Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

On a Stabilization Problem of Nonlinear Programming Neural Networks

Yuancan Huang *

* *Intelligent Robot Institute, Beijing Institute of Technology, Beijing, 100081 China, (Tel: +86-10-68913111; e-mail: yuancanhuang@bit.edu.cn)*

Chuang Yu **

** *Department of Information, Shenyang Ligong University, Shenyang, 110168, China, (Tel: +86-24-24682012; e-mail: yuchuang2002@163.com)*

Lingyun Zhu ***

*** *Intelligent Robot Institute, Beijing Institute of Technology, Beijing, 100081 China, (Tel: +86-10-68913111; e-mail: wtzly11@gmail.com)*

Abstract: Intrinsically, Lagrange multipliers in Nonlinear Programming Theory play a regulating role in the process of searching the optima of constrained optimization problems. Hence, they may be regarded as control input variables as those in control systems. From this new perspective, it is showed that synthesizing nonlinear programming neural networks can be formulated to solve servomechanism problems. In this paper, under the second-order sufficient assumptions of nonlinear programming problems, a dynamic output feedback control law is proposed to stabilize the corresponding nonlinear programming neural networks. Moreover, their asymptotical stability is proved by the Lyapunov First Approximation Principle.

1. PROBLEM FORMULATION

In this paper, we study the general nonlinear programming problem with mixed constraints:

$$\begin{aligned} &minimize \quad f(x) \\ &subject\ to\ \ h(x) = 0, g(x) \leq 0, \end{aligned} \tag{1}$$

where $f(x) : R^n \rightarrow R$, $h(x) : R^n \rightarrow R^m$, and $g(x) : R^n \rightarrow R^r$ are scalar objective function and vector equality and inequality constraint functions, respectively. Its feasible set is defined as $X = \{x : h(x) = 0, g(x) \leq 0, x \in R^n\}$.

Definition 1. Let x^* be a vector satisfying the constraint conditions, then $J(x^*)$ denotes a set of index j for which $g_j(x^*) = 0$, namely

$$J(x^*) = \{j \mid g_j(x^*) = 0, j = 1, 2, \cdots, r\}. \tag{2}$$

If the gradients $\nabla h_i(x^*), i = 1, 2, \cdots, m, \nabla g_j(x^*), j \in J(x^*)$ are linearly independent, then x^* is called regular point.

By introducing the vector of slack variables $z = col(z_1, z_2, \cdots, z_r)$ and the functions

$$\begin{aligned} &\bar{f}(x, z) = f(x) \\ &\bar{h}_i(x, z) = h_i(x), \quad i = 1, 2, \cdots, m \\ &\bar{g}_j(x, z) = g_j(x) + z_j^2, \quad j = 1, 2, \cdots, r. \end{aligned} \tag{3}$$

The problem (1) may be rewritten as

$$\begin{aligned} &minimize \quad \bar{f}(x, z) \\ &subject\ to\ \ \bar{h}(x, z) = 0, \bar{g}(x, z) = 0. \end{aligned} \tag{4}$$

It is clear that the two problems (1) and (4) are equivalent in the sense that x^* be a local minimum for the original problem (1) if and only if (x^*, z^*),$z_j^* = [-g_j(x^*)]^{-1/2}$, $j = 1, 2, \cdots, r$, is a local minimum for the problem (4).

The Lagrangian function of the problem (1) is defined as

$$\bar{L}(x, z, \lambda, \mu) = \bar{f}(x, z) + \sum_{i=1}^{m} \lambda_i \bar{h}_i(x, z) + \sum_{j=1}^{r} \mu_j \bar{g}_j(x, z), \tag{5}$$

where λ and μ are multiplier vectors.

After a direct calculation, it is derived that a variant of Karush-Kuhn-Tucker Theorem furnishes the necessary conditions for some x^* being a local minimum of the problem (1) in this case [7, 6, 8, 10].

Theorem 2. Let x^* be a local minimum of the problem (1) and assume that x^* is a regular point. Then there exists the unique vectors λ^* and μ^* such that

$$\begin{aligned} &\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) + \sum_{j=1}^{r} \mu_j^* \nabla g_j(x^*) = 0 \\ &2\mu_j^* z_j^* = 0, \quad j = 1, 2, \cdots, r \\ &\bar{h}(x^*, z^*) = 0, \bar{g}(x^*, z^*) = 0. \end{aligned} \tag{6}$$

The fundamental idea using neural networks to solve nonlinear programming problems is to construct a continuous-time dynamical system whose equilibria coincide with the KKT pair

★ This work was supported by the National 863 Project under Grant 20060104Z2056.

(x^*, λ^*, μ^*) in Karush-Kuhn-Tucker Theorem[1]-[5]. Obviously, the constructed continuous-time dynamical system must be asymptotically stable in order to settle down to its equilibria ultimately. Observe that, if the multipliers λ_i's and μ_j's in Lagrangian function are taken as the control variables u_i's and v_j's in the nonlinear programming neural networks, respectively, then the stabilizability of the nonlinear programming neural networks will be boiled down to the solvability of the following servomechanism problem without disturbances:

$$\dot{x} = -\nabla f(x) - \sum_{i=1}^{m} u_i \nabla h_i(x) - \sum_{j=1}^{r} v_j \nabla g_j(x) \tag{7}$$
$$\dot{z} = -\begin{pmatrix} 2v_1 z_1 \\ \vdots \\ 2v_r z_r \end{pmatrix}$$
$$e_1 = h(x)$$
$$e_2 = \begin{pmatrix} g_1(x) + z_1^2 \\ \vdots \\ g_r(x) + z_r^2 \end{pmatrix}.$$

where x represents the n-dimensional state vector, z the r-dimensional state vector, u the m-dimensional input vector, v the m-dimensional input vector, and e the $(m + r)$-dimensional output vector representing the tracking errors. Thus, we have bonded the construction of the stable nonlinear programming neural networks to the well-developed nonlinear servomechanism theory.

The rest of the paper is organized as follows. Section 2 aims to introduce a general theory for this servomechanism. A construction of the stable nonlinear programming neural networks is given in Section 3. Then, an illustrative example is given in Section 4. Finally, in Section 5 we conclude this paper with some remarks.

2. A GENERAL THEORY FOR SERVOMECHANISM PROBLEM

Consider the general nonlinear system described by

$$\dot{x} = b(x, u) \tag{8}$$
$$e = c(x, u),$$

where $x(t)$ is the n-dimensional state vector, $u(t)$ the m-dimensional input vector, and $e(t)$ the m-dimensional output vector representing the tracking errors.

The class of control laws considered here are described by

$$u = \gamma(x, \xi, e) \tag{9}$$
$$\dot{\xi} = \eta(x, \xi, e)\ \xi(0) = \xi_0,$$

where ξ is the compensator state vector of dimension ν. The aforementioned controller is referred to as a dynamic state controller. The closed-loop system can be written as

$$\dot{x} = b(x, \gamma(x, \xi, e))\ x(0) = x_0 \tag{10}$$
$$\dot{\xi} = \eta(x, \xi, e)\ \xi(0) = \xi_0$$
$$e = c(x, u).$$

All the functions involved in this setup are assumed to be sufficiently smooth and defined globally on the appropriate Euclidean spaces, with the value zero at the equilibrium $x_e^* = col(x^*, \xi^*)$ of the system (10), that is, $b(x^*, \xi^*, 0) = 0$, $\eta(x^*, \xi^*, 0) = 0$ and $c(x^*, \xi^*) = 0$. The servomechanism problem was intensively studied in the literature[11]-[15].

Definition 3. Local Simplified Servomechanism Problem(LSSP): Find a controller of the form (9) such that the closed-loop system (10) satisfies the following two properties.

- P1: The eigenvalues of its linearized part at its equilibria have negative real parts.
- P2: For x_e belong to sufficiently small neighborhood of the equilibrium x_e^*, $\lim_{t\to\infty} e(t) = 0$.

Note that the system (9) may be written in the form

$$\dot{x} = Ax + Bu + \phi(x, u) \tag{11}$$
$$e = Cx + Du + \psi(x, u),$$

where $\phi(x, u)$ and $\psi(x, u)$ vanish at the equilibrium with their first order derivatives, and A, B, C, D are matrices defined by

$$A = \frac{\partial b}{\partial x}(x^*, u^*)\ \ B = \frac{\partial b}{\partial u}(x^*, u^*)$$
$$C = \frac{\partial h}{\partial x}(x^*, u^*)\ \ D = \frac{\partial h}{\partial u}(x^*, u^*). \tag{12}$$

To give an account of the solvability condition for the LSSP, let us first state two standard assumption as follows.

- A1: The pair (A, B) is stabilizable.
- A2: The pair (A, C) is detectable.

From [16]-[17], there is

Theorem 4. If Assumptions A1 and A2 are hold, then the LSSP is solvable by the linear dynamic output feedback control law, namely,

$$u = K\xi \tag{13}$$
$$\dot{\xi} = L\xi + Qe.$$

3. STABILIZATION OF NONLINEAR PROGRAMMING NEURAL NETWORKS

Unfortunately, although we have given the dynamic output feedback control law (13) for solving the LSSP, it is useless for stabilizing nonlinear programming neural networks since the equilibria of nonlinear programming neural networks cannot be known *a priori*. Therefore, a new output feedback control law which is independent of these equilibria must be sought to stabilize nonlinear programming neural networks. One of such candidate dynamic output feedback control law for stabilizing the system (7) is written as

$$u = \lambda + ce_1, v = \mu + ce_2 \tag{14}$$
$$\dot{\lambda} = e_1, \dot{\mu} = e_2,$$

where c is a positive parameter. Hence, the overall closed-loop system is given

$$\dot{x} = -\nabla f(x) - \sum_{i=1}^{m}(\lambda_i + ce_1^i)\nabla e_1^i - \sum_{j=1}^{r}(\mu_j + ce_2^j)\nabla e_2^j \tag{15}$$
$$\dot{z} = -\begin{pmatrix} 2\mu_1 z_1 \\ \vdots \\ 2\mu_r z_r \end{pmatrix}$$

$$\dot{\lambda} = e_1, \dot{\mu} = e_2$$
$$e_1 = h(x)$$
$$e_2 = \begin{pmatrix} g_1(x) + z_1^2 \\ \vdots \\ g_r(x) + z_r^2 \end{pmatrix}.$$

One more assumption is appended in order to show the stability of the closed-loop system:

A3: Let x^* be regular point for the nonlinear programming problem (1). If there exists vectors λ^* and μ^* satisfying

$$\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) + \sum_{j=1}^{r} \mu_j^* \nabla g_j(x^*) = 0 \quad (16)$$
$$h(x^*) = 0, g(x^*) \leq 0$$
$$\mu^* \geq 0, \mu_j^* g_j(x^*) = 0, j = 1, 2, \cdots, r,$$

as well as the strict complementarity condition

$$\mu_j^* > 0, \forall j \in J(x^*). \quad (17)$$

Assume that for every $y \neq 0$ such that $\nabla h_i(x^*)^T y = 0, i = 1, 2, \cdots, m$, and $\nabla g_j(x^*)^T y = 0$ for every $j \in J(x^*)$, it follows that

$$y^T [\nabla^2 f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 h_i(x^*) + \sum_{j=1}^{r} \mu_j^* \nabla^2 g_j(x^*)] y > 0. \quad (18)$$

Assumption A3 is referred generally as the second-order sufficient condition for x^* being a strict local minimum of the nonlinear programming problem (1).

Let

$$\nabla_{xx}^2 L(x, \lambda, \mu) = \nabla^2 f(x) + \sum_{i=1}^{m} \lambda_i \nabla^2 h_i(x) + \sum_{j=1}^{r} \mu_j \nabla^2 g_j(x).$$

Before the stability result is stated, we introduce a lemma[7, 9].

Lemma 5. Let P be a symmetric $n \times n$ matrix and Q a positive semidefinite symmetric $n \times n$ matrix. Assume that $y^T P y > 0$ for every $y \neq 0$ satisfying $y^T Q y = 0$, then there exists a scalar $c > 0$ such that

$$P + cQ > 0. \quad (19)$$

From Lemma (5), we easily show that, if Assumption A3 hold, there exists a $\bar{c} > 0$ such that

$$\nabla_{xx}^2 L(x^*, \lambda^*, \mu^*) \quad (20)$$
$$+c \left[\sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) \nabla h_i(x^*)^T + \sum_{j=1}^{r} \mu_j^* \nabla g_j(x^*) \nabla g_j(x^*)^T \right] > 0$$

for any $c > \bar{c}$.

For simplicity of notation, let

$$Q = \nabla_{xx}^2 L(x^*, \lambda^*, \mu^*) \quad (21)$$
$$+c \left[\sum_{i=1}^{m} \lambda_i^* \nabla h_i(x^*) \nabla h_i(x^*)^T + \sum_{j=1}^{r} \mu_j^* \nabla g_j(x^*) \nabla g_j(x^*)^T \right].$$

Proposition 6. Assume that Assumption A3 is satisfied. Then the servomechanism problem (7) is solved by the dynamic output feedback control law (14).

proof: We linearize the closed-loop system at its equilibrium point (x^*, λ^*, μ^*). Taking the KKT conditions into account, the linearized part is given as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{\lambda} \\ \dot{\mu} \end{bmatrix} = H \begin{bmatrix} x - x^* \\ z - z^* \\ \lambda - \lambda^* \\ \mu - \mu^* \end{bmatrix} \quad (22)$$

where

$$H = \begin{bmatrix} G & F \\ -F^T & \end{bmatrix}.$$

with

$$G = \begin{bmatrix} -Q & & & \\ & -2\mu_1^* & & \\ & & \ddots & \\ & & & -2\mu_r^* \end{bmatrix}$$

$$F = \begin{bmatrix} \nabla h(x^*) & \nabla g(x^*) & & \\ & 2z_1^* & & \\ & & \ddots & \\ & & & 2z_r^* \end{bmatrix}.$$

Now we shall show that the real part of every eigenvalue of H is negative.

For any complex vector v, denotes by v^H its complex conjugate transpose, and for any complex number α, denotes by $\Re(\alpha)$ its real part. Let β be an eigenvalue of H, and nonzero vector $p = col(z, w)$ be a corresponding eigenvector. We have

$$\Re(p^H H p) = \Re(\beta)(|z|^2 + |w|^2). \quad (23)$$

Expanding the left-hand side of the above equation, we obtain

$$\Re(p^H H p) = \Re\{z^H G z + z^H F w - w^H F^T z\}. \quad (24)$$

Since there is $\Re(z^H F w) = \Re(w^H F^T z)$, it follows from Eqs. (23) and (24) that

$$\Re(\beta)(|z|^2 + |w|^2) = \Re(z^H G z) \leq 0. \quad (25)$$

Then we derive that either $\Re(\beta) < 0$ or $z = 0$. However, if $z = 0$, the following equation

$$H \begin{bmatrix} z \\ w \end{bmatrix} = \beta \begin{bmatrix} z \\ w \end{bmatrix} \quad (26)$$

yields

$$Fw = 0. \quad (27)$$

If w is partitioned into two parts $w = col(w_1, w_2)$ with appropriate dimensions, Eq. (27) may be rewritten

$$\nabla h(x^*) w_1 + \nabla g(x^*) w_2 = 0 \quad (28)$$
$$\begin{bmatrix} 2z_1^* & & \\ & \ddots & \\ & & 2z_r^* \end{bmatrix} w_2 = 0.$$

From the second part of the last equation, we derive that $w_2^j = 0, j \ni J(x^*)$. Since $\nabla h_i(x^*), i = 1, 2, \cdots, m, \nabla g_j(x^*), j \in J(x^*)$ has full row rank, it follows that $w_1 = 0$ and $w_2^j =$

$0, j \in J(x^*)$. Hence, we have that $w = 0$. This contradicts our earlier assumption that p is a nonzero vector. Consequently, we must have $\Re(\beta) < 0$. Thus (x^*, λ^*, μ^*) is the asymptotically exponentially stable point of the closed-loop system (15), that is, the servomechanism problem (7) is solvable.

4. AN ILLUSTRATIVE EXAMPLE

An example from Matlab is taken showing how to construct nonlinear programming neural networks with the proposed approach.

$$\begin{aligned} &\textit{minimize} \quad f(x) = \exp(x_1)(4x_1^2 + 2x_2^2 + 4x_1x_2 + 2x_2 + 1) \\ &\textit{subject to} \quad x_1 + x_2 - x_1x_2 \geq 1.5 \\ &\qquad\qquad\quad x_1x_2 \geq -10 \end{aligned}$$

where its optimal solution locates at the point $(-9.5474, 1.0474)$ and both of constraints are active.

The corresponding differential equations are

$$\begin{aligned} \dot{x}_1 &= -\frac{\partial f}{\partial x_1} - (\mu_1 + ce_1)\frac{\partial g_1}{\partial x_1} - (\mu_2 + ce_2)\frac{\partial g_2}{\partial x_1} \\ \dot{x}_2 &= -\frac{\partial f}{\partial x_2} - (\mu_1 + ce_1)\frac{\partial g_1}{\partial x_2} - (\mu_2 + ce_2)\frac{\partial g_2}{\partial x_2} \\ \dot{z}_1 &= -2\mu_1 z_1, \dot{\mu}_1 = e_1, e_1 = g_1 + z_1^2 \\ \dot{z}_2 &= -2\mu_2 z_2, \dot{\mu}_2 = e_2, e_2 = g_2 + z_2^2, \end{aligned}$$

Figure 1 shows that the trajectories of state variables in the nonlinear programming neural network tend ultimately to the optimal solutions.

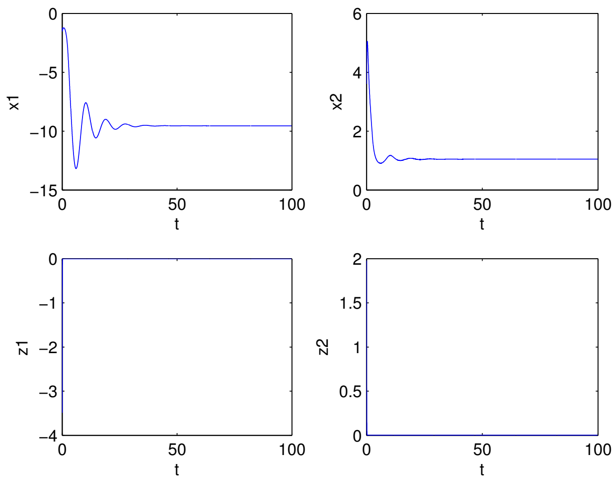

Fig. 1. Trajectories of x_1, x_2, z_1, z_2.

5. CONCLUSIONS AND DISCUSSIONS

Reconsidering Lagrange multiplier in Nonlinear Programming Theory as control input variables in control systems, a new approach to study the stabilization of nonlinear programming neural networks is proposed in this paper. We conclude that the stabilizability of nonlinear programming neural networks is boiled down to the solvability of servomechanism problems without disturbances. Under second-order sufficient assumption, a dynamic output feedback control law with linear form is used to stabilize nonlinear programming neural networks, and, by the Lyapunov First Approximation Principle, the neural networks are shown to be locally asymptotically stable. Finally, an illustrative example is given to show the feasibility of the proposed approach.

REFERENCES

[1] Y.C. Huang. A Novel Method to Handle Inequality Constraints for Convex Programming Neural Network. *Neural Processing Letters*, 16:17–27, 2002.

[2] Y.C. Huang. Lagrange-Type Neural Networks for Nonlinear Programming Problems with Inequality Constraints. *Proceeding of the 44th Conference on Decision and Control*, 2005.

[3] M.P. Kennedy, and L.O. Chuo. Neural networks for nonlinear programming. *IEEE Trans. on Circuits and Systems*, 35:554–562, 1988.

[4] A. Cichocki, and R. Unbehauen. *Neural Networks for Optimization and Singal Processing*. Chichester, Wiley, 1993.

[5] S. Zhang, and A.G. Constantinides. Lagrange Programming Neural Networks. *IEEE Trans. on Circuits and Systems*, 35:554–562, 1988.

[6] D.P. Bertsekas. *Nonlinear Programming*. 2nd edn. Massachusetts Athena Scientific, Belmont, 1999.

[7] D.P. Bertsekas. *Constrainted Optimization and Lagrange Methods*. Academic Press, New York, 1982.

[8] M. Avriel. *Nonlinear Programming Analysis and Methods*. Englewood Cliffs, New Jersey, 1976.

[9] J. Nocedal, and S. J. Wright. *Numerical Optimization*. Springer-Verlag, New York:, 1999.

[10] R. T. Rockafellar. Lagrange Multiplier and Optimality. *SIAM Review*, 35:183–238, 1993.

[11] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, New York, 1995.

[12] A. Isidori, and C.I. Byrnes. Output Regulation of Nonlinear Systems. *IEEE Trans. on Autom. Control*, 35:880–885, 1990.

[13] J. Huang, and Z.Y. Chen. A General Framework for Tackling the Output Regulation Problem. *IEEE Trans. on Autom. Control*, 49:2203–2218, 2004.

[14] C.I. Byrnes, and A. Isidori. Limit Sets, Zero Dynamics, and Internal Models in the Problem of Nonlinear Output Regulation. *IEEE Trans. on Autom. Control*, 4:1712–1723, 2003.

[15] C.I. Byrnes, and A. Isidori. Output Regulation for Nonlinear Systems: An Overview. *Int. J. Robust Nonlinear Control*, 10:323–337, 2000.

[16] E.J. Davison. The Robust Control of a Servomechanism Problem for Linear Time-Invariant Multivariable Systems. *IEEE Trans. on Autom. Control*, 21:25–34, 1976.

[17] B.A. Francis. The Linear Multivariable Regulator Problem. *SIAM J. Control Optimiz.*, 14:486–505, 1977.

[18] B.A. Francis, and W.M. Wonham. The Internal Model Principle of Control Theory. *Automatica*, 12:457–465, 1976.