선형계획 문제의 해를 구하는 신경회로

*장석호, *강성귀, **남부희, **이정문
*강원대학교 전자공학과, **강원대학교 제어계측공학과

Neural Networks for Solving Linear Programming Problems and Linear Systems

*S. H. Chang, *S. G. Kang, **J. M. Lee and **B. H. Nam
*Dept. of Electronics Eng., **Dept. of Control and Instrumentation Eng.
Kangwon National University

ABSTRACT

The Hopfield model is defined as an adaptive dynamic system. In this paper we propose a modified neural network which is capable of solving linear programming problems and a set of linear equations. The model is directly implemented from the given system, and solves the problem without calculating the inverse of the matrices. We get the better stability results by the addition of scaling property and by using the nonlinearities in the linear programming neural networks.

I. 서론

Tank 와 Hopfield[1]는 처음으로 선형계획 문제의 해를 구하는 신경회로를 제안하였다. 후에 그들의 모델은 Kennedy 와 Chua[2, 3]에 의하여 수정되었으며, Kennedy 와 Chua는 해의 정확도가 페널티 파라미터에 의해서 조정될 수 있음을 보였다. 그들의 결과는 Maa 와 Shanblatt[5]에 의해 최적화 이론의 견지에서 정당화 되었으며 선형시스템문제와 자승계획문제와 같은 다른 최적화 문제에도 적용되었다. Aourid et al.[4]는 실제로 선형계획 신경회로를 구현하는 데에 있어 해가 안정점 근처에서 진동하는 것을 제어하기 위해 비선형성을 조절하는 선형함수를 제안했다.

본 논문에서 제안한 모델은 기존의 페널티 함수의 동작 영역을 확장하여 표준형 LP(*Linear Programming*)문제를 해결할 수 있으며 선형 시스템 문제도 직접 입력하여 해를 구할 수 있다.

II. 기존의 선형계획 신경회로

The Tank 와 Hopfield 모델은 다음과 같은 정준형 선형계획 문제의 해를 구하도록 고안되었다.

$$\text{Min} \sum_{i=1}^{n} a_i v_i$$
subject to
$$Dv - b \geq 0$$
$$v_i \geq 0, \qquad i = 1, \dots, n$$

여기서 D 는 m x n 행렬이고, $v \in R^n$, $b \in R^m$, $a^T = [a_1 a_2 \cdots a_n]$, $D^T = [d_1 d_2 \cdots d_m]$, $d_j^T = [d_{j1}\ d_{j2} \cdots d_{jn}]$이다.

Tank 와 Hopfield 모델의 회로 방정식은 다음과 같이 쓸 수 있다.

$$C\frac{du_i}{dt} = -a_i - \frac{u_i}{R} - \sum_{j=1}^{m} d_{ji} g(e_j(v)),$$
$$\text{for } i = 1, \cdots, n \tag{1}$$

변수 증폭기는 아래와 같은 관계는

$$v_i = h(u_i) = k\,u_i\,, \qquad i = 1, \dots, n \tag{2}$$

만약 제약조건의 만족도를 정의하면

$$e_j(v) = d_j^T v - b_j \qquad j = 1, \dots, m$$
$$= \sum_{i=1}^{n} d_{ji} v_i - b_j, \tag{3}$$

페널티 증폭기 g_1은 (4)식과 같은 입출력 관계 특징이 있다.

$$g_1(e_j(v)) = \begin{cases} 0 & \text{if } e_j(v) > 0 \\ -e_j(v) & \text{if } e_j(v) \leq 0\,. \end{cases} \tag{4}$$

그러나 함수 g_1은 Kennedy와 Chua에 의해 (5)식과 같이 g_2로 수정된다.

$$g_2(e_j(v)) = \begin{cases} 0 & \text{if } e_j(v) > 0 \\ e_j(v) & \text{if } e_j(v) \leq 0 \end{cases} \tag{5}$$

만족하지 않는 제약조건들에 대해 Tank와 Hopfield의 비선형성은 Kennedy와 Chua의 비선형과 반대 방향으로 페널티를 가한다.

Aourid et al.는 신경회로의 안정점과 문제의 해사이의 차이를 줄이기 위해 선형함수 L(•)를 제안했다. 즉, 안정점근처의 진동을 제어하기 위한것이다.

그들의 선형계획 신경회로는 다음형태를 갖는다.

$$C_i\frac{du_i}{dt} = -\frac{u_i}{R_i} - L\Big(a_i + \sum_{j=1}^{m} d_{ji} g_j(e_j(v))\Big)$$
$$\text{with } L(x) = \frac{x}{l}, \quad l > 1 \quad \text{for } i = 1, \cdots, n \tag{6}$$

선형함수의 부가는 보다 나은 안정성을 낳고, l이 클수록 신경회로는 더욱 안정하게 된다.

Cichocki와 Unbehauen[6]는 선형계획 신경회로는 표준형의 LP문제를 대상으로 한다.

$$\text{minimize } f(x) = a^T v$$

subject to

$$e_j(v) = \sum_{i=1}^{n} d_{ji} v_i - b_j,\ v_i \geq 0$$

for $i = 1,2, \cdots, n$ and $j = 1,2, \cdots, m$.

변형된 Lagrange multiplier 방법에 의해 에너지함수는

$$E(v) = \sum_{i=1}^{n} a_i v_i + \sum_{j=1}^{m} \lambda_j e_j + \frac{1}{2}\alpha \sum_{j=1}^{m} \lambda_j^2 \tag{7}$$

여기서 α는 is 규칙(regularization) 매개변수이다. 신경회로의 동특성 방정식은

$$\frac{dv_i}{dt} = -\mu_i (c_i + \sum_{j=1}^{m} d_{ji}\lambda_j),\ v_i(0) = v_i^{(0)},\ v_i \geq 0$$

$$\frac{d\lambda_j}{dt} = -\rho_j (\sum_{i=1}^{m} d_{ji} v_i - b_j - \alpha\lambda_j)),\ \lambda_j(0) = \lambda_j^{(0)} \tag{8}$$

여기서

$$\mu_i = \frac{1}{\tau_{1i}} > 0,\ \rho_j = \frac{1}{\tau_{2j}} > 0$$

$\tau_{1i}$and τ_{2j}는 적분기의 적분 시정수이다.

III. 제안된 선형계획 신경회로

우리는 다음과 같은 LP 문제를 고려한다.

등식의 제약조건을 갖는 스칼라 가격함수 $\phi(x)$를 최소화한다.

$$\text{minimize } \phi(x) = a^T v$$

subject to

$$Dv = b$$

여기서

$$a^T = [a_1\ a_2\ \cdots\ a_n]$$
$$v^T = [v_1\ v_2\ \cdots\ v_n]$$
$$b^T = [b_1\ b_2\ \cdots\ b_m]$$

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{mi} & \cdots & d_{mn} \end{bmatrix} = \begin{bmatrix} d_1^T \\ d_2^T \\ \vdots \\ d_m^T \end{bmatrix}$$

이제, 우리는 표준형 선형계획문제를 직접 풀 수 있는 신경회로를 제안한다.

동특성 방정식은

$$C\frac{du_i}{dt} = -\frac{u_i}{R} - L(\ a_i\ ) - \sum_{j=1}^{m} d_{ji} g(e(v)) + f(v_i)$$

for $i = 1, \cdots, n$ (9)

여기서

$$L(x) = \frac{x}{l},\quad l > 1 \tag{10}$$

(7)식에서 정의된 페널티 함수 g 와 f는 각각 다음과 같이 정의된다.

$$g(e_j(v)) = g(d_j^T v - b_j)$$
$$= \begin{cases} K_1(d_j^T - b_j) & \text{if } (d_j^T - b_j) \leq 0 \\ K_2(d_j^T - b_j) & \text{if } (d_j^T - b_j) > 0 \end{cases} \tag{11}$$

그리고

$$f(v_i) = \begin{cases} K_3 v_i & \text{if } v_i < 0 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

함수 f는 변수의 비음수 제약조건을 보장하기 위해 1 사분면에서 동작하는 페널티함수이다. 여기서 다음과 같이 가정한다.

$$K_1 = K_2 = K_3 = K > 0\ .$$

이제 전체 에너지함수의 형태는

$$E = \frac{1}{l}\sum_{i=1}^{n} a_i v_i + \sum_{i=1}^{n}\frac{1}{R}\int_0^{v_i} h^{-1}(v)\, dv + \sum_{j=1}^{m}\int_0^{e_j(v)} g(x)\, dx + \sum_{i=1}^{n}\int_0^{v_i} f(x)\, dx \tag{13}$$

여기서 $h^{-1}(v)$가 점근증가 함수이므로 둘째 항은 비음수이고, 네째항 역시 수동성 때문에 비음수이다($xg(x) \geq 0$, $xf(x) \geq 0$).

안정도 해석을 위해, 시간에 대하여 1차미분하면

$$\frac{dE}{dt} = \frac{1}{l}\sum_{i=1}^{n} a_i \frac{dv_i}{dt} + \sum_{i=1}^{n}\frac{h^{-1}(v_i)}{R}\frac{dv_i}{dt}$$
$$+ \sum_{i=1}^{n}(\sum_{j=1}^{m} d_{ji} g(e(v)) + f(v_i))\frac{dv_i}{dt}$$
$$= \sum_{i=1}^{n}\left(\frac{a_i}{l} + \frac{u_i}{R} + \sum_{j=1}^{m} d_{ji} g(e_j(v)) + f(v_i)\right)\frac{dv_i}{dt}$$
$$= -\sum_{i=1}^{n} C\frac{dh^{-1}(v_i)}{dv_i}\left(\frac{dv_i}{dt}\right)^2$$

여기서

$$\left(\frac{dv_i}{dt}\right)^2 \geq 0,\quad \frac{h^{-1}(v_i)}{dv_i} > 0 \text{ for } v_i \neq 0.$$

$\frac{dE}{dt} \leq 0$. 그러므로 제안된 선형계획 신경회로는 완전안정하다.

IV. 모의실험 결과

모의실험은 IBMPC486상에서 386MATLAB에 의해 프로그램되었다. 선형계획 신경회로의 회로 방정식을 적분하기 위해 4차 룬게-쿠타방법이 사용되었다.

샘플링 시간은 10^{-6}[sec]이고, 모든 증폭기는 매단계마다 동기화 된다고 가정하였다.한다. 입력저항과 케패시턴스는 각각 $R = 10^5$ ohms, $C = 10^{-4}$ farads이다.

A. 선형 계획 문제

제안된 신경회로는 등식의 제약조건을 가진 LP문제의 해를 구할 수 있다. 또한 부등식의 제약조건을 가진 선형계획문제도 슬랙변수를 첨가한다면 표준형의 LP문제로 변형되므로 해를 구할 수 있다.

다음의 LP 문제를 고려하자.

$$\text{Minimize } f(x) = -2x_1 - x_2$$

subject to

$$\begin{aligned} x_1 &\geq 0 \\ x_2 &\geq 0 \\ -x_1 + 2x_2 &\leq 2 \\ x_1 + x_2 &\leq 4 \\ x_1 &\leq 3 \end{aligned}$$

이 예제는 슬랙 변수의 첨가로 표문형 LP문제가 된다.

$$\text{Minimize } f(x) = -2x_1 - x_2$$
subject to
$$\begin{aligned} -x_1 + 2x_2 + x_3 &= 2 \\ x_1 + x_2 + x_4 &= 4 \\ x_1 + x_5 &= 3 \\ x_i \geq 0 \quad i &= 1,\cdots,5 \end{aligned}$$

여기서 x_3, x_4, x_5는 슬랙 변수이다. 신경회로로 부터 얻어진 값은

$$x^T = [\ 3.0033\ 1.0001\ 3.0030\ -0.0017\ -0.0016]$$

B. 선형 시스템 문제

제안된 선형계획 신경회로에서 a = 0으로 하고 변수의 비음수이 없다면 Dv = b와 같은 제약조건만을 가진 선형계획 문제 즉, 선형시스템 문제가 된다. 동특성 방정식은

$$C\frac{du_i}{dt} = -\frac{u_i}{R} - \sum_{j=1}^{m} d_{ji} g(e(v)) \qquad (14)$$

이 때 에너지 함수는

$$E = \sum_{i=1}^{n} \frac{1}{R} \int_0^{v_i} h^{-1}(v)\, dv + \sum_{j=1}^{m} \int_0^{e_j(v)} g(x)\, dx \qquad (15)$$

E를 시간에 대해 일차 미분하면 $\frac{dE}{dt} \leq 0$. 그러므로 역시 선형 시스템 문제의 해를 구하는 신경회로는 완정하다.

우리는 D와 v의 계수를 직접 제안된 선형계획 신경회로에 입력하여 D의 역행렬을 구하지 않고 선형 시스템 문제의 해를 구할 수 있다.

Dv = b 문제에서 D가 full rank이면, 전역 최소치의 값을 구할 수 있다. 만약 rank(D) < n 이면, 초기치에 따라 국부 최소치중 하나의 값을 얻게 된다. rank(D) = n < m의 경우는 최소자승문제가 되는데 우리는 점근 안정치를 얻을 수 있다. 예로서 $D_{n\times n}x = b$(D : full rank)인 경우를 생각하자.

$$\begin{aligned} 2x_2 + x_4 &= 0 \\ 2x_1 + 2x_2 + 3x_3 + 2x_4 &= -2 \\ 4x_1 - 3x_2 + 4x_4 &= -7 \\ 6x_1 + x_2 - 6x_3 - 5x_4 &= 6 \end{aligned}$$

구하여진 해는

$$x^T = [-0.50000\ 1.00000\ 0.33333\ -2.00000].$$

V. 결론

본 논문에서 우리는 선형계획 문제와 선형시스템 문제의 해를 구할 수 있는 변형된 신경회로를 제안했다. 우리의 모델은 선형계획 문제와 선형시스템 문제를 변형없이 하나의 모델로서 해결할 수 있다. 앞으로 신경회로의 스케일링 효과는 다른 많은 분야에서 응용되어 탁월한 효과를 얻을 수 있을 것이다.

참고문헌

[1] D. W. Tank and J. J. Hopfield, "Simple Neural Optimization Networks: An A/D Converter, Signal Decision Circuits, and a Linear Programming Circuit", IEEE Trans. on Circuits and Systems, vol CAS-33, May 1986.

[2] M. P. Kennedy and L. O. Chua, "Unifying the Tank and Hopfield Linear Programming Circuit and Nonlinear Programming Circuit of Chua and Lin", IEEE Trans. on Circuits and Systems, vol. CAS-34, No.2, pp. 210-214, 1987.

[3] M. P. Kennedy and L. O. Chua, "Neural Networks for Nonlinear Programming", IEEE Trans. on Circuits and Systems, No.5, pp. 554-562, May 1988.

[4] M. Aourid, D. Mukhedkar, B. Kaminska, " Convergence and Stability Study of Hopfield's Neural Network for Linear Programming", IJCNN 92, pp. IV-525 ~ 531.

[5] C. Y. Maa and M. A. Shanblatt, "Linear and Quadratic Programming Neural Network Analysis", IEEE Trans. on Neural Networks, Vol. 3, No. 4, pp. 580-594, July 1992.

[6] A. Cichocki and R. Unbehauen, "Neural Networks for Solving Systems of Linear Equations and Related Problems", IEEE Trans. on CAS, Vol. 39, No. 2, Feb. 1992.

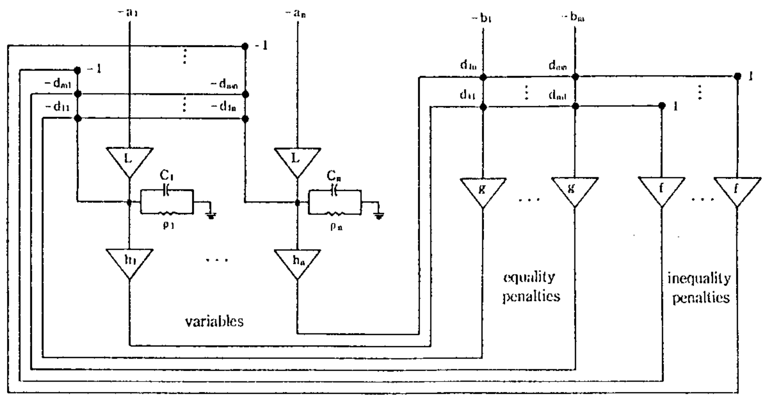

Fig. 1. The proposed linear programming neural networks