Systems Engineering and Computer Science Graduate Program

Alberto Luiz Coimbra Institute for Graduate Studies and Research in Engineering

FEDERAL UNIVERSITY OF RIO DE JANEIRO

TECHNICAL REPORT

TR-PESC/Coppe/UFRJ- 797/2025

A Computational Experiment: Hyperbolic Multiplier Algorithm for Mathematical Programming with Complementarity Constraints

Authors:

Lennin Mallma Ramirez[1*], Igor P. S. Pereira[2†], Nelson Maculan[1,3†]

[1*]Programa de Engenharia de Sistemas e Computação

COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.

[2]Programa de Engenharia Química

COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.

[3]Instituto de Matemática

Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.

*Corresponding author(s). E-mail(s): lenninmr@cos.ufrj.br

Contributing authors: ipereira@peq.coppe.ufrj.br; maculan@cos.ufrj.br;

[†]These authors contributed equally to this work.

Abstract

In this note, we use the Dislocation Hyperbolic Augmented Lagrangian Algorithm (DHALA), also known as the Hyperbolic Multiplier Algorithm (HyMA), to solve computationally the mathematical programming with complementarity (or equilibrium) constraints. Furthermore, the subproblem generated by our algorithm is solved with a second-order algorithm and a derivative-free algorithm.

Keywords: Augmented Lagrangian, Complementarity constraints, Python

Publication Date: 05/05/2025

A Computational Experiment: Hyperbolic Multiplier Algorithm for Mathematical Programming with Complementarity Constraints

Lennin Mallma Ramirez[1*], Igor P.S. Pereira[2†] and Nelson Maculan[3†]

[1*]Programa de Engenharia de Sistemas e Computação (COPPE), Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.
[2]Programa de Engenharia Química (COPPE), Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.
[3]Programa de Engenharia de Sistemas e Computação , IM (COPPE), Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brazil.

*Corresponding author(s). E-mail(s): lenninmr@cos.ufrj.br;
Contributing authors: ipereira@peq.coppe.ufrj.br; maculan@cos.ufrj.br;
†These authors contributed equally to this work.

Abstract

In this note, we use the Dislocation Hyperbolic Augmented Lagrangian Algorithm (DHALA), also known as the Hyperbolic Multiplier Algorithm (HyMA) to solve computationally the mathematical programming with complementarity (or equilibrium) constraints. Furthermore, the subproblem generated by our algorithm is solved with a second-order algorithm and a derivative-free algorithm.

Keywords: Augmented Lagrangian, Complementarity constraints, Python

1 Introduction

The nonlinear optimization problem (P) with inequality constraints, is as follows

$$\min_{x\in\mathbb{R}^n} f(x)$$

$$
\begin{aligned}
s.t. \quad g(x) \leq 0, \\
x \in X = \{x \in \mathbb{R}^n : lb \leq x \leq ub\},
\end{aligned}
$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^m$, are continuously differentiable, and X is a box-constraint with $lb, ub \in \mathbb{R}^n$. In this note, we are interested in solving the following problem: mathematical programming with complementarity constraints (MPCC), which is formulated as follows:

$$
\begin{aligned}
\min_{x \in \mathbb{R}^n} f(x) \\
s.t. \quad G(x) \geq 0, \\
D(x) \geq 0, \\
G_l(x) D_l(x) \leq 0, \ l = 1, ..., q, \\
x \in X = \{x \in \mathbb{R}^n : lb \leq x \leq ub\}.
\end{aligned}
$$

where $f : \mathbb{R}^n \to \mathbb{R}$, and $G, D : \mathbb{R}^n \to \mathbb{R}^q$ are differentiable functions, $lb, ub \in \mathbb{R}^n$, and X is a box-constraint. The MPCC models are interesting since they have several applications, some of these applications can be seen in electricity market [3], variational inequality [9], stochastic mathematical programs [19], bilevel programming [6], multiobjective optimization [7] and chemical engineering [17].
In this note, we solve the MPCC problem using DHALA [15], and the subproblem generated by our algorithm is solved using the L-BFGS-B [4]. On the other hand, in [14], the hyperbolic penalty algorithm is considered to solve MPCC.

Notation: Some denotations that will be considered throughout this work are: If $x \in \mathbb{R}^n$, $x = (x_1, ..., x_n)^T$ we denote $x_+ = (\max\{0, x_1\}, ..., \max\{0, x_n\})^T$. If $g : \mathbb{R}^n \to \mathbb{R}^m$, then $g(x)_+ = (\max\{0, \ g_1(x)\}, \max\{0, \ g_2(x)\}, ..., \max\{0, \ g_m(x)\})$. The symbol $|| \cdot ||$ denotes the Euclidean norm, $|| \cdot ||_\infty = \max_{1 \leq i \leq n} \mid x_i \mid$ is the infinity norm, $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \geq 0\}$ and $\mathbb{R}^n_{++} = \{x \in \mathbb{R}^n : x > 0\}$.

2 Basic Results

We define the dislocation hyperbolic penalty function (DHPF) for inequality constraints as follows

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

where $p : \mathbb{R} \times \mathbb{R}_{++} \times \mathbb{R}_{++} \to \mathbb{R}$. This function is a smoothing of the exact penalty function studied in [22]. For more details on this function p, see A.E. Xavier (1982) [20] and (1992)[21]. In the work [15] a convergence analysis is shown for the case of the

Hyperbolic Augmented Lagrangian Function. We redefine the function p as follows

$$p(g_i(x), \lambda_i, \tau) = \frac{1}{\tau} h\left(\lambda_i g_i(x)\tau\right), \; i = 1, ..., m, \tag{1}$$

where $h : \mathbb{R} \to \mathbb{R}$, is defined as $h(t) = t + \sqrt{t^2+1} - 1$. We are going to consider function (1) to define the Dislocation Hyperbolic Augmented Lagrangian Function (DHALF) of problem (P) by $\mathcal{L}_H : \mathbb{R}^n \times \mathbb{R}^m_{++} \times \mathbb{R}_{++} \to \mathbb{R}$,

$$\mathcal{L}_H(x, \lambda, \tau) = f(x) + \sum_{i=1}^{m} \frac{1}{\tau}\, h\left(\lambda_i g_i(x)\tau\right), \tag{2}$$

for all $\tau > 0$, where this parameter is a penalty and smoothing parameter. Next we introduce our algorithm DHALA.

2.1 Dislocation Hyperbolic Augmented Lagrangian Algorithm

Algorithm 1 DHALA [15]

Step 0. Set $k = 0$. Let $(x^0, \lambda^0, \tau^0) \in \mathbb{R}^n \times \mathbb{R}^m_{++} \times \mathbb{R}_{++}$, $0 < \beta < 1$, and $1 < \alpha$. Define, $V_i^0 = \max\{g_i(x^0), 0\}, \forall i = 1, ..., m$.

· **Step 1.** Find $x^k \in \mathbb{R}^n$ as an approximate solution of the subproblem

$$min_{x\in\Omega}\; \mathcal{L}_H(x, \lambda^k, \tau^k). \tag{3}$$

Step 2. Updating of Lagrange multipliers:

$$\lambda_i^{k+1} = \lambda_i^k h'\left(\lambda_i^k g_i(x^k)\tau^k\right), \;\; i = 1, ..., m. \tag{4}$$

Step 3. Update the penalty parameter. Define

$$V_i^k = \lambda_i^k g_i(x^k), \; i = 1, ..., m.$$

If,

$$\max\left\{\left\|g(x^k)_+\right\|_\infty, \left\|V^k\right\|_\infty\right\} \leq \beta \max\left\{\left\|g(x^{k-1})_+\right\|_\infty, \left\|V^{k-1}\right\|_\infty\right\},$$

then $\tau^{k+1} = \tau^k$. Else, $\tau^{k+1} = \alpha\tau^k$.

Step 4. Check convergence:

$$\frac{\left|f(x^k) - f(x^{k-1})\right|}{\left|f(x^{k-1})\right| + 1} \leq Tol \quad and \quad \|\; g_+(x^k)\; \|_\infty \leq Tol,$$

if it has converged, stop. Else $k \leftarrow k+1$ and go to Step 1.

---

3 Computational Experiments

The algorithm DHALA were coded in Python and implemented on a computer with the following characteristics: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz. In Step 1, we used the L-BFGS-B to solve the subproblem. By default, all initial points $x^0 \in \mathbb{R}^n$ considered in the problems will be generated randomly. In what follows, we will maintain the following notation: It is the total number of iterations considered to converge, k iteration number, Tol tolerance, x^* optimal solution, λ^* final multiplier, $time(s)$ measurement of time in seconds, $f(x^*)$ value of the objective function at the optimal solution, λ^0 is the initial multiplier, τ^0 initial penalty/smoothing parameter, also consider that $\alpha = 10$ and $\beta = 0.5$. The stopping criterion it is defined by default for all problems as follows $\frac{|f(x^k)-f(x^{k-1})|}{|f(x^{k-1})|+1} \leq Tol$ and $\left\|g_+(x^k)\right\|_\infty \leq Tol$.

3.1 Problems

Let us consider the following problems, which are known in the literature.

Problem 1 (scholtes3) See [1] and [13]

$$
\begin{aligned}
\min_{x\in\mathbb{R}^2} f(x) = \frac{1}{2}((x_1-1)^2 + (x_2-1)^2)& \\
s.t. \quad g_1(x) = -x_1 \leq 0,& \\
g_2(x) = -x_2 \leq 0,& \\
g_3(x) = x_1x_2 \leq 0,& \\
x \in X = \{x \in \mathbb{R}^2 : -10^6 \leq x_j \leq 10^6, j = 1,2\}.&
\end{aligned}
$$

The solutions are $x^* = (1,0)$ or $x^* = (0,1)$ and $f(x^*) = 0.5$.

Problem 2 (scale4) See [11]

$$
\begin{aligned}
\min_{x\in\mathbb{R}^2} f(x) = (100x_1-1)^2 + (100x_2-1)^2& \\
s.t. \quad g_1(x) = -x_1 \leq 0,& \\
g_2(x) = -x_2 \leq 0,& \\
g_3(x) = x_1x_2 \leq 0,& \\
x \in X = \{x \in \mathbb{R}^2 : -10^6 \leq x_j \leq 10^6, j = 1,2\}.&
\end{aligned}
$$

The solutions are $x^* = (0.01,\ 0)$ or $x^* = (0,\ 0.01)$ and $f(x^*) = 1.0$.

Problem 3 (scale5) See [11]

$$
\begin{aligned}
\min_{x\in\mathbb{R}^2} f(x) = 100((x_1-1)^2 + (x_2-1)^2)& \\
s.t. \quad g_1(x) = -x_1 \leq 0,& \\
g_2(x) = -x_2 \leq 0,&
\end{aligned}
$$

$$g_3(x) = x_1 x_2 \leq 0,$$
$$x \in X = \{x \in \mathbb{R}^2 : -10^6 \leq x_j \leq 10^6, j = 1, 2\}.$$

The solutions are $x^* = (1,0)$ or $x^* = (0,1)$ and $f(x^*) = 100.0$.

Problem 4 See [16]

$$\min_{x \in \mathbb{R}^2} f(x) = x_1 + \frac{1}{2} x_2^2$$
$$s.t. \quad g_1(x) = -x_1 \leq 0,$$
$$g_2(x) = -x_2 \leq 0,$$
$$g_3(x) = x_1 x_2 \leq 0,$$
$$x \in X = \{x \in \mathbb{R}^2 : -10^6 \leq x_j \leq 10^6, j = 1, 2\}.$$

Has a strongly stationary point $x^* = (0,0)$.

3.2 Results

Table 1 Problem 1, $\lambda^0 = 3$, $\tau^0 = 100$, $Tol : 1E-3$

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k) - f(x^{k-1}) \rvert}{\lvert f(x^{k-1}) \rvert + 1}$	$\lVert g_+(x^k) \rVert_\infty$
000	+4.994344e-01	1.000000e+02	2.502828e-01	5.665458e-04
001	+5.000035e-01	1.000000e+02	3.795042e-04	3.480947e-06

$It = 2$
$time(s) = 0.01918459$
$f(x^*) = 0.50000348$
$x^* = (-3.481e-06,\ 1.000e+00)$
$\lambda^* = (2.499e+00,\ 1.670e-05,\ 3.498e+00)$
$g(x^*) = (3.481e-06,\ -1.000e+00,\ -3.481e-06)$

3.3 Some Comments

Computational experiments show that the DHALA algorithm apparently converges toward strict complementarity condition (see [12], [18] and [10]), see comments in Tables 1, 2 and 3. Regarding the Problem 4, we show 4 tables that show our experiments.

- Table 4: the subproblem is solved with the Trust Region algorithm (second-order algorithm).
- Table 5: The subproblem is solved with the L-BFGS-B (second-order algorithm) and we consider the multistart strategy, with 50 points.

Table 2 Problem 2, $\lambda^0 = 3000$, $\tau^0 = 10$, $Tol : 1E-3$

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k)-f(x^{k-1})\rvert}{\lvert f(x^{k-1})\rvert+1}$	$\lVert g_+(x^k)\rVert_\infty$
000	+9.645416e-02	1.000000e+01	6.345153e-01	6.090138e-05
001	+2.464043e-01	1.000000e+01	1.367592e-01	4.211989e-05
002	+7.038716e-01	1.000000e+01	3.670296e-01	1.480642e-05
003	+1.000001e+00	1.000000e+02	1.737980e-01	5.630292e-09
004	+1.000000e+00	1.000000e+02	5.258047e-07	3.787079e-10

$It = 5$

$time(s) = 0.03519821$

$f(x^*) = 1.00000009$

$x^* = (-3.787e-10,\ 1.000e-02)$

$\lambda^* = (2.726e-02,\ 2.585e-02,\ 2.000e+04)$

$g(x^*) = (3.787e-10-1.000e-02-3.788e-12)$

Table 3 Problem 3, $\lambda^0 = 3$, $\tau^0 = 10$, $Tol : 1E-2$

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k)-f(x^{k-1})\rvert}{\lvert f(x^{k-1})\rvert+1}$	$\lVert g_+(x^k)\rVert_\infty$
000	+1.696531e-01	1.000000e+01	9.941808e-01	9.425983e-01
001	+6.397660e-01	1.000000e+01	4.019250e-01	8.900824e-01
002	+2.294973e+00	1.000000e+02	1.009416e+00	7.972333e-01
003	+7.488570e+00	1.000000e+03	1.576218e+00	6.504398e-01
004	+2.102292e+01	1.000000e+04	1.594421e+00	4.566869e-01
005	+4.795857e+01	1.000000e+05	1.223073e+00	2.604199e-01
006	+1.000000e+02	1.000000e+06	1.062969e+00	2.042497e-08
007	+1.000000e+02	1.000000e+06	0.000000e+00	2.042497e-08

$It = 8$

$time(s) = 0.03287101$

$f(x^*) = 99.99999592$

$x^* = (2.042e-08,\ 1.000e+00)$

$\lambda^* = (5.072e-06,\ 8.664e-08,\ 7.526e+02)$

$g(x^*) = (-2.042e-08,\ -1.000e+00,\ 2.042e-08)$

- Table 6: The subproblem is solved with the L-BFGS-B (second-order algorithm) and we consider the multistart strategy, with 20 points.
- Table 7: The subproblem is solved with the Nelder-Mead algorithm (a derivative-free algorithm) and we consider the multistart strategy, with 20 points.

Apparently, computational experiments show us that the sequence generated by our algorithm converges to a strong stationarity (S-stationary) or Mordukhovich stationary (M-stationary). On the other hand, mathematical programming problems with complementarity constraints are solved using an augmented Lagrangian algorithm

Table 4 Problem 4, $\lambda^0 = 30$, $\tau^0 = 100$, $Tol : 1E-1$, Subproblem solved with: Trust Region Algorithm [5]

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k)-f(x^{k-1})\rvert}{\lvert f(x^{k-1})\rvert+1}$	$\lVert g_+(x^k)\rVert_\infty$
000	+1.165878e-03	1.000000e+02	1.165878e-03	7.257575e-06

$It = 1$

$time(s) = 0.28706312$

$f(x^*) = 0.00116588$

$x^* = (0.001,\ 0.006)$

$\lambda^* = (1.194,\ 0.041,\ 30.653)$

$g(x^*) = (-1.146e-03,\ -6.334e-03,\ 7.258e-06)$

Table 5 Problem 4, $\lambda^0 = 30$, $\tau^0 = 10000$, $Tol : 1E-3$, Subproblem solved with: L-BFGS-B [4], and in the subproblem we consider the multistart strategy, with 50 points

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k)-f(x^{k-1})\rvert}{\lvert f(x^{k-1})\rvert+1}$	$\lVert g_+(x^k)\rVert_\infty$
000	+1.953418e-02	1.000000e+02	1.953418e-02	5.565766e-04
001	+8.646192e-04	1.000000e+02	1.831185e-02	7.263023e-07
002	+2.464962e-05	1.000000e+02	8.392439e-04	8.312372e-10

$It = 3$

$time(s) = 2.48027468$

$f(x^*) = 0.00002465$

$x^* = (2.465e-05,\ 3.372e-05)$

$\lambda^* = (1.000e+00,\ 1.095e-04,\ 3.056e+00)$

$g(x^*) = (-2.465e-05,\ -3.372e-05,\ 8.312e-10)$

based on the quadratic penalty function (ALGENCAN), see [1] and [2]. The subproblem generated by ALGENCAN is solved using a second-order algorithm, and in this way the sequence generated by this algorithm converges towards a M-stationary, see [1].

Funding

This work has been partially supported by FAPERJ (grant E-26/205.684/2022), COPPETEC Foundation and CNPq.

References

[1] Andreani R., Secchin L.D, Silva P.J.S.: Convergence properties of a second order augmented Lagrangian method for mathematical programs with

Table 6 Problem 4, $\lambda^0 = 30$, $\tau^0 = 100$, $Tol : 1E-1$, Subproblem solved with: L-BFGS-B Algorithm [4], and in the subproblem we consider the multistart strategy, with 20 points

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k)-f(x^{k-1})\rvert}{\lvert f(x^{k-1})\rvert+1}$	$\lVert g_+(x^k)\rVert_\infty$
000	+1.165887e-03	1.000000e+02	1.165887e-03	7.257647e-06

$It = 1$
$time(s) = 0.73982024$
$f(x^*) = 0.00116589$
$x^* = (0.001,\ 0.006)$
$\lambda^* = (1.194,\ 0.041,\ 30.653)$
$g(x^*) = (-1.146e-03,\ -6.334e-03,\ 7.258e-06)$

Table 7 Problem 4, $\lambda^0 = 30$, $\tau^0 = 100$, $Tol : 1E-1$, Subproblem solved with: Nelder-Mead Algorithm [8], and in the subproblem we consider the multistart strategy, with 20 points

k	$f(x^*)$	τ^k	$\frac{\lvert f(x^k)-f(x^{k-1})\rvert}{\lvert f(x^{k-1})\rvert+1}$	$\lVert g_+(x^k)\rVert_\infty$
000	+1.166212e-03	1.000000e+02	1.166212e-03	7.215356e-06

$It = 1$
$time(s) = 0.30188274$
$f(x^*) = 0.00116621$
$x^* = (0.001,\ 0.006)$
$\lambda^* = (1.193,\ 0.042,\ 30.649)$
$g(x^*) = (-1.146e-03,\ -6.294e-03,\ 7.215e-06)$

complementarity constraints. SIAM Journal on Optimization. **28**, 2574–2600 (2018)

[2] Andreani R., Fazzio N.S., Schuverdt M.L., Secchin L.D: A sequential optimality condition related to the quasi-normality constraint qualification and its algorithmic consequences. SIAM J. Optim. **29**, 743–766 (2019)

[3] Allevi E., Aussel D., Riccardi R.: On an equilibrium problem with complementarity constraints formulation of pay-as-clear electricity market with demand elasticity. J Glob Optim. **70**, 329–346 (2018)

[4] Byrd R.H., Lu P., Nocedal J., Zhu C.: A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing. **16**, 1190–1208 (1995)

[5] Coleman T.F., Li Y.: An interior trust region approach for nonlinear minimization subject to bounds. **6**, 418–445 (1996)

[6] Dempe S., Zemkoho A.B.: The bilevel programming problem: reformulations, constraint qualifications and optimality conditions. Math. Program., **138**, 447–473 (2013).

[7] Mordukhovich B.S.: Multiobjective optimization problems with equilibrium constraints. Math. Program., **117**, 331–354 (2007).

[8] Nelder J.A., Mead R.: A Simplex Method for Function Minimization. The Computer Journal, **7**, 308–313 (1965)

[9] Nabetani K., Tseng P., Fukushima M.: Parametrized variational inequality approaches to generalized Nash equilibrium problems with shared constraints. Comput Optim Appl, **48**, 423–452 (2011)

[10] Demiguel V., Friedlander M.P., Nogales F.J., Scholtes S.: A two-sided relaxation scheme for mathematical programs with equilibrium constraints. SIAM J. Optim., **16**, 587–609 (2005)

[11] Helou E.S., Santos S.A., Simões L.E.A.: A sequential optimality condition for Mathematical Programs with equilibrium constraints based on a nonsmooth formulation. IMA Journal of Numerical Analysis. **43**, 1586–1615 (2023)

[12] Izmailov A.F., Solodov M.V.: Examples of dual behaviour of Newton-type methods on optimization problems with degenerate constraints. Comput Optim Appl., **42**, 231–264 (2009)

[13] Izmailov A.F., Solodov M.V., EI Uskov: Global convergence of augmented Lagrangian methods applied to optimization problems with degenerate constraints, including problems with complementarity constraints. SIAM Journal on Optimization, **22**, 1579–1606 (2012)

[14] Melo T.M.M., Matias J.L.H., Monteiro T.T.: A penalty method for solving the MPCC problem. Journal of Mathematical Analysis, **7**, 91–101 (2016)

[15] Mallma Ramirez L., Maculan N., Xavier A.E., Xavier V.L.: Dislocation hyperbolic augmented Lagrangian algorithm for nonconvex optimization. RAIRO-Oper. Res., **57**, 2941–2950 (2023)

[16] Ralph D., Wright S.J.: Some properties of regularization and penalization schemes for mpecs. Optimization Methods and Software. **19**, 527–556 (2004)

[17] Raghunathan A.U., Biegler L.T.: Mathematical programs with equilibrium constraints (MPECs) in process engineering. Computers and Chemical Engineering. **27**,) 1381–1392 (2003)

[18] Scholtes S.: Convergence properties of a regularization scheme for mathematical programs with complementarity constraints. SIAM J. Optim. **11**, 918–936 (2001)

[19] Shapiro A., Xu H.: Stochastic mathematical programs with equilibrium constraints, modelling and sample average approximation. Optimization. **57**, 395–418 (2008)

[20] Xavier A.E.: Penalização Hiperbólica: Um Novo Método para Resolução de Problemas de Otimização (Ms.c. thesis), Federal University of Rio de Janeiro/COPPE, Rio de Janeiro, Brazil. (1982)

[21] Xavier A.E.: Penalização Hiperbólica (Ph.D. thesis), Federal University of Rio de Janeiro/COPPE, Rio de Janeiro, Brazil. (1992)

[22] Zangwill W.I.: Non-linear programming via penalty functions. Management Sciences. **13**, 344–358 (1967)