

HYPERBOLIC AUGMENTED LAGRANGIAN ALGORITHM FOR MULTIOBJECTIVE OPTIMIZATION PROBLEMS

APPOLINAIRE TOUGMA^{1*}, AUGUSTIN KABORÉ² AND KOUNHINIR SOMÉ³

ABSTRACT. In this article, we propose an adaptation of the Hyperbolic Augmented Lagrangian algorithm to solve multiobjective optimization problems. After conducting a literature review, we provide a definition of the Hyperbolic Augmented Lagrange method. Under clearly defined assumptions, we demonstrate that any limit point of a sequence generated by the proposed approach is feasible and constitutes a Pareto-optimal solution. To assess the effectiveness of the proposed algorithm, we incorporate two variants of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method, thereby offering two approaches for solving well-known test problems from the literature. We employ metrics to evaluate the performance of these two approaches.

1. INTRODUCTION

In multiobjective optimization, the objective is to optimize multiple objectives simultaneously. Thus, there is not a single solution that can satisfy all conflicting objective functions simultaneously. In this case, the concept of Pareto optimality is used, demonstrating an optimal solution to a multiobjective problem. In recent years, multiobjective optimization concepts have proven valuable for solving a wide variety of problems in various domains, including physics, economics, transportation, social choice, and many others [27, 38, 39].

Numerous methods have been proposed in the literature to address optimization problems. These include approximation techniques such as heuristics and/or metaheuristics [17], as well as exact methods [15, 19, 20, 21, 23, 24, 29]. It's worth noting that the majority of these methods are geared towards solving unconstrained problems.

Among the exact methods, we can mention the quasi-Newton method, widely used for solving optimization problems, whether they are single-objective or multiobjective without constraints. In this method, the search direction is determined from a quadratic model of the objective function, where certain approximations replace the true Hessian at each iteration. Several schemes for approximating the Hessian, coupled with an appropriate linear search technique, have improved the accuracy of curvature approximations, making the methods more effective in

Date: Received: Jan 23, 2024; Accepted: Mar 15, 2024.

* Corresponding author.

2020 *Mathematics Subject Classification.* 49M37, 65K05, 90C29.

Key words and phrases. multiobjective quasi-Newton method, penalty function, Pareto stationary point, Pareto front, BFGS method.

terms of convergence towards the solution. For more information, readers are recommended to refer to the following references [10, 30, 31, 32, 33, 34], and the references therein.

Given that real-world optimization problems often involve constraints, research efforts have been dedicated to solving multiobjective optimization problems under constraints. The main objective of such research is to develop methods capable of effectively handling nonlinear optimization problems with constraints. Faced with the challenges posed by constraints, many methods use penalty functions to integrate objective functions and constraints. This transformation allows us to convert the original problem into an unconstrained optimization problem.

Among these approaches, the Augmented Lagrangian method has emerged as a promising technique for solving constrained optimization problems. It effectively deals with constraints through an iterative process, improving the accuracy of the obtained solutions. For more information, readers can refer to the following references and those references therein [1, 2, 3, 4, 5, 6, 7, 11, 12, 22, 25, 41].

In 2001, Xavier et al. [42] proposed a hyperbolic penalty function for solving optimization problems under inequality constraints. In 2013, Costa, M. Fernanda et al [13] drew inspiration from this method, introducing a hyperbolic augmented Lagrangian function for solving single-objective optimization problems. They combined a heuristic algorithm (Artificial Fish Swarm (AFS) algorithm) to solve the sub-problem of hyperbolic augmented Lagrangian. Another version of this approach was proposed by the same authors in 2016 [37], also incorporating the AFS algorithm. In 2022, Lennin Mallma Ramirez [35, 36] studied another version of a hyperbolic augmented Lagrange algorithm for solving single-objective optimization problems under inequality constraints as part of his thesis.

In this work, we propose a hyperbolic augmented Lagrange method for solving multiobjective optimization problems. The main contributions and highlights of the article are as follows:

- development of a new augmented Lagrange algorithm for addressing multiobjective problems.
- proposal of a convergence study for the algorithm,
- comparative study of the proposed algorithm against variants of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) method for solving the sub-problem of the proposed algorithm.

In the remainder of this article, we provide preliminaries in Section 2, which includes a literature review on the definitions and basic properties of multiobjective optimization. In Section 3, we introduce the Hyperbolic Augmented Lagrange method for solving multiobjective optimization problems, along with the main algorithm named MOHALA. In Section 4, we present the theoretical convergence results of MOHALA. In Section 5, we demonstrate the application of MOHALA to structural problems, followed by a numerical convergence study. Finally, in Section 6, we conclude with remarks on future research directions.

2. PRELIMINARIES

In this work, we consider multiobjective optimization problems defined as follows:

$$\begin{aligned} \min \quad & F(x) = (f_1(x), f_2(x), \dots, f_q(x)) \\ \text{s.t.} \quad & \begin{cases} g_i(x) \leq 0 & \forall i \in I = \{1, \dots, p\} \\ x \in \mathbb{R}^n, \end{cases} \end{aligned} \quad (2.1)$$

In this context, $F : \mathbb{R}^n \rightarrow \mathbb{R}^q$ represents a vector function with components $f_1(x), f_2(x), \dots, f_p(x)$, and $g : \mathbb{R}^n \rightarrow \mathbb{R}^p$ is another vector function with components indexed by $I = \{1, 2, \dots, p\}$. The feasible space $\mathcal{X}$ of problem (2.1) is defined as $\mathcal{X} = \{x \in \mathbb{R}^n : g(x) \leq 0\}$. Throughout our study, we adopt the following conventions: $\mathbb{R}_{++}$ denotes the set of positive real numbers, $\mathbb{R}^n$ denotes the set of column vectors of dimension n , $\text{Im}(A)$ represents the image space of a matrix $A \in \mathbb{R}^{q \times n}$, and $\|\cdot\|$ denotes the Euclidean norm. The unit vector of dimension q is denoted by e .

For any vectors $v = (v_1, v_2, \dots, v_n)^\top$ and $\omega = (\omega_1, \omega_2, \dots, \omega_n)^\top$, we define the following conventions for inequalities:

- (i) $v < \omega \Leftrightarrow v_i < \omega_i$ for all $i = 1, 2, \dots, n$
- (ii) $v \leq \omega \Leftrightarrow v_i \leq \omega_i$ for all $i = 1, 2, \dots, n$

Definition 2.1 ([28]). A point $x^* \in \mathcal{X}$ is called Pareto optimal for problem (2.1) if there does not exist another point $x \in \mathcal{X}$ such that:

$$F(x) \leq F(x^*) \text{ and } F(x) \neq F(x^*).$$

We present the following definition, which proposes more relaxed conditions that are practical to achieve.

Definition 2.2 ([28]). A point $x^* \in \mathcal{X}$ is called Weakly Pareto optimal for problem (2.1) if there does not exist another point $x \in \mathcal{X}$ such that:

$$F(x) < F(x^*). \quad (2.2)$$

Taking into account these two definitions, we present the following lemma extracted from [11], which establishes equivalences in the Pareto optimality definition.

Lemma 2.3. A point $x^* \in \mathcal{X}$ is said to be:

(a): a Pareto optimum for (2.1) if and only if, for all $y \in \mathcal{X}$, at least one of the following relations is satisfied:

- (i): $\max_{j=1,2,\dots,q} \left\{ f_j(y) - f_j(x^*) \right\} > 0,$
- (ii): $\min_{j=1,2,\dots,q} \left\{ f_j(y) - f_j(x^*) \right\} \geq 0.$

(b): a weak Pareto optimum for (2.1) if and only if, for all $y \in \mathcal{X}$, we have

$$\max_{j=1,2,\dots,q} \left\{ f_j(y) - f_j(x^*) \right\} \geq 0$$

We also define $x^* \in \mathcal{X}$ as a local Pareto point (or weak local Pareto point) if there exists a neighborhood $V(x^*) \in \mathcal{X}$ such that x^* is a Pareto optimal point (or a weak Pareto optimal point) for F restricted to $V(x^*)$. In this context, we use a partial order induced by $\mathbb{R}_{++}$, where $F(x) < F(x^*) \Leftrightarrow F(x) - F(x^*) \in -(\mathbb{R}_{++}^q)$.

The following relation expresses a necessary, but generally not sufficient, condition for weak Pareto optimality.

$$-(\mathbb{R}_{++}) \cap \text{Im}(JF(x^*)) = \emptyset, \quad (2.3)$$

where $JF(\cdot)$ represents the Jacobian matrix of F . A point $x^* \in \mathcal{X}$ is considered stationary for F if it satisfies the relation (2.3). Subsequently, a Pareto optimality condition is formulated in the following definition.

Definition 2.4 ([19]). A point $x^* \in \mathcal{X}$ is considered a Pareto-stationary point for problem (2.1) if, for any

$$d \in \{d \in \mathbb{R}^n \mid \exists \bar{t} > 0 : x^* + td \in \mathcal{X} \forall t \in [0, \bar{t}]\},$$

the following inequality is satisfied:

$$\max_{j=1;q} \nabla f_j(x^*)^\top d \geq 0. \quad (2.4)$$

Note that if x^* is not a Pareto-stationary point, there exists an admissible direction d such that $\max_{j=1;q} \nabla f_j(x^*)^\top d < 0$.

3. HYPERBOLIQUE AUGMENTED LAGRANGIAN FUNCTION FOR MULTIOBJECTIVE OPTIMIZATION PROBLEMS

In this section, we discuss the implementation of the augmented hyperbolic Lagrangian method to solve multiobjective optimization problems.

3.1. Principe. The fundamental concept of the Hyperbolic Augmented Lagrangian method is to penalize the inequality constraints of a constrained optimization problem. It is imperative to emphasize that this approach does not convert the multiobjective problem into a single-objective problem, thereby enabling us to directly address Pareto points during each iteration. Considering the problem (2.1), the main idea of the Hyperbolic Augmented Lagrangian method is to add a penalty term to each objective function given by the function

$$\mathcal{P}(x, \delta^k, \tau^k, \mu^k) = \left[\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x) + \sqrt{(g_i(x))^2 + (\mu^k)^2} \right) \right]. \quad (3.1)$$

The behavior of the function $\mathcal{P}(x, \delta^k, \tau^k, \mu^k)$ can be observed in Figure 1. It is noted that for all $i = \{1, \dots, p\}$, when $g_i(x) \rightarrow -\infty$, the penalty function P approaches the line $d(g_i(x)) = 0$ (the horizontal asymptote), and when $g_i(x) \rightarrow +\infty$, the penalty function P approaches the curve $d(g_i(x)) = \frac{1}{2} \delta_i^k [g_i(x)]_+^2 + 2\tau g_i(x)$.

The formulation (2.1) can be reformulated using equation (3.1) as follows:

$$\min_{x \in \mathbb{R}^n} \mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k) \quad (3.2)$$

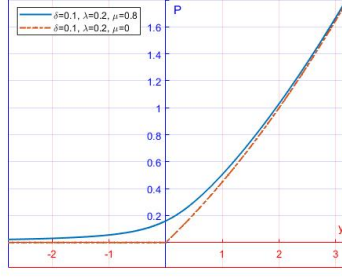


FIGURE 1. Hyperbolic Penalty

where

$$\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k) = F(x) + \left[\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x) + \sqrt{(g_i(x))^2 + (\mu^k)^2} \right) \right] \cdot e, \quad (3.3)$$

with $[g_i(x)]_+ = \max\{0, g_i(x)\}$, $\delta = (\delta_1, \dots, \delta_p)$ represents the Lagrange multiplier associated with the inequality constraints $g_i(x) \leq 0$, τ and μ represent penalty parameters with different roles: τ is the classical increasing penalty parameter, and μ , decreasing, aims to improve the accuracy of the approximation. The crucial aspect that distinguishes HALM scalar lies in the update phase of x^k and the first term of the penalty function.

The Lagrange multiplier update paradigm is used following the principle given in [13], given by the following relation for all $i = \{1, \dots, p\}$

$$\delta_i^{k+1} = \begin{cases} \min \left\{ \delta_i^k + \tau^k \left(1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + (\mu^k)^2}} \right), \delta^+ \right\} & \text{if } g_i(x) > 0, \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

Admissibility and complementarity at iteration k are evaluated using the relation $\|V^{k+1}\| \leq \nu \|V^k\|$, where $V_k = \min\{-g_i(x^k), \delta_i^{k+1}\}$ with δ_i being the multiplier corresponding to g_i calculated at x^k and ν a constant in the interval $[0, 1)$.

3.2. Algorithm. The algorithm for the Hyperbolic Augmented Lagrangian method is presented as follows:

Algorithm 1: Hyperbolic Augmented Lagrangian Algorithm for Multi-objective Optimization Problems.

Data: $\tau^1 > 0; \sigma > 1, \mu^1 \geq 0; \zeta < 1/\sigma; \nu \in [0; 1); \delta^+ > 0; \delta_i^1 \in [0; \delta^+];$
 $x_1 \in \mathbb{R}^n$

```

1 for  $k = 1, 2, 3, \dots$  do
2   Find  $x^k \in \mathbb{R}^n$  a  $\epsilon_k$ -Pareto-Point of

       
$$\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k) = F(x) + \left[ \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x)]_+^2 + \right. \\
       \left. \tau^k \sum_{i=1}^p \left( g_i(x) + \sqrt{(g_i(x))^2 + (\mu^k)^2} \right) \right] \cdot e, \quad (3.5)$$


3   for  $i = 1, 2, 3, \dots, p$  do
4     if  $g_i(x^k) > 0$  then
5       Set  $\delta_i^{k+1} = \min \left\{ \delta_i^k + \tau^k \left( 1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + (\mu^k)^2}} \right), \delta^+ \right\}$ 
6     else
7       Set  $\delta_i^{k+1} = 0$ 
8     Set  $V_i^k = \min \left\{ -g_i(x^k), \delta_i^{k+1} \right\}$ 
9   Set  $\mu^{k+1} = \zeta \cdot \mu^k$ 
10  if  $\|V^k\| \leq \nu \|V^{k-1}\|$  then
11    Set  $\tau_{k+1} = \tau_k$ 
12  else
13    Set  $\tau_{k+1} = \sigma \tau_k$ 

```

The algorithm begins by entering the initial parameters and an initial point x^0 . In the first "for" loop at step 2, the algorithm starts by searching for a point x^k ϵ_k -Pareto by solving the function (3.3). Once the point x^k is determined, the update of the Lagrange multipliers δ_i associated with the constraints g_i for $i = \{1, \dots, p\}$ is performed from step 3 to step 7. The update of the decreasing penalty parameter μ^{k+1} is carried out at step 9, and the increasing one (τ^{k+1}) at steps 11 to 13, following the relation $\|V^k\| \leq \nu \|V^{k-1}\|$, verifying admissibility and complementarity at each iteration k .

In the next section, we will present the theoretical results regarding the admissibility and optimality of the sequences generated by Algorithm 1.

4. CONVERGENCE ANALYSIS

Before starting the analysis, we present the following assumptions.

Assumption 1. The objective function F exhibits compact level sets in a multi-objective context. This means that for every $z \in \mathbb{R}^q$, the set $T_0 = \{x \in \mathbb{R}^n \mid F(x) \leq z\}$ is compact.

Assumption 2. The functions f_j , $i = 1, \dots, p$, and g_i , $i = 1, \dots, m$, are continuous in a neighborhood of x^* .

Assumption 3. The sequence $\{x^k\}$ generated by Algorithm 1 is well-defined, and there exists a set of indices $K \subseteq \mathbb{N}$ such that $\lim_{k \in K} x^k = x^*$.

Assumption 4. The sequence $\{\epsilon_k\}$ satisfies $\lim_{k \rightarrow \infty} \epsilon_k = 0$ for all $k \in K$.

Assumption 5. $\{\tau^k \mu^k\}$ is a bounded and monotonically decreasing sequence of non-negative real numbers.

We now begin the examination of the convergence of Algorithm 1 with the following Proposition, demonstrating that at each iteration k of Algorithm 1, the Lagrange multipliers associated with the inequality constraints are positive.

Proposition 4.1. Let $\delta^k = (\delta_1^k, \dots, \delta_p^k) \in \mathbb{R}^p$ be the Lagrange multiplier associated with the inequality constraints $g(x) = (g_1(x), \dots, g_p(x))$ generated by Algorithm 1 at each iteration k . If $\delta^k \in \mathbb{R}_{++}^p$ then $\delta^{k+1} \in \mathbb{R}_{++}^p$.

Proof. Suppose $\mu > 0$ is fixed. Since we have $0 < \mu^2$, we can obtain for all $i = \{1, \dots, p\}$ the following:

$$(g_i(x^k))^2 < (g_i(x^k))^2 + \mu^2 \quad (4.1)$$

$$\Rightarrow |g_i(x^k)| < \sqrt{(g_i(x^k))^2 + \mu^2} \quad (4.2)$$

$$\Rightarrow -\sqrt{(g_i(x^k))^2 + \mu^2} < g_i(x^k) < \sqrt{(g_i(x^k))^2 + \mu^2} \quad (4.3)$$

$$\Rightarrow -1 < \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + \mu^2}} < 1 \quad (4.4)$$

$$\Rightarrow 0 < 1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + \mu^2}} < 2 \quad (4.5)$$

$$\Rightarrow 0 < \tau^k \left(1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + \mu^2}} \right) < 2\tau^k \quad (4.6)$$

$$\Rightarrow \delta_i^k < \delta_i^k + \tau^k \left(1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + \mu^2}} \right) < 2\tau^k + \delta_i^k \quad (4.7)$$

since $\delta_i^k > 0$ by assumption for all $i = 1, 2, \dots, p$ and according to equation (3.4), and the definition of Algorithm 1, $\delta_i^{k+1} > 0$ for $g_i(x^k) > 0$ and $\delta_i^{k+1} = 0$ for $g_i(x^k) < 0$, $i = 1, 2, \dots, p$. $\square$

The following theorem shows that every sequence generated by Algorithm 1 has a limiting point that is admissible.

Theorem 4.2 (Admissibility). *Suppose that hypotheses 1 to 4 are satisfied. Let x^* be an accumulation point of the sequence $\{x^k\}$ generated by Algorithm 1. Then, x^* satisfies the admissibility condition for problem (2.1), i.e., $g(x^*) \leq 0$.*

Proof. Since $\mathbb{R}^n$ is a closed set and $x^k \in \mathbb{R}^n$, we have $x^* \in \mathbb{R}^n$. We consider two cases regarding the penalty parameter sequence τ_k :

- (a): the sequence $\{\tau^k\}$ is bounded,
- (b): the sequence $\{\tau^k\}$ is unbounded.

In case (a), according to the definition of the penalty parameter sequence τ^k , there exists a positive integer k_0 such that for every $k \geq k_0$, we have $\tau^k \rightarrow \tau^{k_0}$, and the condition $\|V^k\| \leq \nu \|V^{k-1}\|$ is satisfied. This implies that either $-g_i(x^k) \rightarrow 0$ or $\delta_i^k \rightarrow 0$ with $g_i(x^k) < 0$ for all $i = \overline{1, p}$. Consequently, $\max \left\{ 0, g_i(x^k) \right\} \rightarrow 0$ for all $i = \overline{1, p}$, and we conclude that x^* is admissible.

In case (b), we assume that $\tau^k \rightarrow +\infty$ and δ_i^k is bounded. Suppose, by contradiction, that every limit point x^* of a sequence $\{x^k\}$ generated by Algorithm 1 is not admissible, and there exists a Pareto optimum $y \in \mathbb{R}^n$ such that $g_i(y) \leq 0$. Therefore, $g_i(x) > 0 \geq g_i(y)$ or $\sum_{i=1}^p \delta_i^k [g_i(x^*)]_+ > \sum_{i=1}^p \delta_i^k [g_i(y)]_+$.

According to Lemma 2.3,

$$\min_{j=1,2,\dots,q} \left\{ f_j(x^k) + \left[\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) \right] - f_j(y) - \left[\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 - \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \right] \right\} \leq 0. \quad (4.8)$$

Since the functions g are continuous, there exists a constant $c > 0$, and for $k \in K$ sufficiently large, we have

$$\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 > \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 + c \quad (4.9)$$

and

$$\begin{aligned} & \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) \\ & > \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) + c. \end{aligned} \quad (4.10)$$

By equation (3.3), we have for all $j = \overline{1, q}$,

$$\begin{aligned}
& f_j(x^k) + \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) > f_j(y) \\
& + \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) + f_j(x^k) - f_j(y) + c. \quad (4.11)
\end{aligned}$$

which implies

$$\begin{aligned}
& f_j(x^k) + \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) - f_j(y) \\
& - \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 - \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) > f_j(x^k) - f_j(y) + c. \quad (4.12)
\end{aligned}$$

Since f_j is continuous for all $j = \overline{1, q}$, for $k \in K$ sufficiently large, $f_j(x^k) - f_j(y) + c > \epsilon_k$ with $\epsilon_k > 0$.

Therefore,

$$\begin{aligned}
& \min_{j=1,2,\dots,q} \left\{ f_j(x^k) + \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) - f_j(y) \right. \\
& \left. - \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 - \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \right\} > \min_{j=1,2,\dots,q} \left\{ f_j(x^k) - f_j(y) + c \right\}. \quad (4.13)
\end{aligned}$$

This contradicts equation (4.8) for $k \in K$ sufficiently large, suggesting that $\lim_{\substack{k \rightarrow +\infty \\ k \in K}} x^k = x^* \in \mathcal{X}$. Consequently, the proof of the theorem is complete. $\square$

Theorem 4.3 (Optimality). *Assuming that hypotheses 1 to 5 are satisfied. Let x^* be an accumulation point of the sequence $\{x^k\}$ generated by algorithm 1. Then, x^* is a weak Pareto point for problem (2.1).*

Proof. Let $K \subseteq \{0, 1, \dots\}$ such that $\lim_{\substack{k \rightarrow +\infty \\ k \in K}} x^k = x^* \in \mathcal{X}$. According to Theorem

4.2, we have $g(x^*) \leq 0$. Now, let's assume by contradiction that x^* is not a weakly Pareto point for problem (2.1). Then, there exists $y \in \mathcal{X}$ such that

$$f_j(y) < f_j(x^*) \quad \forall j = 1, 2, \dots, q. \quad (4.14)$$

According to Lemma 2.3, we have

$$\begin{aligned}
& \min_{j=1,2,\dots,q} \left\{ f_j(x^k) + \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) - \right. \\
& \left. f_j(y) - \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 - \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \right\} \leq 0. \quad (4.15)
\end{aligned}$$

which implies

$$\begin{aligned} \min_{j=1,2,\dots,q} \left\{ f_j(x^k) - f_j(y) \right\} &\leq \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(y)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \\ &\quad - \frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 - \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right). \end{aligned} \quad (4.16)$$

Since $y \in \mathcal{X}$, we obtain

$$\begin{aligned} \min_{j=1,2,\dots,q} \left\{ f_j(x^k) - f_j(y) \right\} &\leq -\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x^k)]_+^2 - \tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) \\ &+ \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \leq -\tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) \\ &\quad + \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right). \end{aligned} \quad (4.17)$$

Now, we consider two cases:

- (a): the sequence $\{\tau^k\}$ is bounded,
- (b): the sequence $\{\tau^k\}$ is unbounded.

In case (a), there exists a positive integer k_0 such that for each $k \geq k_0$, $\tau^k \rightarrow \tau^{k_0}$, and we have $\tau^{k_0} \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \leq p\tau^{k_0}\mu^k$. Hence,

$$\min_{j=1,2,\dots,q} \left\{ f_j(x^k) - f_j(y) \right\} \leq -\tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) + p\tau^k\mu^k. \quad (4.18)$$

Since x^* is feasible, and using the fact that $\lim_{k \in K} \mu^k = 0$ for $k \in K$ sufficiently large, we obtain

$$\min_{j=1,2,\dots,q} \left\{ f_j(x^*) - f_j(y) \right\} \leq 0. \quad (4.19)$$

This contradicts (ii) of Lemma 2.3.

For case (b), using equation (4.17), we get

$$\begin{aligned} \min_{j=1,2,\dots,q} \left\{ f_j(x^k) - f_j(y) \right\} &\leq -\tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) \\ &+ \tau^k \sum_{i=1}^p \left(g_i(y) + \sqrt{(g_i(y))^2 + (\mu^k)^2} \right) \leq -\tau^k \sum_{i=1}^p \left(g_i(x^k) + \sqrt{(g_i(x^k))^2 + (\mu^k)^2} \right) \\ &\quad + p\tau^k\mu^k \leq p\tau^k\mu^k. \end{aligned} \quad (4.20)$$

Now, by taking the limit for k sufficiently large and using $\lim_{k \in K} \tau^k \mu^k = 0$, and recalling once again that $g_i(x^k) \leq 0$, we obtain

$$\min_{j=1,2,\dots,q} \left\{ f_j(x^*) - f_j(y) \right\} \leq 0. \quad (4.21)$$

This contradicts (ii) of Lemma 2.3. Consequently, the proof of the theorem is complete. $\square$

5. A PRACTICAL ALGORITHM

5.1. Practical algorithm for solving the subproblem. In this section, we address the resolution of the subproblem at step 2 of Algorithm 1. Here, we assume that the objective functions f_j , $\forall j$, and the constraints g_i , $\forall i$ are continuous and differentiable.

The assumptions of differentiability imply that the augmented multiobjective Lagrangian is continuously differentiable with respect to x for $\mu^k > 0$. More specifically, the gradient of its j -th component, denoted as $\nabla \mathcal{L}_{\rho^k}^j$, is expressed as

$$\begin{aligned} \nabla \mathcal{L}_{\rho^k}^j = \nabla f_j(x^k) + \sum_{i=1}^p \delta_i^k \left[\max \{0, g_i(x^k)\} \nabla g_i(x^k) \right] + \\ \tau^k \sum_{i=1}^p \left[\nabla g_i(x^k) \left(1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + (\mu^k)^2}} \right) \right]. \end{aligned} \quad (5.1)$$

Since the subproblem (3.5) is multiobjective and unconstrained, we opt for a resolution using the Quasi-Newton multiobjective method. This method provides an exact approach to solving unconstrained multiobjective problems. It relies on determining the descent direction from a quadratic model of the objective functions, where certain approximations replace the true Hessian at each iteration. As a reminder, we revisit the following definitions:

Definition 5.1. A point $x^* \in \mathbb{R}^n$ is said to be Pareto-Stationary for the problem $\min_{x \in \mathbb{R}^n} F(x)$ if

$$\min_{\substack{d \in \mathbb{R}^n, \\ \|d\| \leq 1}} \max_{j=1,q} \nabla f_j(x^*)^\top d + \frac{1}{2} d^\top B_j d \geq 0, \quad (5.2)$$

where B_j is an approximation of the Hessian matrix associated with the function f_j at x^* , and d is a descent direction. Let $\theta(x)$ and $d(x)$ be defined as follows:

$$\theta(x) = \min_{\substack{d \in \mathbb{R}^n, \\ \|d\| \leq 1}} \max_{j=1,q} \nabla f_j(x)^\top d + \frac{1}{2} d^\top B_j d, \quad (5.3)$$

$$d(x) = \arg \min_{\substack{d \in \mathbb{R}^n, \\ \|d\| \leq 1}} \max_{j=1, q} \nabla f_j(x)^\top d + \frac{1}{2} d^\top B_j d. \quad (5.4)$$

The stopping condition is based on Lemma 2 presented in [31], which establishes a relationship between the stationarity of a point x , $d(x)$, and $\theta(x)$.

Lemma 5.2 ([31]). *Let $B_j(x)$ be a positive definite matrix for every $x \in \mathbb{R}^n$, and consider θ defined in 5.3. Then,*

- (1) *for all $x \in \mathbb{R}^n$, $\theta(x) \leq 0$,*
- (2) *the following conditions are equivalent,*
 - (a) *the point x is not stationary,*
 - (b) *$\theta(x) < 0$,*
 - (c) *$d(x) \neq 0$,*
- (3) *the function θ is continuous.*

The update of the Hessian matrix at each iteration in the Quasi-Newton method typically follows the BFGS formula. Various variants of the Quasi-Newton method are listed in the literature, distinguished by how the matrix B_j^{k+1} is updated. Among these variants, we mention two examples that we will use for solving the subproblem of equation (3.5) in Algorithm 1.

- **Classic version of the BFGS method:** The update of the matrix B_j^{k+1} at each iteration is given by the following formula:

$$B_j^{k+1} = B_j^k - \frac{B_j^k s_k s_k^T B_j^k}{s_k^T B_j^k s_k} + \frac{y_j^k (y_j^k)^T}{s_k^T y_j^k}.$$

where $y_j^k = \nabla f_j(x^{k+1}) - \nabla f_j(x^k)$, $s_k = x^{k+1} - x^k$

- **Self-Scaling BFGS (SS-BFGS):** The update formula is defined as follows:

$$B_j^{k+1} = \frac{s_k^\top y_j^k}{s_k^T B_j^k s_k} \left(B_j^k - \frac{B_j^k s_k s_k^\top B_j^k}{s_k^\top B_j^k s_k} \right) + \frac{y_j^k (y_j^k)^\top}{s_k^\top y_j^k}.$$

For solving the subproblem in Algorithm 1 as mentioned earlier, we apply the variants of the BFGS method presented above. The following algorithm describes the steps for solving the subproblem (3.5) using either the BFGS or SS-BFGS method in Algorithm 1.

Algorithm 2: Practical Hyperbolic Augmented Lagrangian Algorithm for Multiobjective Optimization Problems.

Data: $B_j = I_n$, $\tau^1 > 0$; $\sigma > 1$, $\mu^1 \geq 0$; $\zeta < 1/\sigma$; $\nu \in [0; 1)$; $\delta^+ > 0$;
 $\delta_i^1 \in [0; \delta^+]$; $x_1 \in \mathbb{R}^n$, $\beta \in (0, 1)$, $\alpha_1 \in (0, 1)$, $\{\epsilon_k\} \in \mathbb{R}$

1 **for** $k = 2, 3, \dots$ **do**
2 Let $\mathcal{L}_{\rho^k}$ be the Hyperbolic Augmented Lagrangian function defined as follows:

$$\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k) = F(x) + \left[\frac{1}{2} \sum_{i=1}^p \delta_i^k [g_i(x)]_+^2 + \tau^k \sum_{i=1}^p \left(g_i(x) + \sqrt{(g_i(x))^2 + (\mu^k)^2} \right) \right] \cdot e, \quad (5.5)$$

3 Set $x^k = \text{MOBFGS}(\mathcal{L}_{\rho^k}(\cdot, \delta^k, \tau^k, \mu^k), x^{k-1}, \epsilon_k)$
4 **for** $i = 1, 2, 3, \dots, p$ **do**
5 **if** $g_i(x^k) > 0$ **then**
6 Set $\delta_i^{k+1} = \min \left\{ \delta_i^k + \tau^k \left(1 + \frac{g_i(x^k)}{\sqrt{(g_i(x^k))^2 + (\mu^k)^2}} \right), \delta^+ \right\}$
7 **else**
8 Set $\delta_i^{k+1} = 0$
9 Set $V_i^k = \min \left\{ -g_i(x^k), \delta_i^{k+1} \right\}$
10 Set $\mu^{k+1} = \zeta \cdot \mu^k$
11 **if** $\|V^k\| \leq \nu \|V^{k-1}\|$ **then**
12 Set $\tau_{k+1} = \tau_k$
13 **else**
14 Set $\tau_{k+1} = \sigma \tau_k$

Whether algorithm 2 is well defined depends on step 3. Thus, the following proposition shows that the sequence $\{x^k\}$ is determined in a finite number of iterations.

Proposition 5.3. *Suppose there exists a constant c such that for every j , $\|B_j(x)\| \leq c$, and assumption 1 is satisfied. Then, at each iteration k of Algorithm 2, the multiobjective BFGS algorithm identifies, in a finite number of iterations, a point x^k that satisfies the ε_k -Pareto-stationary condition for $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$.*

Proof. Referring to [34], Theorem 5, it is established that the multiobjective BFGS algorithm generates a sequence of points, and each accumulation point

is Pareto-critical if there exists a constant c such that $\|B_j(\cdot)\| \leq c$ and under assumption 1, i.e., the objective functions have bounded level sets.

Firstly, let's prove that, under the assumption of the proposition, the hyperbolic augmented Lagrangian function $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$ indeed has bounded level sets. Let $z \in \mathbb{R}$. It is known that T_0 is bounded. Now consider the level set $T_1 = \{x \in \mathbb{R}^n \mid \mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k) \leq z\}$. By the definition of $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$, for all $x, \delta^k, \tau^k, \rho^k$, and μ^k , we have $F(x) \leq \mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$. Therefore, $T_1 \subseteq T_0$. Since z is arbitrary and any subset of a bounded set is bounded, the hyperbolic augmented Lagrangian function $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$ has bounded level sets.

Now we need to demonstrate that the multiobjective BFGS algorithm finds an ε_k -Pareto-stationary point for $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$ in a finite number of iterations. Assume by contradiction that the algorithm generates an infinite sequence $\{x^t\}$ such that, for every t , x^t is not ε_k -Pareto-stationary for $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$. Since the level sets of the hyperbolic augmented Lagrangian function are bounded, there exists a subset $K \subseteq \{0, 1, \dots\}$ such that $x^t \rightarrow x^*$ as $t \rightarrow \infty$, $t \in K$, where x^* is a Pareto-critical point for $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$, i.e., according to Definition 5.1, for $x^* \in \mathcal{X}$, $\max_j \nabla f_j(x^*)^\top d + \frac{1}{2} d^\top B_j d \geq 0$ for every $d \in \mathbb{R}^n$, $\|d\| \leq 1$. Considering the continuity of the max operator and ∇f_j for $j = 1, \dots, q$, we can write $\lim_{\substack{t \rightarrow \infty, \\ t \in K}} \max_j \nabla f_j(x^t)^\top d + \frac{1}{2} d^\top B_j d \geq 0 > -\varepsilon_k$ for every $d \in \mathbb{R}^n$, $\|d\| \leq 1$. For $t \in$

K sufficiently large, it must then be true that $\max_j \nabla f_j(x^t)^\top d + \frac{1}{2} d^\top B_j d > -\varepsilon_k$ for every $d \in \mathbb{R}^n$, $\|d\| \leq 1$, i.e., x^t is ε_k -Pareto-stationary for $\mathcal{L}_{\rho^k}(x, \delta^k, \tau^k, \mu^k)$. Therefore, we obtain a contradiction, and the thesis is finally proven. $\square$

5.2. Numerical Simulation. We implemented Algorithm 2 with the following parameters: $\mu_0 = 1$, $\rho_0 = 10^6$, $\tau_0 = 0.9$, $\gamma = 10$, $\sigma_0 = 10$, and $\mu_{\max} = 10^4$.

Table 1 presents the assortment of test problems used in this study. The first column lists the problem titles, followed by the number of variables in the second column, the number of objectives in the third, details of related constraints in the fourth, and finally, the origins of the problems are described in the fifth column.

To experiment with the 2 algorithm, we ran tests using an HP EliteBook laptop equipped with an Intel Core i7-3687U processor, with a base frequency between 2.10 GHz and 2.60 GHz, and 4 GB RAM.

TABLE 1. List of multiobjective optimization problems

Problems	n	q	Parameters bounds	Sources
BNH1	2	2	$[0, 5]^2$	[26]
SCH	1	2	$[-4, 4]$	[38]
M-LAP1	2	2	$[-7, 1]^2$	[11]
M-LAP2	a	2	$[0, 1]^n$	[11]
JOS1	a	2	$[0, 1]^n$	[26]
ZLT1	a	2	$[-1000, 1000]^n$	[26]
MLF1	1	2	$[0, 20]$	[26]

M-MLF2	1	2	$[0, 20]$	[26]
LRS1	2	2	$[-50, 50]$	[26]
MHHM1	1	3	$[0, 1]$	[26]
MHHM2	2	3	$[0, 1]$	[26]
IKK1	1	3	$[-50, 50]$	[26]
VFM1	2	3	$[-2, 2]$	[26]
DGO1	1	2	$[-10, 13]$	[26]
SSFYY1	1	2	$[-100, 100]$	[26]
VU1	2	2	$[-3, 3]$	[26]
BK1	2	2	$[-5, 10]^2$	[26]
SSFYY1	1	2	$[-100, 100]$	[26]
VU2	2	2	$[-3, 3]^2$	[26]
COSH	1	2	$[0, 5]$	[40]
DEB	2	2	$[0.1, 1] \times [0, 1]$	[16]
BNH2	2	2	$[-20, 20]^2$	[26]
FF1	2	2	$[-2, 2]^2$	[26]
FON	a	2	$[-2, 2]^n$	[26]
WIT1-6	2	2	$[-2, 2]^2$	[9]
Min-Ex	2	2	$[0.1, 1] \times [0, 5]$	[26]
Imbalance1-2	2	2	$[-2, 2]^2$	[26]
SSFYY2	2	2	$[-100, 100]^2$	[26]

where

$$a \in \{2, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50\};$$

To compare the different variants of the BFGS method, which we name MOHALA-BFGS (or MOHALA in the legend of the figures) for solving the subproblem using the classical BFGS method, and MOHALA-SS-BFGS (or MOSSHALA in the legend of the figures) for solving the subproblem using the SS-BFGS method in solving the subproblem at step 3 of Algorithm 2, we use the performance profiles developed by Elizabeth D. Dolan and Jorge J. Moré [18] for the purity and dispersion measures, namely the Δ -Spread and Γ -Spread metrics.

The purity metric assesses the quality of the Pareto front produced by an algorithm, indicating the percentage of non-dominated solutions generated by the custodioDirectMultisearchMultiobjective2011b method. Mathematically, the purity metric is expressed as follows:

$$Purity(S) = \frac{|F_{p,s} \cap F_p|}{|F_{p,s}|},$$

where $F_{p,s}$ denotes the solutions generated by a method $s \in S$ for a problem $p \in P$, S representing the set of methods and P the set of test problems. F_p represents the set of solutions generated by all methods for problem p ($F_p = \bigcup F_{p,s}$) without any dominated points.

The dispersion metrics used are Γ -Spread and Δ -Spread. The Γ -Spread metric measures the maximum spacing of solutions generated by a method [14]. It is given by the following formula:

$$\Gamma\text{-Spread}(S) = \max_{j \in \{1, \dots, q\}} \left(\max_{i \in \{0, \dots, N\}} \{\delta_{i,j}\} \right);$$

where N represents the number of solutions generated by a method, m is the number of objective functions, and $\delta_{i,j} = f_j(x^{k+1}) - f_j(x^k)$ with the values of $f_j(x^k)$ arranged in ascending order.

The Δ -Spread metric evaluates the distribution of solutions produced by a method [14]. Its calculation is determined by the following formula:

$$\Delta\text{-Spread}(S) = \max_{j \in \{1, \dots, q\}} \frac{\delta_{0,j} + \delta_{N,j} + \sum_{i=1}^{N-1} |\delta_{i,j} - \bar{\delta}_{i,j}|}{\delta_{0,j} + \delta_{N,j} + (N-1)\bar{\delta}_{i,j}};$$

where $\bar{\delta}_{i,j}$ is the average of $\delta_{i,j}$ with $j = 1, \dots, N-1$. $\delta_{0,j}$ and $\delta_{N,j}$ represent the extreme points indexed by 0 and $N+1$. We therefore used the methodology suggested in [14] to find the most extreme points for problems that have no analytic front. We began by removing the dominant points from the union of all these fronts. The largest pairwise distance measured using $f_j(\cdot)$ was selected for each component of the objective function.

We then evaluate the performance of the four metrics presented above using the performance profiles proposed in [8, 18]. It is noteworthy to recall that performance profiles are depicted by a diagram depicting a cumulative distribution function, named $\rho(\tau)$, which is defined as follows:

$$\rho_s(\tau) = \frac{1}{|P|} |\{p \in P : r_{p,s} \leq \tau\}|$$

where $r_{p,s} = \frac{t_{p,s}}{\min \{t_{p,\bar{s}} : \bar{s} \in S\}}$. For the Purity metric, we set $t_{p,s} = 1/t_{p,s}$ since performance profiles are used for metrics where the lower value indicates better performance.

5.2.1. Comparative study with different variants MOHALA-BFGS and MOHALA-SS-BFGS.

We begin the study by presenting the performance profiles of the various performance metrics. It should be noted that the two approaches to solving the subproblem of Algorithm 2, namely the MOHALA-BFGS method and the MOHALA-SS-BFGS method, are executed by first randomly generating a set of 100 points in the feasible space. This set is then considered as the starting set for exploration.

The Figure 2 represents the performance profile on the set of considered test problems for the three metrics, namely the Purity metric, the Δ -Spread metric, and the Γ -Spread metric. We can observe that the MOHALA-BFGS and

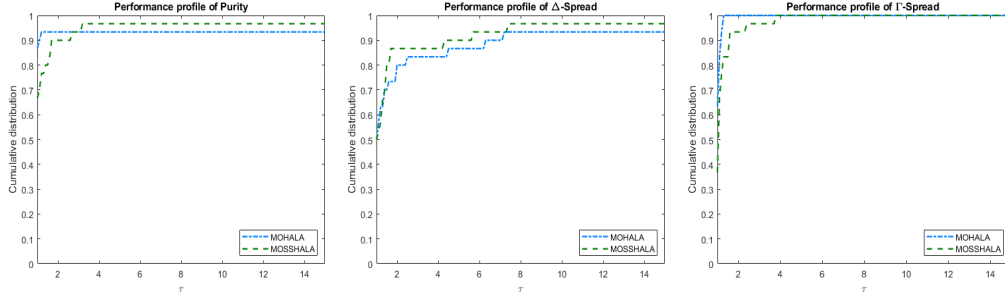


FIGURE 2. Performance profiles of MOHALA-BFGS and MOHALA-SS-BFGS

MOHALA-SS-BFGS methods are incomparable on the Γ -Spread metric. Therefore, for the Purity and Δ -Spread metrics, we notice that the MOHALA-SS-BFGS method is significantly better than the MOHALA-BFGS method.

Table 2 presents the performance measures on the metrics for the 7 considered test problems for the MOHALA-BFGS and MOHALA-SS-BFGS methods. It can be observed that, out of the 7 test problems, the MOHALA-SS-BFGS method is better than the MOHALA-BFGS method on the BNH2 problem and remains incomparable on the other problems. For the Δ -Spread metric, the MOHALA-BFGS method achieves a percentage of 57.14% compared to 42.86% for MOHALA-SS-BFGS. For the Γ -Spread metric, the MOHALA-SS-BFGS method achieves a percentage of 57.14% compared to 42.86% for MOHALA-BFGS.

TABLE 2. Purity and Spread values for 07 problems for MOHALA-BFGS and MOHALA-SS-BFGS.

Methods	MOHALA-BFGS			MOHALA-SS-BFGS		
	Purity	Δ -Spread	Γ -Spread	Purity	Δ -Spread	Γ -Spread
BHN1	1.0	0.9271	11.7635	1.0	0.9241	11.7694
BNH2	0.8700	1.2328	80.5346	0.9700	1.9564	46.3629
FON	1.0	0.9412	0.0712	1.0	0.9381	0.1010
FF1	1.0	1.0296	0.0831	1.0	1.0988	0.1381
Imbalance1	1.0	1.0732	1.5552E+03	1.0	1.1312	964.9984
ZLT1	1.0	1.0282	0.1424	1.0	1.2192	0.2064
COSH	1.0	1.0069	3.5877	1.0	0.9631	2.2784

6. CONCLUSION

In this paper, we introduced a novel method based on the Hyperbolic Augmented Lagrangian for solving multiobjective optimization problems. It is a method that does not transform the multiobjective problem into a single-objective problem. After presenting the method, we subsequently proposed an algorithm

for which we demonstrated the feasibility and optimality of the solutions generated by the proposed method. Additionally, we conducted numerical experiments on benchmark problems, where the subproblem of the proposed algorithm was solved using two variants of the BFGS method. The analysis reveals that the MOHALA-SS-BFGS method is superior to the MOHALA-BFGS method in generating non-dominated solutions and distributing solutions.

Future research will involve applying the MOHALA method to solve large-scale multiobjective optimization problems and extending it to multidimensional variational problems.

REFERENCES

1. Andreani, R., Birgin, E., Martínez, J. & Schuverdt, M. (2008). On Augmented Lagrangian Methods with General Lower-Level Constraints. *SIAM Journal On Optimization*. **18**(4), 1286-1309. <https://doi.org/10.1137/060654797>
2. Bertsekas, D. (2015). Convex Optimization Algorithms. Athena Scientific. <https://api.semanticscholar.org/CorpusID:118800102>
3. Bertsekas, D. (1997). Nonlinear Programming. *Journal Of The Operational Research Society*. **48**(3), 334-334. <https://doi.org/10.1057/palgrave.jors.2600425>
4. Birgin, E. & Martínez, J. (2016). On the Application of an Augmented Lagrangian Algorithm to Some Portfolio Problems. *EURO Journal On Computational Optimization*. **4**(1), 79-92. <https://doi.org/10.1007/s13675-015-0052-9>
5. Birgin, E., Martínez, J. & Raydan, M. (2000). Nonmonotone Spectral Projected Gradient Methods on Convex Sets. *SIAM Journal On Optimization*. **10**(4), 1196-1211. <https://doi.org/10.1137/S1052623497330963>
6. Birgin, E., Martínez, J. & Prudente, L. (2015). Optimality Properties of an Augmented Lagrangian Method on Infeasible Problems. *Computational Optimization And Applications*. **60**(3), 609-631. <https://doi.org/10.1007/s10589-014-9685-5>
7. Birgin, E. & Martínez, J. (2014). Practical Augmented Lagrangian Methods for Constrained Optimization. Society for Industrial and Applied Mathematics. <https://doi.org/10.1137/1.9781611973365>
8. Brás, C. & Custódio, A. (2020). On the Use of Polynomial Models in Multiobjective Directional Direct Search. *Computational Optimization And Applications*. **77**, 897-918. <https://doi.org/10.1007/s10589-020-00233-8>
9. Chen, J., Tang, L. & Yang, X. (2023). A Barzilai-Borwein Descent Method for Multiobjective Optimization Problems. *European Journal Of Operational Research*. **311**(1), 196-209. <https://doi.org/10.1016/j.ejor.2023.04.022>
10. Chen, H., Wu, H., Chan, S. & Lam, W. (2020). A Stochastic Quasi-Newton Method for Large-Scale Nonconvex Optimization With Applications. *IEEE Transactions On Neural Networks And Learning Systems*. **31**, 4776-4790. <https://doi.org/10.1137/140954362>
11. Cocchi, G. & Lapucci, M. (2020). An Augmented Lagrangian Algorithm for multiobjective Optimization. *Computational Optimization And Applications*. **77**(1), 29-56. <https://doi.org/10.1007/s10589-020-00204-z>
12. Cocchi, G., Lapucci, M. & Mansueto, P. (2021). Pareto Front Approximation through a Multi-Objective Augmented Lagrangian Method. *EURO Journal On Computational Optimization*. **9**, 100008. <https://doi.org/10.1016/j.ejco.2021.100008>
13. Costa, M., Rocha, A. & Fernandes, E. (2014). An Artificial Fish Swarm Algorithm Based Hyperbolic Augmented Lagrangian Method. *Journal Of Computational And Applied Mathematics*. **259**, 868-876. <https://doi.org/10.1016/j.cam.2013.08.017>
14. Custódio, A., Madeira, J., Vaz, A. & Vicente, L. (2011). Direct Multisearch for Multiobjective Optimization. *SIAM J. Optim.* **21**, 1109-1140. <https://doi.org/10.1137/10079731X>

15. Das, I. & Dennis, J. (1998). Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems. *SIAM Journal On Optimization*. **8**(3), 631-657. <https://doi.org/10.1137/S1052623496307510>
16. Deb, K. (1999). Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems. *Evolutionary Computation*. **7**, 205-230. <https://doi.org/10.1162/evco.1999.7.3.205>
17. Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. John Wiley & Sons. https://doi.org/10.1007/978-0-85729-652-8_1
18. Dolan, E. & Moré, J. (2002). Benchmarking Optimization Software with Performance Profiles. *Mathematical Programming*. **91**, 201-213. <https://doi.org/10.1007/s101070100263>
19. Fliege, J. & Svaiter, B. (2000). Steepest Descent Methods for Multicriteria Optimization. *Mathematical Methods Of Operations Research (ZOR)*. **51**(8), 479-494. <https://doi.org/10.1007/s001860000043>
20. Fliege, J., Drummond, L. & Svaiter, B. (2009). Newton's Method for Multiobjective Optimization. *SIAM Journal On Optimization*. **20**(2), 602-626. <https://doi.org/10.1137/08071692X>
21. Fukuda, E. & Drummond, L. (2011). On the Convergence of the Projected Gradient Method for Vector Optimization. *Optimization. A Journal Of Mathematical Programming And Operations Research*. **60**, 1009-1021. <https://doi.org/10.1080/02331934.2010.522710>
22. Galvan, G. & Lapucci, M. (2019). On the Convergence of Inexact Augmented Lagrangian Methods for Problems with Convex Constraints. *Operations Research Letters*. **47**(3), 185-189. <https://doi.org/10.1016/j.orl.2019.03.006>
23. Graña Drummond, L. & Svaiter, B. (2005). A Steepest Descent Method for Vector Optimization. *Journal Of Computational And Applied Mathematics*. **175**, 395-414. <https://doi.org/10.1016/j.cam.2004.06.018>
24. Hestenes, M. (1969). Multiplier and Gradient Methods. *Journal Of Optimization Theory And Applications*. **4**(5), 303-320. <https://doi.org/10.1007/BF00927673>
25. Hug, R., Maitre, E. & Papadakis, N. (2020). On the Convergence of Augmented Lagrangian Method for Optimal Transport between Nonnegative Densities. *Journal Of Mathematical Analysis And Applications*. **485**(2), 123811. <https://doi.org/10.1016/j.jmaa.2019.123811>
26. Huband, S., Hingston, P., Barone, L. & While, L. (2006). A Review of Multiobjective Test Problems and a Scalable Test Problem Toolkit. *IEEE Transactions On Evolutionary Computation*. **10**(10), 477-506. <https://doi.org/10.1109/TEVC.2005.861417>
27. Lian, S., Meng, S. & Wang, Y. (2018). An Objective Penalty Function-Based Method for Inequality Constrained Minimization Problem. *Mathematical Problems In Engineering*. **2018**, pp. e7484256. <https://doi.org/10.1155/2018/7484256>
28. Miettinen, K. (1999). Nonlinear Multiobjective Optimization. Springer Science & Business Media. <https://doi.org/10.1007/978-1-4615-5563-6>
29. Mita, K., Fukuda, E. & Yamashita, N. (2019). Nonmonotone Line Searches for Unconstrained Multiobjective Optimization Problems. *Journal Of Global Optimization*. **75**(1), 63-90. <https://doi.org/10.1007/s10898-019-00802-0>
30. Morovati, V., Basirzadeh, H. & Pourkarimi, L. (2018). Quasi-Newton Methods for Multiobjective Optimization Problems. *4OR*. **16**, 261-294. <https://doi.org/10.1007/s10288-017-0363-1>
31. Povalej, Ž. (2014). Quasi-Newton's Method for Multiobjective Optimization. *Journal Of Computational And Applied Mathematics*. **255**, pp. 765-777. <https://doi.org/10.1016/j.cam.2013.06.045>
32. Prudente, L. & Souza, D. (2022). A Quasi-Newton Method with Wolfe Line Searches for Multiobjective Optimization. *Journal Of Optimization Theory And Applications*. **194**, 1107-1140. <https://doi.org/10.1007/s10957-022-02072-5>

33. Kumar, K., Ghosh, D., Upadhayay, A., Yao, J. & Zhao, X. (2023). Quasi-Newton methods for multiobjective optimization problems: A systematic review.. *Applied Set-Valued Analysis & Optimization*. **5(2)**. <https://doi.org/10.23952/asvao.5.2023.2.12>
34. Qu, S., Goh, M. & Chan, F. (2011). Quasi-Newton Methods for Solving Multiobjective Optimization. *Operations Research Letters*. **39**, 397-399. <https://doi.org/10.1016/j.orl.2011.07.008>
35. Ramirez, L., Maculan, N., Xavier, A. & Xavier, V. (2023). Dislocation Hyperbolic Augmented Lagrangian Algorithm for Nonconvex Optimization. *RAIRO - Operations Research*. **57**, 2941-2950. <https://doi.org/10.1051/ro/2023153>
36. Ramirez, L. (2022). The Hyperbolic Augmented Lagrangian Algorithm. Universidade Federal do Rio de Janeiro UFRJ/COPPE.
37. Rocha, A., Costa, M. & Fernandes, E. (2016). A Shifted Hyperbolic Augmented Lagrangian-based Artificial Fish Two-Swarm Algorithm with Guaranteed Convergence for Constrained Global Optimization. *Engineering Optimization*. **48**, 2114-2140. <https://doi.org/10.1080/0305215X.2016.1157688>
38. Tougma, A., Some, K. (2024) Inexact Exponential Penalty Function with the Augmented Lagrangian for Multiobjective Optimization Algorithms, *Journal of Applied Mathematics*, vol. 2024, Article ID 9615743, 19 pages.<https://doi.org/10.1155/2024/9615743>
39. Tougma, A., Yougbaré, W. & Zongo, M. (2021). Performance Analysis of Higher Education Institutions using AHP and PROMETHEE. *AADM*. **27(2)**, 265-284. <http://dx.doi.org/10.17654/DM027020265>
40. Tougma, A., Somé, K. & Compaoré, A. (2023). Extension of the Projected Gradient and Armijo's Rule Concepts for Solving Convex Nonlinear Multiobjective Optimization Problems. *Applied Analysis And Optimization*. **7(3)**, 263-278.
41. Upadhayay, A., Ghosh, D., Ansari, Q. & Jauny. (2022). Augmented Lagrangian Cone Method for Multiobjective Optimization Problems with an Application to an Optimal Control Problem. *Optimization And Engineering*. **24(3)**, 1633-1665. <https://doi.org/10.1007/s11081-022-09747-y>
42. Xavier, A. (2001). Hyperbolic Penalty: A New Method for Nonlinear Programming with Inequalities. *International Transactions In Operational Research*. **8**, 659-671. <https://doi.org/10.1111/1475-3995.t01-1-00330>

LABORATOIRE DE MATHÉMATIQUES, INFORMATIQUE ET APPLICATION (LAMIA), UNIVERSITÉ NORBERT ZONGO, BP 376 KOUDOUGOU, BURKINA FASO.

Email address: appolinaire.tougma19@gmail.com

LABORATOIRE DE MATHÉMATIQUES, INFORMATIQUE ET APPLICATION (LAMIA), UNIVERSITÉ NORBERT ZONGO, BP 376 KOUDOUGOU, BURKINA FASO.

Email address: kaboreaugustin19@gmail.com

LABORATOIRE DE MATHÉMATIQUES, INFORMATIQUE ET APPLICATION (LAMIA), UNIVERSITÉ NORBERT ZONGO, BP 376 KOUDOUGOU, BURKINA FASO.

Email address: sokous11@gmail.com