ARTICLE IN PRESS

Neurocomputing (xxxx) xxxx–xxxx

Contents lists available at ScienceDirect

Neurocomputing

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

A projection type steepest descent neural network for solving a class of nonsmooth optimization problems

M.J. Ebadi[a], Alireza Hosseini[b,c,*], M.M. Hosseini[a]

[a] Department of Mathematics, Yazd University, P.O. Box 89195-741, Yazd, Iran
[b] School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, P.O. Box 14115-175, Tehran, Iran
[c] School of Mathematics, Institute for Research in Fundamental Sciences (IPM), P. O. Box 19395-5746, Tehran, Iran

ARTICLE INFO

Communicated by Ding Wang

Keywords:
Recurrent neural network
Nonsmooth optimization
Global convergence
Stability
Differential inclusion
Solution trajectory.

ABSTRACT

In this paper, a new one layer recurrent neural network is proposed to solve nonsmooth optimization problems with nonlinear inequality and linear equality constraints. Model is based on a differential inclusion and combines steepest descent and gradient projection methods simultaneously. Any solution trajectory of the introduced differential inclusion converges globally to the optimal solution set of the corresponding optimization problem. Comparing with the existing models for solving nonsmooth optimization problems, there does not exist any penalty parameter in the structure of the new model and the model has simple structure. Moreover, the optimal solution of the original optimization problem is equivalent to the equilibrium point of the proposed neural network. Some illustrative examples are presented to show the effectiveness and performance of the proposed neural network.

1. Introduction

Consider the following constrained nonlinear optimization problem:

$$\min \quad f(x) \text{subject to} \quad g_i(x) \le 0,\ i = 1, \dots, m \qquad Ax = b, \tag{1}$$

where $x = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$ is the vector of decision variables, $f: \mathbb{R}^n \to \mathbb{R}$ is a locally Lipschitz function which is not convex generally and may be nonsmooth and $g(x) = \max\{g_i(x) | i = 1, \dots, m\}$, where $g_i (i = 1, 2, \dots, m): \mathbb{R}^n \to \mathbb{R}$ are nonsmooth functions. Assume g is convex (note that if g_is are convex functions, then g is convex too). $A \in R^{n\times m}$ is a full row-rank matrix (i.e. rank(A) = $m \le n$) and $b = (b_1, b_2, \dots, b_m)^T \in R^m$. $\Omega_1 = \{x \in R^n: g(x) \le 0\}$, $\Omega_2 = \{x \in R^n: Ax{=}b\}$ and $\Omega = \Omega_1 \bigcap \Omega_2$ is the feasible region of problem (1). Moreover assume that the objective function f is convex over Ω_1 (convexity of f over $\mathfrak{R}^n$ is not needed). It means that the problem (1)can be a nonconvex programming problem.

Obviously, problem (1) is equivalent to the following problem:

$$\min \quad f(x) \text{subject to} \quad g(x) \le 0, \qquad Ax = b. \tag{2}$$

Constrained optimization problems have many applications in science and engineering such as robot control, optimal control, signal processing, manufacturing system design and pattern recognition [1–6]. Real-time solutions of the optimization problems are often needed. Applying recurrent neural networks to solving dynamic optimization problems is one of the possible and very promising approaches. Among parallel computational models for solving constrained optimization problems, recurrent neural networks have been utilized and received a great deal of attention over the past recent decades (e.g., see [7–10] and references therein). In 1986, as a first attempt, a recurrent neural network model based on gradient method proposed by Tank and Hopfield [7] for solving linear programming problems. The design and applications of recurrent neural networks for optimization have been widely investigated. For example Kennedy and Chua [8] presented a neural network for solving nonlinear programming problems. The structure of the proposed neural network model is based on Newton-type descent, which contains finite penalty parameters and generates only approximate solutions. If the penalty parameter is chosen very large, a significant problem will arise for such neural network. From then on, many neural networks have been proposed and well developed for solving various kinds of optimization problems. For example, Lagrangian neural networks which are constructed to solve nonlinear programming problems with equality constraints [11,12], a projection-type neural network which is designed to solve nonlinear convex programming problems [13], a generalized neural network [10] and so on. Moreover, many other methods have been proposed in recent years to avoid using penalty parameters (e.g., see [14–17]). Xia and Wang [18] introduced a recurrent neural network to solve nonlinear convex programming with nonlinear inequality constraints without using

* Corresponding author at: School of Mathematics, Statistics and Computer Science, College of Science, University of Tehran, P.O. Box 14115-175, Tehran, Iran.
E-mail addresses: hosseini.alireza@ut.ac.ir, a.r_hosseini@khayam.ut.ac.ir (A. Hosseini).

http://dx.doi.org/10.1016/j.neucom.2017.01.010
Received 29 April 2016; Received in revised form 24 November 2016; Accepted 3 January 2017
0925-2312/

Please cite this article as: Ebadi, M.J., Neurocomputing (2017), http://dx.doi.org/10.1016/j.neucom.2017.01.010

any penalty parameter. Any solution state of the neural neural network when the objective function is convex and all constraint functions are strictly convex is globally convergent to exact optimal solutions in finite time. To prevent the strict condition on the constraint function in [18], Yang and Cao [19] extended the proposed neural network in [18] under the convexity assumptions of the objective and constraint functions. Moreover, we can point to a projection-type neural network which is proposed by Gao [14] for solving nonlinear convex programming problems with bound constraints. Among optimization problems, nonsmooth ones are attractive for researchers because of their important role in engineering applications, such as manipulator control, sliding mode, signal processing and so on (see [20–23] and the references therein). To speak generally, solving this kind of problems is difficult even in unconstrained cases. Forti et al. [10] proposed a generalized neural network for solving a much wider class of nonlinear nonsmooth programming problems with inequality constraints in real time which is an extension of the Kennedy and Chua's network [8] from smooth to nonsmooth case and described by a differential inclusion. The model is based on penalty method and the solution trajectories converge only when the penalty parameter tends to infinity. Xue and Bian [24,25] extended the neural network presented in [10] for nonsmooth convex and nonconvex optimization problems containing inequality and affine equality constraints based on the subgradient and penalty parameter methods. A recurrent neural network for nonsmooth convex optimization problem with a larger class of constraints has been proposed by Cheng et al. [26] which is not based on any penalty parameter, but with a complex structure. To reduce the model complexity, some recurrent neural networks have been proposed for solving nonsmooth convex optimization problems with linear equality [27], bound [28] and both linear equality and bound constraints [29]. Recently, Liu and Wang [30] and Qin and Xue [31] presented one layar and two layer neural network models to solve nonsmooth optimization problem (1) respectively. In [30] the authors proposed a model with penalty parameters in its structure. To guarantee convergence of the state variables to optimal solution set, some assumptions on inequality constraints are needed. Moreover, in order to determine a suitable penalty value we should estimate an upper bound of the Lipschitz constants of the constraint functions over a compact set. Recently, Hosseini et al. [32] proposed a recurrent neural network to solve pseudoconvex optimization problems. The model is a penalty-based method and convergence to optimal solution can be guaranteed only under some limiting assumptions. Moreover, a steepest descent neural network model has been proposed in [33]. This model can be applied for optimization problems with nonsmooth and nonlinear inequality constraints. This neural network model is not penalty based. However we can not use this method to solve problems with linear equality constraints. To overcome the above difficulties and reduce the model complexity, in this paper, we propose a new one layer recurrent neural network to solve a class of nonlinear nonsmooth optimization problems with nonlinear inequality and linear equality constraints. Objective function can be nonconvex, however it must be convex over the region Ω_1. The model is based on a differential inclusion and applies gradient projection and steepest descent approaches in its structure. We prove the global convergence of the proposed neural network and show the stability of the dynamical system. In the structure of the new model, there is not any penalty parameter, therefore starting by any initial state, solution trajectory of the designed differential inclusion converges to an element of the optimal solution set of the corresponding optimization problem. The reminder of the paper is organized as follows: The related preliminaries and some definitions are given in Section 2. In Section 3, the neural network model for solving optimization problem (2) is constructed. We prove the stability and globally convergence of the proposed neural network in Section 4. Some illustrative examples are given in Section 5. Analog circuit for the proposed neural network is designed in Section 6. Finally, in Section 7 some conclusions are presented.

2. Preliminaries

We present some definitions and lemmas for the convenience of the later discussion. Throughout this paper, $\|\cdot\|_1$ and $\|\cdot\|_2$ denote l_1 and l_2 norms of a vector in $\mathbb{R}_n$, respectively.

Definition 2.1. [34] Suppose that X and Y are two sets. A map F from X to Y is said a set valued map, if it associates a subset $F(x)$ of Y with any $x \in X$.

Definition 2.2. [34] A set valued map$F$ with nonempty values is upper semicontinuous (U.S.C.) at $x^0 \in X$, if for any open set N containing $F(x^0)$ there exists a neighborhood M of x^0, such that $F(M) \subset N$. Also, F is U.S.C. if it is U.S.C. at every $x^0 \in X$.

Definition 2.3. [34] A function $f: \mathbb{R}^n \to \mathbb{R}$ is said to be Lipschitz near $x \in \mathbb{R}^n$, if for any given $\epsilon > 0$ there exists $\delta > 0$, such that for any $x_1, x_2 \in \mathbb{R}^n$, satisfying $\|x_1 - x\|_2 < \delta$ and $\|x_2 - x\|_2 < \delta$, we have $|f(x_1) - f(x_2)| \le \epsilon\|x_1 - x_2\|_2$. We say that f is locally Lipschitz on $\mathbb{R}_n$, if f is Lipschitz near any point $x \in \mathbb{R}^n$.

Definition 2.4. [35] Suppose that f is Lipschitz near $x \in \mathbb{R}^n$. The generalized directional derivative of f at x in the direction of any vector $v \in \mathbb{R}^n$, is given by

$$f^0(x; v) = \limsup_{y \to x, t \downarrow 0} \frac{f(y + tv) - f(y)}{t},$$

and the Clarke generalized gradient of f at x is defined as

$$\partial f(x) = \{y \in \mathbb{R}^n: f^0(x; v) \ge y^T v, \forall\, v \in \mathbb{R}^n\}.$$

Definition 2.5. [35] A function f, which is locally Lipschitz near $x \in \mathbb{R}^n$, is called regular at x if we have

(1) for any direction $v \in \mathbb{R}^n$, the one-sided directional derivative $f'(x; v)$ which is given by

$$f'(x; v) = \lim_{\xi \to 0^+} \frac{f(y + \xi v) - f(y)}{\xi},$$

exists;

(2) for all v, $f'(x; v) = f^0(x; v)$.

Lemma 2.6 (*Chain rule Clarke [35]*). *If $W: \mathbb{R}^n \to \mathbb{R}$ is regular at $x(t)$, $x(t): \mathbb{R} \to \mathbb{R}^n$ is differentiable at t and Lipschitz near t, then*

$$\dot{W}(x(t)) = \langle \xi, \dot{x}(t) \rangle, \quad \forall\, \xi \in \partial W(x(t)) \quad \text{for a. e. } t \in [0, \infty).$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product.

3. Neural network model

3.1. Neural network structure

Assumption 3.1. One of the following assumptions holds:

(a) There exists at least one optimal solution of problem (2) at the interior of the region Ω_1.
(b) g is strictly convex over Ω_1.

Theorem 3.2. *Suppose that Assumption 3.1 holds. Then, x^* is an optimal solution of problem (2), if there exists $y^* \in R^m$ such that (x^*, y^*) satisfies the following equations*

$$0 \in T(x^*) - A^T y^*,$$
$$0 = Ax^* - b. \tag{3}$$

Where $T(x^*) = \{\Psi(g(x^*))[\eta - \gamma] - \eta: \eta \in \partial f(x^*), \gamma \in \partial g(x^*)\}$ (for simplicity we write $T(x^*) = \Psi(g(x^*))[\partial f(x^*) - \partial g(x^*)] - \partial f(x^*)$) and