Applied Mathematics and Computation 251 (2015) 284–299

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

# A system of nonsmooth equations solver based upon subgradient method

Qiang Long [a], Changzhi Wu [b,*], Xiangyu Wang [b,c]

[a] School of Science, Southwest University of Science and Technology, Sichuan, China
[b] Australasian Joint Research Centre for Building Information Modelling, School of Built Environment, Curtin University, Perth, WA 6845, Australia
[c] Department of Housing and Interior Design, Kyung Hee University, Seoul, Republic of Korea

ARTICLE INFO

*Keywords:*
System of equations
Subgradient method
Bilevel programming
Nonlinear complementarity problem

ABSTRACT

In this paper, a subgradient method is developed to solve the system of (nonsmooth) equations. First, the system of (nonsmooth) equations is transformed into a nonsmooth optimization problem with zero minimal objective function value. Then, a subgradient method is applied to solve the nonsmooth optimization problem. During the processes, the pre-known optimal objective function value is adopted to update step sizes. The corresponding convergence results are established as well. Several numerical experiments and applications show that the proposed method is efficient and robust.



## 1. Introduction

In this paper, we consider the following system of nonlinear equations

$$\boldsymbol{H}(x) = \begin{pmatrix} H_1(x) \\ H_2(x) \\ \vdots \\ H_m(x) \end{pmatrix} = \mathbf{0}, \tag{1}$$

where $H_1, H_2, \ldots, H_m : \mathbb{R}^n \to \mathbb{R}$ are locally Lipschitz continuous but not necessarily differentiable. We call these equations *the system of nonsmooth equations* or *nonsmooth equation system*.

The solution of the system of nonsmooth equations has attracted intensive research in optimization society. This is because many well-known problems in mathematical programming, such as nonlinear complementarity problems (NCP), variational inequality (VI), Karush–Kuhn–Tucker (KKT) systems, bilevel programming problems can be equivalently transformed to a sequence of nonsmooth equation systems [24]. The most popular method for solving the system of nonsmooth equations is Newton method and its variants. Using Newton method to solve the system of equations is extended from the classic Newton–Raphson method for smooth univariate equation [2]. For a univariate equation $f(x) = 0$, the Newton–Raphson iteration is

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}. \tag{2}$$

* Corresponding author.
*E-mail address:* c.wu@curtin.edu.au (C. Wu).

http://dx.doi.org/10.1016/j.amc.2014.11.064
0096-3003/

Through replacing the derivative $f'(x_k)$ by Jacobian of the system of equations, we have the Newton iteration for Problem (1),

$$x_{k+1} = x_k + \delta_k, \tag{3}$$

where $\delta_k$ solves the system

$$J(\boldsymbol{H}(x_k))\delta_k = -\boldsymbol{H}(x_k). \tag{4}$$

The matrix $J(\boldsymbol{H}(x_k))$ is the Jacobian matrix of $\boldsymbol{H}(x)$ at point $x_k$, the system (4) is called Jacobian system [4]. The Jacobian system can be solved by (linear) Krylov method [33,34] approximately. Peter [23] proposed a hybrid method based upon Kryolv method and Powell's dogleg strategy to solve the system (4) without explicit expression of Jacobian matrix. However, in the nondifferentiable case, the Jacobian $J(\boldsymbol{H}(x_k))$ may not exist. There are two different approaches to overcome this difficulty [29]. The first one is to replace the Jacobian matrix by $V_k \in \partial\boldsymbol{H}(x_k)$ [12,30], where $\partial\boldsymbol{H}(x_k)$ is the generalized Jacobian of $\boldsymbol{H}$ at $x_k$ in the sense of Clarke [6,7]. In another approach [21,22], $\boldsymbol{H}$ is additionally assumed to be directionally differentiable and $d_k$ is determined by solving

$$\boldsymbol{H}(x_k) + \boldsymbol{H}'(x_k, d_k) = \boldsymbol{0}. \tag{5}$$

However, solving Problem (5) itself is not easy, since it is a system of nonlinear equations. Some other Newton methods or their variants to solve the system of nonsmooth equations refer to [10,11,23,31,32]. A good survey of Newton method for the system of equations refers to [28], where the smooth Newton method was introduced.

Quasi-Newton method and its variants is another type of numerical approach to solve the system of nonsmooth equations. In [36], the authors presented a unified realization of quasi-Newton methods for solving several standard problems including complementarity problems, variational inequality and the KKT system. Only a linear equation is involved in each step in the proposed method. In [13], BFGS method is adopted to solve KKT system. Given the reformulation of the KKT system as a nonsmooth equation system, the original KKT system is split successively into subproblems such that each subproblem has a particular structure. Then, the BFGS method is applied to solve each of them.

There are still some other numerical methods to solve the system of nonlinear equations. For example, Potra and Qi [26] introduced a generalization of the secant method for semismooth equation systems. Pseudo-transient continuation is a Newton-like iterative method for computing steady state solution of differential equations in cases where the initial data are far from a steady state. In [9], the authors showed how steady-state solution to certain algebraic differential equation with nonsmooth dynamics could be computed with the method of pseudo-transient continuation. In [27], two kinds of trust region methods were introduced to solve nonsmooth equation systems. One was an extension of the classical Levenberg–Marquardt method. It approximates the locally Lipschitzian function by a smooth one. Then, the derivative of the smooth function is used. The global convergence for this algorithm is established under regular condition. The other one approximates the Lipschitz function successively in which their derivatives are used. In both algorithms, the objective functions of subproblems are quadratic functions.

Actually, a system of equations, whether smooth or nonsmooth, is equivalent to an unconstrained optimization problem with zero minimal objective function value. Thus, solving the system of nonsmooth equations (1) equals to solve the following unconstrained optimization problems

$$\begin{cases} \text{Minimize} & \|\boldsymbol{H}(x)\| \\ \text{Subject to} & x \in \mathbb{R}^n \end{cases} \tag{6}$$

or

$$\begin{cases} \text{Minimize} & \frac{1}{2}\|\boldsymbol{H}(x)\|^2 \\ \text{Subject to} & x \in \mathbb{R}^n \end{cases}. \tag{7}$$

Here $\|\cdot\|$ stands for Euclidian norm. Obviously, $x^*$ is a solution of the system of nonsmooth equations if and only if $x^*$ is the solution of Problem (6) or (7). This idea is applied by Wu [38], who used filled function method [39,37] to solve Problem (7); and Long [14], who used quasisecant method [1] to solve Problem (6). One important information of Problem (6) and (7) is that their minimal objective function value, which is zero, is already known. Using this information to design an efficient numerical algorithm to solve Problem (1) is the core of this paper.

The remain of this paper are arranged as follows. In Section 2, we review some basic definitions of nonsmooth analysis and the general process of subgradient method. In Section 3, we provide an algorithm based upon subgradient method for solving the system of nonsmooth equations and prove its convergence. In Section 4, we test the proposed method by some well-known academic benchmarks and analyze the numerical results. In Section 5, we apply the proposed method to solve the sub-problems arisen in bilevel programming problem and nonlinear complementarity problem. Section 6 concludes the paper.

## 2. Preliminaries

### *2.1. Definitions for nonsmooth equation*

We first introduce some basic definitions and theorems in nonsmooth analysis. Suppose that $f : \mathbb{R}^n \to \mathbb{R}$ is a continuous but not necessarily differentiable function, $x \in \mathbb{R}^n$ is a vector in $\mathbb{R}^n$. The *directional derivative* of $f$ at $x$ in the direction $v \in \mathbb{R}^n$ is defined by