ADV MATH SCI JOURNAL
Advances in Mathematics: Scientific Journal **12** (2023), no.8, 701–718
ISSN: 1857-8365 (printed); 1857-8438 (electronic)
https://doi.org/10.37418/amsj.12.8.1

# A NEW HYBRIDIZATION FOR IMPROVING THE CONVERGENCE OF THE MOMA-PLUS METHOD

Abdoulaye Compaoré[1], Alexandre Som, and Kounhinir Somé

ABSTRACT. We propose in this article a hybridization of the algorithm of the MOMA-Plus method and that of the Differential Evolution method. This hybridization consists of defining a simplex around an efficient solution generated by MOMA-plus and applying the Differential Evolution algorithm to find a better solution than that obtained by MOMA-plus. The results interpreted through a performance study of the solutions obtained on multiobjective optimization test problems show that this hybridization improves the convergence of the basic MOMA-plus algorithm. Moreover, a better complexity than that of basic MOMA-plus is obtained.

## 1. INTRODUCTION

Optimization is one of the main characteristics in decision-making at the collective level globally and at the individual level in particular. It is a question of finding the optimum of a function (often vectorial) and subject (or not) to constraints. Modeling an optimization problem leads to a multiple and conflicting objective problem or a single objective problem. The first is mathematically ill-posed due to the non-uniqueness of the solutions. This gives rise to the existence

[1]*corresponding author*
2020 *Mathematics Subject Classification.* 90-08, 90-10, 90C29,90C56, 90C59.
*Key words and phrases.* MOMA-plus method, Differential Evolution algorithm, metaheuristic, hybrid method, Multiobjective optimization.
*Submitted:* 02.07.2023; *Accepted:* 18.07.2023; *Published:* 12.08.2023.

of compromise solutions called Pareto solutions. However, the search for Pareto solutions is a very difficult task, hence the development of new, very powerful resolution techniques. They could be grouped into two large groups: exact methods and metaheuristics [4, 6]. In this work we are only interested in metaheuristics, more specifically, we are interested in the MOMA-plus metaheuristic developed by K. SOME et al. [20] and the Differential Evolution metaheuristic developed by R. Storn and K. Price [22].

The MOMA-plus method uses the technique of aggregation of objective functions, then a penalty function in order to make a multi-objective optimization problem with constraints into a single-objective optimization problem without constraints. Then it uses an Alienor transformation technique to reduce the multivariate optimization problem to a single variable problem. Subsequently, MOMA-plus was subsequently used in several works such as [9–14, 17, 18, 21]. However, the use of the Alienor transformation made the problem multimodal and impacted convergence as well as complexity. Hence the need to improve this convergence and this complexity using hybridization with the Differential Evolution method.

Indeed, the Differential Evolution method is a metaheuristic which is similar to the Pattern search and the genetic algorithm in the sense that it uses similar operators such as selection, crossover and mutation for the search of the optimal solution. These operators, like the genetic algorithm, use binary or real coding. It is based on the vector mutation characterized by a disturbance of the selected individuals. However, such algorithms make few or no assumptions about the underlying optimization problem (just like MOMA-plus) and can quickly explore very large design spaces. Also the Differential Evolution method is arguably one of the most versatile and stable population-based search algorithms that exhibits robustness to multimodal problems, as evidenced by the multiple works carried out [1–3, 7, 8].

Also, the use of the Nelder-Mead algorithm only provides local optima, hence, in our hybridization, the principle is that the solution found by the Nelder-Mead algorithm, in the MOMA-plus method, will be used as a starting point for the application of the Differential Evolution metaheuristic in order to search for a near and better solution. Subsequently, an even larger search procedure will be launched on the search space of the penalized function.

To better expose our work, we go in Section 2, to present the preliminaries, in Section 3, to present the theoretical results of our hybridization and to finish, in Section 4, we will present the numerical results.

## 2. PRELIMINARIES

### 2.1. **Basics of Multiobjective Optimization.**

Consider the following problem:

$$\begin{array}{ll} \min & f_1(x_1,\ldots,x_n) \\ \min & f_2(x_1,\ldots,x_n) \\ \vdots & \\ \min & f_p(x_1,\ldots,x_n) \\ s.t: & \begin{cases} g_1(x_1,\ldots,x_n) \leq 0, \\ \vdots \\ g_m(x_1,\ldots,x_n) \leq 0, \\ x_i \in \mathbb{R}, i=1,2,\ldots,n. \end{cases} \end{array} \tag{2.1}$$

For the resolution of a problem of the type (2.1), the use of an aggregation function allowing to transform a multiobjective optimization problem in a general way into a single-objective optimization problem. In the context of this work, it is Tchebycheff function denoted Tch and defined from $\mathbb{R} \times \mathbb{R}^p \times \mathbb{R}^p$ to $\mathbb{R}$ by

$$\text{Tch}(f(x_1,\ldots,x_n),\lambda,\overline{z}) = \max_{j=1,\ldots,p}\{\lambda_j|f_j(x_1,\ldots,x_n)-\overline{z}_j|\} \tag{2.2}$$

which is used. In the relation (2.2), $\overline{z}=(\overline{z}_1,\overline{z}_2,\ldots,\overline{z}_p)$ denotes the ideal point of (2.1) and the $\lambda_j \in [0,1]$ are weighting weights such that $\sum\limits_{j=1}^{m}\lambda_j = 1$.

### 2.2. **MOMA-plus algorithm.**

The principle of the MOMA-plus [20] method is to reduce a multi-objective optimization problem with constraints of several variables to a single-objective optimization problem without constraints of a single variable. For this, it uses successively the weighted distance of Tchebycheff, a penalization function, the Alienor transformation and the Nelder-Mead algorithm [16].

The commonly used Alienor transformation is that of Confé-Cherruault [15] which is defined as follows:

$$x_i = h_i(\theta) = \frac{1}{2}\Big[(b_i - a_i)\cos(\omega_i\theta + \phi_i) + a_i + b_i,\Big],\ x_i \in [a_i, b_i],\ i = 1, \dots, n, \tag{2.3}$$

with $\theta_{max} = \dfrac{(b-a)\theta^1 + (b+a)}{2}$ and $\theta^1 = \dfrac{2\pi - \phi_1}{\omega_1}$.

The algorithm of the MOMA-Plus method is defined below [20]:

**Algorithm 1** MOMA-Plus Algorithm

Begin
$f(x_1, \dots, x_n) \longleftarrow \text{Tch}(f(x_1, \dots, x_n), \lambda, \overline{z})$
$g(x_1, \dots, x_n) \longleftarrow g_1(x_1, \dots, x_n) + |g_1(x_1, \dots, x_n)|$.
**for** $i = 2$ to $m$ **do**
  $g(x_1, \dots, x_n) \longleftarrow g(x_1, \dots, x_n) + g_i(x_1, \dots, x_n) + |g_i(x_1, \dots, x_n)|$.
**end for**
$L(x_1, \dots, x_n) \longleftarrow f(x_1, \dots, x_n) + \eta.g(x_1, \dots, x_n)$.
**for** $i = 1$ to $n$ **do**
  $x_i = h_i(\theta)$.
**end for**
$\overline{f}(\theta) \longleftarrow L(h_1(\theta), h_2(\theta), \dots, h_n(\theta))$,
$\theta^* \longleftarrow \arg\min \overline{f}(\theta)$.
**for** $i = 1$ to $n$ **do**
  $x_i = h_i(\theta^*)$
**end for**
**Ensure:** Show all $x$ solutions of the problem which is a compromise for the generated $\lambda_k$ weights.
End

**Remark 2.1.**

- *In Algorithm 1, the operation $\theta^* \longleftarrow \arg\min \overline{f}(\theta)$ is performed with the Nelder-Mead algorithm [16].*
- *The complexity of Algorithm 1 is:*

$$T = K.\mathcal{O}(\max\{p^2, m, n, w^2\}), \tag{2.4}$$

*where $p, m, n$ and $w$ are respectively the size of the objective functions, the constraints, of the decision variable and the size of the simplex of the Nelder-Mead algorithm. $K$ is the size of the weights.*

2.3. **Differential Evolution algorithm.**
The principle of the Differential Evolution algorithm is based on an initial population generated on which an evolutionary process is applied which optimizes a problem by iteratively improving a chosen based candidate solution [22]:

(1) A population of $n$ individuals $x_i$ , $i = 1, 2, \ldots, k$, $k \geq 4$ is generated according to the relation:

$$x_i^t = \{x_{1,i}^t, x_{2,i}^t, \ldots, x_{d,i}^t\}, \text{ to the generation } t,\ t \geq 0, \tag{2.5}$$

$x_i^t$ is the chromosome and $d$ is the dimension of the search space.

(2) **The mutation:** For each $x_i$ at the generation $t$, we perform a random choice of three vectors $x_h, x_q$ and $x_r$

$$V_i^{t+1} = x_h^t + F(x_q^t - x_r^t), \tag{2.6}$$

where F is a constant chosen from the interval $[0, 2]$ and $h, q, r \in \{1, 2, \ldots, k\}$.

(3) **Crossing:**
- the first step is to choose a crossover parameter $C_r \in [0, 1]$;
- the crossing is done under two processes which are described as follows:
  - the binomial way performs the crossover operator on each component of the variable according to the $j$-th component of $V_i$:

$$u_{j,i}^{t+1} = \begin{cases} V_{j,i}^t, & r \leq C_r, \\ x_{j,i}^t, & \text{Otherwise}, \end{cases},\ j = 1, 2, \ldots, d, \text{ and } r \in [0, 1]\,. \tag{2.7}$$

  - at the level of the exponential pathway, a part of the donor vector is selected, let $k \in [0, d-1]$ and $L \in [0, d]$ and we have:

$$u_{j,i}^{t+1} = \begin{cases} V_{j,i}^t, & j = k, \ldots, k - L + 1, \\ x_{j,i}^t, & \text{Otherwise}. \end{cases} \tag{2.8}$$

(4) **The selection:** The selection process used is identical to that of the genetic algorithm. It results in the process below:

$$x_i^{t+1} = \begin{cases} u_i^{t+1}, & f(u_i^{t+1}) \leq f(x_i^t) \\ x_i^t, & \text{Otherwise}. \end{cases} \tag{2.9}$$

The algorithm of the Differential Evolution method is defined below:

**Algorithm 2** ALGORITHM OF THE DIFFERENTIAL EVOLUTION METHOD

Begin
**Require:** Population initialization
**Require:** Set constant $F \in [0, 2]$,
**while** stopping criterion not reached **do**
  **for** $i = 1$ to $n$ **do**
    **for** each $x_i$ **do**
      choose randomly $x_p,\ x_q$ and $x_r$;
      Generate a new vector $V$ by the relation (2.6),
      Generate $J_r \in \{1, 2, \ldots, d\}$ for permutation,
      Generate $r_i \in [0; 1]$,
    **end for**
    **for** $j = 1$ to $d$ **do**
      $u_{j,i}^{t+1} = \begin{cases} V_{j,i}^{t+1}, & if\, r_i \leq C_r \text{ and } j = J_r \\ x_{j,i}^{t}, & if\, r_i > C_r \text{ and } j \neq J_r \end{cases};$
    **end for**
    Select and update solution by (2.9).
  **end for**
**end while**
End

2.4. **Performance evaluation of a metaheuristic.**

To evaluate the performance of a metaheuristic, we use performance metrics. The metrics we will use are the convergence and the diversity of the Pareto optimal solutions obtained. Convergence is calculated with the following relation [4]:

$$\frac{\sqrt{\sum_{i=1}^{\Pi} d_i^2}}{\Pi} \tag{2.10}$$

and reflects the gap separating the Pareto front and the analytical front.

Diversity is defined by the following relation [4]:

$$\frac{d_f + d_l + \sum_{i=1}^{\Pi-1} |d_i - \overline{d}|}{d_f + d_l + (\Pi - 1)\overline{d}}. \tag{2.11}$$

It evaluates the distributivity of the solutions on the analytical front.

In the relations (2.10) and (2.11), $\Pi$ denotes the number of solutions provided by our method. $d_f$ and $d_l$ define respectively the Euclidean distances separating the upper and lower extremal solutions provided by our method. $d_i$ is the Euclidean distance between two consecutive solutions, $\overline{d}$ is the arithmetic average of all the solutions provided by our method.

In practice, a good convergence or a good distributivity is the one which converges towards zero.

## 3. Results of hybridization

### 3.1. Principles.

The hybridization consists, from the solution provided by the algorithm of Nelder-Mead in Algorithm 1, to define a simplex of 12 elements in the neighborhood of the solution $\theta^*$.

### 3.2. Algorithme.

The hybridization algorithm is defined by:

---

**Algorithm 3** Hybrid Algorithm

---

Begin
$f(x_1,\dots,x_n) \longleftarrow \mathrm{Tch}(f(x_1,\dots,x_n),\lambda,\overline{z})$
$g(x_1,\dots,x_n) \longleftarrow g_1(x_1,\dots,x_n) + |g_1(x_1,\dots,x_n)|$.
**for** $i = 2$ to $m$ **do**
  $g(x_1,\dots,x_n) \longleftarrow g(x_1,\dots,x_n) + g_i(x_1,\dots,x_n) + |g_i(x_1,\dots,x_n)|$.
**end for**
$L(x_1,\dots,x_n) \longleftarrow f(x_1,\dots,x_n) + \eta * g(x_1,\dots,x_n)$.
**for** $i = 1$ to $n$ **do**
  $x_i = h_i(\theta)$,
**end for**
$\overline{f}(\theta) \longleftarrow L(h_1(\theta), h_2(\theta),\dots,h_n(\theta))$,
$\theta^* \longleftarrow \arg\min \overline{f}(\theta)$,
**for** $i = 1$ to $N$ **do**
  $\theta_i = \theta^* + \Delta$;
**end for**
$\overline{x} = \dfrac{1}{N}\sum_{i=1}^{N}\theta_i$

---

**Algorithm 4** Hybrid Algorithm (continued)

**Require:** Construction of a simplex $\chi$ of 12 elements around $\theta^*$ and $\overline{x}$.
**Require:** Define constant $F \in [0, 2]$;
  **while** stopping criterion not reached **do**
    **for** $i = 1$ to $n$ **do**
      **for** each $x_i$ **do**
        choose randomly $x_p,\ x_q\ et\ x_r,$
        $V_i^{t+1} \longleftarrow \theta^* + F(x_q^t - x_r^t),$
        Generate $J_r \in \{1, 2, \ldots, d\}$ for permutation
        Generate $r_i \in [0, 1]$
      **end for**
      **for** $j = 1$ to $d = |\chi|$ **do**
        $u_j^{t+1} \longleftarrow \begin{cases} V_j^{t+1}, & if\ r_i \leq C_r \text{ and } j = J_r, \\ \theta^* \in \chi, & if\ r_i > C_r \text{ and } j \neq J_r, \end{cases}$
      **end for**
      if $L(u_j^{t+1}) \leq L(\theta^*)$ do
      $x_j^{t+1} \longleftarrow u_j^{t+1},$
      else $x_j^{t+1} \longleftarrow \theta^*,$
      Select and update the solution by relation (2.9).
    **end for**
  **end while**
  **for** $i = 1$ to $n$ **do**
    $x_i = h_i(x_j^{t+1})$
  **end for**
  End

**Theorem 3.1.** *The complexity of Algorithm 3 is:*

$$T_1 = T + K.(\mathcal{O}(n) + \mathcal{O}(n^2) + \mathcal{O}\Big(\frac{l!}{2!(l-2)!}\Big) + \mathcal{O}(N) + \mathcal{O}(1)). \tag{3.1}$$

*Proof.* We will proceed step by step through the different phases of hybridization which are defined as follows:

(1) Definition of the elements in the neighborhood of $\theta^*$ characterized by a discretization:

$$x_i = \theta^* + \Delta i, \quad i = \{1, 2, \ldots, N\}, \tag{3.2}$$

where $\Delta$ is the discretization step and N the size of the defined elements.

(2) We define the center of gravity of all the elements by the relation:

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i. \tag{3.3}$$

This operation has a complexity at worst equal to $\mathcal{O}(N)$.

(3) Subsequently, we define a simplex of 12 elements from the optimum $\theta^*$ and the center of gravity $\overline{x}$. which gives us the set

$$\mathcal{X} = \{\overline{x}, x_a, x_{a_1}, x_{a_2}, x_b, x_{b_1}, x_c, x_{c_1}, x_{c_2}, x_d, x_{d_1}, x_{d_2}, x_{d_3}\}.$$

The elements of this simplex are defined by the following relations:

$$x_a = \theta^* + \rho(x_{best} - x_N) \tag{3.4}$$

$$x_{a_1} = \overline{x} + \rho(\overline{x} - x_N) \tag{3.5}$$

$$x_{a_2} = \overline{x} + \rho(\overline{x} - \theta^*) \tag{3.6}$$

$$x_b = \theta^* + \chi(x_1 - \theta^*) \tag{3.7}$$

$$x_{b_1} = \overline{x} + \chi(x_1 - \theta^*) \tag{3.8}$$

$$x_c = \theta^* + \gamma(\theta^* - x_N) \tag{3.9}$$

$$x_{c_1} = \overline{x} + \gamma(\overline{x} - \theta^*) \tag{3.10}$$

$$x_{c_2} = \overline{x} + \gamma(\overline{x} - x_1) \tag{3.11}$$

$$x_d = \overline{x} + \rho(\theta^* - x_1) \tag{3.12}$$

$$x_{d_1} = \overline{x} + \chi(\theta^* - x_1) \tag{3.13}$$

$$x_{d_2} = \overline{x} + \chi(\overline{x} - x_1) \tag{3.14}$$

$$x_{d_3} = \overline{x} + \sigma(\overline{x} - x_N). \tag{3.15}$$

The creation of this simplex is of constant complexity, since the number of solutions does not vary.

(4) The random choice of the values $x_q$ and $x_r$ defined in the relation (2.6) will be made in the set $\mathcal{X}$. The complexity at worst of this operation is a combination of 2 in $l$ defined by the relation $\dfrac{l!}{2!(l-2)!}$, where $l = card(\mathcal{X})$.

(5) Application of crossover and mutation operators based on conditions stated in the description of the stages 3 of the Differential Evolution method to the selected individual. The mutation operator that we are going to use is of the binomial type (2.7) because of its simplicity of implementation. It is defined by:

$$V_i^{t+1} = \theta^* + F(x_q^t - x_r^t),\ F \in [0,2]\,. \tag{3.16}$$

Complexity at worst is $\mathcal{O}(N)$.

(6) Determining the best $\theta^*$ and using of the dominance in the sense of Pareto between the best solution and $\theta^*$, to find the optimum, after having transformed them into an element of $\mathbb{R}^n$ is an operation of complexity at the worst $\mathcal{O}(n^2)$.

□

## 4. Numerical results

### 4.1. **Test problems.**

Our hybridization has been tested on Zitzler test problems below for a simulation.

Table 1: Multi-objective test problems

| index | Multiobjective problems | n | bounds |
|---|---|---|---|
| $T_1$ | $\begin{cases} \min f_1(x_1,x_2) = x_1, \\ \min f_2(x_1,x_2) = \dfrac{1+x_2}{x_1}, \\ 0.1 \le x_1 \le 1, \\ 0 \le x_2 \le 5. \end{cases}$ | 2 | $x_1, x_2 \in [0,1]\,.$ |

| | | | |
|---|---|---|---|
| $T_2$ | $\begin{cases} \min f_1(x) = x^2, \\ \min f_2(x) = (x-2)^2, \\ -5 \leq x \leq 5. \end{cases}$ | 1 | $x \in [0,4]$ . |
| $T_3$ | $\begin{cases} \min f_1(x) = x_1, \\ \min f_2(x) = g(x).\left(1 - \sqrt{\frac{f_1(x)}{g(x)}}\right), \\ g(x) = 1 + \dfrac{9}{n-1}.\sum\limits_{i=2}^{n} x_i, \\ x = (x_1, x_2, \ldots, x_n) \in [0,1]^n. \end{cases}$ | 30 | $x_i \in [0,1]$ . |
| $T_4$ | $\begin{cases} \min f_1(x) = x_1, \\ \min f_2(x) = g(x).\left(1 - \left(\dfrac{f_1(x)}{g(x)}\right)^2\right), \\ g(x) = 1 + \dfrac{9}{n-1} \times \sum\limits_{i=2}^{n} x_i, \\ x = (x_1, x_2, \ldots, x_n) \in [0,1]^n. \end{cases}$ | 30 | $x_i \in [0,1]$ . |
| $T_5$ | $\begin{cases} \min f_1(x) = x_1, \\ \min f_2(x) = g(x).h(x), \\ g(x) = 1 + \dfrac{9}{n-1}.\sum\limits_{i=2}^{n} x_i, \\ h(x) = 1 - \sqrt{\dfrac{f_1(x)}{g(x)}} - \dfrac{f_1(x)}{g(x)} \sin(10\pi f_1(x)), \\ x = (x_1, x_2, \ldots, x_n) \in [0,1]^n \end{cases}$ | 30 | $x_i \in [0,1]$ . |
| $T_6$ | $\begin{cases} \min f_1(x) = x_1, \\ \min f_2(x) = g(x).\sqrt{1 - \dfrac{f_1(x)}{g(x)}}, \\ g(x) = 1 + \dfrac{9}{n-1}.\sum\limits_{i=2}^{n} x_i, \\ x = (x_1, x_2, \ldots, x_n) \in [0,1]^n. \end{cases}$ | 30 | $x_i \in [0,1]$ . |

## 4.2. Pareto fronts with hybridization.

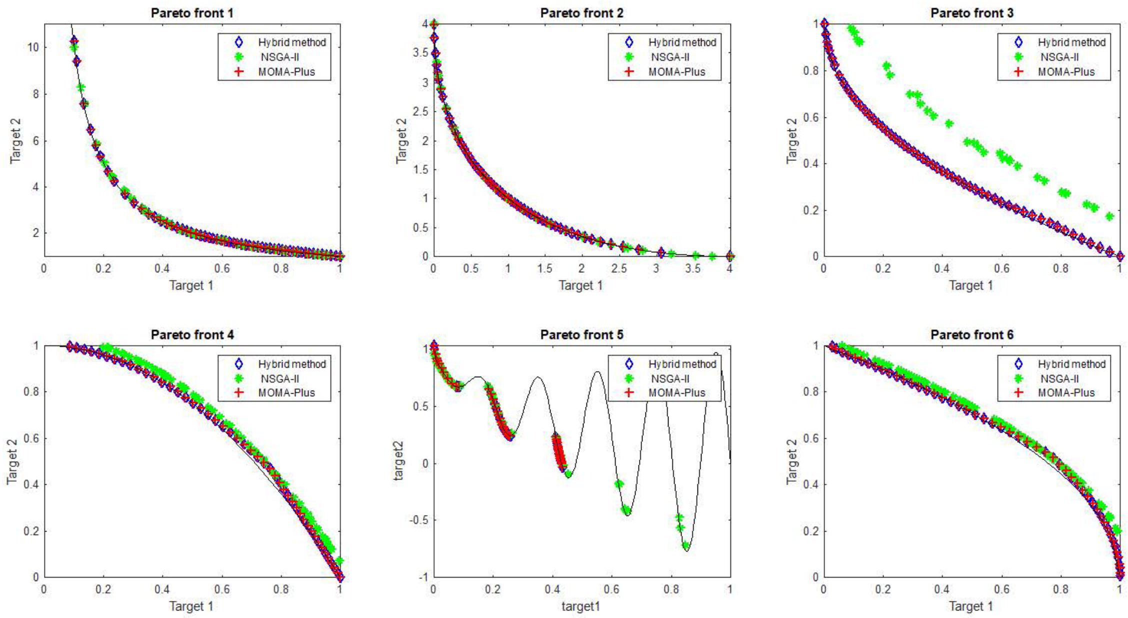


FIGURE 1. Pareto fronts from simulations

## 4.3. Performance results.

TABLE 2. Hybridization convergence index

| Convergence | $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ |
|---|---|---|---|---|---|---|
| $Hybrid$ | 0.0281 | 0.0011 | 0.0014 | 0.0018 | 0.0029 | 0.0018 |
| MOMA-Plus | 0.0691 | 0.0053 | 0.0137 | 0.0042 | 0.0599 | 0.0046 |
| NSGA-II | 0.0324 | 0.0056 | 0.0175 | 0.0025 | 0.0124 | 0.0038 |

We note that the values of the hybridization convergence index are better than those provided by the two methods MOMA-plus and NSGA-II. Consequently, these converge better than the last two methods.

TABLE 3. Hybridization diversity index

| Diversity | $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ |
|---|---|---|---|---|---|---|
| $Hybrid$ | 0.9819 | 0.9846 | 0.9825 | 0.9823 | 0.9824 | 0.9822 |
| MOMA-Plus | 1.1833 | 0.5537 | 0.3498 | 0.0309 | 0.9835 | 0.9820 |
| NSGA-II | 0.0290 | 0.0183 | 1.0023 | 0.0319 | 0.9710 | 0.0201 |

Rankings of methods on each test problem at the convergence level and diversity are recorded in the tables below: this classification is based on the value of the convergence and diversity metrics:

TABLE 4. Rank on convergence

| Convergence | $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ |
|---|---|---|---|---|---|---|
| $Hybrid$ | 1 | 1 | 1 | 1 | 1 | 1 |
| MOMA-Plus | 4 | 4 | 3 | 4 | 4 | 4 |
| NSGA-II | 3 | 3 | 4 | 3 | 3 | 3 |

TABLE 5. Diversity Rank

| Diversity | $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ |
|---|---|---|---|---|---|---|
| $Hybrid$ | 2 | 3 | 2 | 3 | 3 | 3 |
| MOMA-Plus | 4 | 2 | 1 | 1 | 2 | 2 |
| NSGA-II | 1 | 1 | 4 | 2 | 1 | 1 |

The sum of the rank values over all test problems for each metric is recorded in the table below

TABLE 6. Joint ranking

| | $Hybrid$ | Moma-Plus | NSGA-II |
|---|---|---|---|
| Convergence | 6 | 22 | 20 |
| Diversity | 16 | 12 | 10 |

4.4. **Comparison of different methods and discussions.**
Following these results, the performance profiles of the methods give us the following figures:

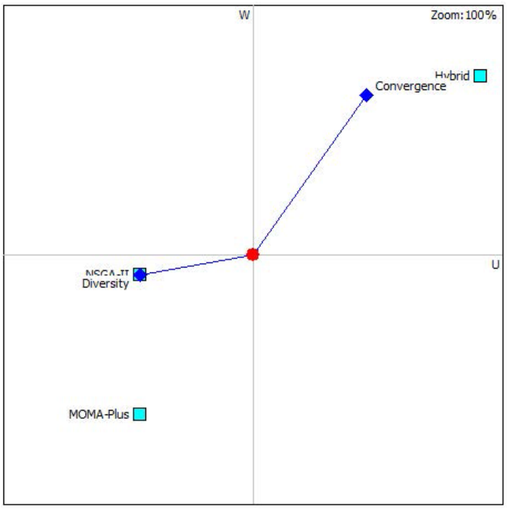


FIGURE 2. Performance analysis using the GAIA method

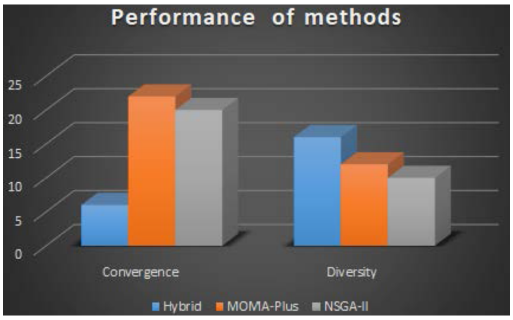


FIGURE 3. Performance histogram

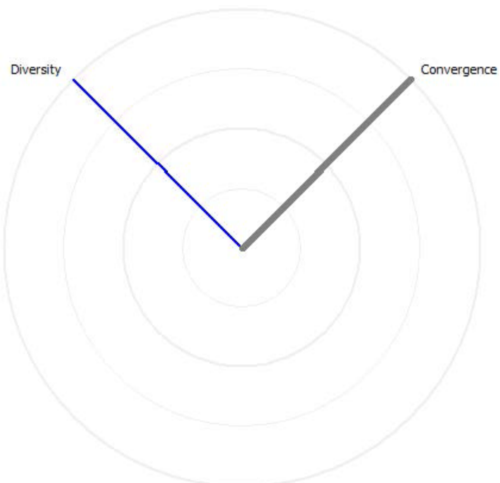


FIGURE 4. Profile of hybrid methods

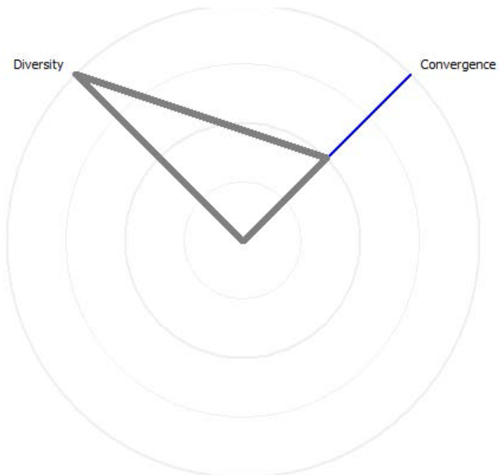


FIGURE 5. NSGA-II method profile

The figures (2), (3), (4) and (5) reflect the profiles of the different methods. The two hybrid methods having the same performance indices on the convergence axis pointing to their positions in the figure (2).This would mean that the two hybrid methods converge better than the MOMA-Plus and NSGA-II methods. The NSGA-II method is the one that has a good diversity according to the figures. Therefore, it has a better distribution of solutions on the Pareto front. This interpretation emerges from the histogram represented by the figure (3), the hybrid method converges better than the other methods. The NSGA-II method has the best diversity. The figure (4) shows that hybrid methods, in addition to being efficient on the convergence of solutions on the Pareto front, have an acceptable indicator on the diversity of solutions on the Pareto front. It is the same for the NSGA-II method for the figure (5) which has an acceptable profile on the convergence of the Pareto solutions on the front.

**Remark 4.1.**
*The main steps that make up this hybridization suggest the possibility of defining another approach in which the algorithm will already be launched from the domain* $[0, \theta_{max}]$ *instead of creating a simplex and the same operators that were used in the first approach are still used. The solution resulting from the operations are compared with* $\theta^*$ *in the sense of Pareto dominance. It is:*

(i) *create an initial population in the interval* $[0, \theta_{max}]$ *following a uniform distribution. This operation has complexity* $\mathcal{O}(k)$*, where* $k$ *is the size of the chosen population;*
(ii) *rank this population, which will find a minimum of the function* $L(\theta)$*. The complexity of this operation is of the order of* $\mathcal{O}(k^2)$*;*
(iii) *apply the crossover and mutation operators for the search for the optimum. To improve the results, the mutation and the crossing will be done from* $\theta^*$ *and the minimum defined by the previous step;*
(iv) *the results of these operations will be compared on the basis of Pareto dominance from the solution* $\theta^*$*, the optimum given by the MOMAPlus method. This operation is of constant complexity.*

*The algorithmic complexity of this approach is defined by the relation:*

$$T_2 = T + K.(\mathcal{O}(k) + \mathcal{O}(k^2) + \mathcal{O}(N)).$$

*The complexities of the two approaches display a satisfactory result on the decidability of these two approaches, the complexities are all polynomial.*

## 5. Conclusion

The study of this hybridization reflects the possibility of combining the MOMA-plus method with a metaheuristic represented by the Differential Evolution method. The results of this work open the field for the development of new metaheuristics for solving optimization problems in general. The results of this hybridization are acceptable, in particular according to the values of the performance indices, because of their convergence towards zero and also in comparison with the MOMA-Plus and NSGA-II methods. So, this work opens up perspectives and fields of work in the direction of alleviating the complexity in the system of recognized and recent methods for the search for optimal solutions.

## References

[1] M.F. Ahmad, N.A.M. Isa, W.H. Lim, K.M. Ang: *Differential Evolution. A recent review based on state-of-the-art work*, Alexandra Engineering Journal, **61** (2022), 3831–3872.

[2] K. Price, R.M. Storn, J.A. Lampinen: *Differential evolution: a practical approach to global optimization,* Springer Science and Business Media, 2006.

[3] A.K. Qin, V.L. Huang, P.N. Suganthan: *Differential evolution algorithm with strategy adaptation for global numerical optimization*, IEEE Trans. Evol. Comput. **13**(2) (2008), 398–417.

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

[5] P. Siarry: *Metaheuristics,* Springer International Publishing Switzerland, 2016.

[6] A. Kaveh; T. Bakhshpoori: *Metaheuristics: outlines, Matlab codes and examples,* Springer, 2019.

[7] S. Das, P.N. Suganthan: *Differential evolution: A survey of the state-of-the-art*, IEEE Trans. Evol. Comput. **15**(1) (2010), 4–31,

[8] M. Pant, H. Zaheer, L. Garcia-Hernandez, A. Abraham: *Differential Evolution: A review of more than two decades of research*, Eng. Appl. Artif. Intell. **90** (2020), art. no. 103479.

[9] A. Compaoré, K. Somé, B. Somé: *New approach to the resolution of triangular fuzzy linear programs : MOMA-plus method,* International Journal of Applied Mathematical Research, **6**(4) (2017) 115–120.

[10] A. COMPAORÉ, K. SOMÉ, J. PODA AND B. SOMÉ: *Efficiency of MOMA-plus Method to Solve Some Fully Fuzzy L-R Triangular Multiobjective Linear Programs*, Journal of Mathematics Research, **10**(2) (2018) 77–87.

[11] A. SOM, K. SOMÉ, A. COMPAORÉ, B. SOMÉ: *Performances assessment of MOMA-Plus method on multiobjective optimization problems*, European Journal of Pure and Applied Mathematics, **13**(1) (2020), 48–68.

[12] A. SOM, K. SOMÉ, A. COMPAORÉ AND B. SOMÉ: *Effect of aggregation function in MOMA-Plus method for obtainig pareto optimal solution*, Applications and Applied Mathematics: An International Journal, **17**(2) (2022), 536–557.

[13] A. SOM, K. SOMÉ, A. COMPAORÉ, B. SOMÉ: *Exponential penalty function with MOMA-plus for the multiobjective optimization problems*, Applied Analysis and Optimization, **5**(3) (2021), 323–334.

[14] A. SOM, K. SOMÉ, A. COMPAORÉ: *Hybrid Method Based On Exponential Penalty Function And Moma-Plus Method For Multiobjective Optimization*, Journal of Computer Science and Applied Mathematics, **5**(2) (2023), 35–51.

[15] B. O. KONFÉ, Y. CHERRUAULT, B. SOMÉ, T. BENNEOUALA: *Alienor method applied to operational research*, Kybernetes, **34** (2005), 1211–1222.

[16] J. NELDER, R. MEAD: *A simplex method for function minimization*, Computer Journal, **7**(4) (1965), 308–313.

[17] J. PODA, K. SOMÉ, A. COMPAORÉ, B. SOMÉ: *Adaptation of the MOMA-Plus method to the resolution of transportation and assignment problems*, Journal of Mathematical Sciences, **22** (2018), 25–44.

[18] J. PODA, K. SOMÉ, A. COMPAORE, B. SOMÉ: *Mobile secondary ideal point and MOMA-plus method in two-phase method for solving bi-objective assigment problems*, Universal Journal of Mathematics and Mathematical Sciences, **12**(1) (2019), 31–52.

[19] K. MIETTINEN: *Nonlinear multiobjective optimization*, Kluwer Accademic Publishers, 1999.

[20] K. SOMÉ, A. COMPAORÉ, W. O. SAWADOGO; B. ULUNGU, B. SOMÉ: *Improving of numerical performance of the MOMA method*, International Journal of Mathematics and Computer Resarch, **5**(5) (2017), 1817–1827.

[21] A. ZOUNGRANA, K. SOMÉ, J. PODA: *Obtaining Optimal Pareto Solutions Using a Hybrid Approach Combining E-Constraint and MOMA-Plus Method*, International Journal of Numerical Methods and Applications, **23**(1) (2023), 107–129.

[22] R. STORN, K. PRICE: *Differential Evolution: a simple and efficient heuristic for global optimization over continuous spaces*, Journal of Global Optimization, **11** (1997), 341–59.

[23] Y. CHERRUAULT: *A new method for global optimization (Alienor)*, Kybernetes, **19**(3) (1989), 19–32.

[24] Y. COLLETTE, P. SIARRY: *Optimisation multiobjectif*, édition Eyrolles, 61, Bld Saint-Germain 75240 Paris Cedex 05, 2002.

[25] M. EHRGOTT: *Multicriteria Optimization*, Second Edition, Springer Berlin. Heidelberg, 2005.

DÉPARTEMENT DE MATHÉMATIQUES, LABORATOIRE DE MATHÉMATIQUES, INFORMATIQUE ET APPLICATIONS
UNIVERSITÉ NORBERT ZONGO
BP 376 KOUDOUGOU,
BURKINA FASO.
*Email address*: ab.compaore1@gmail.com

DÉPARTEMENT DE MATHÉMATIQUES, LABORATOIRE D'ANALYSE NUMÉRIQUE, D'INFORMATIQUE ET DE BIOMATHÉMATIQUES
UNIVERSITÉ JOSEPH KI-ZERBO
03 BP 7021 OUAGADOUGOU,
BURKINA FASO.
*Email address*: irenotesom89@gmail.com

DÉPARTEMENT DE MATHÉMATIQUES, LABORATOIRE DE MATHÉMATIQUES, INFORMATIQUE ET APPLICATIONS
UNIVERSITÉ NORBERT ZONGO
BP 376 KOUDOUGOU,
BURKINA FASO.
*Email address*: sokous11@gmail.com