Journal of Mathematical Analysis
ISSN: 2217-3412, URL: http://www.ilirias.com
Volume 7 Issue 1 (2016), Pages 91–101.

A PENALTY METHOD FOR SOLVING THE MPCC PROBLEM

TEÓFILO M. M. MELO, JOÃO L. H. MATIAS, M. TERESA T. MONTEIRO

Abstract. The main goal of this work is to solve Mathematical Program with Complementarity Constraints (MPCC) using the penalty technique from the nonlinear optimization. The hyperbolic penalty method is used to solve the nonlinear reformulation of the MPCC in which the complementarity constraints are gathered into a single constraint and included in the penalty function. Three algorithms were implemented in MATLAB language using the penalty technique, two of them use the hyperbolic penalty method in two different approaches and the other implements the l_1 penalty. Numerical experiments are performed using a set of AMPL test problems from the MacMPEC database. A performance comparative analysis with respect to some metrics is carried out.

1. Introduction

In this paper we consider the Mathematical Program with Complementarity Constraints (MPCC) which arises from many applications in Engineering and Economics [8, 21]. They are so predominant in these areas since the concept of complementarity is related with the notion of system equilibrium. The complementarity constraints may come from: i) the stationary conditions of an optimization problem, ii) a game, or iii) a variational inequality. This kind of problem is very difficult to solve because the usual constraint qualifications necessary to guarantee the algorithms convergence fail in all feasible points [3]. This complexity arises from the disjunctive nature of the complementarity constraints.

There have been proposed some nonlinear approaches to solve MPCC based on its equivalent nonlinear problem (NLP) reformulation, starting with the penalty approaches [2,4,15,23,25], the relaxation schemes [5,16,17,20,27] and the smoothing schemes [7,13,19]. We also emphasize the work [24] that uses interior point methods, the "elastic mode" for nonlinear programming in conjunction with a sequential quadratic programming (SQP) algorithm [1] and [10], where SQP is guaranteed to, under relatively mild conditions, quadratically converge near a stationary point.

In this work we apply the hyperbolic penalty method [28] to solve the NLP reformulation of the MPCC problem where the complementarity constraints are gathered into a single constraint. This method combines features of both exterior and interior penalty methods and has the feature of being completely differentiable.

2010 *Mathematics Subject Classification.* 90C33; 90C30; 49M37.
Key words and phrases. MPCC; Penalty method; NLP.

Submitted June 08, 2015. Revised Dec 15, 2015. Published Jan 04, 2016.

Three algorithms were implemented in MATLAB language using the penalty strategy, two of them using the hyperbolic penalty and the other the l_1 penalty.

This paper is organized as follows. Next section presents the MPCC problem. The hyperbolic penalty method and the MATLAB algorithms are presented in Section 3. The numerical experiments carried out with the algorithms are reported in Section 4. Some conclusions and future work are summarized in Section 5.

2. Problem definition

In this work we consider the MPCC defined as follows:

$$\begin{array}{ll} \min\limits_x & f(x) \\ s.t. & c_i(x) = 0, i \in E, \\ & c_i(x) \geq 0, i \in I, \\ & 0 \leq x_1 \perp x_2 \geq 0, \end{array} \tag{MPCC}$$

where f and c are the nonlinear objective function and constraints functions, respectively, assumed to be twice continuously differentiable functions. A decomposition $x = (x_0, x_1, x_2)$ of the variables is used, where $x_0 \in \mathbb{R}^n$ (control variables) and $(x_1, x_2) \in \mathbb{R}^{2q}$ (state variables). E and I are two disjoined finite index sets with cardinality p and m, respectively. The expressions $0 \leq x_1 \perp x_2 \geq 0 : \mathbb{R}^{2q} \to \mathbb{R}^q$ are the q complementarity constraints.

One common approach to solve (MPCC) is to consider its equivalent nonlinear program (MPCC-NLP):

$$\begin{array}{ll} \min\limits_x & f(x) \\ s.t. & c_i(x) = 0, i \in E, \\ & c_i(x) \geq 0, i \in I, \\ & x_1 \geq 0,\, x_2 \geq 0, \\ & x_{1j} x_{2j} \leq 0, j \in Q. \end{array} \tag{MPCC-NLP}$$

In this approach the complementarity constraints are replaced by a set of nonlinear inequalities, such as $x_{1j}\, x_{2j} \leq 0, j \in Q$, where Q is a finite index set with cardinality q. Another NLP approach used by Ralph and Wright [25] is to consider the following problem formulation (MPCC-NLPC), where the complementarity constraints are gathered into a single constraint:

$$\begin{array}{ll} \min\limits_x & f(x) \\ s.t. & c_i(x) = 0, i \in E, \\ & c_i(x) \geq 0, i \in I, \\ & x_1 \geq 0,\; x_2 \geq 0, \\ & x_1^T x_2 \leq 0. \end{array} \tag{MPCC-NLPC}$$

The main difficulty on solving (MPCC-NLP) and (MPCC-NLPC) is that no feasible point satisfies the inequalities strictly, this implies that the Mangasarian-Fromovitz constraint qualification (MFCQ) is violated at every feasible point [26].

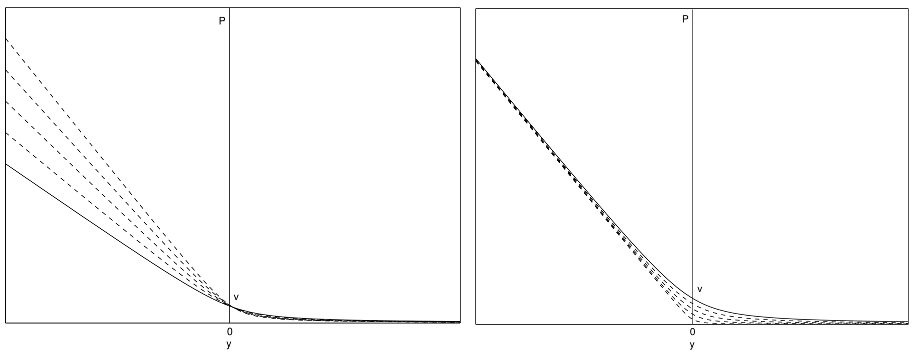
3. Penalty method

In this section we present the hyperbolic penalty method developed by Xavier [28] for solving nonlinear problems with inequality constraints of greater than or equal type.

3.1. **Hyperbolic penalty method.** The method adopts the function:

$$P(y,r,v) = -ry + \sqrt{r^2y^2+v^2}, \tag{3.1}$$

with $r, v \geq 0$, $r \to +\infty$ and $v \to 0$. The graphic representation is an hyperbole with asymptotes forming angles $(\pi - r)$ and zero with the horizontal axis with v intercepting the ordinates axis. Figure 1 presents the geometric idea of the hyperbolic penalty method, i.e., a two phases penalty approach. In the first stage represented in Figure 1(a), the initial parameter r increases, thus causing a reduction in the penalty to the points outside the feasible region. Figure 1(b) presents the second phase where r remains constant and the values of v decrease sequentially.



(a) First phase. (b) Second phase.

FIGURE 1. Geometric idea.

This penalty method can be situated in the context of exterior and interior penalties, where in the first phase, the hyperbolic penalty method is similar to exterior penalty methods, such as the quadratic loss function [9], and in the second phase, the behavior resembles the barrier methods such as, for example, the logarithmic barrier function [12].

The derivative of function (3.1) with respect to y takes the form:

$$P'_y(y,r,v) = -r + \frac{r^2y}{\sqrt{r^2y^2+v^2}},$$

varies in the range $(-2r, 0)$ with $P'(0,r,v) = -r$ for $v > 0$. When parameter r increases, the derivative $P'_y(y,r,v)$ decreases for points $y < \bar{y}$ and increases for points $y > \bar{y}$, where $\bar{y} = \frac{v}{r}\sqrt{\frac{-1+\sqrt{5}}{2}}$. More properties concerning to function (3.1) and corresponding proofs can be found in [28].

3.2. **Approach to MPCC.** In this work we apply the hyperbolic penalty to solve problem formulation (MPCC-NLPC) by penalizing the complementarity term. A sequence of the following nonlinear constrained optimization problem (P1) is solved:

$$\begin{array}{ll} \min\limits_{x} & P_1(x,r,v) \\ s.t. & c_i(x) = 0, i \in E, \\ & c_i(x) \geq 0, i \in I, \\ & x_1 \geq 0, \quad x_2 \geq 0, \end{array} \tag{P1}$$

where,

$$P_1(x, r, v) = f(x) + r(x_1^T x_2) + \sqrt{r^2(-x_1^T x_2)^2 + v^2},$$

with $r, v \geq 0$, $r \to \infty$ and $v \to 0$. For comparative purpose the following problem formulation (P2) based on the l_1 penalty is considered:

$$\begin{array}{ll} \min\limits_{x} & f(x) + r x_1^T x_2 \\ s.t. & c_i(x) = 0, i \in E, \\ & c_i(x) \geq 0, i \in I, \\ & x_1 \geq 0, \;\; x_2 \geq 0, \end{array} \tag{P2}$$

with $r > 0$ and $r \to \infty$. This formulation is a special case of [1] where the complementarity constraints are removed from the set of constraints and are included in the penalty function as a penalty term. Furthermore, we also include for the comparative analysis another approach using the hyperbolic penalty method. In this case the following penalty function is used:

$$P_3(x, r, v) = f(x) + \sum_{j \in Q} -r(-x_{1j} x_{2j}) + \sqrt{r^2(-x_{1j} x_{2j})^2 + v^2}, \tag{3.2}$$

with $r, v \geq 0$, $r \to \infty$ and $v \to 0$ for solving problem (MPCC-NLP) in which the complementarity constraints are replaced by a set of nonlinear inequalities $x_{1j}\, x_{2j} \leq 0,\ j \in Q$.

3.3. **MATLAB algorithms.** Three algorithms were implemented in MATLAB language for solving the (MPCC) equivalent problems (MPCC-NLPC) and (MPCC-NLP) using the penalty strategy. The corresponding code is available on `http://www.norg.uminho.pt/tm/.algorithms/`. We start presenting the algorithm named (A1) that applies the penalty approach to the problem (P1).

Algorithm 1 (A1)

Take initial values x_0, $r_0 > 0$, $v_0 > 0$, ρ_1, ρ_2 and tolerances $\epsilon_1, \epsilon_2, \epsilon_3$.
Read problem information: [x0,lb,ub,vd,cl,cu,cv]=amplfunc(prob_name).
Initialize internal, external iterations ($k = 0$) and function evaluations counters.
repeat
    Built the penalty function $P_1(x_k, r_k, v_k)$ and constraints.
    Built the gradient of the penalty function $P_1(x_k, r_k, v_k)$ and constraints.
    Run [x,fval,exitflag,output,lambda]=
    fmincon('fph_v1',x_k,...,lb,ub,'frest',options,cl,cu,lb,ub,cv,r,v).
    Estimate the vector of Lagrange multipliers ($\perp$).
    Update internal iterations and function evaluations counters.
    if $x_{1j} x_{2j} > \frac{-v_k}{1000}, j \in Q$ **then**
        $v_{k+1} = v_k \times \rho_1,\ 0 < \rho_1 < 1.$
    else
        $r_{k+1} = r_k \times \rho_2,\ \rho_2 > 1.$
    end if
    $x_{k+1} = x.$
    $k = k + 1.$
until x_{k+1} verify the stopping criterium.

Notice that, the feasibility test carried out at step 10 concerns only to the complementarity constraints and k at step 16 is the external iterations counter. The stopping criterium consists on the disjunction of four conditions, i.e., the iterative process ends if one of them is verified:

$$\begin{array}{ll} \text{I.} & \|\nabla\mathcal{L}(x_k,\lambda_k)\| \le \epsilon_1 \quad \wedge \quad \|h(x_k)\| \le \epsilon_2 \\ \text{II.} & \dfrac{\|x_{k+1}-x_k\|}{\|x_{k+1}\|} \le \epsilon_3 \quad \wedge \quad \dfrac{\|f_{k+1}-f_k\|}{\|f_{k+1}\|} \le \epsilon_3 \quad \wedge \quad \|h(x_k)\| \le \epsilon_2 \\ \text{III.} & r_k \ge r_{\max} \\ \text{IV.} & v_k \le v_{\min} \end{array} \tag{3.3}$$

where $\|\nabla\mathcal{L}(x_k,\lambda_k)\|$ is the norm of the gradient of the Lagrangian function, $\|h(x_k)\|$ is the norm of the violation of the complementarity constraints, $r_{\max}$ and $v_{\min}$ are the limits of the penalty parameters.

To evaluate the stopping criterium it was necessary estimating the Lagrange multipliers of the complementarity constraints. Consider the Lagrangian function of the (MPCC) in the following nomenclature:

$$\begin{aligned} \mathcal{L}(x,\lambda,\lambda_\perp,\lambda_{ub},\lambda_{lb}) &= f(x) + \sum_{i\in E\cup I} \lambda_i c_i(x) + (\lambda_\perp)(x_1^T x_2) + \lambda_{ub}^T(x-ub) + \\ &\quad + \lambda_{lb}^T(-x+lb), \end{aligned}$$

where λ, $\lambda_\perp$, λ_{ub} and λ_{lb} denote the Lagrange multipliers of the equality and inequality, complementarity, upper and lower bound constraints, respectively. Its gradient is given by:

$$\nabla\mathcal{L}(x,\lambda,\lambda_\perp,\lambda_{ub},\lambda_{lb}) = \nabla f(x) + \sum_{i\in E\cup I} \lambda_i \nabla c_i(x) + (\lambda_\perp)\nabla(x_1^T x_2) + \lambda_{ub} - \lambda_{lb}.$$

The Lagrangian function of problem (P1) is defined by:

$$\begin{aligned} \mathcal{L}_{P_1}(x,\lambda,\lambda_\perp,\lambda_{ub},\lambda_{lb},r,v) &= f(x) + r(x_1^T x_2) + \sqrt{r^2(x_1^T x_2)^2 + v^2} + \\ &\quad + \sum_{i\in E\cup I} \lambda_i c_i(x) + \lambda_{ub}^T(x-ub) + \lambda_{lb}^T(-x+lb). \end{aligned}$$

The corresponding gradient will come as:

$$\begin{aligned} \nabla\mathcal{L}_{P_1}(x,\lambda,\lambda_\perp,\lambda_{ub},\lambda_{lb},r,v) &= \nabla f(x) + r\nabla(x_1^T x_2) + \frac{r^2(x_1^T x_2)\nabla(x_1^T x_2)}{\sqrt{r^2(x_1^T x_2)^2 + v^2}} + \\ &\quad + \sum_{i\in E\cup I} \lambda_i c_i(x) + \lambda_{ub} - \lambda_{lb}. \end{aligned}$$

To estimate a formula for the Lagrange multipliers of the complementarity constraints we start with the following equality at the solution point x^*:

$$\nabla\mathcal{L}(x^*,\lambda,\lambda_\perp,\lambda_{ub},\lambda_{lb}) = \nabla\mathcal{L}_{P_1}(x^*,\lambda,\lambda_\perp,\lambda_{ub},\lambda_{lb},r,v).$$

From this equality we obtained the following estimation formula for the Lagrange multiplier of the complementarity constraint for problem (P1):

$$\lambda_\perp = r + \frac{r^2(x_1^T x_2)}{\sqrt{r^2(x_1^T x_2)^2 + v^2}}.$$

We now present the algorithm named (A2) that uses the problem formulation (P2) and is based on the l_1 penalty.

Algorithm 2 (A2)

Take initial values x_0, $r_0 > 0$, ρ_3 and tolerances $\epsilon_1, \epsilon_2, \epsilon_3$.
Read problem information: [x0,lb,ub,vd,cl,cu,cv]=amplfunc(prob_name).
Initialize internal, external iterations ($k = 0$) and function evaluations counters.
repeat
    Built the penalty function $P_2(x_k, r_k, v_k)$ and constraints.
    Built the gradient of the penalty function $P_2(x_k, r_k, v_k)$ and constraints.
    Run [x,fval,exitflag,output,lambda]=
    fmincon('fpl1_v1',x_k,...,lb,ub,'frest',options,cl,cu,lb,ub,cv,r).
    Estimate the vector of Lagrange multipliers ($\perp$).
    Update internal iterations and function evaluations counters.
    $r_{k+1} = r_k \times \rho_3,\ \rho_3 > 1$.
    $x_{k+1} = x$.
    $k = k + 1$.
until x_{k+1} verify the stopping criterium.

The stopping criterium for (A2) consists on the disjunction of the conditions I, II, and III from (3.3). To estimate the Lagrange multiplier of the complementarity constraint, the same strategy as in algorithm (A1) was used, i.e., it was assumed the following:

$$\nabla \mathcal{L}(x^*, \lambda, \lambda_\perp, \lambda_{ub}, \lambda_{lb}) = \nabla \mathcal{L}_{P_2}(x^*, \lambda, \lambda_\perp, \lambda_{ub}, \lambda_{lb}, r).$$

From this equality it was obtained the estimation formula $\lambda_\perp = r$. The Lagrangian function of problem (P2) is defined by:

$$\begin{aligned} \mathcal{L}_{P_2}(x, \lambda, \lambda_\perp, \lambda_{ub}, \lambda_{lb}, r) &= f(x) + r(x_1^T x_2) + \sum_{i \in E \cup I} \lambda_i c_i(x) + \lambda_{ub}^T (x - ub) + \\ &+ \lambda_{lb}^T(-x + lb), \end{aligned}$$

and its corresponding gradient as follows:

$$\nabla \mathcal{L}_{P_2}(x, \lambda, \lambda_\perp, \lambda_{ub}, \lambda_{lb}, r, v) = \nabla f(x) + r \nabla (x_1^T x_2) + \sum_{i \in E \cup I} \lambda_i c_i(x) + \lambda_{ub} - \lambda_{lb}.$$

The third algorithm named (A3) uses our second approach with the hyperbolic penalty method wherein we apply the penalty function (3.2). This algorithm carries out the same steps as algorithm (A1) and shares the same feasibility test for the complementarity constraints and stopping criterium (3.3). More details about this work could be found in [22].

4. Numerical experiments

This section summarizes the numerical experiments with algorithms (A1), (A2) and (A3) for solving problems formulations (MPCC-NLPC) and (MPCC-NLP).

4.1. **Computational environment.** The computational experiments use MATLAB version 8.1 for OS X operating system in version 10.9.4. with a 2.26 GHz Intel Core 2 Duo with 8 GB of RAM. An AMPL interface was implemented to link the problem in the format *.nl to MATLAB [14]. To evaluate the algorithms performance it was selected a set of AMPL [11] test problems from MacMPEC library containing a vast collection of MPCC problems of different sizes and different origins [18]. Table 1 summarizes the features of 102 test problems used in the numerical experiments and presents the quantile of the distribution of the number of variables, general constraints and complementarity constraints.

TABLE 1. Test problems features.

Quantile	# variables	# constraints	# complementarities
1-9	27	50	27
10-99	66	36	67
100-999	9	16	8

The algorithms (A1), (A2) and (A3) were implemented using two iterative procedures. The inner iterative procedure is performed by the *fmincon* routine from MATLAB optimization toolbox that uses the SQP method. The external iterative procedure is carried out by algorithms (A1), (A2) and (A3).

The initial parameters $r_0 = 1$ and $r_{\max} = $ 1e16 were used in both algorithms (A1) and (A2), additionally, $v_{\min} = $ 1e-16 was considered in algorithm (A1). To update the penalty parameters in algorithm (A1) it was considered $\rho_1 = 0.1$ and $\rho_2 = \sqrt{10}$, and in the algorithm (A2) was used $\rho_3 = 5$. Algorithm (A3) considers the initial parameter $r_0 = 7$ using the same values as in algorithm (A1). The constants $\epsilon_1 = 1e-4$ and $\epsilon_2 = \epsilon_3 = 1e-5$ were considered in algorithms (A1), (A2) and (A3).

TABLE 2. Stopping criterium.

Stopping criterium	1	2	3	4
(A1)	92	10	0	0
(A2)	93	9	0	-
(A3)	92	9	0	1

Table 2 reports the total number of problems that verified each condition of the stopping criterium. The results allow to conclude that the Lagrange multipliers estimation of the complementarity constraints performed well in all the approaches. Notice that algorithm (A2) does not have condition IV of (3.3).

4.2. **Performance profiles.** The algorithm (A1) was compared with the algorithms (A2) and (A3) using an application (*m-file*) developed by Dolan and Moré [6], which allows to obtain performance profiles graphics of several codes relatively to one metric.

In this work four metrics to compare the performance of algorithms were used: the number of internal iterations and function evaluations, concerning the internal iterative process (*fmincon*), the number of external iterations and time (CPU) in seconds for the external cycle of the implemented algorithm.

The graphics of the performance profiles are presented in Figures 2, 3, 4 and 5 corresponding to external and internal iterations, function evaluations and CPU time, respectively, using the *log* scale.

By observing the Figure 2 it is possible to visualize that algorithm (A3) presents the highest probability of being the optimal solver for about 83% of the problems, followed by algorithm (A2) with 73% of the problems and algorithm (A3) with 58% of the problems. For the metric of internal iterations, the algorithms (A2) and (A3) had the best performance relatively to algorithm (A1). However it is not possible to visualize which one presents the best behavior as show in Figure 3, since the lines that represent algorithms (A2) and (A3), cross.

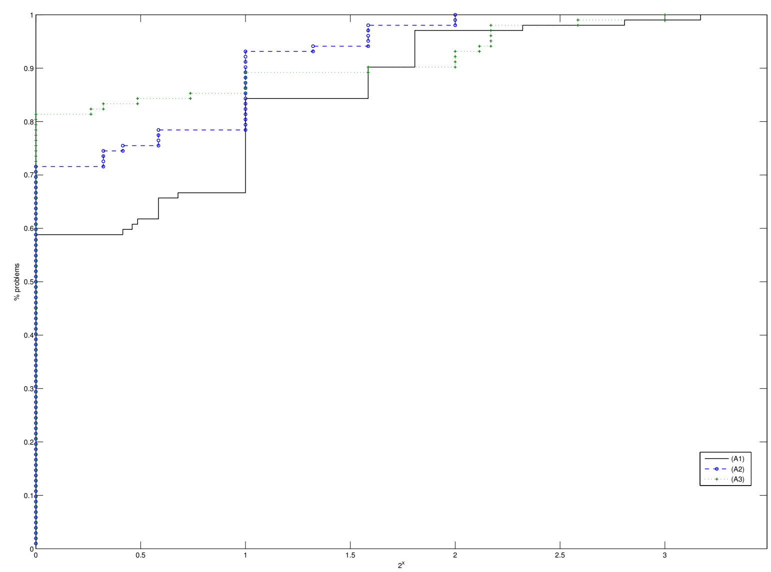


FIGURE 2. External iterations.

In terms of function evaluations, as shown in Figure 4, it is clear that algorithm (A2) presents the highest probability of being the optimal solver for about 60% of the problems. It is also possible to realize that both algorithms (A1) and (A3) had a similar behavior. From Figure 5 is possible to see that algorithm (A3) presents the best behavior having the highest probability of being the optimal solver for about 50% of the problems with respect to CPU time. It is also possible to observe that algorithm (A1) and (A2) presented a similar performance.

5. Conclusions and future work

In this work three algorithms for solving the MPCC problem using two approaches of its NLP equivalent problem formulations are presented. The algorithm (A1) is still in an improvement phase but some conclusions can already be taken. The numerical results presented good accuracy in the solution when compared with the one provided from the set of MacMPEC problems. From the performance profiles carried out, it was noticed that the algorithm (A1) has a similar behavior in terms of CPU time when compared with algorithm (A2) which solves the same NLP reformulation of the MPCC, however in the other metrics analyzed, it did not

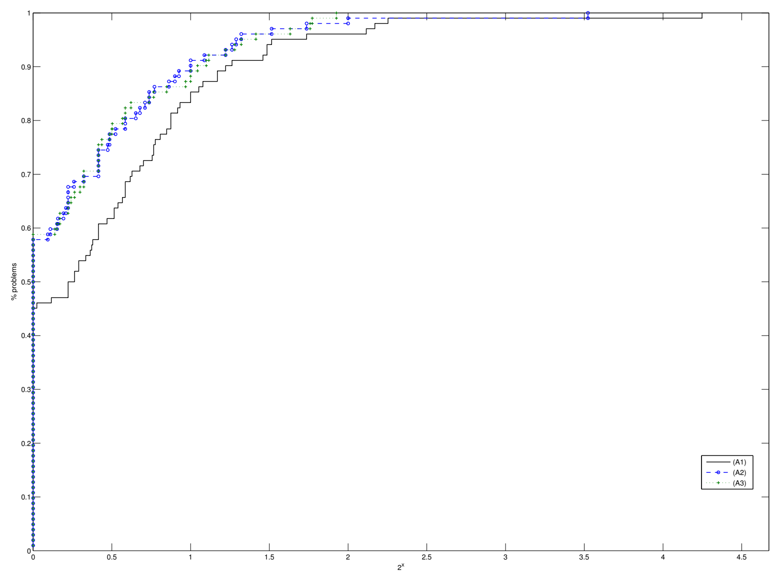


FIGURE 3. Internal iterations.

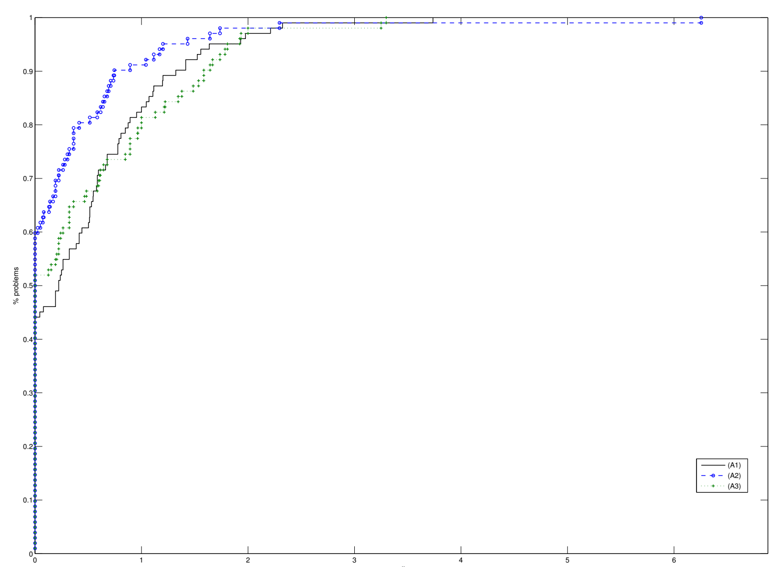


FIGURE 4. Function evaluations.

perform as well. As future work we intend to implement a new feasibility test to update the penalty parameters on the algorithm (A1) and test a set of larger scale problems.

Acknowledgments. The third author was supported by ALGORITMI R&D Center under the project PEst-UID/CEC/00319/2013 financed by Fundação para a Ciência e Tecnologia (FCT-Portugal).

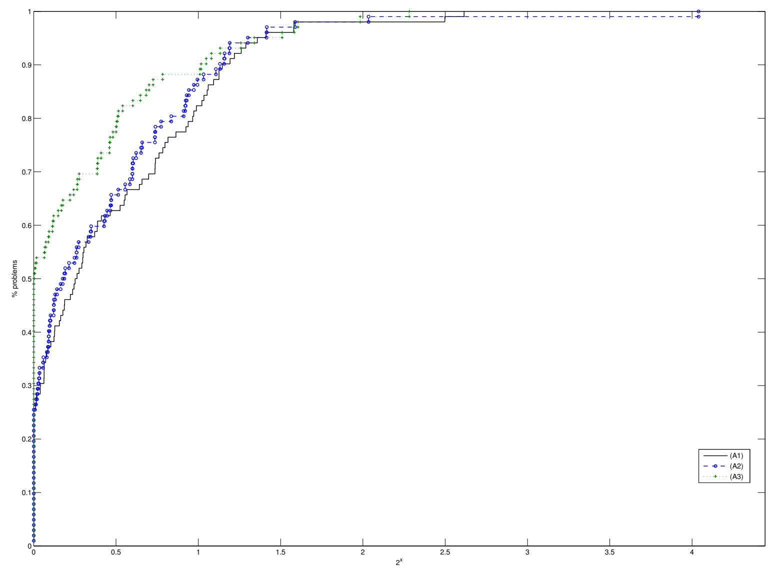


Figure 5. CPU time.

The work is also supported by PEst-OE/MAT/UI4080/2014. The authors are very grateful to all the financial support. The authors are very grateful to two referees for many constructive comments and remarks.

References

[1] M. Anitescu, P. Tseng, D. Ralph, *Elastic mode algorithms for mathematical programs with equilibrium constraints: global convergence and stationarity proprieties.* Mathematical Programming, **110** (2006) 337–371.

[2] M. Benharrat, D. F. M. Torres, *Optimal control with time delays via the penalty method.* Math. Probl. in Eng., **2014** (2014) 1–9.

[3] X. Chen, M. Florian, *The nonlinear bilevel programming problem: formulations, regularity and optimality conditions.* Optimization, **32** (1995) 193–209.

[4] Z. Coulibaly, D. Orban, *An l1 elastic interior-point method for mathematical programs with complementarity constraints.* SIAM Journal on Optimization, **22** (2012) 187–211.

[5] V. Demiguel, M. Friedlander, F. Nogales, S. Scholtes, *A two-sided relaxation scheme for mathematical programs with equilibrium constraints.* SIAM Journal on Optimization, **16** (2005) 587–609.

[6] E. Dolan, J. Moré, *Benchmarking optimization software with performance profiles.* Mathematical Programming, **91** (2002) 201–213.

[7] F. Facchinei, H. Jiang, L. Qi, *A smoothing method for mathematical programs with equilibrium constraints.* Technical Report 03.96, Universita di Roma, "La Sapienza", 1996.

[8] M. Ferris, J. Pang, *Engineering and economic applications of complementarity constraints.* SIAM Review, **39** (1997) 669–713.

[9] V. Fiacco, P. McCormick, *The sequential unconstrained minimization technique for nonlinear programming: a primal method.* Management Science, **10** (1964) 360–364.

[10] R. Fletcher, S. Leyffer, D. Ralph, S. Scholtes, *Local convergence of SQP methods for mathematical programs with equilibrium constraints.* SIAM Journal on Optimization, **17** (2006) 259–286.

[11] R. Fourer, B. Kernighan, *AMPL: a modeling language for mathematical programming.* Duxburg Press, Massachusetts, 1993.

[12] R. Frisch, *The logarithmic potential method of convex programming.* University Institute of Economics, 1995.

[13] M. Fukushima, J. Pang, *Convergence of a smoothing continuation method for mathematical programs with complementarity constraints.* Lectures Notes in Economics and Mathematical Systems, Springer-Verlag, **447** (1999) 99–110.
[14] M. Gay, *Hooking your solver to AMPL.* Tech. Rep., Bell Laboratories, Murray Hill, NY, USA, 1997.
[15] X. Hu, D. Ralph, *Convergence of a penalty method for mathematical programs with complementarity constraints.* Journal of Optimization Theory and Applications, **123** (2004) 365–390.
[16] A. Kadrani, J. Dussault, A. Benchakroun, *A new regularization scheme for mathematical programs with complementarity constraints.* SIAM Journal on Optimization, **20** (2009) 78–103.
[17] C. Kanzow, A. Schwartz, *Convergence properties of the inexact Lin-Fukushima relaxation method for mathematical programs with complementarity constraints.* Computational Optimization & Applications, **59** (2014) 249–262.
[18] S. Leyffer, https://wiki.mcs.anl.gov/leyffer/index.php/MacMPEC, 2015.
[19] Y. Li, T. Tan, X. Li, *A log-exponential smoothing method for mathematical programs with complementarity constraints.* Applied Mathematics and Computation, **218** (2012) 5900–5909.
[20] G. Lin, M. Fukushima, *A modified relaxation scheme for mathematical programs with complementarity constraints.* Ann. Oper. Res., **133** (2005) 63–84.
[21] Z. Luo, J. Pang, D. Ralph, *Mathematical programs with equilibrium constraints.* Cambridge University Press 1996.
[22] T. Melo, M. Teresa T. Monteiro, J. Matias, *Solving MPCC problem with the hyperbolic penalty function.* Numerical Analysis and Applied Mathematics ICNAAM 2011, AIP Conf. Proc. **1389** (2011) 763-766.
[23] M. T. T. Monteiro, J. F. P. Meira, *A penalty method and a regularization strategy to solve MPCC.* Int. J. Comput. Math., **88** (2011) 145–149.
[24] A. Raghunathan, L. Biegler, *Interior point methods for mathematical programs with complementarity constraints.* CAPD technical report, Department of Chemical Engineering, Carnegie Mellon University, June 2003
[25] D. Ralph, S. Wright, *Some proprieties of regularization and penalization schemes for MPECs.* Optimization Methods and Software, **19** (2004) 527–556.
[26] H. Scheel, S. Scholtes, *Mathematical programs with complementarity constraints: stationarity, optimality and sensitivity.* Mathematics of Operations Research, **25** (2000) 1–22.
[27] S. Scholtes, *Convergence proprieties of a regularization scheme for mathematical programs with complementarity constraints.* SIAM Journal on Optimization, **11** (2001) 918–936.
[28] A. Xavier, *Hyperbolic penalty: a new method for nonlinear programming with inequalities.* Intl. Trans. in Op. Res., **8** (2000) 659–671.

Teófilo M. M. Melo
CIICESI, School of Technology and Management of Felgueiras,
Polytechnic of Porto, 4610-156 Felgueiras, Portugal
E-mail address: tmelo@estgf.ipp.pt

João L. H. Matias
Centre of Mathematics, CM-UTAD, Universidade de Trás-os-Montes e Alto Douro,
UTAD, Escola de Ciências e Tecnologia, 5001-801 Vila Real, Portugal
E-mail address: j_matias@utad.pt

M. Teresa T. Monteiro
R&D Algoritmi Centre, Department of Production and Systems, University of Minho,
Campus of Gualtar, 4710-057 Braga, Portugal
E-mail address: tm@dps.uminho.pt