

光滑拉格朗日神经网络解决非光滑最优化问题*

喻昕, 于琰, 谢緬

(广西大学 计算机与电子信息学院, 南宁 530004)

摘要: 针对目标函数是局部 Lipschitz 函数, 其可行域由一组等式约束光滑凸函数组成的非光滑最优化问题, 通过引进光滑逼近技术将目标函数由非光滑函数转换成相应的光滑函数, 进而构造一类基于拉格朗日乘子理论的神经网络, 以寻找满足约束条件的最优解。证明了神经网络的平衡点集合是原始非光滑最优化问题关键点集合的一个子集; 当原始问题的目标函数是凸函数时, 最小点集合与神经网络的平衡点集合是一致的。通过仿真实验验证了理论结果的正确性。

关键词: 局部 Lipschitz 函数; 光滑逼近技术; 平衡点集合; 最小点集合

中图分类号: TP183

文献标志码: A

文章编号: 1001-3695(2014)05-1349-04

doi:10.3969/j.issn.1001-3695.2014.05.015

Smoothing Lagrange neural network for constrained nonsmooth optimization problems

YU Xin, YU Yan, XIE Mian

(School of Computer, Electronics & Information, Guangxi University, Nanning 530004, China)

Abstract: The objective function of the nonsmooth optimization problems was locally Lipschitz and the feasible set of that consisted of a set of equality constrained smoothing and convex function. The nonsmooth function was converted into smooth function by being applied with the smoothing approximate techniques. Moreover, it modeled the Lagrange neural network by a class of differential equations, which could be implemented easily. The methodology was based on the Lagrange multiplier theory in optimization and sought to provide solutions satisfying the necessary conditions of optimality. It proved that any equilibrium point of the network was a subset to the critical point set of primal problems, when the objective function of primal problems was convex, the minimum set coincided with the equilibrium point set of the network. Finally, it presented a simulation experiment to illustrate above theoretical finding.

Key words: locally Lipschitz function; smoothing approximate techniques; equilibrium point set; minimum set

0 引言

多年来,研究者对用神经网络解决约束最优化问题已有深入的研究。1986年 Tank 等人^[1]介绍了用 Hopfield 神经网络解决线性规划问题,激发了很多研究者去构造多种神经网络来解决最优化问题。Kennedy 等人^[2]提出了一种解决非线性规划问题的神经网络。然而长久以来,研究者们发现采用这些神经网络解决最优化问题都要求目标函数和约束函数是光滑的,因为在这种网络中要用到目标函数和约束函数的梯度,而在很多重要的应用中,最优化问题往往是非光滑的。因此,学习用神经网络来解决非光滑最优化问题是十分必要的。

在文献[3]中, Forti 等人介绍了一种广义神经网络用于解决一类非光滑非线性规划问题。Li 等人^[4]提出用广义梯度神经网络解决在封闭凸子集上的非光滑凸和非凸非线性规划问题。Liu 等人在文献[5,6]中提出了一类基于投影运算的单层反馈神经网络来解决目标函数和约束函数分别为不同形式的非光滑最优化问题,分析了网络的稳定性及其收敛性。无论待解决的最优化问题是否光滑,大部分的神经网络技术解决约束最优化问题都是带惩罚项的。例如在文献[7,8]中, Xue 等人

利用惩罚函数和微分包含技术提出了一种解决非光滑凸和非光滑非凸最优化问题的基于子梯度法神经网络。不幸的是惩罚函数法有很多致命的缺点,例如惩罚函数法只能给出近似解,当惩罚参数是无限大时无法用电路实现,并且使用惩罚函数法在计算上有很大的困难。在文献[9]中, Zhang 等人提出了一类新型神经网络用于解决一般非线性规划问题,这种方法是基于著名的拉格朗日乘子理论,通过把约束合并到一个修正目标函数中来处理约束。这种方法避免了计算上的困难。此外,文献[10~12]也研究了利用拉格朗日神经网络解决不等式约束的最优化问题。然而它们所解决优化问题的目标函数都仅限于光滑函数,所以有待于将这类神经网络的应用扩展到非光滑目标函数的研究中。

在文献[13]中, Bian 等人提出了一种光滑神经网络解决非 Lipschitz 约束最优化问题,并介绍了其应用。与文献[13]不同的是,本文研究了一种光滑拉格朗日神经网络以解决非光滑约束最优化问题。这种神经网络包含两种神经元,即变量神经元和拉格朗日神经元。变量神经元负责寻找原问题的最小点,此神经元最终会给出神经网络的平衡点,以此来为寻找原始问题的最小点作进一步的判断;而拉格朗日神经元则负责将动态轨迹快速地引入可行域。求解过程如下:a)采用光滑逼近技

收稿日期: 2013-07-17; **修回日期:** 2013-08-27 **基金项目:** 国家自然科学基金资助项目(61063045); 广西科技攻关基金资助项目(桂科攻11107006-1); 广西教育厅资助项目(TLZ100715)

作者简介: 喻昕(1973-),男,重庆人,教授,博士,主要研究方向为人工神经网络、互连网络、智能计算等(yuxin21@126.com); 于琰(1986-),女,山东烟台人,硕士研究生,主要研究方向为智能计算; 谢緬(1990-),女,湖北潜江人,硕士研究生,主要研究方向为智能计算。

术将原始问题的目标函数由非光滑的局部 Lipschitz 函数转换成光滑函数(即二次连续可微函数);b)根据变换后的光滑函数建立拉格朗日神经网络模型,并利用神经网络的动态行为找到其平衡点。文中证明了神经网络的平衡点集合是原始非光滑最优化问题关键点集合的一个子集,并且当原始问题的目标函数是凸函数时,原始问题的最小点集合与神经网络的平衡点集合是一致的,这为寻找原始非光滑问题的最优解提供了理论依据。

1 预备知识

本章介绍本文中要用到的一些定义和性质。首先,给出以下等式约束的非线性规划问题:

$$\begin{cases} \min & f(x) \\ \text{subject to} & h(x) = 0 \end{cases} \quad (1)$$

其中: $x = (x_1, x_2, \dots, x_n)^T \in R^n$; $f: R^n \rightarrow R$ 和 $h: R^n \rightarrow R^m$ 分别是目标函数和约束函数且 $m \leq n$ 。函数 h 由 $h_1, h_2, \dots, h_m$ 组成。这里 f 是局部 Lipschitz 函数, h 是光滑凸函数(即二次连续可微)。

由约束函数定义的可行区域为 $S = \{x \in R^n; h_j(x) = 0, j = 1, 2, \dots, m\}$ 。因此,本文假设 S 是非空有界的,并且是内部非空的集合。如果 $f(x)$ 是局部 Lipschitz 函数,则 $f(x)$ 几乎处处可微。用 D_f 表示 f 可微点的集合,那么 Clarke 广义梯度^[14] 定义为

$$\partial f(x) = \text{co} \left\{ \lim_{x_k \rightarrow x; x_k \in D_f} \nabla f(x_k) \right\} \quad (2)$$

命题 1^[14] 对任意固定的 $x \in R^n$, 满足以下结论:

- a) $\partial f(x)$ 是 R^n 上的非空凸紧集;
- b) $\partial f(x)$ 在 x 上是半连续的。

定义 1 对于非光滑最优化问题式(1),有以下两个定义:

- a) 关键点集合 C 为

$$C = \{x \in S; 0 \in \partial f(x) + N_S(x)\} \quad (3)$$

其中: $N_S(x)$ 是 x 在 S 上的正规锥,从文献[14]可知

$$N_S(x) = \text{cl} \left(\bigcup_{v \geq 0} v \partial d_S(x) \right) \quad (4)$$

其中: cl 表示弱闭包, $d_S(x) = \min_{u \in S} \|x - u\|^2$ 是全局 Lipschitz 函数。

- b) 全局最小点集合 M 为

$$M = \arg \min_{x \in S} f(x) \neq \varnothing \quad (5)$$

值得注意的是,当问题式(1)为凸函数时,关键点集合与全局最小点集合是一致的,即 $M = C$ 。

在过去的几十年里,研究者们提出了很多种光滑逼近技术用来解决非光滑最优化问题。光滑逼近技术的主要思想就是用带参数的光滑函数来近似非光滑函数。

定义 2 令 $g: R^n \rightarrow R$ 是局部 Lipschitz 函数,把函数 $\tilde{g}: R^n \times [0, +\infty)$ 叫做函数 g 的光滑函数,如果 $\tilde{g}$ 满足以下条件:

- a) 对任意给定的 $\mu > 0$, $\tilde{g}(\cdot, \mu)$ 在 R^n 上是连续可微的,并且对任意给定的 $x \in R^n$, $\tilde{g}(\cdot, \mu)$ 在 $[0, +\infty)$ 内是可微的;

- b) 对任意给定的 $x \in R^n$, $\lim_{\mu \downarrow 0} \tilde{g}(x, \mu) = g(x)$;

- c) 对于正常数 $\kappa_{\tilde{g}} > 0$,有以下关系式:

$$\|\nabla_{\mu} \tilde{g}(x, \mu)\| \leq \kappa_{\tilde{g}}, \forall \mu \in [0, +\infty), x \in R^n;$$

- d) $\{\lim_{z \rightarrow x, \mu \downarrow 0} \nabla_z \tilde{g}(z, \mu)\} \subseteq \partial g(x)$ 。

对于给定的 $x \in R^n$,由定义 2 中的 b) 和 c),可以得到以下结论:

$$\lim_{z \rightarrow x, \mu \downarrow 0} \tilde{g}(z, \mu) = g(x) \quad (6)$$

更多有关构造光滑函数的结论在文献[15]中可以查阅,它们对于构造光滑函数有很大的指导作用。本文始终运用光滑函数 $\tilde{f}(x, \mu)$ 来近似非光滑函数 $f(x)$ 。

定义 3 拉格朗日函数 $L_1: R^{n+m} \rightarrow R$ 由非光滑函数 $f(x)$ 和约束条件 $h(x) = 0$ 组合而成,定义为

$$L_1(x, \lambda) = f(x) + \lambda^T h(x)$$

其中: $\lambda \in R^m$ 是拉格朗日乘子。

定义 4 拉格朗日函数 $L_2: R^{n+m} \rightarrow R$ 由光滑函数 $\tilde{f}(x, \mu)$ 和约束条件 $h(x) = 0$ 组合而成,定义为

$$L_2(x, \lambda) = \tilde{f}(x, \mu) + \lambda^T h(x)$$

其中: $\lambda \in R^m$ 是拉格朗日乘子。

一阶最优化条件可以描述为: $L_2(x, \lambda)$ 在 x 和 λ 上的稳定点为 (x^*, λ^*) ,用公式表示如下:

$$\nabla_x L_2(x^*, \lambda^*) = \nabla f(x^*, \mu) + \nabla h(x^*) \lambda^* = 0$$

$$\nabla_{\lambda} L_2(x^*, \lambda^*) = h(x^*) = 0$$

注:

$$\nabla_x L_2(x, \lambda) = \left[\frac{\partial L_2(x, \lambda)}{\partial x_1} \frac{\partial L_2(x, \lambda)}{\partial x_2} \dots \frac{\partial L_2(x, \lambda)}{\partial x_n} \right]^T$$

$$\nabla h = [\nabla h_1, \dots, \nabla h_m]$$

2 主要定理及其证明

2.1 定义拉格朗日神经网络

本文中光滑拉格朗日神经网络的瞬时行为定义为以下两个微分等式:

$$\frac{dx}{dt} = -\nabla_x L_2(x, \lambda) \quad (7)$$

$$\frac{d\lambda}{dt} = \nabla_{\lambda} L_2(x, \lambda) \quad (8)$$

其状态方程为

$$\frac{dx}{dt} = -\nabla_x \tilde{f}(x(t), \mu(t)) - \lambda \nabla_x h(x(t)) \quad (9)$$

$$\frac{d\lambda}{dt} = h(x(t)) \quad (10)$$

更精确的形式为

$$\frac{dx_i}{dt} = -\frac{\partial f}{\partial x_i} - \sum_{j=1}^m \lambda_j \frac{\partial h_j}{\partial x_i} \quad i = 1, 2, \dots, n \quad (11)$$

$$\frac{d\lambda_j}{dt} = h_j \quad j = 1, 2, \dots, m \quad (12)$$

其中: $\tilde{f}: R^n \times [0, +\infty) \rightarrow R$ 是 $f(x)$ 的光滑函数; $\mu: [0, +\infty) \rightarrow [0, +\infty)$ 是一个连续可微且递减函数,满足 $\lim_{t \rightarrow \infty} \mu(t) = 0$ 。

注:本文选择 $\mu(t)$ 为

$$\mu(t) = e^{-t} / (1 + e^{-t})^{\alpha} \quad (13)$$

这里的参数 α 是一个正数。

命题 2^[9] 如果点对 (x^*, λ^*) 使神经网络达到平衡状态,且向量 x^* 满足 $h(x^*) = 0$,那么 $\nabla h_1(x^*), \dots, \nabla h_m(x^*)$ 是线性无关的。

定义 5 拉格朗日神经网络的平衡点集合 ε_{δ} 为

$$\varepsilon_{\delta} = \{x^* \in R^n; \nabla_x L_2(x^*, \lambda^*) = 0, \nabla_{\lambda} L_2(x^*, \lambda^*) = 0\} \quad (14)$$

2.2 相关定理及证明

下面给出本文重要的定理及其证明过程。

定理1 对给定的 $x \in R^n$, 拉格朗日神经网络的平衡点集合 $\mathcal{E}_\delta$ 与原始问题式(1)的关键点集合 C 满足关系: $\mathcal{E}_\delta \subseteq C$ 。

证明 假设 x^* 是神经网络的任意一个平衡点, 即 $x^* \in \mathcal{E}_\delta$, 由一阶最优化条件知: 存在点对 (x^*, λ^*) 使得拉格朗日函数 $L_2(x, \lambda)$ 达到稳定状态, 所以满足等式 $\nabla_x L_2(x^*, \lambda^*) = 0$ 和 $\nabla_\lambda L_2(x^*, \lambda^*) = 0$, 即

$$\begin{cases} 0 = \nabla_x f(x^*, \mu) + \nabla_x h(x^*) \lambda^* \\ 0 = h(x^*) \end{cases} \quad (15)$$

对拉格朗日函数 $L_1(x, \lambda)$ 求 x 方向导数

$$\nabla_x L_1(x, \lambda) = \partial f(x) + \lambda^T \nabla h(x) \quad (16)$$

由 Clarke 广义梯度知:

$$\partial f(x) = \text{co} \left\{ \lim_{x_k \rightarrow x; \mu_k \rightarrow 0} \nabla f(x_k) \right\}$$

又由定义2知:

$$\{ \lim_{z \rightarrow x; \mu \downarrow 0} \nabla_z f(z, \mu) \} \subseteq \partial f(x)$$

所以 $\nabla_x f(x^*, \mu) \in \partial f(x^*)$ (17)

$$\nabla_x f(x^*, \mu) + \nabla_x h(x^*) \lambda^* \in \partial f(x^*) + \nabla_x h(x^*) \lambda^* \quad (18)$$

由式(15)可得

$$0 \in \partial f(x^*) + \nabla_x h(x^*) \lambda^* \quad (19)$$

由于 x^* 是神经网络的平衡点, 由命题2知, 方向导数 $\nabla h_1(x^*), \dots, \nabla h_m(x^*)$ 是线性无关的。根据文献[14]知:

$$N_S(x) = \{ \sum_{i=1}^m v_i \nabla h_i(x); v_i \geq 0, i=1, 2, \dots, m \} \quad (20)$$

因此, 必存在一组 v_i 使得 $\lambda_i^* = v_i (i=1, 2, \dots, m)$, 从而有

$$\nabla h(x^*) \lambda^* \in N_S(x^*) \quad (21)$$

则 $\partial f(x^*) + \nabla h(x^*) \lambda^* \in \partial f(x^*) + N_S(x^*)$ (22)

结合式(3)(19)和(22)可得 $x^* \in C$, 其中 C 是问题式(1)的关键点集合。由于 $x^* \in \mathcal{E}_\delta$, 所以 $\mathcal{E}_\delta \subseteq C$, 其中 $\mathcal{E}_\delta$ 是神经网络的平衡点集合。

定理2 当原始问题式(1)是凸函数时, 神经网络的平衡点集合 $\mathcal{E}_\delta$ 与原始问题式(1)的全局最小点集合 M 是一致的, 即 $M = \mathcal{E}_\delta$ 。

证明 定义神经网络的 Lyapunov 能量函数 $E(x, \lambda)$ 为

$$E(x, \lambda) = \frac{1}{2} |\nabla_x L_2(x, \lambda)|^2 + \frac{1}{2} |h(x)|^2 \quad (23)$$

对 $E(x, \lambda)$ 求 t 的导数

$$\frac{dE(x, \lambda)}{dt} = -\nabla_x L_2(x, \lambda) \nabla_{xx}^2 L_2(x, \lambda)^T \nabla_x L_2(x, \lambda) \quad (24)$$

由文献[9]知能量函数满足 $dE(x, \lambda)/dt \leq 0$, 又由式(23)知 $E(x, \lambda) \geq 0$, 即能量函数有下界。因此根据 Lyapunov 能量函数的稳定性, 此拉格朗日神经网络可以达到全局稳定状态。

假设 x^* 是神经网络的一个平衡点, 由式(14)知 $x^* \in \mathcal{E}_\delta$, 则存在点对 (x^*, λ^*) 使得式(15)成立。所以当神经网络达到平衡点 x^* 时, $dE(x, \lambda)/dt = 0$, 即神经网络的能量达到了下界值, 此时神经网络达到了稳定状态。由于原始问题式(1)是凸函数, 则由原始问题的目标函数变换而来的光滑函数 $\tilde{f}(x, \mu)$ 也是凸函数。由 Lyapunov 的全局稳定性知, 从任一初始点出发都能找到一个点对 (x^*, λ^*) , 使得神经网络能够达到全局稳定状态。由定理1知此时的 x^* 也是原始问题式(1)的关键点。由于原始问题式(1)是凸函数, 关键点集合与全局最小点集合是一致的, 所以 x^* 也是原始式(1)的全局最小点。因此, $M = \mathcal{E}_\delta$ 成立。

推论 a) 当 $\nabla_x L_2(x^*, \lambda) = 0$ 时, $0 \in \nabla_x L_1(x^*, \lambda)$, 则 x 趋近于极值点 x^* ; b) 当 $\nabla h_1(x^*), \dots, \nabla h_m(x^*)$ 线性无关时,

λ 也趋近于极值点 λ^* 。

证明 a) 当 $\nabla_x L_2(x^*, \lambda) = 0$ 时, 由式(24)知, 能量函数满足 $dE(x, \lambda)/dt = 0$, 此时神经网络达到稳定状态, 这时由于 $0 \in \nabla_x L_1(x^*, \lambda)$, 所以可以推知 x 达到了极限值 x^* 。

b) 当 $\nabla h_1(x^*), \dots, \nabla h_m(x^*)$ 线性无关时, 有

$$\nabla_x L_2(x^*, \lambda) = \nabla f(x^*, \mu) + \nabla h(x^*) \lambda = 0$$

通过定理1得

$$0 \in \nabla_x L_1(x^*, \lambda) = \partial f(x^*) + \nabla h(x^*) \lambda$$

由此可知 λ 被唯一确定。因此, $d\lambda/dt \rightarrow 0$, 使得 λ 有一个极限值 λ^* 。

3 仿真实验结果及分析

3.1 仿真实验过程

仿真实验是在 MATLAB 9.0 平台下进行的。实验包括两个实例: 例1是求解非光滑函数在等式约束下的最优化问题, 例2是求解非光滑函数在不等式约束下的最优化问题。

例1 等式约束

$$\begin{aligned} \min \quad & f(x) = |x_1 - x_2| \\ \text{subject to} \quad & x_1 - 2x_2 + 3 = 0 \end{aligned}$$

问题分析: 由给出的非光滑问题可以得到, 当 $x_1 = x_2$ 时 $f(x) = 0$ 达到最小值, 在满足约束条件 $x_1 - 2x_2 + 3 = 0$ 时, $x_1 = x_2 = 3$, 所以原始问题在满足约束条件下仅有一个最小点为 $x = (3, 3)^T$, 即最小点集合为 $M = \{(3, 3)^T\}$ 。

实验步骤如下:

a) 利用光滑逼近技术^[15]将非光滑函数 $f(x)$ 转换成相应的光滑函数 $\tilde{f}(x, \mu)$ 。选择 $\mu(t) = e^{-t}/(1 + e^{-t})^2$, 则

当 $\mu_0 > 0$ 时, $(s)_+$ 的光滑函数为

$$\varphi(s, \mu_0) = \int_{-\infty}^{\infty} (s - \mu_0 t) + \mu(t) dt = \mu_0 \ln(1 + e^{s/\mu_0})$$

由 $|s| = (s)_+ + (-s)_+$ 可以得到 $|s|$ 的光滑函数为

$$\begin{aligned} \psi(s, \mu_0) &= \varphi(s, \mu_0) + \varphi(-s, \mu_0) = \\ &= \mu_0 \ln(2 + e^{s/\mu_0} + e^{-s/\mu_0}) \end{aligned}$$

所以, 综上可知 $f(x) = |x_1 - x_2|$ 的光滑函数为

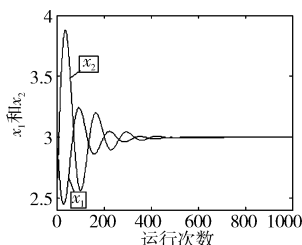
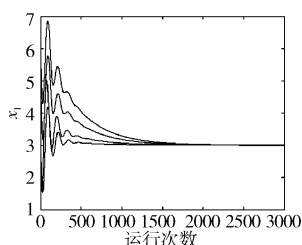
$$\tilde{f}(x, \mu_0) = \mu_0 \ln(2 + e^{(x_1 - x_2)/\mu_0} + e^{-(x_1 - x_2)/\mu_0})$$

b) 通过拉格朗日神经网络模型的式(7)和(8), 建立具体的神经网络学习规则^[16]为

$$\begin{aligned} x_j(k+1) &= x_j(k) - \mu_x \left[\frac{\partial \tilde{f}(x(k))}{\partial x_j} + \sum_{i=1}^m \lambda_i(k) \frac{\partial h_i(x(k))}{\partial x_j} \right] \\ \lambda_j(k+1) &= \lambda_j(k) + \mu_\lambda h_j(x(k)) \end{aligned}$$

其中: $\mu_x = 0.05$; $\mu_\lambda = 0.01$ 是学习率参数, 约束条件个数 $i = 1, \dots, m$, 运行次数 $j = 1, \dots, n$ 。

c) 用 MATLAB 9.0 软件编程来模拟神经网络的行为。当任意选取一个初始点为 $x = (3, 2.5)^T$, $\lambda = 0.6$, 并且假定 $\mu_0 = 2$ (满足 $\mu_0 > 0$) 时, 最终当网络达到平衡时收敛到平衡点 $x^* = (3, 3)^T$ 。图1给出了 x_1 和 x_2 的轨迹图。另外, 为了体现本例子选取初始点的任意性, 任取四个初始点分别为 $(3, 2.5)$, $(3.75, 2.25)$, $(3.5, 3)$ 和 $(2.5, 2)$, 最终 x 收敛到平衡点仍然是 $x^* = (3, 3)^T$ 。图2给出了这四个点中 x_1 的轨迹。因此, 得到平衡点集合为 $\mathcal{E}_\delta = \{(3, 3)^T\}$, 这与问题分析中的最小点集合是一致的。

图1 例1中 x_1 和 x_2 的轨迹图2 例1中任意四个初始点 x_1 的轨迹

例2 不等式约束

$$\begin{aligned} \min \quad & f(x) = |x_1 - x_2| \\ \text{subject to} \quad & x_1 - 2x_2 + 3 \leq 0 \\ & x_1 - x_2 \leq 0 \end{aligned}$$

问题分析:由给出的非光滑问题可以得到,在满足约束条件 $x_1 - 2x_2 + 3 \leq 0$ 和 $x_1 - x_2 \leq 0$ 时,当 $x_1 = x_2$ 时 $f(x) = 0$ 达到最小值。所以原始问题的最小点集合是在满足约束条件且 $x_1 = x_2$ 构成的集合。

实验步骤如下:

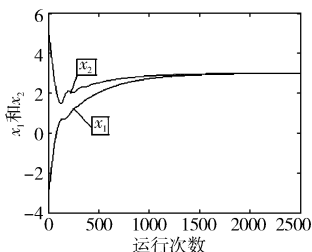
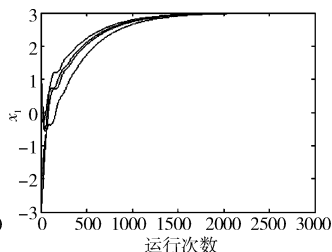
a)利用光滑逼近技术将非光滑函数转换成光滑函数,变换过程同例1。

b)构造不等式约束的神经网络模型,建立学习规则^[16]:

$$\begin{aligned} x_j(k+1) &= x_j(k) - \mu_x \left[\frac{\partial f(x(k))}{\partial x_j} + \sum_{i=1}^m \lambda_i(k) \frac{\partial h_i(x(k))}{\partial x_j} \right] \\ \lambda_j(k+1) &= \max\{0, \lambda_i(k) + \mu_\lambda h_j(x(k))\} \end{aligned}$$

其中: $\mu_x = 0.05$, $\mu_\lambda = 0.01$ 是学习率参数,约束条件个数 $i = 1, \dots, m$,运行次数 $j = 1, \dots, n$ 。

c)用MATLAB 9.0 软件编程来模拟神经网络的行为。当任意选取一个初始点为 $x = (-3, 5)^T$, $\lambda = (0.6, 0.2)^T$,且假定 $\mu_0 = 2$ (满足 $\mu_0 > 0$)时,最终当网络达到平衡时收敛到平衡点 $x^* = (3, 3)^T$ 。图3给出了 x_1 和 x_2 的轨迹。同样,为了体现本例子选取初始点的任意性,任取四个初始点分别为 $(-3, 5)$, $(-2, 2)$, $(0, 1)$ 和 $(1.5, -1)$,最终 x 收敛到平衡点为 $x^* = (3, 3)^T$ 。图4给出了这四个点中 x_1 的轨迹。因此,由实验得到的平衡点符合问题分析中的最小点集合的条件。

图3 例2中 x_1 和 x_2 的轨迹图4 例2中任意四个初始点 x_1 的轨迹

3.2 实验结果分析

函数 $f(x) = |x_1 - x_2|$ 在满足 $x_1 - 2x_2 + 3 = 0$ 的前提下,当 x_1 和 x_2 值相等时, $f(x)$ 就会得到最小值0。因此,当把非光滑函数 $f(x)$ 变换成光滑函数 $\tilde{f}(x, \mu_0)$ 后,本研究希望 $\tilde{f}(x, \mu_0)$ 在等式 $x_1 - 2x_2 + 3 = 0$ 约束下达到最小值时, x_1 和 x_2 值也是相等的。从例1的仿真结果中可以看出,当 x_1 和 x_2 的特征曲线趋于不变时,神经网络达到稳定状态,这时 x_1 的值与 x_2 相等,这与期望的结果是相符的。因此,验证了本文所提出的用光滑拉格朗日神经网络来解决等式约束非光滑目标函数优化问题的有效性。在例2中,当约束条件是不等式时,从仿真结果图2可以看出,在可行域内当神经网络达到稳定时, x_1 与 x_2 的值

相等,这时的 x_1 和 x_2 使得原问题 $f(x) = |x_1 - x_2|$ 达到了最小值。因此,例2从不等式约束的角度验证了本文结论的正确性。

4 结束语

本文介绍了一类约束非光滑最优化问题,这类问题在工程学、自然科学、经济学等方面都有广泛的应用。通过引进光滑逼近技术和光滑拉格朗日神经网络来解决这类约束非光滑最优化问题。同时利用了Clarke广义梯度、一阶最优化条件、Lyapunov能量函数稳定性等理论证明了由光滑函数构造的光滑拉格朗日神经网络的平衡点集合是原问题关键点集合的子集。最后,通过MATLAB编程进行仿真实验,验证了理论结果。

参考文献:

- [1] TANK D W, HOPFIELD J J. Simple 'neural' optimization networks: an A/D converter, signal decision circuit, and a linear programming circuit[J]. IEEE Trans on Circuits System, 1986, 33(5):533-541.
- [2] KENNEDY M P, CHUA L O. Neural networks for nonlinear programming[J]. IEEE Trans on Circuits System, 1988, 35(5):554-562.
- [3] FORTI M, NISTRI P, QUINCAMPOIX M. Generalized neural network for nonsmooth nonlinear programming problems[J]. IEEE Trans on Circuits and Systems—I: Regular Papers, 2004, 51(9):1741-1754.
- [4] LI Guo-cheng, SONG Shi-ji, WU Cheng. Generalized gradient projection neural networks for nonsmooth optimization problems[J]. Science China Information Sciences, 2010, 53(5):990-1005.
- [5] LIU Qing-shan, WANG Jun. A one-layer projection neural network for nonsmooth optimization subject to linear equalities and bound constraints[J]. IEEE Trans on Neural Networks and Learning Systems, 2013, 24(5):812-824.
- [6] LIU Qing-shan, WANG Jun. Finite-time convergent recurrent neural network with a hard-limiting activation function for constrained optimization with piecewise-linear objective functions[J]. IEEE Trans on Neural Networks, 2011, 22(4):601-613.
- [7] XUE Xiao-ping, BIAN Wei. Subgradient-based neural networks for nonsmooth convex optimization problems[J]. IEEE Trans on Circuits and Systems—I: Regular Papers, 2008, 55(8):2378-2391.
- [8] BIAN Wei, XUE Xiao-ping. Subgradient-based neural networks for nonsmooth nonconvex optimization problems[J]. IEEE Trans on Neural Network, 2009, 20(6):1024-1038.
- [9] ZHANG Sheng-wei, CONSTANTINIDES A G. Lagrange programming neural networks[J]. IEEE Trans on Circuits and Systems—II: Analog and Digital Signal Processing, 1992, 39(7):441-452.
- [10] HUANG Yuan-can, YU Chang. Improved Lagrange nonlinear programming neural networks for inequality constraints[C]//Proc of International Joint Conference on Neural Networks. 2007.
- [11] HUANG Yuan-can. Lagrange-type neural networks for nonlinear programming problems with inequality constraints[C]//Proc of the 44th IEEE Conference on Decision and Control, and the European Control Conference. 2005:4129-4133.
- [12] HUANG Yuan-can. A novel method to handle inequality constraints for convex programming neural network[J]. Neural Processing Letters, 2002, 16(1):17-27.
- [13] BIAN Wei, CHEN Xiao-jun. Smoothing neural network for constrained non-Lipschitz optimization with applications[J]. IEEE Trans on Neural Network and Learning Systems, 2012, 23(3):399-411.
- [14] CLARKE F H. Optimization and non-smooth analysis[M]. New York: Wiley, 1983:30-57.
- [15] CHEN Xiao-jun. Smoothing methods for nonsmooth, nonconvex minimization[EBOL]. (2012) [2013-05-20]. <http://www.polyu.edu.hk/ama/staff/xjchen/MPB2012.pdf>.
- [16] HAM F M, KOSTANIC I. 神经计算原理[M]. 叶世伟, 王海娟, 译. 北京: 机械工业出版社, 2007:178-213.