

HALA in Convex Optimization

Lennin Mallma Ramirez

PESC/Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil
lenninmr@cos.ufrj.br

Alexandre Belfort de Almeida Chiacchio

GTA/DEL/Poli/Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil
alexandrebelmont@poli.ufrj.br

Nelson Maculan

PESC/Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil
maculan@cos.ufrj.br

Rodrigo de Souza Couto

PEE/GTA/Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil
rodrigo.couto@coppe.ufrj.br

Adilson Elias Xavier

PESC/Universidade Federal do Rio de Janeiro
Rio de Janeiro, Brazil
adilson.xavier@gmail.com

Vinicius Layter Xavier

IME/Universidade do Estado do Rio de Janeiro
Rio de Janeiro, Brazil
viniciuslx@ime.uerj.br

RESUMO

O Algoritmo Lagrangiano Aumentado Hiperbólico (HALA) é um novo algoritmo recentemente proposto na literatura, para resolver o problema de programação não linear convexa restrita. Apresentamos o resultado da convergência e algumas propriedades deste algoritmo. Neste trabalho, mostramos principalmente vários experimentos computacionais usando o HALA.

PALAVRAS CHAVE. Problema convexo. Algoritmo Lagrangeano aumentado. Programação não linear

Tópicos: Programação não linear

ABSTRACT

The Hyperbolic Augmented Lagrangian Algorithm (HALA) is a new algorithm recently proposed in the literature, for solving the convex constrained nonlinear programming problem. We present the convergence result and some properties of this algorithm. In this work, we mainly show various computational experiments using HALA.

KEYWORDS. Convex problem. Augmented Lagrangian algorithm. Nonlinear Programming

Paper topics: Mathematical Programming

1. Introduction

In this work, we are interested in the nonlinear programming problem subject to inequality constraints

$$\min f(x) \text{ s.t. } g_i(x) \geq 0, \quad i = 1, \dots, m, \quad (1.1)$$

where f and g_i , $i = 1, \dots, m$ are real-valued functions defined on $\mathbb{R}^n$.

The augmented Lagrangian algorithms that solve problems similar to (1.1), are: Hestenes [1969], Powell [1969] and Rockafellar [1974]. On the other hand, the nonquadratic Lagrangian algorithms also solve this problem, see for example Kort e Bertsekas [1973], Polyak [1992], Tseng e Bertsekas [1993] and Polyak [2001].

In Xavier [1982], the Hyperbolic Penalty Algorithm (HPA) is proposed. This algorithm is based on the hyperbolic penalty function (HPF). Later, this function is studied in Xavier [1992] and Xavier [2001]. HPA induces ideas to build a new algorithm of augmented Lagrangian type see Xavier [1992]. On the other hand, the function HPF was important for the construction of the hyperbolic smoothing clustering algorithm (HSCA), see Xavier [2010]. The computational results of HSCA are interesting see Xavier e Oliveira [2005], Xavier e Xavier [2011], Bagirov et al. [2020], Bagirov et al. [2015], Souza et al. [2011], Xavier e Xavier [2016] and Xavier et al. [2015]. Due to what was mentioned above, we are interested in continuing to study the HPF through an augmented Lagrangian algorithm.

In Ramirez [2022], the hyperbolic augmented Lagrangian algorithm (HALA) is studied and a result of convergence is assured. HALA is mainly inspired by the works of **Adilson Elias Xavier** (see Xavier [1982] and Xavier [1992]). This algorithm is based on HPF. HPF has the characteristic of being continuously differentiable, and this offers us the possibility to use the best unconstrained minimization techniques which use second-order derivatives. HPF is important in the construction of the hyperbolic augmented Lagrangian function (see Xavier [1992]). Now, in this work, after approximately 30 years, we return to the hyperbolic augmented Lagrangian function to study it.

In this work, we mainly present several computational experiments to see the performance of HALA. The problems we solve are convex problems known in the literature. We will observe that our algorithm converges towards the optimal solution, despite our considering a fixed penalty parameter. We also compare our computational results obtained with another augmented Lagrangian algorithm.

This work is organized as follows: In Section 2, we present some basic results that will be important for the understanding of our work. In Section 3, we present HALA and also study some features of this algorithm. In Section 4, we analyze the convergence of HALA. In Section 5, we computationally illustrate our theoretical results and guarantee the performance of HALA. In Section 6, some conclusions about our work are presented. In Section 7, we propose some future works.

2. Background

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

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

where

$$S = \{x \in \mathbb{R}^n \mid g_i(x) \geq 0, \quad i = 1, \dots, m\}$$

is the convex feasible set of the problem (P) and where the function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex, $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$, are concave functions, and we assume that f and g_i are continuously differentiable. 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 $g_i(\hat{x}) > 0$, $i = 1, \dots, m$.

The Lagrangian function of the problem (P) is $L : \mathbb{R}^n \times \mathbb{R}_+^m \rightarrow \mathbb{R}$, and it is defined as

$$L(x, \lambda) = f(x) - \sum_{i=1}^m \lambda_i g_i(x), \quad (2.2)$$

where, $\lambda_i \geq 0$, $i = 1, \dots, m$, are called dual variables. Due to assumption **C2**, the following results will occur: there exists $\lambda^* = (\lambda_1^*, \dots, \lambda_m^*)$, such that, the Karush-Kuhn-Tucker (KKT) conditions hold true,

$$\nabla_x L(x^*, \lambda^*) = \nabla f(x^*) - \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) = 0, \quad (2.3)$$

$$\lambda_i^* g_i(x^*) = 0, \quad i = 1, \dots, m, \quad (2.4)$$

$$g_i(x^*) \geq 0, \quad i = 1, \dots, m, \quad (2.5)$$

$$\lambda_i^* \geq 0, \quad i = 1, \dots, m. \quad (2.6)$$

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

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

It is known that Λ^* is a bounded set (and hence compact set) due to **C2**. The dual function $\Phi : \mathbb{R}_+^m \rightarrow \mathbb{R}$ is defined as

$$\Phi(\lambda) = \inf_{x \in \mathbb{R}^n} L(x, \lambda), \quad (2.7)$$

and the dual problem consists of finding

$$(D) \quad \lambda \in \Lambda^* = \operatorname{argmax}\{\Phi(\lambda) \mid \lambda \in \mathbb{R}_+^m\}.$$

2.1. Hyperbolic Penalty

The hyperbolic penalty method is meant to solve the problem (P). The penalty method adopts the hyperbolic penalty function (HPF)

$$P(y, \lambda, \tau) = -\lambda y + \sqrt{(\lambda y)^2 + \tau^2}, \quad (2.8)$$

where $P : \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++} \rightarrow \mathbb{R}$. The HPF is originally proposed in Xavier [1982] and studied in Xavier [2001]. The HPF is equivalent to a smoothing of the penalty studied by Zangwill [1967]. Let us consider the following properties of the function P :

P0) $P(y, \lambda, \tau)$ is continuously differentiable at y , for $\tau > 0$ and λ fixed.

P1) $P(y, \lambda, \tau)$ is a strictly decreasing function of y , i.e.,

$$\nabla_y P(y, \lambda, \tau) = -\lambda \left(1 - \frac{\lambda y}{\sqrt{(\lambda y)^2 + \tau^2}} \right) \leq 0,$$

for $\tau > 0$ and $\lambda > 0$.

P2) $P(y, \lambda, \tau)$ is convex function of y , i.e.,

$$\nabla_{yy}^2 P(y, \lambda, \tau) = \frac{\lambda^2 \tau^2}{\left((\lambda y)^2 + \tau^2 \right)^{\frac{3}{2}}} > 0.$$

3. Hyperbolic Augmented Lagrangian Algorithm

We define the Hyperbolic Augmented Lagrangian Function (HALF) of problem (P) by $L_H : \mathbb{R}^n \times \mathbb{R}_{++}^m \times \mathbb{R}_{++} \rightarrow \mathbb{R}$,

$$\begin{aligned} L_H(x, \lambda, \tau) &= f(x) + \sum_{i=1}^m P(g_i(x), \lambda_i, \tau) \\ &= f(x) + \sum_{i=1}^m \left(-\lambda_i g_i(x) + \sqrt{(\lambda_i g_i(x))^2 + \tau^2} \right), \end{aligned} \quad (3.9)$$

where $\tau > 0$ is the penalty parameter. Note that this function belongs to the class C^∞ if the involved functions $f(x)$ and $g_i(x)$, $i = 1, \dots, m$, do too. Now, let us consider the following assumption.

C3. For every $\tau > 0$ and $\lambda > 0$, the level set

$$M = \{x \in \mathbb{R}^n \mid L_H(x, \lambda, \tau) \leq \alpha\},$$

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

Remark 3.1 The assumption **C3** is verified if in particular the function f is strongly convex. That way L_H will also be strongly convex. Other works that also use the strong convexity assumptions are: Kort e Bertsekas [1973] and Auslender et al. [1999].

Now we present HALA to solve the nonlinear problem (P).

3.1. Algorithm HALA

Step 1. Let $k := 0$. Take initial values $\lambda^0 = (\lambda_1^0, \dots, \lambda_m^0) \in \mathbb{R}_{++}^m$ and $\tau \in \mathbb{R}_{++}$.

Step 2. Solve the unconstrained minimization problem:

$$\begin{aligned}
 x^{k+1} &\in \operatorname{argmin}_{x \in \mathbb{R}^n} L_H(x, \lambda^k, \tau) \\
 &= \operatorname{argmin}_{x \in \mathbb{R}^n} \left\{ f(x) + \sum_{i=1}^m \left(-\lambda_i^k g_i(x) + \sqrt{(\lambda_i^k g_i(x))^2 + \tau^2} \right) \right\}.
 \end{aligned}$$

Step 3. Update the Lagrange multipliers:

$$\lambda_i^{k+1} = \lambda_i^k \left(1 - \frac{\lambda_i^k g_i(x^{k+1})}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2 + \tau^2}} \right), \quad i = 1, \dots, m. \quad (3.10)$$

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**.

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

$$L_H(x^{k+1}, \lambda^k, \tau) = \min_{x \in \mathbb{R}^n} L_H(x, \lambda^k, \tau).$$

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

$$\nabla f(x^{k+1}) - \sum_{i=1}^m \lambda_i^k \left(1 - \frac{\lambda_i^k g_i(x^{k+1})}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2 + \tau^2}} \right) \nabla g_i(x^{k+1}) = 0. \quad (3.11)$$

Substituting (3.10) in (3.11), we have:

$$\nabla_x L_H(x^{k+1}, \lambda^k, \tau) = \nabla f(x^{k+1}) - \sum_{i=1}^m \lambda_i^{k+1} \nabla g_i(x^{k+1}) = \nabla_x L(x^{k+1}, \lambda^{k+1}) = 0, \quad (3.12)$$

for any $\tau > 0$. On the other hand, we know that

$$\Phi(\lambda^{k+1}) = f(x^{k+1}) - \sum_{i=1}^m \lambda_i^{k+1} g_i(x^{k+1}). \quad (3.13)$$

From (3.13) we obtain

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

where $\partial \Phi(\lambda^{k+1})$ is the subdifferential of $\Phi(\lambda)$ at $\lambda = \lambda^{k+1}$. In the following remark, we analyze what happens with Lagrange multipliers (3.10) in **Step 3** depending on the type of restriction. We define the following sets of indices

$$I_0(x) = \{i \in \{1, \dots, m\} \mid g_i(x) = 0\},$$

$$I_-(x) = \{i \in \{1, \dots, m\} \mid g_i(x) < 0\},$$

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

Remark 3.2 Let $\{\lambda^k\}$ be a sequence generated by HALA such that $\lambda_i^k > 0$, $i = 1, \dots, m$ and let $\tau > 0$ fixed. Let us consider the following cases:

(c1) If $i \in I_0(x)$, then we have at the k -th iteration $g_i(x^{k+1}) = 0$, then by (3.10), we get,

$$\lambda_i^{k+1} = \lambda_i^k.$$

(c2) If $i \in I_+(x)$, then we have at the k -th iteration $g_i(x^{k+1}) > 0$, then by (3.10), we get,

$$\lambda_i^k > \lambda_i^{k+1}.$$

(c3) If $i \in I_-(x)$, then we have at the k -th iteration $g_i(x^{k+1}) < 0$, then by (3.10), we get,

$$\lambda_i^k < \lambda_i^{k+1}.$$

Proposition 3.1 Let $\{\lambda^k = (\lambda_1^k, \dots, \lambda_m^k) \mid k = 1, 2, \dots\} \subset \mathbb{R}^m$. If

$$\lambda^k \in \mathbb{R}_{++}^m, \quad \text{then} \quad \lambda^{k+1} \in \mathbb{R}_{++}^m, \quad i = 1, \dots, m.$$

Proof. We know that

$$\left(\lambda_i^k g_i(x^k)\right)^2 < \left(\lambda_i^k g_i(x^k)\right)^2 + \tau^2, \quad i = 1, \dots, m,$$

it follows that

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

$$0 < 1 - \frac{\lambda_i^k g_i(x^k)}{\sqrt{(\lambda_i^k g_i(x^k))^2 + \tau^2}} < 2, \quad i = 1, \dots, m,$$

so,

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

By the above inequality and by (3.10), we get $\lambda_i^{k+1} > 0$, $i = 1, \dots, m$. ■

Remark 3.3 By C3 and Proposition 3.1, we obtain that HALA is well defined.

Proposition 3.2 The sequence $\{\lambda^k\}$ generated by the HALA is bounded.

Proof. See Proposition 3.2.3 of Ramirez [2022]. ■

Lemma 3.1 Let $d > 0$ and a sequence $\{a^k\} \subset \mathbb{R}_+$. If

$$\lim_{k \rightarrow \infty} \left(a^k / \sqrt{a^k + d}\right) = 0, \quad \text{then} \quad \lim_{k \rightarrow \infty} a^k = 0.$$

Proof. See Lemma 3.2.1 of Ramirez [2022]. ■

Theorem 3.1 *Let the sequences $\{x^k\}$ and $\{\lambda^k\}$ be generated by HALA. Then*

$$\lim_{k \rightarrow \infty} (\lambda_i^k g_i(x^k)) = 0, \quad i = 1, \dots, m.$$

Proof. Let $\tau > 0$ be fixed. Since $\Phi(\cdot)$ is concave, we have

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

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

$$\sum_{k=1}^{\infty} \sum_{i=1}^m \left(\frac{(\lambda_i^k g_i(x^{k+1}))^2}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2 + \tau^2}} \right) \leq \lim_{k \rightarrow \infty} (\Phi(\lambda^k) - \Phi(\lambda^1)) \leq f^* - \Phi(\lambda^1) < \infty.$$

Therefore, for the test of comparison, we obtain

$$\lim_{k \rightarrow \infty} \sum_{i=1}^m \left(\frac{(\lambda_i^k g_i(x^{k+1}))^2}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2 + \tau^2}} \right) = 0. \quad (3.14)$$

We note each term in the summation of (3.14) is nonnegative, thus

$$\lim_{k \rightarrow \infty} \left(\frac{(\lambda_i^k g_i(x^{k+1}))^2}{\sqrt{(\lambda_i^k g_i(x^{k+1}))^2 + \tau^2}} \right) = 0, \quad i = 1, \dots, m. \quad (3.15)$$

In (3.15), we can apply the Lemma 3.1 with $a^k = (\lambda_i^k g_i(x^{k+1}))^2$ and $d = \tau^2$ and, thus, we obtain $\lim_{k \rightarrow \infty} (\lambda_i^k g_i(x^{k+1}))^2 = 0$, $i = 1, \dots, m$, so,

$$\lim_{k \rightarrow \infty} (\lambda_i^k g_i(x^{k+1})) = 0, \quad i = 1, \dots, m.$$

Then we can obtain that

$$\lim_{k \rightarrow \infty} (\lambda_i^{k+1} g_i(x^{k+1})) = 0, \quad i = 1, \dots, m.$$

For more details see Ramirez [2022]. ■

4. Main Result

In this section, we are going to consider the following assumption.

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

A similar assumption to **C4** can also be seen in Hartman [1975] and Nguyen e Strodiot [1979].

Theorem 4.1 *The convex problem (P) satisfies C1, C2, C3 and C4. Let sequences $\{x^k\}$ and $\{\lambda^k\}$ generated by HALA. Then, any limit point of a subsequence $\{x^k\}$ and $\{\lambda^k\}$ are an optimal solution-Lagrange multiplier pair for the problem (P).*

Proof. See Theorem 3.3.1 of Ramirez [2022]. ■

Remark 4.1 *Let us note that if (x^*, λ^*) is a point of KKT, then we obtain that (where τ is fixed)*

$$\begin{aligned} L_H(x^*, \lambda^*, \tau) &= f(x^*) + \sum_{i=1}^m \left(-\lambda_i^* g_i(x^*) + \sqrt{(\lambda_i^* g_i(x^*))^2 + \tau^2} \right) \\ &= f(x^*) + \sum_{i=1}^m \tau \\ &= f(x^*) + m\tau. \end{aligned}$$

Also note the following, if

$$\nabla L_H(x^*, \lambda^*, \tau) = \nabla f(x^*) - \sum_{i=1}^m \lambda_i^* \left(1 - \frac{\lambda_i^* g_i(x^*)}{\sqrt{(\lambda_i^* g_i(x^*))^2 + \tau^2}} \right) \nabla g_i(x^*) = 0.$$

so,

$$\nabla L_H(x^*, \lambda^*, \tau) = \nabla f(x^*) - \sum_{i=1}^m \lambda_i^* \nabla g_i(x^*) = \nabla L(x^*, \lambda^*) = 0.$$

Remark 4.2 *We can rewrite the function L_H as follows*

$$\begin{aligned} L_H(x, \lambda, \tau) &= f(x) + \sum_{i=1}^m \left(-\lambda_i g_i(x) + \sqrt{(\lambda_i g_i(x))^2 + \tau^2} \right) \\ &= f(x) + \sum_{i=1}^m \tau \left(-\frac{\lambda_i g_i(x)}{\tau} + \sqrt{\left(\frac{\lambda_i g_i(x)}{\tau} \right)^2 + 1} \right). \end{aligned}$$

Thus,

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

where $h : \mathbb{R} \rightarrow \mathbb{R}$ is defined as $h(t) = -t + \sqrt{t^2 + 1}$. This is a more “compact” way of working with the function L_H .

5. Computational Experiments

HALA was implemented with Python, using `numpy` and `scipy` modules. All variables used were typed as 64bit floating point using `numpy`. In addition, the **Step 2** was implemented using `scipy.optimize.minimize()` function, with either `method='BFGS'` or `method='L-BFGS-B'` flag, that signifies using either Broyden, Fletcher, Goldfarb, and Shanno (BFGS) algorithm, a Quasi-Newton method, or Bounded Limited Memory BFGS (L-BFGS-B). It is important to note that the bounds on the variables were not passed as arguments in the function call, but as additional restrictions, so in practice, when calling `method='L-BFGS-B'`, it uses the L-BFGS algorithm (Byrd et al. [1995]).

All experiments were run on a Notebook with Microsoft Windows 11 Home 10.0.22621 operating system, CPU i7-8565U, and 16GB RAM, with stopping criteria $\|x^k - x^{k+1}\| < 10^{-3}$ and feasibility of x^{k+1} . The descriptions of the columns of Table 1 are below:

P The problem name.

n The number of variables.

m The number of restrictions.

IT External iterations of da Fonseca Prudente [2012].

HALA External iterations of HALA.

Found objective function value at x found by HALA.

Sol $f(x^*)$ objective function value at x^* , the actual solution.

τ Penalty Parameter value used.

Method Algorithm used for internal iterations, the **Step 2** of HALA. Either BFGS or L-BFGS.

From the Table 1, we can see that the number of iterations of HALA is similar to the algorithm studied in da Fonseca Prudente [2012]. Our algorithm HALA converges to the exact solution within the precision of the computer, all this despite considering a fixed penalty parameter.

Table 1

P	n	m	AL	HALA	Found	Sol	τ	Method
HS1	2	1	0	2	0.00	0.00	1e-08	BFGS
HS2	2	1	0	2	0.05	0.05	1e-08	BFGS
HS3	2	1	0	2	0.00	0.00	1e-08	L-BFGS
HS10	2	1	8	7	-1.00	-1.00	1e-08	L-BFGS
HS11	2	1	9	3	-8.50	-8.50	1e-07	L-BFGS
HS12	2	1	3	7	-30.00	-30.00	1e-08	L-BFGS
HS21	2	5	1	3	-99.96	-99.96	1e-08	BFGS
HS22	2	2	8	8	1.00	1.00	1e-08	BFGS
HS30	3	7	1	8	1.00	1.00	1e-08	BFGS
HS34	3	8	3	7	-0.83	-0.83	1e-07	BFGS
HS35	3	4	3	6	0.11	0.11	1e-08	L-BFGS
HS43	4	3	3	2	-44.00	-44.00	1e-04	L-BFGS
HS64	3	4	10	9	6299.84	6299.84	1e-06	BFGS
HS65	3	7	3	5	0.95	0.95	1e-08	BFGS
HS66	3	8	3	2	0.52	0.52	1e-06	BFGS
HS72	4	10	14	17	727.68	727.68	1e-04	BFGS
HS76	4	7	3	9	-4.68	-4.68	1e-08	L-BFGS
HS100	7	4	3	2	680.63	680.63	1e-01	BFGS
HS110	10	20	0	2	-45.78	-45.78	1e-08	BFGS
HS113	10	8	3	17	24.31	24.31	1e-01	BFGS

6. Conclusions

A new augmented Lagrangian type function is studied in this work. Through the computational experiments, we observed the performance of HALA. We observe that the number of iterations used by our algorithm is similar to that of another augmented Lagrangian algorithm, despite our consideration of a fixed penalty parameter.

7. Future Works

A future work is to study the dislocation hyperbolic augmented Lagrangian function (DHALF) defined as follows:

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

Let us observe that a point (x^*, λ^*) of KKT in the function is as follows:

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

So,

$$L_H(x^*, \lambda^*, \tau) = f(x^*).$$

We will investigate if there is any significant difference in solving HALF or DHALF. On the other hand, other future works are: finding adequate conditions to update the penalty parameter

and extending the convergence theory of HALA for the nonconvex case.

acknowledgements The first author is supported by PDR10/FAPERJ Brazil, the third author by COPPETEC Foundation, Brazil and the fourth by CAPES and FAPERJ, Brazil.

References

- Auslender, A., Teboulle, M., e Ben-Tiba, S. (1999). Interior proximal and multiplier methods based on second order homogeneous kernels. *Mathematics of Operations Research*, 24(3):645–668.
- Bagirov, A., Karmitsa, N., e Taheri, S. (2020). Partitional clustering via nonsmooth optimization. *Clustering Via Optimization*. Springer.
- Bagirov, A. M., Ordin, B., Ozturk, G., e Xavier, A. E. (2015). An incremental clustering algorithm based on hyperbolic smoothing. *Computational Optimization and Applications*, 61:219–241.
- Byrd, R. H., Lu, P., Nocedal, J., e Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on scientific computing*, 16(5):1190–1208.
- da Fonseca Prudente, L. (2012). *Inviabilidade em mÃ©todos de Lagrangiano Aumentado*. PhD thesis, Universidade Estadual de Campinas (UNICAMP). Instituto de Matemática.
- Hartman, J. K. (1975). Iterative determination of parameters for an exact penalty function. *Journal of Optimization Theory and Applications*, 16:49–66.
- Hestenes, M. R. (1969). Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320.
- Kort, B. e Bertsekas, D. (1973). Multiplier methods for convex programming. In *1973 IEEE Conference on Decision and Control including the 12th Symposium on Adaptive Processes*, p. 428–432. IEEE.
- Nguyen, V. e Strodiot, J. (1979). On the convergence rate for a penalty function method of exponential type. *Journal of Optimization Theory and Applications*, 27:495–508.
- Polyak, R. (1992). Modified barrier functions (theory and methods). *Mathematical programming*, 54:177–222.
- Polyak, R. A. (2001). Log-sigmoid multipliers method in constrained optimization. *Annals of operations Research*, 101:427–460.
- Powell, M. J. (1969). A method for nonlinear constraints in minimization problems. *Optimization*, p. 283–298.
- Ramirez, L. M. (2022). *The Hyperbolic Augmented Lagrangian Algorithm (The Carioca Algorithm)*. PhD thesis, Universidade Federal do Rio de Janeiro.
- Rockafellar, R. T. (1974). Augmented lagrange multiplier functions and duality in nonconvex programming. *SIAM Journal on Control*, 12(2):268–285.
- Souza, M., Xavier, A. E., Lavor, C., e Maculan, N. (2011). Hyperbolic smoothing and penalty techniques applied to molecular structure determination. *Operations Research Letters*, 39(6): 461–465.

- Tseng, P. e Bertsekas, D. P. (1993). On the convergence of the exponential multiplier method for convex programming. *Mathematical programming*, 60(1-3):1–19.
- Xavier, A. E. (1982). *Penalização hiperbólica: Um novo método para resolução de problemas de otimização*. PhD thesis, M. Sc. Thesis-COPPE-UFRJ, Rio de Janeiro.
- Xavier, A. E. (1992). *Penalização hiperbólica*. PhD thesis, DSc. Thesis-COPPE-UFRJ, Rio de Janeiro.
- Xavier, A. E. (2001). Hyperbolic penalty: a new method for nonlinear programming with inequalities. *International Transactions in Operational Research*, 8(6):659–671.
- Xavier, A. E. (2010). The hyperbolic smoothing clustering method. *Pattern Recognition*, 43(3):731–737.
- Xavier, A. E., Gesteira, C. M., e Xavier, V. L. (2015). Solving the continuous multiple allocation p-hub median problem by the hyperbolic smoothing approach. *Optimization*, 64(12):2631–2647.
- Xavier, A. E. e Oliveira, A. A. F. D. (2005). Optimal covering of plane domains by circles via hyperbolic smoothing. *Journal of Global Optimization*, 31(3):493–504.
- Xavier, A. E. e Xavier, V. L. (2011). Solving the minimum sum-of-squares clustering problem by hyperbolic smoothing and partition into boundary and gravitational regions. *Pattern Recognition*, 44(1):70–77.
- Xavier, A. E. e Xavier, V. L. (2016). Flying elephants: a general method for solving non-differentiable problems. *Journal of Heuristics*, 22:649–664.
- Zangwill, W. I. (1967). Non-linear programming via penalty functions. *Management science*, 13(5):344–358.