American Journal of Computational Mathematics, 2015, 5, 239-242
Published Online September 2015 in SciRes. http://www.scirp.org/journal/ajcm
http://dx.doi.org/10.4236/ajcm.2015.53020

Levenberg-Marquardt Method for Mathematical Programs with Linearly Complementarity Constraints

Cong Zhang[1*], Limin Sun[1], Zhibin Zhu[2], Minglei Fang[3]

[1]Department of Mathematics and Computer Science, Huarui College, Xinyang Normal University, Xinyang, China
[2]School of Mathematics & Computational Science, Guilin University of Electronic Technology, Guilin, China
[3]College of Science, Anhui University of Science and Technology, Huainan, China
Email: [*]zhcopt@126.com

Received 17 June 2015; accepted 17 August 2015; published 20 August 2015

Abstract

In this paper, a new method for solving a mathematical programming problem with linearly complementarity constraints (MPLCC) is introduced, which applies the Levenberg-Marquardt (L-M) method to solve the B-stationary condition of original problem. Under the MPEC-LICQ, the proposed method is proved convergent to B-stationary point of MPLCC.

Keywords

Mathematical Programs with Linear Complementarity Constraints, MPEC-LICQ, B-Stationarity, Levenberg-Marquardt Method

1. Introduction

The mathematical program with equibrium constraints (MPEC) has extensive application in area engineering design and economic model [1]. It has been an active research topic in recent years. In this paper, we consider the mathematical programming problem with linearly complementarity constraints (MPLCC), which is a special case of the MPEC:

[*]Corresponding author.

How to cite this paper: Zhang, C., Sun, L.M., Zhu, Z.B. and Fang, M.L. (2015) Levenberg-Marquardt Method for Mathematical Programs with Linearly Complementarity Constraints. *American Journal of Computational Mathematics*, **5**, 239-242.
http://dx.doi.org/10.4236/ajcm.2015.53020

$$\begin{aligned} \min \quad & f(x,y) \\ s.t. \quad & Ax \le b \\ & w = Nx + My + q \\ & 0 \le w \perp y \ge 0 \end{aligned} \tag{1.1}$$

where $f: R^{n+m} \to R$ is twice continuously differential real-valued function; $A \in R^{p\times n}$, $N \in R^{m\times n}$ and $M \in R^{m\times m}$ are given matrices; b and q are given *p*, *m* dimensional vectors, respectively.

Complementarity constraints in MPEC are known to be difficult to treat. Research work on the MPEC includes the monograph of Luo *et al*. [1] in which Bouligand stationary condition is introduced that provides a comprehensive study on MPEC. Based on different formulations, there are many algorithms such as Fukushima [2], Zhu [3], Zhang [4] [5], Jiang [6], Tao [7], and Jian [8]. Notice that B-stationary condition is a stronger stationary point. Differing from the approaches mentioned above, we directly introduce L-M technique, without any reformulation or relax form, to solve the B-stationary condition of MPLCC (1.1).

The plan of the paper is as follows: in Section 2, some preliminaries and model we used are presented; in Section 3, the algorithm is proposed.

2. Preliminaries

For reader's convenience, we use following notation throughout this paper:

$$z = (x,y,w), \quad s = (x,y), \quad A^{\mathrm{T}} = \left(a_1^{\mathrm{T}}, a_2^{\mathrm{T}}, \cdots, a_p^{\mathrm{T}}\right),$$

$$b^{\mathrm{T}} = \left(b_1, b_2, \cdots, b_p\right), \quad L_1 = \{1,2,\cdots,p\}, \quad I = \{l \in L_1 : a_l x - b_l = 0\},$$

$$L_2 = \{1,2,\cdots,m\}, \quad I_y = \{i \in L_2 : y_i = 0\}, \quad I_w = \{i \in L_2 : w_i = 0\}.$$

Let *F* denote the feasible set of problem (1.1).

Now we give two definitions as follow.

Definition 2.1. *Let* z^* *be a feasible point of MPLCC* (1.1), *we say that MPEC linear independence constraint qualification is satisfied at* z^* *if the gradient vectors*

$$\begin{pmatrix} A_S^{\mathrm{T}} \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} N^{\mathrm{T}} \\ M^{\mathrm{T}} \\ -I \end{pmatrix} \begin{pmatrix} 0 \\ \operatorname{diag}(e_{1i}) \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \operatorname{diag}(e_{2i}) \end{pmatrix}$$

is linearly independent, where $e_{1i} = \begin{cases} 1, & i \in I_y, \\ 0, & i \in L_2 \setminus I_y, \end{cases}$, $e_{2i} = \begin{cases} 1, & i \in I_w, \\ 0, & i \in L_2 \setminus I_w. \end{cases}$

Definition 2.2. *Under the MPEC-LICQ, a feasible point z is a B-stationary of problem* (1.1) *if there exist multiplier vectors* $\lambda \in R^p$, $\mu \in R^q$ *and* $u, v \in R^m$ *such that*

$$\nabla f(z) + \begin{pmatrix} A_S^{\mathrm{T}} \\ 0 \\ 0 \end{pmatrix} \lambda + \begin{pmatrix} N^{\mathrm{T}} \\ M^{\mathrm{T}} \\ -I \end{pmatrix} \mu + \begin{pmatrix} 0 \\ \operatorname{diag}(e_{1i}) \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 \\ 0 \\ \operatorname{diag}(e_{2i}) \end{pmatrix} v = 0, \tag{2.1}$$

$$\lambda \ge 0, \quad z \in F, \quad \lambda^{\mathrm{T}}(Ax - b) = 0, \tag{2.2}$$

$$u_i = 0, \quad i \in L_2 \setminus I_y, \tag{2.3}$$

$$v_i = 0, \quad i \in L_2 \setminus I_w, \tag{2.4}$$

$$u_i = 0, \quad v_i = 0, \quad i \in I_y \cap I_w. \tag{2.5}$$

As we know, most of the works on MPLCC want to get the B-stationary point of problem (1.1), so we also put emphasis on trying to construct a method to obtain the B-stationary of MPLCC (1.1). Now we rewrite the conditions (2.1)-(2.5) in term of lagrange multipliers as follow:

$$Q(\Omega)=\begin{pmatrix} \nabla f(z)+\begin{pmatrix} A_S^{\mathrm{T}} \\ 0 \\ 0 \end{pmatrix}\lambda+\begin{pmatrix} N^{\mathrm{T}} \\ M^{\mathrm{T}} \\ -I \end{pmatrix}\mu+\begin{pmatrix} 0 \\ \operatorname{diag}(e_{1i}) \\ 0 \end{pmatrix}u+\begin{pmatrix} 0 \\ 0 \\ \operatorname{diag}(e_{2i}) \end{pmatrix}v \\ \lambda_i(a_i x-b_i) \\ Nx+My+q-w \\ u_j y_j \\ v_j w_j \end{pmatrix}=0 \tag{2.6}$$

subject to:

$$Ax\le b,\quad y\ge 0,\quad w\ge 0,\quad \lambda\ge 0,\quad y_j w_j\le 0, \tag{2.7}$$

and

$$v_l\ge 0 \text{ and } u_l\ge 0 \text{ when } y_l=w_l=0 \text{ for some } l\in L_2, \tag{2.8}$$

where $\Omega=(z,\lambda,\mu,u,v)$, $j\in L_2$.

Remark: In (2.7) we replace $y_j w_j=0$ with $y_j w_j\le 0$, because it will be convenient for our computing.

3. The Description of Algorithm

Without any reformulation and relaxing techniques, we now use L-M method to solve the nonlinear systems (2.6). Firstly, let J be the Jacobian of $G(\Omega)$ at Ω. For an approximate solution z^k of (2.6), in order to produce an improving direction, we consider the following system of linear equations

$$\left(J_k^{\mathrm{T}}J_k+\sigma_k I\right)d=-J_k^{\mathrm{T}}G(\Omega) \tag{3.1}$$

$$\sigma_k=\theta\|G_k\|+(1-\theta)\left\|J_k^{\mathrm{T}}G_k\right\|,$$

where $G_k=G(\Omega^k)$, θ is a constant.

Lemma 3.1. *The coefficient matrix of* (*L* − *M*) *is positive definite, and furthermore*, (*L* − *M*) *method has unique solution.*

According to the constraint conditions, we now find a step length for current iterated point. First, we consider computing the step length of (x,y,w,λ). In the first place, for each constraint in (2.7), we should use the Ω^k and d^k to computer a step length:

$$\alpha_{xi}=\begin{cases} 1, & a_i d_x^k\le 0, \\ \min\left(1,\max\left(0,-\dfrac{a_i x-b_i}{a_i d_x^k}\right)\right), & a_i d_x^k>0. \end{cases} \tag{3.2}$$

$$\alpha_x=\min\left(\alpha_{xi}, i\in L_1\right)$$

where d_x^k is the element of d^k. Similar to the discussion of step length about *x*, we can obtain the step length $\alpha_{1y},\alpha_{1w},\alpha_\lambda$ about (y,w,λ).

As to calculating the step length for the constraint $y_j w_j\le 0$, we get the solution to the equation $\left(y_j+\alpha d_y^k\right)\left(w_j+\alpha d_w^k\right)=0$ with α as its variable, then α_j is as follows:

$$\alpha_j=\begin{cases} \max(m_1,m_2), & \text{the equation has two solutions,} \\ \min(1,\max(0,\alpha)), & \text{the equation has one solution and } d_y^k d_w^k<0, \\ 1, & \text{otherwise,} \end{cases} \tag{3.3}$$

$$\alpha_{2w}=\alpha_{2y}=\min\left(\alpha_j, j\in L_2\right),$$

so

$$\alpha_y=\min\left(\alpha_{1y},\alpha_{2y}\right),\ \alpha_w=\min\left(\alpha_{1w},\alpha_{2w}\right).$$

Secondly, we will consider the step length of (μ,u,v). Based on the step length that we obtain above, we can compute the value of y_{new}, w_{new}. If there is some *i* that $(y_{new})_i = (y_{new})_i = 0$, then the step length of corresponding variables u_i, v_i is obtained by the same way in (3.2) in order to satisfy the constraints (2.8); otherwise the step lengths of *u*, *v* are set to 1. The step length of μ is set to 1.

In this paper, we take $\|G(\Omega)\|^2$ as the merit function.

Lemma 3.2. *Let* d *be computed from* (3.1), *then* $d^{\mathrm{T}}\nabla\|G(\Omega)\|^2 \le 0.$

Proof. In view of Equation (3.1) and the positive definition of matrix $(J_k^{\mathrm{T}} J_k + \sigma_k I)$, we have

$$d^{\mathrm{T}}\nabla\|G(\Omega)\|^2 = 2d^{\mathrm{T}} J_k G_k = -2d^{\mathrm{T}}(J_k^{\mathrm{T}} J_k + \sigma_k I)d \le 0.$$

Now we present the algorithm.

Algorithm A:

Step 0: Given a feasible initial point Ω, let $k=1$;

Step 1: If $\|G(\Omega)\|^2 < \epsilon$, then stop; else get the d^k for (3.1);

Step 2: Compute the step length θ_k;

Step 3: $\Omega^{k+1} = \Omega^k + \mathrm{diag}(\theta_k)d^k$, go to Step 1, where $\theta_k = (\alpha_x, \alpha_y, \alpha_w, 1, \alpha_\lambda, \alpha_u, \alpha_v)^{\mathrm{T}}$.

Theorem 3.1. *Suppose that* Ω *is generated by Algorithm A and converges to* $\bar{\Omega}$; *if* $z^k \in F$ *for infinitely many k, let the MPEC-LICQ hold on* $\bar{z}$, *then* $\bar{z}$ *is a B-stationary point of problem* (1.1).

Proof. From the construction of the algorithm, we have $z^k \in F$ for sufficient large *k* and $\bar{z} \in F$. And because the MPEC-LICQ holds on $\bar{z}$, then $\bar{z}$ is a B-stationary point of problem (1.1).

Funding

This work was supported in part by the National Natural Science Foundation (No. 11361018), the Natural Science Foundation of Guangxi Province (No. 2014GXNSFFA118001), Key Program for Science and Technology in Henan Education Institution (No. 15B110008) and Huarui College Science Foundation (No. 2014qn35) of China.

References

[1] Luo, Z.Q., Pang, J.S. and Ralph, D. (1996) Mathmetical Programs with Equilibrium Constraints. Cambridge University Press, Cambridge. http://dx.doi.org/10.1017/CBO9780511983658

[2] Fukushima, M., Luo, Z.Q. and Pang, J.S. (1998) A Globally Convergent Sequential Quadratic Programming Algorithm for Mathematical Programs with Linear Complementarity Constraints. *Computational Optimization and Application*, **10**, 5-34. http://dx.doi.org/10.1023/A:1018359900133

[3] Zhu, Z.B. and Zhang, K.C. (2006) A Superlinearly Convergent SQP Algorithm for Mathematical Programs with Linear Complementarity Constraints. *Application and Computation*, **172**, 222-244. http://dx.doi.org/10.1016/j.amc.2005.01.141

[4] Zhang, C., Zhu, Z.B., Chen, F.H. and Fang, M.L. (2010) Sequential System of Linear Equations Algorithm for Optimization with Complementary Constraints. *Mathematics Modelling and Applied Computing*, **1**, 71-80.

[5] Zhang, C., Zhu, Z.B. and Fang, M.L. (2010) A Superlinearly Convergent SSLE Algorithm for Optimization Problems with Linear Complementarity Constraints. *Journal of Mathematical Science*: *Advance and Application*, **6**, 149-164.

[6] Jiang, H. (2000) Smooth SQP Methods for Mathematical Programs with Nonlinear Complementarity Constraints. *SIAM Journal of Optimization*, **10**, 779-808. http://dx.doi.org/10.1137/S1052623497332329

[7] Tao, Y. (2006) Newton-Type Method for a Class of Mathematical Programs with Complementarity Constrains. C*omputers and Mathematics with Applications*, **52**, 1627-1638. http://dx.doi.org/10.1016/j.camwa.2006.09.002

[8] Jian, J.B. (2005) A Superlinearly Convergent Implicit Smooth SQP Algorithm for Mathematical Programs with Nonlinear Complemetarity Constraints. *Computational Optimization and Applications*, **31**, 335-361. http://dx.doi.org/10.1007/s10589-005-3230-5