Neurocomputing 129 (2014) 298–305

Contents lists available at ScienceDirect

Neurocomputing

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

Recurrent networks for compressive sampling

Chi-Sing Leung [a,*], John Sum [b], A.G. Constantinides [c]

[a] The Department of Electronic Engineering, City University of Hong Kong, Kowloon Tong, KLN, Hong Kong
[b] Institute of Technology Management, National Chung Hsing University, Taiwan
[c] Imperial College, UK

ARTICLE INFO

Article history:
Received 14 April 2013
Received in revised form
31 July 2013
Accepted 19 September 2013
Communicated by D. Wang
Available online 1 November 2013

Keywords:
Neural circuit
Stability

ABSTRACT

This paper develops two neural network models, based on Lagrange programming neural networks (LPNNs), for recovering sparse signals in compressive sampling. The first model is for the standard recovery of sparse signals. The second one is for the recovery of sparse signals from noisy observations. Their properties, including the optimality of the solutions and the convergence behavior of the networks, are analyzed. We show that for the first case, the network converges to the global minimum of the objective function. For the second case, the convergence is locally stable.

1. Introduction

Nonlinear constrained optimization problems have been studied over several decades [1–3]. Conventional ways for solving them are based on numerical methods [1,2]. As mentioned by many neural network pioneers [3–7], when realtime solutions are required, the neural circuit approach [3–5] is more effective. In the neural circuit approach, we do not solve them in a digital computer. Instead, we set up an associated neural circuit for the given constrained optimization problem. After the neural circuit settles down at one of the equilibrium points, the solution is obtained by measuring the neuron output voltages at this stable equilibrium point. So, one of the important issues is the stability of equilibrium points.

In the past two decades, many results of neural circuits were reported. For instance, Hopfield [4] investigated a neural circuit for solving quadratic optimization problems. In [5,8–10] a number of models were proposed to solve various nonlinear constrained optimization problems. The neural circuit approach is able to solve many engineering problems effectively. For example, it can be used for optimizing microcode [11]. Besides, it is able to search the maximum of a set of numbers [12–14]. In [15], the Lagrange programming neural network (LPNN) model was proposed to solve general nonlinear constrained optimization problems. For many years, many neural models for engineering problems were addressed. However, little attention has been paid to analog neural circuits for compressing sampling.

In compressing sampling [16–18], a sparse signal is sampled (measured) by a few random-like basis functions. The task of compressing sampling is to recover the sparse signal from the measurements. Compressive sampling can also be applied to a non-sparse signal by increasing the sparsity of the signal. It can be done by using some transform coding techniques. Conventional approaches for recovering sparse signals are based on the Newton's method [19–21].

As the neural circuit approach is a good alternative for nonlinear optimization, it is interesting to investigate the ways to apply the neural circuit approach for compressing sampling. This paper proposes two analog neural network models, based on LPNNs, for compressive sampling. One is for the standard recovery. Another one is for the recovery from noisy measurements. Since the norm-1 measure is not twice differentiable, it is difficult to construct a neural circuit. Hence this paper proposes an approximation for the objective function. With the approximation, the hyperbolic tangent function, which is a commonly used activation function in the neural network community, is involved. We use experiments to investigate how the hyperbolic tangent parameter affects the performance of the proposed models. Since the convergence and stability of neural circuits are important issues, this paper investigates the stability and optimality of our approaches. For the case of the standard recovery, we show that the proposed neural model converges to the global minimum of the objective function. For the case of the recovery from noisy measurements, we show that the neural model is locally stable.

This paper is organized as follows. In Section 2, the backgrounds of compressive sampling and LPNNs are reviewed. Section 3 formulates neural models for compressive sampling. The theoretical analysis on the proposed neural models is also presented.

* Corresponding author.
E-mail address: eeleungc@cityu.edu.hk (C.-S. Leung).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.09.028

Section 4 presents our simulation results. Section 5 concludes our results.

2. Background

2.1. Compressive sampling

In compressive sampling [16–18], we would like to find a sparse solution $\boldsymbol{x} \in \Re^n$ of an underdetermined system, given by

$$\boldsymbol{b} = \boldsymbol{\Phi}\boldsymbol{x}, \tag{1}$$

where $\boldsymbol{b} \in \Re^m$ is the observation vector, $\boldsymbol{\Phi} \in \Re^{m \times n}$ is the measurement matrix with a rank of m, $\boldsymbol{x} \in \Re^n$ is the unknown sparse vector to be recovered, and $m < n$. In a more precise way, the sparsest solution is defined as

$$\min \quad \|\boldsymbol{x}\|_0 \tag{2a}$$

$$\text{subject to} \quad \boldsymbol{b} = \boldsymbol{\Phi}\boldsymbol{x}. \tag{2b}$$

Unfortunately, problem (2) is NP-hard. Therefore, we usually replace the l_0-norm measure with the l_1-norm measure. The problem for recovering sparse signals becomes

$$\min \quad \|\boldsymbol{x}\|_1 \tag{3a}$$

$$\text{subject to} \quad \boldsymbol{b} = \boldsymbol{\Phi}\boldsymbol{x}. \tag{3b}$$

This problem is known as basis pursuit (BP) [17,19]. Let $\boldsymbol{\phi}_j$ be the j-th column of $\boldsymbol{\Phi}$. Define the mutual coherence [18] of $\boldsymbol{\Phi}$ as

$$\mu(\boldsymbol{\Phi}) = \max_{i \neq j} \frac{|\boldsymbol{\phi}_i^T \boldsymbol{\phi}_j|}{\|\boldsymbol{\phi}_i\|_2 \|\boldsymbol{\phi}_j\|_2}. \tag{4}$$

If the cardinality of the true solution obeys $\|\boldsymbol{x}\|_0 < \frac{1}{2}(1 + 1/\mu(\boldsymbol{\Phi}))$, then the solution of (3) is exactly the same as that of (2). When the measurement matrix has independent Gaussian entries and the number m of measurements is greater than $2k \log(n/k)$, the BP [22] can reconstruct the sparse signal with high probability.

When there is measurement noise in $\boldsymbol{b}$, the sampling process becomes

$$\boldsymbol{b} = \boldsymbol{\Phi}\boldsymbol{x} + \boldsymbol{\xi}, \tag{5}$$

where $\boldsymbol{\xi} = [\xi_1, \xi_2, \ldots, \xi_m]^T$, and ξ_i's are independent identical random variables with zero mean and variance σ^2. In this case, our problem becomes

$$\min \quad \|\boldsymbol{x}\|_1 \tag{6a}$$

$$\text{subject to} \quad \|\boldsymbol{b} - \boldsymbol{\Phi}\boldsymbol{x}\|^2 \le m\sigma^2. \tag{6b}$$

2.2. Lagrange programming neural networks

The LPNN approach aims at solving a general nonlinear constrained optimization problem, given by

$$\text{EP}: \ \min \quad f(\boldsymbol{x}) \tag{7a}$$

$$\text{subject to} \quad \boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}, \tag{7b}$$

where $\boldsymbol{x} \in \Re^n$ is the state of system, $f : \Re^n \to \Re$ is the objective function, and $\boldsymbol{h} : \Re^n \to \Re^m$ $(m < n)$ describes the m equality constraints. The two functions f and $\boldsymbol{h}$ are assumed to be twice differentiable.

The LPNN approach first sets up a Lagrange function for EP, given by

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) = f(\boldsymbol{x}) + \boldsymbol{\lambda}^T \boldsymbol{h}(\boldsymbol{x}), \tag{8}$$

where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_m]^T$ is the Lagrange multiplier vector. The LPNN approach then defines two kinds of neurons: variable neurons and Lagrange neurons. The variable neurons hold the state variable vector $\boldsymbol{x}$. The Lagrange neurons hold the Lagrange multiplier vector $\boldsymbol{\lambda}$.

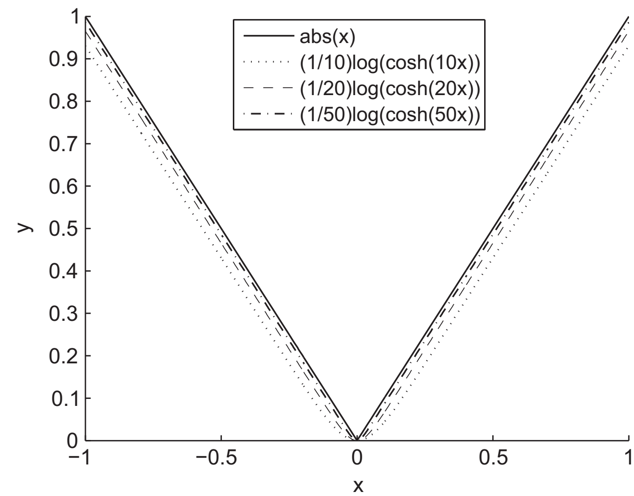

Fig. 1. $y = (1/a)\ln(\cosh(ax))$.

The LPNN approach defines two updating equations for those neurons, given by

$$\frac{1}{\varepsilon}\frac{d\boldsymbol{x}}{dt} = -\nabla_x \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}) \tag{9a}$$

$$\frac{1}{\varepsilon}\frac{d\boldsymbol{\lambda}}{dt} = \nabla_\lambda \mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}), \tag{9b}$$

where ε is the time constant of the circuit. The time constant depends on the circuit resistance and the circuit capacitance. Without loss of generality, we set ε to 1. The variable neurons seek for a state with the minimum cost in a system while the Lagrange neurons are trying to constrain the system state of the system such that the system state falls into the feasible region. With (9), the network will settle down at a stable state [15] if the network satisfies some conditions.

3. LPNNs for compressive sampling

3.1. Sparse signal

From (3) and (7), one may suggest that we can set up a LPNN to solve problem (3). However, the absolute operator $|\cdot|$ is not differentiable at $x_i = 0$. Hence, we need an approximation. In this paper, we use the following approximation:

$$|x_i| \approx \frac{\ln(\cosh(a x_i))}{a}, \tag{10}$$

where $a > 1$. Fig. 1 shows the shape of $(1/a)\ln(\cosh(ax))$. From the figure, the approximation[1] is quite accurate for a large a.

Thanks to the property of $(1/a)\ln(\cosh(ax))$, the hyperbolic tangent function is involved in the dynamic equations of the neural circuit. It should be noticed that the hyperbolic tangent function [23] is the commonly used activation in the neural network community. Also, comparators or amplifiers [24] are associated with the hyperbolic-tangent relation between input and output for differential bipolar pairs.

[1] In Section 4, we will use simulation examples to show the effectiveness of this approximation.