*),\epsilon,\eta) \leq L(f(x),\epsilon,\eta)$. That is equivalent to

$$f_k(x^*) + \eta \sum_{\substack{j=1 \\ j\neq k}}^{m+p} \Big(f_j(x^*) - \epsilon_j + g_j(x^*) + |f_j(x^*) - \epsilon_j + g_j(x^*)| \Big)$$
$$\leq f_k(x) + \eta \sum_{\substack{j=1 \\ j\neq k}}^{m+p} \Big(f_j(x) - \epsilon_j + g_j(x) + |f_j(x) - \epsilon_j + g_j(x)| \Big).$$

Let us suppose that x and x^* are taken in χ then, $f_j(x) - \epsilon_j + |f_j(x) - \epsilon_j| = 0$ and $g_j(x) + |g_j(x)| = 0$. In this case, $f_k(x^*) \leq f_k(x)$. Hence, $x^* \in \chi_E^\epsilon$. ■

As the problem 2 is unconstrained single-objective optimization problem, then any good global numerical optimization method can be use to achieve the solutions.

The following Theorem 3 guarantees that the computational complexity of MOGWAT method is polynomial.

Theorem 3. *Let M, n and $MaxInt$ be respectively the sizes of the population of solutions, the number of variables, and the number of iterations. Therefore, the computational complexity of the MOGWAT method is $O(M.n.MaxIter)$.*

Proof of Theorem 3: The computational complexity of the aggregation stage is $O(1)$ because it only consists of selecting one objective function. The complexity of the penalty operation is $O(m+p-1)$ because there are $m+p-1$ constraint functions. HmGWOGA method initialization stage requires $O(M.N)$ as the computational time, where M is the size of the population of solutions and n is the number of variables. In order to compute the control parameters of HmGWGA and the update of the position of the grey wolf, the computational complexity is $O(M.n)$[32]. The complexity of the operations for the computation of the fitness value of each grey wolf is also $O(M.n)$. Hence, the sum of these complexities gives $O(M.n.MaxIter)$ as the computational complexity of MOGWAT. ■

As the computational complexity of the NSGA-II method is $O(p.n^2)$ with p the number of objective functions and n the number of decision variables[31], we can conclude that MOGWAT is the best option.

B. Numerical performance results

The numerical performance study of the MOGWAT method has been done on the computational time, the convergence of the obtained solutions and the distribution of the obtained solutions. All of these parameters have been evaluated by using some test problems taken into the literature. In this work, these test problems have been shared out into three groups.

1) **Results of Zitzler-Deb-Thiele test problems:** this is the representation of optimal Pareto solutions of each test problem (see Fig. 2 and Fig. 3), the computational time of the used methods (see Table IV, Table V, Table X and Table XI), the convergence parameters (see Table VI and Table VII) and distribution parameters (see Table VIII and Table IX). As MOGWAT is a stochastic algorithm, we computed for each problem one hundred solutions in ten times. This allows us to calculate the average and variance of the performance parameters for all these ten times. We compare our results to those of NSGA-II because all of these test problems have been solved by this method [31], [32].

In the following we will sometimes set A=MOGWAT and B=NSGA-II.

Here are the results for the problems with analytic Pareto front. For them, we have plotted in the same figure the analytic Pareto front and those given by MOGWAT method.

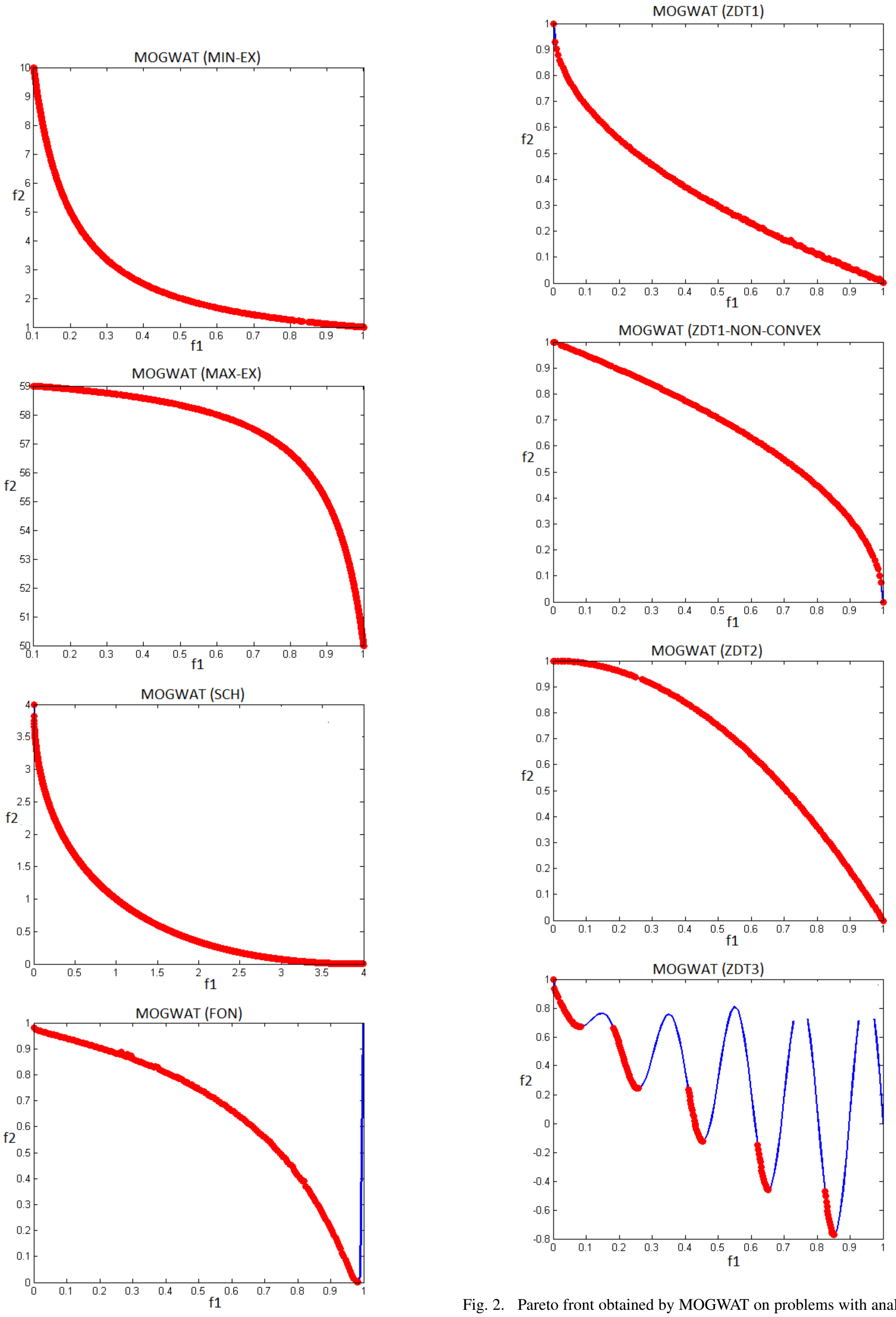


Fig. 2. Pareto front obtained by MOGWAT on problems with analytic front

TABLE IV
COMPUTATIONAL TIME FOR ONE HUNDRED SOLUTIONS

		PL1	**PL2**	**PL3**	**PL4**
A	$\overline{\tau}$	31.7	39.2	29.9	54.5
	σ_{τ}^2	00.0	00.0	00.00	00.1
B	$\overline{\tau}$	50.8	59.5	49.7	94.9
	σ_{τ}^2	00,2	05.8	04.2	05.8

TABLE V
COMPUTATIONAL TIME FOR ONE HUNDRED SOLUTIONS

		PL5	**PL6**	**PL7**	**PL8**
A	$\overline{\tau}$	174.4	150.9	176.0	193.8
	σ_{τ}^2	000.0	000.0	000.0	000.2
B	$\overline{\tau}$	217.1	173.5	224.1	220.6
	σ_{τ}^2	001.4	007.3	001.5	005.5

TABLE VI
GENERATIONAL DISTANCE FOR CONVERGENCE PERFORMANCE

		PL1	**PL2**	**PL3**	**PL4**
A	$\overline{\gamma}(e-4)$	01.5	01.8	00.2	44.0
	$\sigma_{\gamma}^2(e-10)$	00.5	00.0	01.3	00.0
B	$\overline{\gamma}(e-4)$	93.0	02.2	75.0	45.0
	$\sigma_{\gamma}^2(e-5)$	00.0	50.0	00.0	0.00

TABLE VII
GENERATIONAL DISTANCE FOR CONVERGENCE PERFORMANCE

		PL5	**PL6**	**PL7**	**PL8**
A	$\overline{\gamma}(e-4)$	04.3	11.0	06.0	14.0
	$\sigma_{\gamma}^2(e-10)$	01.5	00.0	00.6	20.2
B	$\overline{\gamma}(e-4)$	09.5	40.0	07.6	27.0
	$\sigma_{\gamma}^2(e-6)$	00.0	00.0	58.4	00.0

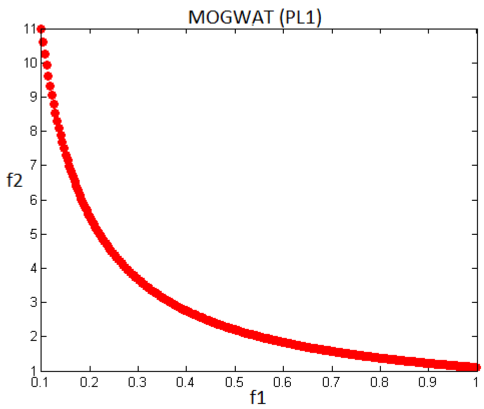
TABLE VIII
SPREAD INDEX FOR DISTRIBUTION PERFORMANCE

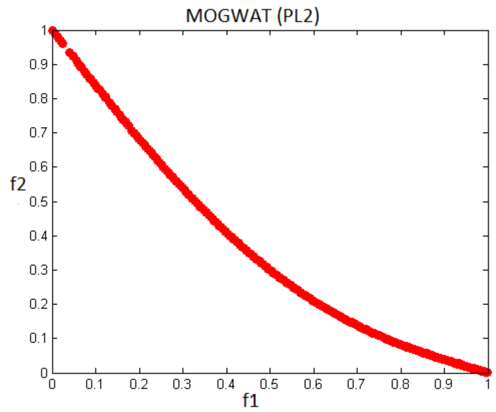
		PL1	PL2	PL3	PL4
A	$\overline{\Delta}(e-4)$	7776	9815	4998	9145
	$\sigma_{\Delta}^2(e-5)$	5560	0000	0000	0000
B	$\overline{\Delta}(e-4)$	6775	6207	5825	6765
	$\sigma_{\Delta}^2(e-6)$	1000	0900	0075	0700

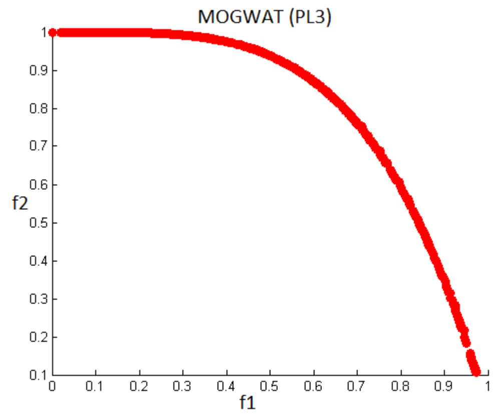
TABLE IX
SPREAD INDEX FOR DISTRIBUTION PERFORMANCE

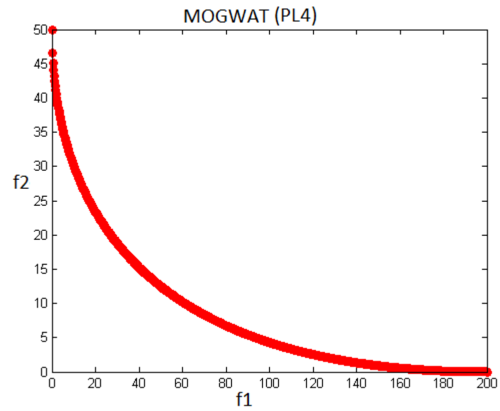
		PL5	PL6	PL7	PL8
A	$\overline{\Delta}(e-4)$	2540.0	2999.0	2633.0	1280.0
	$\sigma_{\Delta}^2(e-7)$	0008.8	0000.0	0006.0	0044.2
B	$\overline{\Delta}(e-4)$	6178.0	6715.0	6087.0	5909.0
	$\sigma_{\Delta}^2(e-6)$	0800.0	0700.0	2800.0	1100.0

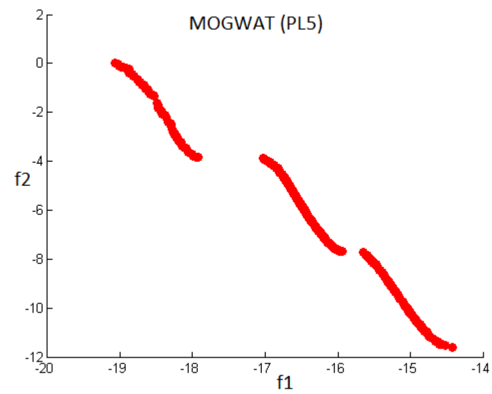
Here are the results for the problem without analytic Pareto front.











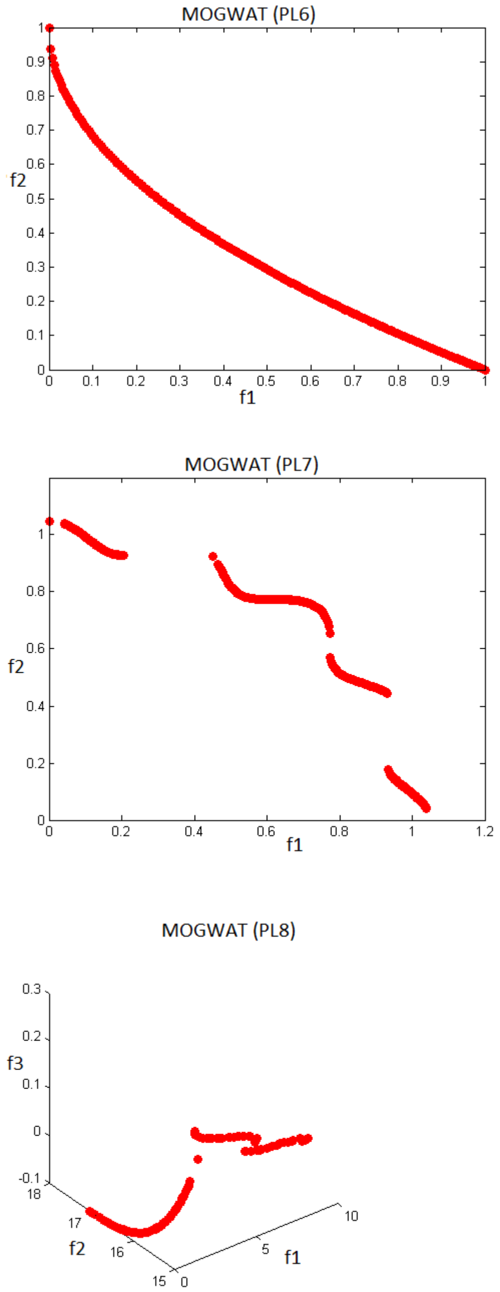


Fig. 3. Pareto front obtained by MOGWAT on problems without analytic front

TABLE X
COMPUTATIONAL TIME FOR ONE HUNDRED SOLUTIONS

		PL9	PL10	PL11	PL12
A	$\overline{\tau}$	40.5	37.0	47.8	46.2
	σ^2_{τ}	00.0	13.0	00.0	01.7
B	$\overline{\tau}$	74.8	60.8	68.3	68.1
	σ^2_{τ}	04.5	17.1	00.6	00.0

TABLE XI
COMPUTATIONAL TIME FOR ONE HUNDRED SOLUTIONS

		PL13	PL14	PL15	PL16
A	$\overline{\tau}$	039.4	122.7	051.1	078.2
	σ^2_{τ}	000.2	000.5	002.9	000.4
B	$\overline{\tau}$	061.6	161.4	065.3	098.5
	σ^2_{τ}	009.3	001.1	001.7	008.2

TABLE XII
VALUE OF CONTRIBUTION METRIC

		PL9	PL10	PL11	PL12
A	$\overline{Cont}_{(A,B)}$	0.5102	0.5127	0.4867	0.6120
	$\sigma^2_{Cont(A,B)}$	0.0000	0.0000	0.0015	0.0007
B	$\overline{Cont}_{(B,A)}$	0.4898	0.4873	0.5133	0.3817
	$\sigma^2_{Cont(B,A)}$	0.0000	0.0000	0.0015	0.0011

TABLE XIII
VALUE OF CONTRIBUTION METRIC

		PL13	PL14	PL15	PL16
A	$\overline{Cont}_{(A,B)}$	0.6507	0.9529	0.5795	0.3664
	$\sigma^2_{Cont(A,B)}$	0.0169	0.0001	0.0018	0.0000
B	$\overline{Cont}_{(B,A)}$	0.3493	0.0471	0.4205	0.6336
	$\sigma^2_{Cont(B,A)}$	0.0169	0.0001	0.0067	0.0000

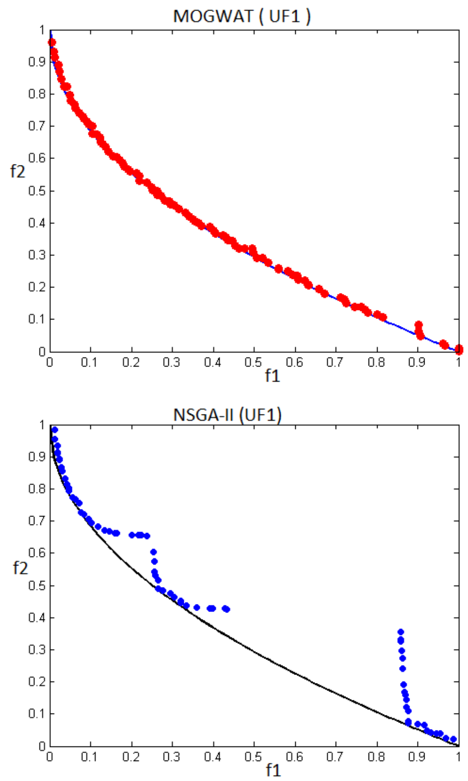
TABLE XIV
VALUES OF C-METRIC

		PL9	PL10	PL11	PL12
A	$\overline{C}_{(A,B)}$	0.0533	0.0004	0.0003	0.3567
	$\sigma^2_{C(A,B)}$	0.0006	0.0000	0.0005	0.0048
B	$\overline{C}_{(B,A)}$	0.0000	0.0000	0.0825	0.0000
	$\sigma^2_{C(B,A)}$	0,0000	0,0000	0.0136	0.0000

TABLE XV
VALUES OF C-METRIC

		PL13	PL14	PL15	PL16
A	$\overline{C}_{(A,B)}$	0.5167	0.9300	0.6307	0.0600
	$\sigma^2_{C(A,B)}$	0.0748	0.0001	0.0009	0.0009
B	$\overline{C}_{(B,A)}$	0.1716	0.0000	0.0234	0.4389
	$\sigma^2_{C(B,A)}$	0.0016	0.0000	0.0011	0.0001

2) **Results of CEC 2009 test problems:** The Pareto optimal solutions obtained by applying the MOGWAT method are given in the Fig. 4



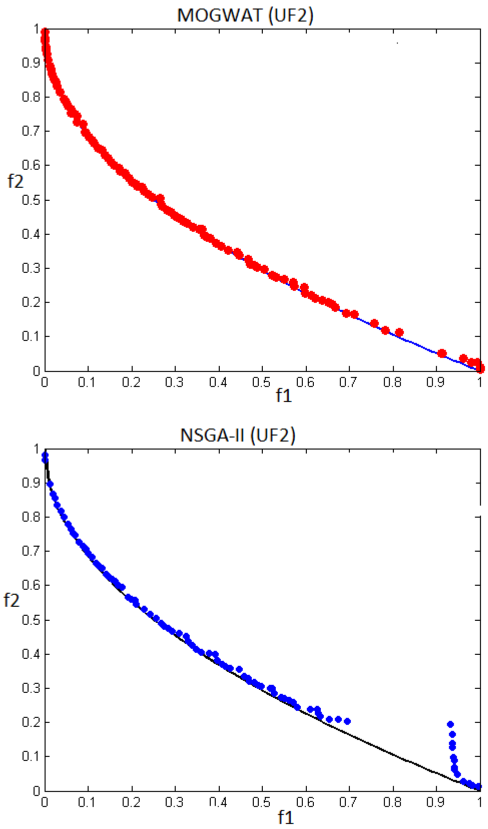


Fig. 4. Pareto front obtained by MOGWAT and NSGA-II for uf1 and uf2

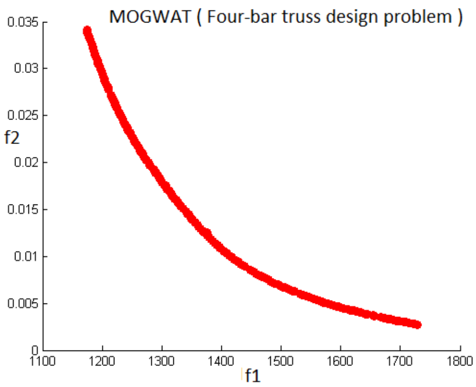
TABLE XVI
VALUE OF METRIC GD FOR UF1, UF2

	MOGWAT		**NSGA-II**	
GD	mean	std	mean	std
Four-bar	0,0097	0,4056	0,0256	0,0256
Cantilever	0,0105	0,1210	0,0135	0,0100

TABLE XVII
VALUE OF METRIC Δ FOR UF1, UF2

	MOGWAT		**NSGA-II**	
Δ	mean	std	mean	std
uf1	0,0367	0,3006	0,2010	0,0034
uf2	0,0502	0,1101	0,12075	0,0219

3) **Results of Engineering problems:**
For the **Example 1**, we have obtained the Pareto optimal solutions with MOGWAT and NSGA-II that we have given in the Fig. 5.



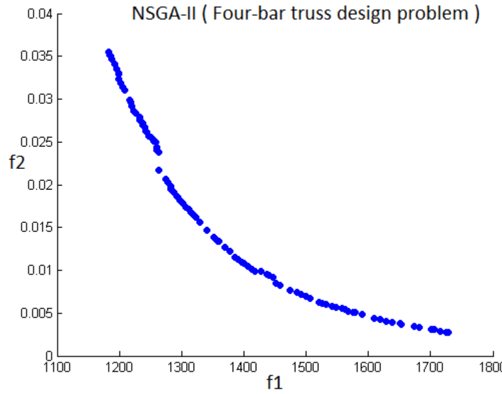


Fig. 5. Pareto front obtained by MOGWAT and NSGA-II on Four-bar truss design problem

For the **Example 2**, MOGWAT and NSGA-II methods have been applied and the Pareto optimal solutions are given in the Fig. 6.

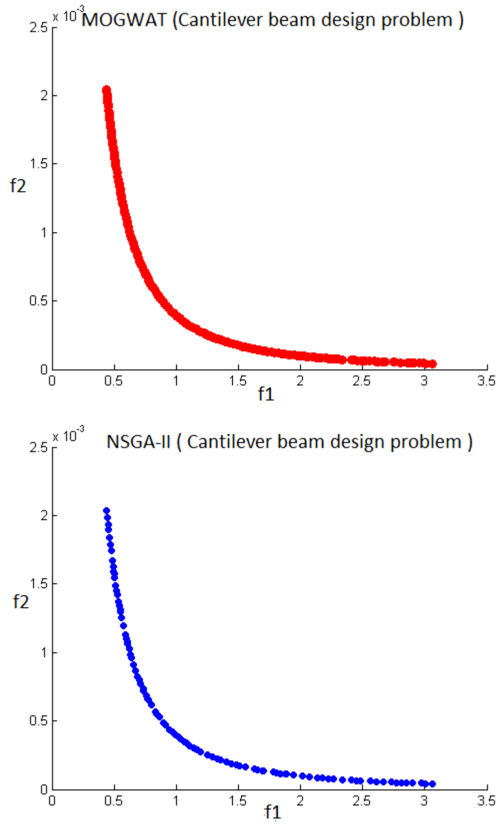


Fig. 6. Pareto front obtained by MOGWAT and NSGA-II on Cantilever beam design problem

TABLE XVIII
VALUE OF SPACING S ON ENGINEERING PROBLEMS

	MOGWAT		**NSGA-II**	
S	mean	std	mean	std
Four-bar	0,0402	0,3646	0,8261	0,0256
Cantilever	0,0135	0,0010	0,0335	0,0099

C. Discussions

According to the test problems with analytic Pareto front, we have the following results: the **Table IV** and **Table V** show that the MOGWAT method works faster than NSGA-II on all of these eight test problems; the **Table VI** and **Table VII** show that the MOGWAT method outperforms NSGA-II on all of these eight test problems in terms of converge; and the **Table VIII** and **Table IX** show that the MOGWAT method is better than NSGA-II on five test problems, namely $PL3$, $PL5$, $PL6$, $PL7$ and $PL8$ in terms of distribution.

According to the test problems without analytic Pareto front, we have the following results: the **Table X** and **Table XI** show that the MOGWAT method works faster than NSGA-II on all of these eight test problems; the **Table XII** and **Table XIII** show that the MOGWAT method is the better than NSGA-II on only five test problems, namely $PL11$, $PL13$, $PL14$, $PL15$ and $PL16$ in terms of contribution; and the **Table XIV** and **Table XV** show that the MOGWAT method outperforms NSGA-II on all of these eight test problems in terms of the number of dominated solutions ratio.

According to the CEC 2009 test problems, we have the following results: the **Table XVI** shows that the MOGWAT method converges better than NSGA-II; and the **Table XVII** shows MOGWAT method has a better distribution than NSGA-II.

According to the engineering problems, the **Table XVIII** shows that the MOGWAT method has a best distribution than NSGA-II.

Finally, with this combination of the ϵ-constraint approach and the HmGWOGA algorithm, we have created a method that is effective and efficient for solving multiobjective optimization problems. This is better than NSGA-II in terms of computational time and convergence of the obtained solutions. However, in terms of distribution, it is better than NSGA-II on some problems but not all. Then, improving the distribution of the solutions using the MOGWAT method will be a topic for future research.

V. Conclusion

A metaheuristic method for solving a nonlinear multiobjective optimization problem was proposed by this work. We named it MOGWAT. It is a combination of the ϵ-constraint approach and the HmGWOGA algorithm. On the one hand, we have demonstrated the theoretical foundation of our algorithm and its good computational complexity by proposing three theorems. On the other hand, we have substantiated its numerical performances by successfully resolving 20 test problems. The results have been compared to those of NSGA-II about the computational time, convergence of solutions, and distribution of solutions. Out of these 16 test problems of Zitler-Deb-Thiele, MOGWAT is faster on the 100%, converges better on also the 100%, and has a better distribution on the 62.5%. Therefore, MOGWAT is better than NSGA-II at 87% of these performance parameters. For the CEC 2009 test problems and Engineering problem, MOWGAT outperforms NSGA-II on convergence and distribution of Pareto optimal solutions. According to these theoretical and numerical results, MOGWAT is the best choice for solving multiobjective optimization problems.

Our future study will focus on enhancing the distribution of our approach.

References

[1] Yiyuan Wang, Xianfeng Ding, Tingting Wei, Jiaxin Li and Chao Min, "Penalty Function Method for Solving Multiobjective Interval Bilevel linear Programming Problem," IAENG International Journal of Computer Science, vol. 50, no. 1, pp209-218, 2023.

[2] Xijun Zhang, Yunfang Zhong, Baoqi Zhang, and Shengyuan Nie, "Optimization of the NSGA-III Algorithm Using Adaptive Scheduling," Engineering Letters, vol. 31, no. 2, pp459-466, 2023.

[3] M. Premkumar, Pradeep Jangir, B. Santhosh Kumar, Mohammad A. Alqudah and Kottakkaran Sooppy Nisar, "Multi-Objective Grey Wolf Optimization Algorithm for Solving Real-World BLDC Motor Design Problem," Computers, Materials and Continua, vol. 70, no. 2, pp2436-2452, 2022.

[4] Zeynab Hoseini, Hesam Varaee, Mahdi Rafieizonooz, Jang-Ho Jay Kim, "A New Enhanced Hybrid Grey Wolf Optimizer (GWO) Combined with Elephant Herding Optimization (EHO) Algorithm for Engineering Optimization," Journal of Soft Computing in Civil Engineering, vol. 06, no. 4, pp1-42, 2022.

[5] Safora Akhavan-Nasab, Zahra Beheshti, "A Condition-based Grey Wolf Optimizer Algorithm for Global Optimization Problems," Journal of Soft Computing and Information Technology, vol. 11, no. 2, pp26-40, 2022.

[6] Xinyang Liu, Yifan Wang and Miaolei Zhou, "Dimensional Learning Strategy-Based Grey Wolf Optimizer for Solving the Global Optimization Problem," Computational Intelligence and Neuroscience, vol. 2022, pp2-31, 2022.

[7] Eslan Mohammed Abdelkader, Abobakr Al-Sakkaf, Nehal Elshaboury and Ghasan Alfalah, "Hybrid Grey Wolf Optimization-Based Gaussian Process Regression Model for Simulating Deterioration Behavior of Highway Tunnel Components," Processes, vol. 10, no. 1, pp1-38, 2022.

[8] Minghui Zhao, Yongpeng Zhao, Changxi Ma, and Xu Wu, "Robust Optimization of Mixed-Load School Bus Route Based on Multi-Objective Genetic Algorithm," Engineering Letters, vol. 30, no. 4, pp1543-1550, 2022.

[9] Wei Zhang, Sai Zhang, Fengyan Wu and Yagang Wang, "Path Planning of UAV Based on Improved Adaptive Grey Wolf Optimization Algorithm," IEEE Access, vol. 9, IAN21082795, pp89400-89411, 2021.

[10] Farshad, Hamid Reza Safavi, Mohamed Abd Elaziz, Shaker H. Ali El-Sappagh, Mohammed Azmi Al-Betar and Tamer Abuhmed, "An Enhanced Grey Wolf Optimizer with a Velocity-Aided Global Search Mechanism," Mathematics, vol. 10, no. 3, pp1-32, 2021.

[11] A. Som, K. Some and A. Compaore, "Exponential penalty function with MOMA-Plus for the Multiobjective optimization problems," Applied Analysis and Optimization, vol.5, no.3, pp323-334, 2021.

[12] Bingzhuang Liu, "A New Family of Smoothing Exact Penalty Functions for the Constrained Optimization Problem," Engineering Letters, vol. 29, no. 3, pp984-989, 2021.

[13] Amir Seyyedabbasi, Farzad Kiani, "I-GWO and Ex-GWO: improved algorithms of the Grey Wolf Optimizer to solve global optimization problems," Engineering with Computer, vol. 10, no. 1, pp509-532, 2021.

[14] Shubham Gupta, Kusum Deep, "Enhanced leadership-inspired Grey wolf optimizer for global optimization problem," Engineering with Computers, vol. 36, no.1, pp1777–1800, 2020.

[15] Yan-e Hou, Lanxue Dang, Yunfeng Kong, Zheye Wang and Qingjie Zhao, "A Hybrid Metaheuristic Algorithm for the Heterogeneous School Bus Routing Problem and a Real Case Study," IAENG International Journal of Computer Science, vol. 47, no. 4, pp775-785, 2020.

[16] Alexandre Som, Kounhinir Some, Abdoulaye Compaore, Blaise Some, "Performances assessment of MOMA-Plus method on multiobjective optimization problems," European Journal of pure and applied mathematics, vol. 13, no.1, pp48-68, 2020.

[17] W. O. Sawadogo, P. O. F. Ouédraogo, K. Somé, N. Alaa and B. Somé, "Modified hybrid grey wolf optimizer and genetic algorithm (HmGWOGA) for global optimization of positive functions," Advance in Differential Equations and Control Processes, vol. 20, no. 2, pp187-206, 2019.

[18] Wen Long, Shaohong Cai, Jianjun Jiao, Mingzhu Tang, "An efficient and robust grey wolf optimizer algorithm for large-scale numerical optimization," Soft Computing, vol. 24, no.2, pp997-1026, 2019.

[19] Muhammed Arif Sen, Mete Kalyoncu, "Grey Wolf Optimizer Based Tuning of a Hybrid LQR-PID Controller for Foot Trajectory Control of a Quadruped Robot," Gazi University Journal of Science, vol. 32, no. 2, pp674-684, 2019.

[20] Fu Yan, Jianzhong Xu, and Kumchol Yun, "Dynamically Dimensioned Search Grey Wolf Optimizer Based on Positional Interaction Information," Hindawi, Complexity, vol. 2019, Article ID 7189653, pp1-36, 2019.

[21] Hossam Faris, Ibrahim Aljarah, Mohammed Azmi Al-Betar, Seyedali Mirjalili. "Grey wolf optimizer: a review of recent variants and applications," Neural Computing and Applications, vol. 30, no. 1, pp413-435, 2018.

[22] Y. Zheng an Z. Meng, "A new augmented Lagrangian objective penalty function for constrained optimization problems," Open Journal of Optimization, vol. 6, no. 1, pp39-46, 2017.

[23] Long Wen, "Grey Wolf Optimizer based on Nonlinear Adjustment Control Parameter," Advances in Intelligent Systems Research, vol. 136, no. 1, pp643-648, 2016.

[24] Seyedali Mirjalili, Seyed Mohammad Mirjalili, Andrew Lewis, "Grey Wolf Optimizer," Advances in Engineering Software, vol. 69, no. 1, pp46-61, 2014.

[25] Kounhinir some, Berthold Ulungu, Wenddabo Olivier Sawadogo, Blaise Some, "A theoretical foundation metaheuristic method to solve some multiobjective optimization problems," International Journal of Applied Mathematical Research, vol.2, no. 4, pp464-475, 2013.

[26] Mohsen Ejday, "Metamodel based multi-objective optimization for the shaping process," Doctorate dissertation, Paris Tech, France, 2011.

[27] Kounhinir Some, Berthold Ulungu, Ibrahim Himidi Mohamed and Blaise Some, "A new method for solving nonlinear multiobjective optimization problems," JP Journal of Mathematical Sciences, vol. 2, no. 1&2, pp1-18, 2011.

[28] T. Cormen; C. Leiserson; R. Rivest and C. Stein, "Introduction to Algorithms," 3Eds., Cambridge, Massachusetts London, England: The MIT Press, 2009.

[29] Yann Collette-Patrick Siarry, "Multiobjective Optimization," Editions eyrolles, 61, Bld Saint-Germain, Paris Cedex 05, 2002.

[30] H. Meunier, "Parallel Evolutionary Algorithms for Multi-Objective Optimization of Mobile Telecommunications Networks," Doctorate dissertation, Université des Sciences et technologiques de Lille, France, 2002.

[31] K. Deb, A. Patrap, S. Agarwal and T. Meyarivan, "A Fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE Transactions on evolutionary computation, vol. 6, no.2, pp182-197, 2002.

[32] Kalynmoy Deb, "Multiobjective optimization using evolutionary algorithms," Chichester, UK: Wiley, 2001.

[33] E. Zitzler, "Evolutionary algorithms for multiobjective optimization: methods and applications," PhD dissertation, Swiss Federal Institute of Technology, Zurich, 1999.

[34] Seyedali Mirjalili, Pradeep Jangir, Shahrzad Saremi, "Multi-objective ant lion optimizer: a multi-objective optimization algorithm for solving engineering problems," Applied intelligence, vol. 46, no.1, pp79-95, 2017.

[35] Ali Sadollaha, Hadi Eskandarb, Joong Hoon Kima, "Water cycle algorithm for solving constrained multi-objective optimization problems," Applied Soft Computing, vol. 27, no.1, pp279-298, 2015.