An International Journal of Optimization and Control: Theories & Applications
ISSN:2146-0957 eISSN:2146-5703
Vol.14, No.2, pp.147-155 (2024)
http://doi.org/10.11121/ijocta.1402

RESEARCH ARTICLE

Dislocation hyperbolic augmented Lagrangian algorithm in convex programming

Lennin Mallma Ramirez[a*], Nelson Maculan[b], Adilson Elias Xavier[a], Vinicius Layter Xavier[c]

[a]*Federal University of Rio de Janeiro, Systems Engineering and Computer Science Program (COPPE), Rio de Janeiro, Brazil*
[b]*Federal University of Rio de Janeiro, Systems Engineering and Computer Science Program-Applied Mathematics (IM/COPPE), Rio de Janeiro, Brazil*
[c]*Rio de Janeiro State University, Institute of Mathematics and Statistics, Graduate Program in Computational Sciences, Rio de Janeiro, Brazil*
lenninmr@cos.ufrj.br, maculan@cos.ufrj.br, adilson.xavier@gmail.com, viniciuslx@ime.uerj.br

ARTICLE INFO

Article History:
Received 7 May 2023
Accepted 5 April 2024
Available Online 7 April 2024

Keywords:
Augmented Lagrangian
Constrained optimization
Convergence
Convex problem

AMS Classification 2010:
90C30; 90C25

ABSTRACT

The dislocation hyperbolic augmented Lagrangian algorithm (DHALA) is a new approach to the hyperbolic augmented Lagrangian algorithm (HALA). DHALA is designed to solve convex nonlinear programming problems. We guarantee that the sequence generated by DHALA converges towards a Karush-Kuhn-Tucker point. We are going to observe that DHALA has a slight computational advantage in solving the problems over HALA. Finally, we will computationally illustrate our theoretical results.

1. Introduction

We are interested in the convex nonlinear programming problem subject to inequality constraints, as follows

$$min\left\{f(x) \mid x \in S\right\}, \qquad (1)$$

where $S = \{x \in I\!R^n \mid g_i(x) \geq 0,\ i = 1, ..., m\}$, f and $g_i,\ i = 1, ..., m$ are real-valued functions defined on $I\!R^n$, and where the functions f and $g_i,\ i = 1, ..., m$ are continuously differentiable. The problem (1) is solved in particular by the augmented Lagrangian methods. The methodology of this method consists of solving a subproblem, that is, minimizing an augmented Lagrangian function (this is an unconstrained optimization problem), in this way, a primal solution is obtained. Subsequently the Lagrange multipliers are estimated. Some augmented Lagrangian algorithms known in the literature are: exponential Lagrangian [1], Log-sigmoid Lagrangian [2], nonlinear rescaling principle [3] and [4].

These augmented Lagrangian algorithms are called nonquadratic augmented Lagrangian algorithms, these algorithms are often C^2 if the objective and constraints are also twice continuously differentiable.

The quadratic augmented Lagrangian is differentiable only once [5]. The hyperbolic augmented Lagrangian algorithm (HALA) also solves the problem (1), see [6], HALA is a nonquadratic augmented Lagrangian. Adilson Elias Xavier introduces the hyperbolic penalty function (HPF) in [7] and the dislocation hyperbolic penalty function (DHPF) in [8]. With this last function, we are going to propose our algorithm called dislocation hyperbolic augmented Lagrangian algorithm (DHALA).

*Corresponding Author

This algorithm has two interesting characteristics: the function DPF is continuously differentiable unlike the classic quadratic penalty function, see for example [5], and the rule for updating its multipliers naturally generates a kind of safeguards for them.

This does not occur with other augmented Lagrangian type algorithms, i.e., to bound the multipliers, these algorithms have to generate a projection on a box, thus limiting the multipliers (see [9]). The main contributions of our work are:

- We introduce the dislocation hyperbolic augmented Lagrangian algorithm (DHALA), to solve the convex optimization problem with constraints. We guarantee that the sequence generated by DHALA converges to a Karush-Kuhn-Tucker (KKT) point. With this new approach, we notice that our algorithm DHALA converges to the solution in less time when compared to HALA (see the computational illustration of this work).
- This algorithm is based on the dislocation hyperbolic augmented Lagrangian function (DHALF). This function belongs to class C^∞ if the involved functions $f(x)$ and $g_i(x),\ i = 1, ..., m$, do too.

The paper is organized as follows: Section 2 presents some basic definitions; Section 3 introduces the algorithm DHALA and assurance its convergence; Section 4 presents computational illustrations of our theoretical results; Section 5 presents some conclusions of our work, and discusses idea for future work.

2. Basic results

Throughout this paper, we are interested in studying the following optimization problem:

$$(P) \qquad x^* \in X^* = argmin\{f(x) \mid x \in S\},$$

where

$$S = \{x \in I\!R^n \mid g_i(x) \geq 0, \ \ i = 1, ..., m\},$$

is the convex feasible set of the problem (P) and where the function $f : I\!R^n \to I\!R$ is convex, $g_i : I\!R^n \to I\!R,\ i = 1, ..., m$, are concave functions, assuming that $f,\ g_i$ are continuously differentiable in that way, (P) is a convex optimization problem. Let us consider the following assumptions:

C1. The optimal set X^* is nonempty, closed, bounded and, consequently, compact.

C2. Slater constraint qualification holds, i.e., there exists $\hat{x} \in S$ which satisfies $g_i(\hat{x}) > 0,\ i = 1, ..., m$.

The Lagrangian function of the problem (P), $L : I\!R^n \times I\!R^m_+ \to I\!R,$ is defined as

$$L(x, \lambda) = f(x) - \sum_{i=1}^{m} \lambda_i g_i(x),$$

where, $\lambda_i \geq 0,\ i = 1, ..., m$, are the Lagrange multipliers. We know that due to assumption **C2**, the following results will occur: there exists $\lambda^* = (\lambda^*_1, ..., \lambda^*_m)$ such that the KKT conditions hold true, i.e.,

$$\nabla_x L(x^*, \lambda^*) = \nabla f(x^*) - \sum_{i=1}^{m} \lambda^*_i \nabla g_i(x^*) = 0,$$
$$\lambda^*_i g_i(x^*) = 0, \quad i = 1, ..., m,$$
$$g_i(x^*) \geq 0, \quad i = 1, ..., m,$$
$$\lambda^*_i \geq 0, \quad i = 1, ..., m.$$

Moreover, the set of optimal Lagrange multipliers λ^* is denoted by

$$\Lambda^* = \left\{ \lambda \in I\!R^m_+ \mid \nabla f(x^*) - \sum_{i=1}^{m} \lambda_i \nabla g_i(x^*) = 0, \quad x^* \in X^* \right\},$$

which is a bounded set (and hence compact set) due to **C2**. The dual function $\Phi : I\!R^m_+ \to I\!R$, is defined as follows

$$\Phi(\lambda) = \inf_{x \in I\!R^n} L(x, \lambda), \tag{2}$$

and the dual problem consists of finding

$$(D) \qquad \lambda \in \Lambda^* = argmax\{\Phi(\lambda) \mid \lambda \in I\!R^m_+\}.$$

2.1. Hyperbolic and dislocation hyperbolic penalty function

The hyperbolic penalty algorithm (HPA) is meant to solve the problem (P). HPA adopts the HPF as

$$P(y, \lambda, \tau) = -\lambda y + \sqrt{(\lambda y)^2 + \tau^2}, \tag{3}$$

where $P : I\!R \times I\!R_+ \times I\!R_{++} \to I\!R$. HPF is originally proposed in [7] and studied in [10]. This function is a smoothing of the exact penalty function studied by Zangwill [11]. HPF is used in HALA. On the other hand, in [8], DHPF is proposed and defined as follows:

$$p(g_i(x), \lambda_i, \tau) = -\lambda_i g_i(x) + \sqrt{(\lambda_i g_i(x))^2 + \tau^2} - \tau, \tag{4}$$

for $i = 1, ..., m$, where $p : I\!R \times I\!R_+ \times I\!R_{++} \to I\!R$. Using (4), we will introduce DHALA in the next section.

3. Dislocation hyperbolic augmented Lagrangian algorithm

In this section, the are going to consider function (4) to define DHALF of problem (P) by $l_H : I\!R^n \times I\!R^m_{++} \times I\!R_{++} \to I\!R$,

$$l_H(x,\lambda,\tau) = f(x) + \sum_{i=1}^{m} p(g_i(x),\lambda_i,\tau)$$

$$= f(x)+\sum_{i=1}^{m}\left(-\lambda_i g_i(x) + \sqrt{(\lambda_i g_i(x))^2+\tau^2} - \tau\right), \tag{5}$$

where $\tau > 0$ is the penalty parameter. Note that this function belongs to class C^∞ if the involved functions $f(x)$ and $g_i(x),\ i = 1,...,m$, do too. We can rewrite (5) as follows:

$$l_H(x,\lambda,\tau)$$

$$= f(x)-\sum_{i=1}^{m}\tau\left(\frac{\lambda_i g_i(x)}{\tau} - \sqrt{\left(\frac{\lambda_i g_i(x)}{\tau}\right)^2+1}+1\right) \tag{6}$$

$$= f(x) - \sum_{i=1}^{m}\tau\, h\left(\frac{\lambda_i g_i(x)}{\tau}\right),$$

where the function $h : I\!R \to I\!R$, is defined as

$$h(t) = t - \sqrt{t^2+1}+1,$$

with $h \in C^\infty$. Henceforth, we will call the h function the dislocation hyperbolic function. Some properties of h are:

(H1) $h(0) = 0$ and $h'(0) = 1$.

(H2) h is increasing, i.e.,

$$h'(t) = 1 - \frac{t}{\sqrt{t^2+1}} > 0, \quad \forall t \in I\!R,$$

where $0 < h'(t) < 2$.

(H3) The h function is strictly concave, i.e.,

$$h''(t) = \frac{-1}{(t^2+1)^{\frac{3}{2}}} < 0, \quad \forall t \in I\!R.$$

By (H2), we get that $0 < h'(t) < 2$, which is also a characteristic similar to log-sigmoid Lagrangian (LST), see [2]. Now we present DHALA to solve the convex nonlinear programming problem (P).

Algorithm DHALA:

Step 1. Let $k := 0$. Take initial values $\lambda^0 = (\lambda_1^0,...,\lambda_m^0) \in I\!R^m_{++}$ and $\tau \in I\!R_{++}$.

Step 2. Solve the unconstrained minimization problem:

$$x^{k+1} \in argmin_{x\in I\!R^n}\ l_H(x,\lambda^k,\tau)$$

$$= argmin_{x\in I\!R^n}\left\{f(x) - \sum_{i=1}^{m}\tau h\left(\frac{\lambda_i g_i(x)}{\tau}\right)\right\}.$$

Step 3. Update the Lagrange multipliers:

$$\lambda_i^{k+1} = \lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right), \quad i = 1,...,m. \tag{7}$$

Step 4. If the pair (x^{k+1},λ^{k+1}) satisfies the stopping criteria, then stop.

Step 5. $k := k+1$. Go to Step 2.

The methodology of our algorithm is as follows: in **Step 2**, we solve a subproblem that is unconstrained; in **Step 3**, the new Lagrange multipliers are estimated and in **Step 4**, a stop condition is considered. The only difference between (3) and (4) is the term $-\tau$. The word "dislocation" in our algorithm comes specifically from this term. Let us consider the following assumption:

C3. For every $\tau > 0$ and $\lambda \in I\!R^m_{++}$, the level set

$$M = \{x \in I\!R^n \ \mid\ l_H(x,\lambda,\tau) \le \alpha\},$$

is bounded for every $\alpha < \infty$.

The Assumption **C3** can be verified when the function f is strongly convex. The strong convexity assumption for f is also studied in [5] and [12].

Remark 1. *From (7) and (H2) we note the following:*

$$\lambda_i^{k+1} = \lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)$$

$$= \lambda_i^k\left(1 - \frac{\lambda_i^k g_i(x^{k+1})}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2+\tau^2}}\right), \quad i = 1,...,m.$$

On the other hand, by **C3**, there exists $x^{k+1} \in I\!R^n$ such that

$$l_H(x^{k+1},\lambda^k,\tau) = min_{x\in I\!R^n} l_H(x,\lambda^k,\tau),$$

$$\nabla_x l_H(x^{k+1},\lambda^k,\tau) = 0 \ \text{ holds, i.e.,}$$

$$\nabla f(x^{k+1})-\sum_{i=1}^{m}\lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\nabla g_i(x^{k+1}) = 0. \tag{8}$$

Substituting (7) in (8), we have

$$\nabla_x l_H(x^{k+1},\lambda^k,\tau) = \nabla f(x^{k+1})-\sum_{i=1}^{m}\lambda_i^{k+1}\nabla g_i(x^{k+1}),$$

$$= \nabla_x L(x^{k+1},\lambda^{k+1}) = 0, \tag{9}$$

for any $\tau > 0$. We observe that x^{k+1} and λ^{k+1} satisfy $\nabla_x L(x^{k+1},\lambda^{k+1}) = 0$, showing that x^{k+1} is the minimizer of $L(x,\lambda^{k+1})$, i.e.,

$$\Phi(\lambda^{k+1}) = L(x^{k+1},\lambda^{k+1}) = \min_{x\in I\!R^n} L(x,\lambda^{k+1})$$

with

$$\lambda^{k+1} \in I\!R^m_{++},$$

which follows that

$$\Phi(\lambda^{k+1}) = f(x^{k+1}) - \sum_{i=1}^{m} \lambda_i^{k+1} g_i(x^{k+1}). \quad (10)$$

From (10), we obtain $-g(x^{k+1})$

$$= \left(-g_1(x^{k+1}), \cdots, -g_m(x^{k+1})\right)^T \in \partial\Phi(\lambda^{k+1}),$$

where $\partial\Phi(v) = \{-v \,:\, v \in \partial(-\Phi)(\lambda)\}$ is the subdifferential of $\Phi(v)$ at $v = \lambda^{k+1}$. In the following remark, we analyze what happens with Lagrange multipliers depending on the type of restriction. Let us define the following sets of indices

$$I_0 = \{i \in \{1, ..., m\} \mid g_i(x) = 0\},$$
$$I_- = \{i \in \{1, ..., m\} \mid g_i(x) < 0\},$$
$$I_+ = \{i \in \{1, ..., m\} \mid g_i(x) > 0\}.$$

Remark 2. *Let $\{\lambda^k\}$ be a sequence generated by DHALA such that $\lambda_i^k > 0, \quad i = 1, ..., m$ and let $\tau > 0$ fixed. Let us consider the following cases:*

- (c1) *If $i \in I_0$, then at the k-th iteration we have $g_i(x^{k+1}) = 0$. Then, from (7) and (H1), we get $\lambda_i^{k+1} = \lambda_i^k$. Thus, $\left(\lambda_i^k - \lambda_i^{k+1}\right) g_i(x^{k+1}) = 0, \ \forall i \in I_0$.*
- (c2) *If $i \in I_+$, then at the k-th iteration we have $g_i(x^{k+1}) > 0$. Then, from (H3), we have*
$$\frac{\lambda_i^k g_i(x^{k+1})}{\tau} > 0, \ i = 1, ..., m,$$
$$\lambda_i^k h'(0) < \lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right), \ i = 1, ..., m.$$
It follows from (7) and (H1) that $\lambda_i^k > \lambda_i^{k+1}$. Thus, $\left(\lambda_i^k - \lambda_i^{k+1}\right) g_i(x^{k+1}) > 0, \ \forall i \in I_+$.
- (c3) *If $i \in I_-$, then at the k-th iteration we have $g_i(x^{k+1}) < 0$. Then, from (7) and following a similar approach to case (c2), we can obtain $\lambda_i^k < \lambda_i^{k+1}$. Thus, $\left(\lambda_i^k - \lambda_i^{k+1}\right) g_i(x^{k+1}) > 0, \ \forall i \in I_-$.*

From the three previous cases, we can note that we have the following

$$\left(\lambda_i^k - \lambda_i^{k+1}\right) g_i(x^{k+1}) \geq 0, \quad i = 1, ..., m. \quad (11)$$

3.1. Convergence esult

We are going to guarantee the convergence of DHALA. This section is mainly based on [3] and [6]. In the following result, we will demonstrate the positivity of the updated Lagrange multipliers.

Proposition 1. *Let*

$$\left\{\lambda^k = (\lambda_1^k, ..., \lambda_m^k) \mid k = 1, 2, ...\right\} \subset I\!R^m.$$

If $\lambda^k \in I\!R^m_{++}$, then $\lambda^{k+1} \in I\!R^m_{++}$.

Proof. We can easily obtain the following, by making $\lambda_i^k > 0, \ i = 1, ..., m$. Thus, from (H2), we have

$$0 < \lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau^k}\right) < 2\lambda_i^k, \ i = 1, ..., m,$$

from the inequality above and (7), we obtain $\lambda_i^{k+1} > 0, \ i = 1, ..., m$. ∎

Remark 3. *From **C3** and Proposition 1, we conclude that DHALA is well defined.*

From Proposition 1, we get

$$0 < \lambda_i^{k+1} < 2\lambda_i^k, \ i = 1, ..., m, \quad (12)$$

see Proposition 3.2.1 of [6]. Since we have (12), (H1), (H2) and (H3), we can see that DHF has similar properties to the Log-Sigmoid transformation (LST), see Section 3 of [2] and Section 3 of [13].

Theorem 1. *Let $\{\lambda^k\}$ be a sequence generated by DHALA. The sequence $\{\Phi(\lambda^k)\}$ is monotone nondecreasing for all $k \in I\!N$.*

Proof. From the concavity of $\Phi(\cdot)$ and since $-g(x^{k+1}) \in \partial\Phi(\lambda^{k+1})$, we have

$$\Phi(\lambda^{k+1}) - \Phi(\lambda^k) \geq \sum_{i=1}^{m} \left(g_i(x^{k+1})\right)\left(\lambda_i^k - \lambda_i^{k+1}\right). \quad (13)$$

From Remark 1, it follows

$$\lambda_i^k - \lambda_i^{k+1} = \frac{\left(\lambda_i^k\right)^2 g_i(x^{k+1})}{\sqrt{\left(\lambda_i^k g_i(x^{k+1})\right)^2 + \tau^2}}, \quad i = 1, ..., m. \quad (14)$$

Then, expression (14) is replaced on the right side of inequality (13), and we get

$$\Phi(\lambda^{k+1}) - \Phi(\lambda^k) \geq \sum_{i=1}^{m} \left(\frac{\left(\lambda_i^k g_i(x^{k+1})\right)^2}{\sqrt{\left(\lambda_i^k g_i(x^{k+1})\right)^2 + \tau^2}}\right) \geq 0. \quad (15)$$

Thus, $\Phi(\lambda^{k+1}) \geq \Phi(\lambda^k)$. ∎

Proposition 2. *The sequence of dual objective function values $\{\Phi(\lambda^k)\}$ is bounded and monotone nondecreasing, hence it converges.*

Proof. By Theorem 1, we obtain $\Phi(\lambda^{k+1}) \geq \Phi(\lambda^k)$, then $\{\Phi(\lambda^k)\}$ is a nondecreasing sequence for all $k \in I\!N$ and considering the weak duality theorem, we obtain $\Phi(\lambda^k) \leq \Phi(\lambda^{k+1}) \leq f^*, \ \forall k$, i.e., $\{\Phi(\lambda^k)\}$ is bounded from above by the optimal value. Then $\{\Phi(\lambda^k)\}$ is convergent. ∎

Proposition 3. *The sequence $\{\lambda^k\}$ generated by the DHALA is bounded.*

Proof. From **C2**, we know that Λ^* is nonempty and compact. So, one level set of $\Phi(\cdot)$ is compact. Then all level sets are compact, see Corollary 8.7.1 of [14]. From Proposition 2, we obtain in particular that $\lambda^k \in \Gamma = \{\lambda \in I\!R^m_+ \mid \Phi(\lambda^0) \leq \Phi(\lambda)\}$ for all $k \in I\!N$. Hence, $\{\lambda^k\}$ is a bounded sequence. ∎

The following result is important to ensure the complementarity condition.

Lemma 1. *Let $d > 0$ and a sequence $\{a^k\} \subset I\!R_+$. If*

$$\lim_{k\to\infty}\left(a^k/\sqrt{a^k+d}\right) = 0, \qquad \textit{then} \qquad \lim_{k\to\infty} a^k = 0.$$

Proof. See, pag. 19, Lemma 3.2.1 of [6]. ∎

The following result is similar to Proposition 4.3 of [3] and letter (c) of the Lemma 3.2 of [1], also see [6].

Theorem 2. *Let the sequences $\{x^k\}$ and $\{\lambda^k\}$ be generated by DHALA. Then,*

$$\lim_{k\to\infty}\left(\lambda_i^k g_i(x^k)\right) = 0, \quad i = 1, ..., m. \tag{16}$$

Proof. Let be $\tau > 0$ be fixed. Since $\Phi(\cdot)$, is concave and $-g_i(x^{k+1}) \in \partial\Phi(\lambda^{k+1})$ we have

$$\Phi(\lambda^k) \leq \Phi(\lambda^{k+1}) + \sum_{i=1}^m \left(-g_i(x^{k+1})\right)\left(\lambda_i^k - \lambda_i^{k+1}\right).$$

Considering the inequality above and the Remark 2, we obtain

$$0 \leq \sum_{i=1}^m \left(\lambda_i^k - \lambda_i^{k+1}\right) g_i(x^{k+1}) \leq \Phi(\lambda^{k+1}) - \Phi(\lambda^k). \tag{17}$$

On the other hand, by (7), we have

$$\lambda_i^{k+1} = \lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right), i = 1, ..., m.$$

In the expression above, we subtract λ_i^k, $i = 1, ..., m$, and perform some operations, as follows for all $i = 1, ..., m$:

$$\lambda_i^{k+1} - \lambda_i^k = \lambda_i^k h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right) - \lambda_i^k,$$

$$\lambda_i^{k+1} - \lambda_i^k = \lambda_i^k\left(h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right) - 1\right),$$

$$\lambda_i^k - \lambda_i^{k+1} = \lambda_i^k\left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right). \tag{18}$$

Replacing (18) in (17), we have

$$\begin{aligned} 0 \leq \sum_{i=1}^m \left(\lambda_i^k\left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right)\right) g_i(x^{k+1}) \\ \leq \Phi(\lambda^{k+1}) - \Phi(\lambda^k). \end{aligned}$$

Rewriting the above:

$$\begin{aligned} 0 \leq \sum_{i=1}^m \left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right)\left(\lambda_i^k g_i(x^{k+1})\right) \\ \leq \Phi(\lambda^{k+1}) - \Phi(\lambda^k). \end{aligned} \tag{19}$$

Let us verify that the series in (19) is convergent:

$$\begin{aligned} 0 \leq \sum_{k=1}^\infty\sum_{i=1}^m \left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right)\left(\lambda_i^k g_i(x^{k+1})\right) \\ \leq \sum_{k=1}^\infty \left(\Phi(\lambda^{k+1}) - \Phi(\lambda^k)\right). \end{aligned}$$

We notice that $\sum_{k=1}^\infty \left(\Phi(\lambda^{k+1}) - \Phi(\lambda^k)\right)$ is a convergent series (i.e., the partial sum is bounded above), so it follows:

$$\begin{aligned} 0 \leq \sum_{k=1}^\infty\sum_{i=1}^m \left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right)\left(\lambda_i^k g_i(x^{k+1})\right) \\ \leq \lim_{k\to\infty}\left(\Phi(\lambda^k) - \Phi(\lambda^1)\right) \\ \leq f^* - \Phi(\lambda^1) < \infty. \end{aligned}$$

Therefore, for the test of comparison, we obtain

$$\begin{aligned} \lim_{k\to\infty}\sum_{i=1}^m \left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right)\left(\lambda_i^k g_i(x^{k+1})\right) \\ = 0. \end{aligned} \tag{20}$$

We note that each term in the summation (20) is nonnegative, thus

$$\lim_{k\to\infty}\left(1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right)\left(\lambda_i^k g_i(x^{k+1})\right) = 0, \tag{21}$$

$\forall i = 1, ..., m$.

Now, let us prove (16) similar the proof by contradiction. What follows is similar argument to Proposition 4.3 of [3]. Suppose that there exists a subsequence (for any fixed i) $U \subset \{1, 2, ...\}$ and an $\epsilon > 0$ such that

$$\mid \lambda_i^k g_i(x^{k+1}) \mid \geq \epsilon > 0, \quad \forall k \in U. \tag{22}$$

Then from (21), we obtain

$$\left\{1 - h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right\}_U \to 0,$$

so,

$$\left\{h'\left(\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right)\right\}_U \to 1.$$

Since we have (H1), we obtain

$$\left\{\frac{\lambda_i^k g_i(x^{k+1})}{\tau}\right\}_U \to 0,$$

thus,

$$\left\{\lambda_i^k g_i(x^{k+1})\right\}_U \to 0. \tag{23}$$

We see that (23) contradicts the expression (22). So, we get

$$\lim_{k\to\infty}\left(\lambda_i^k g_i(x^{k+1})\right) = 0,\ i = 1, ..., m. \tag{24}$$

Because $\Phi(\cdot)$ is a concave function and from Remark 2, we get

$$\Phi(\lambda^{k+1}) - \Phi(\lambda^k)$$

$$\geq \sum_{i=1}^{m}\left(g_i(x^{k+1})\right)\left(\lambda_i^k - \lambda_i^{k+1}\right) \geq 0. \tag{25}$$

From Proposition 2 we know that $\{\Phi(\lambda^k)\}$ is convergent, so it follows: $\lim_{k\to\infty}\left\{\Phi(\lambda^{k+1}) - \Phi(\lambda^k)\right\} = 0$, and from (25) we obtain

$$\lim_{k\to\infty}\sum_{i=1}^{m}\left(g_i(x^{k+1})\right)\left(\lambda_i^k - \lambda_i^{k+1}\right) = 0. \tag{26}$$

Since $\left(g_i(x^{k+1})\right)\left(\lambda_i^k - \lambda_i^{k+1}\right) \geq 0$ from (26) and (24), it follows that

$$\lim_{k\to\infty}\left(\lambda_i^{k+1} g_i(x^{k+1})\right) = 0,\ i = 1, ..., m. \tag{27}$$

∎

Let us consider the last assumption:

C4. The whole sequence to be $\left\{x^k\right\}$ is convergent to $\tilde{x}$, where $\tilde{x}$ is assumed a feasible point, i.e., $g_i(\tilde{x}) \geq 0,\ i = 1, ..., m$.

A similar assumption to **C4** can also be seen in [15]. Finally, we ensure that the subsequence generated by the algorithm DHALA converges to a KKT point.

Theorem 3. *The convex problem (P) satisfies* **C1**, **C2**, **C3** *and* **C4**. *Let sequences $\{x^k\}$ and $\{\lambda^k\}$ be generated by DHALA. Then, any limit point of a subsequence $\{x^k\}$ and $\{\lambda^k\}$ is an optimal solution Lagrange multiplier pair for the problem (P).*

Proof. Let $\tau > 0$ be fixed. By hypothesis, we have that $\lim_{k\to\infty} x^k = \bar{x}$ and $\lim_{k\to\infty}\lambda^k = \bar{\lambda}$. Henceforth, we can consider the following convergent subsequences $\lim_{k\to\infty} x^k = \bar{x}$ and $\lim_{k\to\infty}\lambda^k = \bar{\lambda}$ with $k \in K_2 \subset I\!N$.

In a previous result, we ensure feasibility, i.e., $g_i(\bar{x}) \geq 0,\ i = 1, ..., m$. From Proposition 1, we obtain, $\lim_{k\to\infty}\lambda_i^k = \bar{\lambda}_i \geq 0,\ i = 1, ..., m$. Passsing the limit in (27), we have $\lim_{k\to\infty}\left(\lambda_i^k g_i(x^k)\right) = \bar{\lambda}_i g_i(\bar{x}) = 0,\ \forall i = 1, ..., m$. Moreover, passing the limit in (9), we obtain

$$\nabla_x L(\bar{x}, \bar{\lambda}) = \nabla f(\bar{x}) - \sum_{i=1}^{m}\bar{\lambda}_i\nabla g_i(\bar{x}) = 0.$$

Thus, $(\bar{x}, \bar{\lambda})$ satisfies the KKT conditions for all $i = 1, ..., m$, and $(\bar{x}, \bar{\lambda})$ is a KKT point. Thus $\bar{x}$ is optimal for the problem (P) and $\bar{\lambda}$ is a Lagrange multiplier. ∎

4. Computational illustration

In this section, we are going to use the algorithms HALA and DHALA to solve the same problems. After obtaining the results, we will observe and comment the differences between these two algorithms.

The computational results presented below were obtained with a preliminary Fortran implementation for HALA and DHALA. The program was compiled with the GNU Fortran compiler version 4:7.4.0-1ubuntu2.3. The numerical Experiments were conducted on a Notebook with operating system Ubuntu 18.04.5, CPU i7-3632QM and 8GB RAM. The unconstrained minimization tasks were carried out by means of a Quasi-Newton algorithm employing the BFGS updating formula, with the function VA13 from HSL library [16]. The algorithm stops when the solution is viable (feasible) and the absolute value of the difference between two consecutives solutions is less than 10^{-7}.
For a better understanding of our work, we are going to present the algorithm HALA (for more details see [6]) below:

Algorithm HALA

Step 1. Let $k := 0$. Take initial values $\lambda^0 = (\lambda_1^0, ..., \lambda_m^0) \in I\!R_{++}^m$ and $\tau \in I\!R_{++}$.

Step 2. Solve the unconstrained minimization problem:

$$x^{k+1} \in argmin_{x\in I\!R^n}\ L_H(x, \lambda^k, \tau)$$

$$= argmin_{x\in I\!R^n}\left\{f(x) + \sum_{i=1}^{m} P(g_i(x), \lambda_i, \tau)\right\}.$$

Step 3. Update the Lagrange multipliers:

$$\lambda_i^{k+1} = \lambda_i^k\left(1 - \frac{\lambda_i^k g_i(x^{k+1})}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2 + \tau^2}}\right), \tag{28}$$

$\forall\ i = 1, ..., m.$

Step 4. If the pair (x^{k+1}, λ^{k+1}) satisfies the stopping criteria, then stop.

Step 5. $k := k + 1$. Go to Step 2.

4.1. Test problems

The problems can be found in the book [17].

Problem 1. *Problem 11 (HS11) of [17].*

$$\min_{x \in I\!R^2} f(x) = (x_1 - 5)^2 + x_2^2 - 25$$
$$s.t. \qquad g_1(x) = -x_1^2 + x_2 \geq 0.$$

Starting with $x^0 = (4.9,\ 0.1)$ (not feasible) and $f(x^0) = -24.98$. The minimum value is $f(x^*) = -8.498464223$.

Problem 2. *Problem 66 (HS66) of [17].*

$$\min_{x \in I\!R^3} f(x) = 0.2x_3 - 0.8x_1$$
$$\begin{aligned} s.t.\ g_1(x) &= x_2 - e^{x_1} \geq 0, \\ g_2(x) &= x_3 - e^{x_2} \geq 0, \\ g_3(x) &= x_1 \geq 0, \\ g_4(x) &= x_2 \geq 0, \\ g_5(x) &= x_3 \geq 0, \\ g_6(x) &= 100 - x_1 \geq 0, \\ g_7(x) &= 100 - x_2 \geq 0, \\ g_8(x) &= 10 - x_3 \geq 0. \end{aligned}$$

Starting with $x^0 = (0, 1.05, 2.9)$ (feasible) and $f(x^0) = 0.58$. The minimum value is $f(x^*) = 0.5181632741$ and the optimal solution is $x^* = (0.1841264879,\ 1.202167873,\ 3.327322322)$.

4.2. Results

Table 1 presents the time used by the algorithm to converge. Both algorithms use 5 iterations to solve problem HS11 and 12 iterations to solve problem HS66.

Table 1. Comparison of HALA and DHALA (time in seconds)

Problem	τ	$HALA$	$DHALA$
HS11	0.10E-01	0.000336	0.000226
HS66	0.10E-02	0.001239	0.000759

- Example 1
  - HALA:
    $x^* = (0.123477247E + 01,$ $0.152466328E + 01)$,

    $\lambda^* = 0.304888381E + 01$,

    $f(x^*) = -0.849846350E + 01$.
  - DHALA:
    $x^* = (0.123477247E{+}01,\ 0.152466328E{+}01)$,

    $\lambda^* = 0.304888381E + 01$,

    $f(x^*) = -0.849846350E + 01$.
- Example 2
  - HALA:
    $x^* = (0.184127435E{+}00,\ 0.120216896E{+}01,\ 0.332732602E + 01)$,

    $\lambda^* = (0.665503228E{+}00,\ 0.199999462E{+}00,\ 0.147229064E{-}05,\ 0.345744295E{-}07,\ 0.451311477E{-}08,\ 0.501818042E{-}11, 0.512143031E{-}11,\ 0.112318220E{-}08)$,

    $f(x^*) = 0.518163256E + 00$.
  - DHALA:
    $x^* = (0.184127436E{+}00,\ 0.120216896E{+}01,\ 0.332732603E + 01)$,

    $\lambda^* = (0.665502922E{+}00,\ 0.199999546E{+}00,\ 0.147229063E{-}05,\ 0.345744284E{-}07,\ 0.451311477E{-}08,\ 0.501818042E{-}11,\ 0.512254052E{-}11,\ 0.112318220E{-}08)$,

    $f(x^*) = 0.518163257E + 00$.

5. Conclusions

In this work, we observed that the convergence of DHALA is similar to the convergence of HALA. The computational illustrations show that DHALA solves the problems in less time when compared to HALA. Additionally both DHALA and HALA solve the same problems in the same number of iterations (see, Table 1). Our algorithms DHALA and HALA converge to the exact solution within the precision of the computer. A limitation of our algorithm is that parameter τ is fixed, despite this limitation, our algorithm converges. For future work, we plan the convergence theory of our algorithm to address multiobjective optimization problems, subsequently applying this expanded framework to the problem investigated in [18]. We are also interested in doing a complexity analysis for our algorithm, similar to work [19]. We also have the interest of solving the subproblem generated by DHALA, with the Quasi Newton algorithm studied in [20].

Acknowledgments

The first author was supported by *Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro* (PDR10/FAPERJ). The second author by COPPETEC Foundation, Brazil. The fourth author by *Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro* (FAPERJ), $E\_26/2021$ - Auxílio Básico à Pesquisa, APQ1.

References

[1] Tseng, P., & Bertsekas, D.P. (1993). On the convergence of the exponential multiplier method for convex programming. *Mathematical Programming*, 60, 1-19. https://doi.org/10.1007/BF01580598

[2] Polyak, R.A. (2001). Log-sigmoid multipliers method in constrained optimization. *Annals of Operations Research*, 101, 427-460. https://doi.org/10.1023/A:1010938423538

[3] Polyak, R., & Teboulle, M. (1997). Nonlinear rescaling and proximal-like methods in convex optimization. *Mathematical Programming*, 76, 265-284. https://doi.org/10.1007/BF02614440

[4] Kort, B.W., & Bertsekas, D.P. (1976). Combined primal-dual and penalty methods for convex programming. *SIAM Journal on Control and Optimization*, 14, 268-294. https://doi.org/10.1137/0314020

[5] Rockafellar, R.T. (1976). Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of Operations Research*, 1, 97-116. https://doi.org/10.1287/moor.1.2.97

[6] Mallma-Ramirez, L. (2022). *The Hyperbolic Augmented Lagrangian Algorithm*. Ph.D. Thesis. Federal University of Rio de Janeiro/COPPE, Rio de Janeiro, Brazil. https://www.cos.ufrj.br/uploadfile/publicacao/3038.pdf

[7] Xavier, A.E. (1982a). *Penalização Hiperbólica-Um Novo Método para Resolução de Problemas de Otimização* . MSc Thesis. Federal University of Rio de Janeiro/COPPE, Rio de Janeiro, Brazil.

[8] Xavier, A.E. (1992). *Penalização Hiperbólica*. Ph.D. Thesis. Federal University of Rio de Janeiro/COPPE, Rio de Janeiro, Brazil.

[9] Kanzow, C., & Steck, D. (2017). An example comparing the standard and safeguarded augmented Lagrangian methods. *Operations Research Letters*, 45, 598-603. https://doi.org/10.1016/j.orl.2017.09.005

[10] Xavier, A. E. (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

[11] Zangwill, W.I. (1967). Non-linear programming via penalty function. *Management Science*, 13, 344-358. https://doi.org/10.1287/mnsc.13.5.344

[12] Auslender, A., Teboulle, M., & Ben-Tiba, S. (1999). Interior proximal and multiplier methods based on second order homogeneous kernels. *Mathematics of Operations Research*, 24, 645-668. https://doi.org/10.1287/moor.24.3.645

[13] Polyak, R.A. (2002). Nonlinear rescaling vs. smoothing technique in convex optimization. *Mathematical Programming*, 92, 197-235. https://doi.org/10.1007/s101070100293

[14] Rockafellar, R.T. (1970). *Convex Analysis.* Princeton University Press, Princeton, NJ.

[15] Hartman, J.K. (1975). Iterative determination of parameters for an exact penalty function. *Journal of Optimization Theory and Applications*, 16, 49-66. https://doi.org/10.1007/BF00935623

[16] Harwell Subroutine Library (2002). A collection of Fortran codes for large scale scientific computation, https://www.hsl.rl.ac.uk/

[17] Hock, W., & Schittkowski, K. (1981). *Test Examples for Nonlinear Programming Codes.* Lecture Notes in Economics and Mathematical Systems 187, Springer-Verlag, Berlin, Germany. https://doi.org/10.1007/978-3-642-48320-2

[18] Tirkolaee, E.B., Goli, A., Gütmen, S., Weber, G.W., & Szwedzka, K. (2023). A novel model for sustainable waste collection arc routing problem: Pareto-based algorithms. *Annals of Operations Research*, 324, 189-214. https://doi.org/10.1007/s10479-021-04486-2

[19] Grapiglia, G.N. (2021). Worst-case evaluation complexity of a quadratic penalty method for nonconvex optimization. *Optimization Methods and Software*, 38, 781-803. https://doi.org/10.1080/10556788.2023.2189711

[20] Belloufi, M., Benzine, R., & Laskri, Y. (2013). Modification of the Armijo line search to satisfy the convergence properties of HS method. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 3, 145-152. https://doi.org/10.11121/ijocta.01.2013.00141

Lennin Mallma Ramirez *received his Ph.D degrees in PESC/COPPE/UFRJ, Brazil (2022). He was a visiting research at the ICTEAM/UCLouvain, Belgium.nHe is currently doing a Postdoc in PESC/COPPE/UFRJ, Rio de Janeiro, Brazil. He is interested in researching continuous optimization algorithms.*
https://orcid.org/0000-0003-3579-1167

Nelson Maculan *Professor Emeritus Federal University of Rio de Janeiro, DHR Management Sciences, University of Paris-Dauphine, PhD Production Engineering, Federal University of Rio de Janeiro. Editor in Chief of RAIRO Operations Research journal.*
https://orcid.org/0000-0002-3897-3356

Adilson Elias Xavier *received his Ph.D degrees in PESC/COPPE/UFRJ, Brazil (1992). He is a professor at PESC/COPPE/UFRJ, Brazil. Working mainly*

on the following topics: Hyperbolic Penalty, Optimization, Nonlinear Programming, Multiplier Methods.
https://orcid.org/0000-0003-4337-8049

Vinicius Layter Xavier *received his Ph.D degrees in PESC/COPPE/UFRJ, Brazil (2016). He is a professor at the IME/UERJ, Rio de Janeiro, Brazil. Working mainly on the following topics: Multivariate Statistics, Cluster Analysis, Nondifferentiable Programming, Min-Sum-Min Problems, Statistical Learning and Pattern Recognition.*
https://orcid.org/0000-0002-7290-0652

An International Journal of Optimization and Control: Theories & Applications (http://www.ijocta.org)