A NONFEASIBLE QUADRATIC APPROXIMATION RECURRENT NEURAL NETWORK FOR EQUALITY CONstrained OPTimization PROBLEMS

(Selected from CEMA’11 Conference)

Maria P. Barbarosou* and Nicholas G. Maratos**

*Hellenic Air-Force Academy, Department of Aeronautical Sciences,
Division of Electronics and Communication Engineering
Dekelia Air Force Base, GR 1010, Greece,
E-mail: mbarbar@teipir.gr.

**National Technical University of Athens,
School of Electrical and Computer Engineering,
9 Iroon Polytechniou St., GR 15773, Athens, Greece,
E-mail: maratos@ece.ntua.gr.

Abstract

Convex optimization techniques are widely used in the design and analysis of communication systems and signal processing algorithms. In this paper a novel recurrent neural network is presented for solving nonlinear strongly convex equality constrained optimization problems. The proposed neural network is based on recursive quadratic programming for nonlinear optimization, in conjunction with homotopy method for solving nonlinear algebraic equations. It constructs generally a non-feasible trajectory which satisfies the constraints as \( t \to \infty \). The boundedness of solutions and the global convergence to the optimal point of the problem are proven. The correctness and the performance of the proposed neural network are evaluated by simulation results on illustrative numerical examples.

1. INTRODUCTION

The use of convex optimization is ubiquitous in communications and signal processing. Many problems in these fields can be converted into convex optimization problems, which greatly facilitate their analysis and numerical solutions [1]-[2].

Consider the following equality constrained optimization problem:

\[
\min \ \{ f(x), \ A^T x - b = 0 \} \tag{1}
\]

where \( f: R^n \to R \), \( A \) an \( nxm \) matrix with \( m < n \) and \( b \) an \( m \) vector. We make the following assumption, standard for quadratic approximation programming:
**Assumption:** (a) The function $f$ is strongly convex and twice continuously differentiable in $\mathbb{R}^n$. (b) The matrix $A$ has full rank.

Since Tank and Hopfield’s pioneering work [3]-[5] on linear programming neural network and analogue circuits, the recurrent neural network approach for solving nonlinear programming has received a great of attention in the last two decades, see [6]-[13] and the references therein. Different approaches towards designing such networks have been developed. Some neural networks employed penalty functions [3]–[7], or the logarithmic barrier function [8], while others [9]–[10] make direct use of the Lagrangian function. In [11] a neural network for solving linear projection equations is described. More recently, neural networks based on gradient projection method for nonlinear programming are designed [12]-[13].

The proposed neural network does not make use of a penalty function or of a projection equation. It solves the problem directly, based on a combination of the recursive quadratic programming [14] and the continuous Newton-Raphson method [14] for solving the constraint equations.

The reminder of the paper is organized as follows. The new neural network description is presented in Section II. In Section III we prove the global convergence to the optimal point of (P). Illustrative examples are given in Section IV. Finally Section V concludes the paper.

2. DERIVATION OF THE PROPOSED NEURAL NETWORK

Let $L(x, \lambda) = f(x) + \lambda^T (A^T x - b)$ be the Lagrangian function for problem (P), where $\lambda \in \mathbb{R}^m$ is the vector of Lagrangian multipliers.

Since $f$ is strongly convex, it is well known [14] that if the optimal point of (P) $x^*$ exists, then it is unique. It is also the only point that satisfies the first order necessary conditions of optimality (Lagrangian conditions) for (P) [14], i.e.

$$\nabla L(x^*, \lambda^*) = 0 \quad \text{and} \quad A^T x^* - b = 0 \quad \text{for some} \quad \lambda^* \in \mathbb{R}^m$$

(2)

where $\nabla L(x, \lambda) = \nabla f(x) + A \lambda$.

In the first instance, we consider the following system consisting of first order partial differential equations and algebraic equations:

$$\nabla L(x(t), \lambda(t)) = e^{-\rho t} \nabla L(x_0, \lambda_0)$$

(3.1)

$$A^T x(t) - b = e^{-\rho t} (A^T x_0 - b)$$

(3.2)
where \((x(t), \lambda(t))\) the solution of system (3) with initial point \((x_o, \lambda_o) = (x(0), \lambda(0))\) and \(\rho\) positive constant. Obviously, the norms of \(\nabla L(x, \lambda)\) and the equality constraints are decreasing along the solution of system (3).

Differentiation of (3) with respect to \(t\) gives:

\[
\frac{\partial^2 L(x, \lambda)}{\partial^2 x} \dot{x} + \frac{\partial^2 L(x, \lambda)}{\partial \lambda \partial x} \dot{\lambda} = -\rho e^{-\rho t} \nabla L(x_o, \lambda_o)
\]

\[
A^T \dot{x} = -\rho e^{-\rho t} (A^T x_o - b)
\]

where \(x, \lambda\) stand for \(x(t)\) and \(\lambda(t)\) respectively. Since

\[
\frac{\partial^2 L(x, \lambda)}{\partial^2 x} = \frac{\partial^2 f(x)}{\partial^2 x} \quad \text{and} \quad \frac{\partial^2 L(x, \lambda)}{\partial \lambda \partial x} = A
\]

the above system in matrix form is written as:

\[
\begin{bmatrix}
\frac{\partial^2 f(x)}{\partial^2 x} & A \\
A^T & 0
\end{bmatrix}
\begin{bmatrix}
\dot{x} \\
\dot{\lambda}
\end{bmatrix}
= -\rho \begin{bmatrix}
\nabla L(x, \lambda) \\
A^T x - b
\end{bmatrix}
\]

(4)

The system (4) is linear with respect to the vector \([x^T, \lambda^T]\). We solve the system via QR decomposition of the matrix \(A\) [15]. Namely, \(A\) is decomposed as:

\[
A = Q \begin{bmatrix} R \\ 0 \end{bmatrix} = [Q_1 \ Q_2] \begin{bmatrix} R \\ 0 \end{bmatrix} = Q_1 R
\]

where \(Q\) is an \(n \times n\) unitary matrix, \(R\) is an \(m \times m\) upper triangular matrix. The matrices \(Q_1\) and \(Q_2\) consist of the first \(m\) and the last \(n-m\) columns of \(A\), respectively. Under the Assumption, the matrix

\[
\begin{bmatrix}
\frac{\partial^2 f(x)}{\partial^2 x} & A \\
A^T & 0
\end{bmatrix}
\]

is invertible. So the system (4) can be solved for \([x^T, \lambda^T]\) yielding:

\[
\dot{x} = -\rho \left[M(x) \nabla f(x) + N(x)(A^T x - b)\right]
\]

(5.1)

\[
\dot{\lambda} = -\rho \left[\lambda + N(x)^T \nabla f(x) + O(x)(A^T x - b)\right]
\]

(5.2)

where:

\[
M(x) = Q_2 \left(Q_2^T \frac{\partial^2 f(x)}{\partial^2 x} Q_2\right)^{-1} Q_2^T, \quad N(x) = \left(I_n - M(x) \frac{\partial^2 f(x)}{\partial^2 x}\right) Q_2 (R^T)^{-1},
\]

\[
O(x) = -R^{-1} Q_1^T \frac{\partial^2 f(x)}{\partial^2 x} N(x).
\]

In the following proposition a Lyapunov function for dynamical system (5) is given.
**Proposition:** Let the Assumption hold, then the function $V : \mathbb{R}^{n+m} \rightarrow \mathbb{R}$ be defined as:

$$V(x, \lambda) = \frac{1}{2} \left( \|\nabla L(x, \lambda)\|^2 + \|A^T x - b\|^2 \right)$$

is decreasing along the solution of (5) and approaches zero as time tends to infinity, where $\| \cdot \|$ is the Euclidean norm.

**Proof:** Finding the directional derivative of $V(x, \lambda)$ in the direction of the solution of (5) we obtain

$$\frac{dV(x, \lambda)}{dt} = \nabla_x V(x, \lambda) \dot{x} + \nabla_\lambda V(x, \lambda) \dot{\lambda} = \left[ \nabla L(x, \lambda) \right]^T \left[ \begin{array}{c} \frac{\partial^2 f(x)}{\partial x} - b \\ A^T \end{array} \right] \lambda$$

where $\nabla_x$ and $\nabla_\lambda$ denote the gradients with respect to $x$ and $\lambda$, respectively. Since the systems (4) and (5) are equivalent, from (4) we have

$$\frac{dV(x, \lambda)}{dt} = -2 \rho \left( \|\nabla L(x, \lambda)\|^2 + \|A^T x - b\|^2 \right) < 0$$

which means that

$$\frac{dV(x, \lambda)}{dt} = -\rho V(x, \lambda) \quad (6)$$

From (6) it follows that the function $V(x, \lambda)$ is decreasing exponentially along the solution of (5). This proves the assertions of the proposition.

The dynamics of the proposed neural network are defined in explicit form, by the system of differential equations (5.1). This is an autonomous dynamical system for $x(t)$, since the multipliers $\lambda(t)$ on its right hand side has been eliminated. A block diagram realization of our neural network is given in Fig.1.
3. GLOBAL CONVERGENCE

The solution of a dynamical system is said to be globally convergent to a point $x^*$ if for any initial point $x_o \in \mathbb{R}^n$, $\lim_{t \to \infty} \{x(t)\} = x^*$. This result can be derived [16] by the boundedness of the solution $\forall x_o$, and the existence of a Lyapunov function with zero derivative at $x^*$.

Theorem: Let the Assumption hold, and let $x^*$ be the unique minimize of problem (P). Then the solution of (5.1) starting from any initial point, is bounded, extends to infinite time and converges to $x^*$, i.e. $\lim_{t \to \infty} \{x(t)\} = x^* \; \forall x_o$.

Proof: The following relationships are used throughout this proof:

$$Q_1Q_1^T + Q_2Q_2^T = I_n, \; Q_1R = A, \; Q_1^TQ_2 = 0, \; Q_1^TQ_1 = I_m, \; A^Tx^* - b = 0.$$  

We shall first show that the solution $x(t)$ of (5.1) is bounded. It holds that

$$x = (Q_1Q_1^T + Q_2Q_2^T)x$$

Premultiplication (5.1) by $Q_1^T$ and after simple algebra, we get

$$\frac{d[Q_1^T(x - x^*)]}{dt} = Q_1^T\dot{x} = -\rho Q_1^T(x - x^*)$$

which is equivalent to $Q_1^T(x - x^*) = C_o e^{-\rho t}$, where $C_o = Q_1^T(x_o - x^*)$. This means that $\|Q_1^T x\|$ is bounded $\forall x_o$, i.e.

$$\|Q_1^T x\| \leq K, \text{for some finite } K$$

(8)

Similarly, premultiplication (5.1) by $Q_2^T \partial^2 f(x)/\partial x^2$, after simple algebra we get

$$\frac{d[Q_2^T \nabla f(x)]}{dt} = Q_2^T \partial^2 f(x)/\partial x^2 \dot{x} = -\rho Q_2^T \nabla f(x)$$

which is equivalent to $Q_2^T \nabla f(x) = C_1 e^{-\rho t}$, where $C_1 = Q_2^T \nabla f(x_o)$. This implies that $\|Q_2^T \nabla f(x)\|$ is bounded $\forall x_o$, i.e.

$$\|Q_2^T \nabla f(x)\| \leq L, \text{for some finite } L$$

(9)

At this point we use the strongly convexity of the objective function, so it holds that [14]

$$\exists m > 0 \text{ such that } (\nabla f(y) - \nabla f(z))^T (y - z) \geq m \|y - z\|^2, \; \forall x, y \in \mathbb{R}^n$$

Then, by choosing $y = x$ and $z = Q_1Q_1^T x$ we obtain

$$(\nabla f(x) - \nabla f(Q_1Q_1^T x))^T Q_2Q_2^T x \geq m \|Q_2^T x\|^2$$

(10)

From (8) and (9) and by using norm properties we get

$$\|\nabla f(x) - \nabla f(Q_1Q_1^T x)^T Q_2Q_2^T x \leq (\|Q_2^T \nabla f(x)\| + \|\nabla f(Q_1Q_1^T x)\|) \|Q_2^T x\|$$
where \( M = L + \max \{\nabla f(Q_1^T x)\} \). Notice that \( f \) is twice differentiable and \( \|Q_1^T x\| \) is bounded, hence Weierstrass’ theorem [14] yields that the quantity \( \max \{\nabla f(Q_1^T x)\} \) exists and is finite.

Then, by (10) and (11), we get \( \|Q_2^T x\| \leq M/m \), which means that \( \|Q_2^T x\| \) is also bounded. From this result and (7) and (8) we deduce that the solution of (5.1) is bounded, hence it extends to infinite time [16]. Since \( x(t) \) is bounded, it can be proved easily from (5.2) that \( \lambda(t) \) is also bounded. Let the set \( D \) be defined as:

\[
D = \left\{ (x, \lambda) \in \mathbb{R}^{n+m} : \frac{dV(x, \lambda)}{dt} = 0 \right\}
\]

where \( V(x, \lambda) \) is the function of Proposition. Then from (6) we have \( D = \{(x, \lambda) \in \mathbb{R}^{n+m} : \|\nabla L(x, \lambda)\| = 0 \text{ and } A^T x - b\| = 0\} \), and from (2) it yields \( D = \{x^*, \lambda^*\} \). Since \( (x(t), \lambda(t)) \) is bounded and satisfies Proposition, from LaSalle’s Theorem [16] it follows that \( (x(t), \lambda(t)) \rightarrow D = \{x^*, \lambda^*\} \) as \( t \rightarrow \infty \). This competes the proof.

4. NUMERICAL EXAMPLES

The performance of our neural network is evaluated by using MATLAB for several test problems. In this section two illustrative examples are given. Example 1 has quadratic objective function and satisfies both parts of Assumption. To demonstrate the effectiveness of our neural network in more general optimization problems, we choose Example 2, whose objective function is a Gaussian as shown in Fig. 2, that is pseudoconvex. So, Example 2 satisfies only the part (b) of Assumption.

![Figure 2. The 2D Gaussian function of Example 2.](image)
Example 1: Consider the following strongly convex problem [6], with \( n = 6 \) and \( m = 3 \):

\[
\min_{x \in \mathbb{R}^n} \{ \|x\|^2 : A^T x - b = 0 \}
\]

where \( b^T = [2 \ -1 \ -4] \) and \( A^T = \begin{bmatrix} 2 & -1 & 4 & 0 & 3 & 3 \\ 5 & 1 & -3 & 1 & 2 & 0 \\ 1 & -2 & 1 & -5 & -1 & 4 \end{bmatrix} \).

This problem has a unique global minimizer at \( x^* = [0.08824674 \ 0.010828343 \ 0.27326648 \ 0.50466163 \ 0.38281032 \ -0.30970696] \), written to eighth decimal place.

The trajectories \( x(t) \) obtained by the proposed neural network with \( \rho = 10 \), starting from five random non-feasible initial points in (-1 1), are shown in Fig.3a. Fig.3b shows the convergence of the cost function for each case. At the end of the simulation, all trajectories reach \( x^* \) with final error \( e = \|x(t) - x^*\| \) of order \( 10^{-6} \).

Figure 3. Example 1: (a) Trajectories \( x(t) \) of the proposed neural network for 5 random initial points and (b) the corresponding cost functions, vs time.
**Example 2:** Consider the following pseudoconvex optimization problem \([9]\), with \(n = 2\) and \(m = 1\):

\[
\min_{x \in \mathbb{R}^2} \left\{ e^{-(x_1^2 + x_2^2)} : A^T x - b = 0 \right\}
\]

where \(A = \begin{bmatrix} 0.787 \\ 0.586 \end{bmatrix}\) and \(b = 0.823\).

This problem has a unique global minimizer at \(x^* = [0.62745172 \ 0.500937796]\), written to eighth decimal place. Fig. 4a shows the trajectories of the proposed neural network with \(\rho = 10\), starting from fifteen non-feasible initial points, random generated from the uniform distribution over \((0,1)\). Fig. 4b shows the convergence of the cost function for each case. At the end of the simulation, all trajectories reach \(x^*\) with final error \(e = \|x(t) - x^*\|\) of order \(10^{-6}\).

![Figure 4](image-url)

**Figure 4.** Example 2: (a) Trajectories \(x(t)\) of the proposed neural network for 15 random initial points and (b) the corresponding cost functions, vs time.
5. CONCLUSIONS

In this paper a recurrent neural network for strongly convex constrained optimization problem is presented, based on quadratic approximation method for nonlinear programming. If initial point is non-feasible, the proposed neural network defines a non-feasible trajectory which satisfies the constraints as $t \to \infty$. Global convergence is proven. Simulation on illustrative numerical examples substantiates the effectiveness and the correctness of the proposed neural network.

REFERENCES


