

論文91-28B-12-9

다중컴퓨터 시스템을 이용한 최적화 신경회로망의 최적 병렬구현

(Optimal Parallel Implementation of an Optimization Neural Network by Using a Multicomputer System)

金 鎮 浩,* 崔 興 文*

(Jin Ho Kim and Heung Moon Choi)

要 約

본 논문에서는 다중 컴퓨터 시스템에서 노드수에 비례하여 속도제고율 S_p 의 선형증가가 가능한 최적화 신경회로망의 최적 병렬 설계 및 그 구현 방안과 성능평가 모델을 제시하였다. 최적화 신경 회로망 고유의 공간적 및 시간적 병렬성을 추출한 다음 이를 파이프라인처리 모델화함으로써, CSP (communicating sequential process) 시스템에서 최대의 속도제고율 및 병렬처리 효율을 얻을 수 있는 병렬처리 방안을 마련하였다. 제안된 방식을 트랜스퓨터시스템에 적용하여 구현한 결과, 최적화 신경회로망의 신경세포의 갯수가 140여개 이상 될때 부터는 다중컴퓨터의 노드갯수에 거의 선형비례하는 S_p 의 증가와, 96% 이상의 병렬처리 효율을 얻을 수 있었다.

Abstract

We proposed an optimal parallel implementation of an optimization neural network with linear increase of speedup by using multicomputer system and presented performance analysis model of the system. We extracted the temporal- and the spatial- parallelism from the optimization neural network and constructed a parallel pipeline processing model using the parallelism in order to achieve the maximum speedup and efficiency on the CSP architecture. The results of the experiments for the TSP using the Transputer system, show that the proposed system gives linear increase of speedup proportional to the size of the optimization neural network for more than 140 neurons, and we can have more than 98% of efficiency upto 16-node system.

I. 서 론

대규모의 병렬분산처리 능력을 갖는 신경회로망을 이용하여 최적화 및 인식 등의 과제들을 고속병렬처리 하기 위한 연구들이 활발히 진행되고 있다.¹⁻¹¹⁾

신경회로망을 이용하여 과제들을 고속병렬처리하기 위해서는 신경회로망의 설계, 시뮬레이션 및 구현

과정을 거쳐야 한다. 일반적으로 신경회로망의 설계 및 시뮬레이션에는 복잡한 계산과정이 요구된다. 이와 같은 신경회로망의 시뮬레이션은 매우 긴 컴퓨터 처리 시간이 요구되므로, 신경세포들의 갯수가 많아 질 경우에는 현실적으로 시뮬레이션의 불가능한 경우도 발생한다. 또한, Stochastic 신경회로망 모델에는 온도 스케줄링 및 잡음발생 방안이 포함되어야 하므로 시뮬레이션을 통한 신경회로망의 설계가 끝났다고 하더라도 그 하드웨어 구현에 많은 어려움이 따르게 된다. 따라서 신경회로망의 설계, 시뮬레이션 및 하드웨어 구현을 원활하게 하기 위한 병렬처리시

* 正會員, 慶北大學校 電子工學科
(Dept. of Elec. Eng., Kyungpook Nat'l Univ.)
接受日字: 1991年 7月 24日

시스템이 절실히 요구되고 있다.

Fukuda 등^[3]은 CCP(communicating concurrent processor)기초로 한 다중컴퓨터 시스템에서 부하균등할당방식을 도입한 신경컴퓨터의 병렬구현에 관하여 연구 발표하였다. 또한 Bagherzadeh 등^[4]은 밀결합 공유메모리를 갖는 다중 컴퓨터 시스템에서 신경회로망을 이용한 TSP 문제의 병렬처리 방안을 제시하였다. 이들 연구에서는 대상 시스템의 통신방식이나 컴퓨터 구조에 따라 서로 다른 신경회로망 구현을 제시하였다. 따라서 신경회로망을 병렬 구현하려는 다중컴퓨터 시스템이 트랜스퓨터와 같은 CSP(communicating sequential process)로 구성된 경우에도 CCP로 구성된 다중컴퓨터 시스템에서와는 달리 나름대로의 효과적인 신경회로망 구현 방안이 별도로 마련되어야 할 것이다. 또한 신경회로망의 병렬구현을 위한 해석적 모델을 이용하여 다중 컴퓨터 시스템의 속도제고율 및 병렬처리 효율 등을 예측할 수 있다면 신경회로망의 설계 및 구현에 관한 연구가 보다 용이해질 수 있을 것이다.

본 논문에서는 다중 컴퓨터 시스템에서 노드수의 증가에 비례하여 속도제고율 Sp의 선형증가가 가능한 최적화 신경 회로망의 최적 병렬설계 및 구현 방안을 제안하고, 그 성능평가 모델을 제시하였다. 먼저 최적화 신경회로망 고유의 공간적 및 시간적 병렬성을 추출한 다음, 이를 병렬 파이프라인 처리 모델화함으로써 CSP 시스템에서 최대 속도제고율 및 병렬처리 효율을 얻을 수 있는 병렬처리 방안을 마련하였다. 또한 성능평가 모델을 구성하여 병렬처리 시스템의 노드수의 증가에 대하여 Sp의 선형증가를 얻으면서도 확장성이 가능한 설계방안을 마련하였다. 제안된 방안을 16개의 트랜스퓨터 T414로 구성된 다중컴퓨터 시스템에 적용하여 그 타당성을 검토해 보았다.

II. 최적화 신경회로망의 설계 및 병렬구현

1. 최적화 신경회로망 모델의 일반화

본 연구에서는 먼저 최적화 신경회로망들의 일반화모델을 구성한 다음 이를 병렬 구현하였다.

최적화 신경회로망들은 신경세포의 출력갱신(output update) 방법에 따라 결정론적(deterministic) 신경회로망과 확률론적(stochastic) 신경회로망 등으로 구분될 수 있다. Hopfield 신경회로망과 같은 결정론적 신경회로망에서 한 신경세포의 동작방정식(motion equation)은 다음과 같이 표현된다. 즉,

$$\frac{du_i(t)}{dt} = -\frac{u_i(t)}{\tau} + \sum W_{ij}V_j(t-1) + I_i \quad (1)$$

이다. 식에서 u_i 및 $V_j(t-1)$ 은 각각 t 및 $t-1$ 번째의 신경세포 입력 및 출력이고, τ 는 시정수, W_{ij} 는 가중치행렬, 그리고 I_i 는 입력 바이어스 전류이다. 식(1)을 Laplace 변환하여 $U_i(s)$ 에 대해 다시 정리하면

$$U_i(s) = \frac{net_i}{s + \frac{1}{\tau}} \quad (2)$$

이 되며 여기서 신경세포의 가중된 입력 $net_i = \sum W_{ij}V_j(t-1) + I_i$ 가 된다. 출력 $V_i(t)$ 는 식(1) 및 (2)로부터 구한 $u_i(t)$ 를 이용하여 다음과 같이 계산될 수 있다. 즉,

$$V_i(t) = f(u_i(t), \lambda) \quad (3)$$

이다. 식(3)에서 λ 는 전달함수의 기울기이다. 식에서 함수 f 는 hard limit, threshold logic 또는 sigmoid 함수로 치환될 수 있다. 신경 세포는 전달함수 f 에 따라 이진값 또는 연속값의 출력을 내게된다. 이성과 같이 결정론적 출력갱신을 하는 신경세포는 그림 1과 같은 블럭 다이어그램으로 모델화할 수 있다.

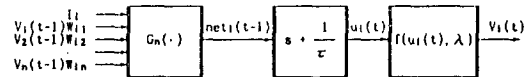


그림 1. 결정론적 출력 갱신을 하는 신경세포모델
Fig. 1. Neuron model with deterministic updates of output.

그림에서 보는 바와 같이 신경세포는 다른 신경세포의 가중된 출력을 받은 다음 크게 세 단계의 순서 처리 과정을 거쳐서 $V_i(t)$ 를 출력한다. 즉, 결정론적 신경회로망의 t 회 갱신에서는 다른 신경세포들의 $t-1$ 회 갱신에서의 출력 $V_j(t-1)$ 를 가중합하여 입력 $u_i(t)$ 를 구하고, sigmoid 등과 같은 출력 함수에 따라 출력 $V_i(t)$ 를 내게 된다.

한편 Boltzmann machine, Gaussian 및 Cauchy machine 등과 같은 확률론적 신경회로망에서는 결정론적 신경회로망의 입력측 또는 출력측에 잡음을 가하여 출력벡터의 확률론적인 변화가 가능하도록 되어 있다. 따라서 확률론적 신경회로망은 다수의 국소최저점들(local minima)이 존재하는 최적화문제에서도 전역 최적해를 효과적으로 구할 수 있다. 확률론적 신경회로망의 신경세포는 그림2와 같이 모델화할 수 있다.

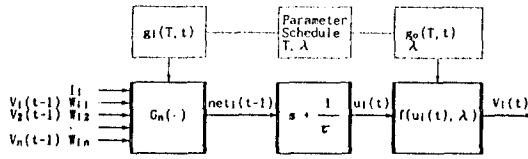


그림 2. 확률론적 동작을 하는 신경세포모델
 Fig. 2. Neuron model with stochastic updates of output.

그림의 변수기획 (parameter schedule)에서는 출력 갱신시마다 잡음의 양을 가변시키는 온도 T와 전달함수의 기울기를 가변시키는 λ 를 기획하여 그에 해당하는 잡음을 $g_i(T, t)$ 또는 $g_o(T, t)$ 를 통해서 신경세포의 입력측 또는 출력측에 인가한다. 따라서 최적화 신경 회로망은 잡음의 인가유무, 인가되는 잡음의 종류 및 인가 방법에 따라 각각 다른 종류의 신경회로망이 된다. 이상을 종합하여 일반화하면 최적화 신경회로망 모델은 다음과 같은 식들로서 표현될 수 있다. 즉,

$$\begin{bmatrix} net_1(t-1) \\ net_2(t-1) \\ \dots \\ net_n(t-1) \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1n} \\ w_{21} & w_{22} & \dots & w_{2n} \\ \dots & \dots & \dots & \dots \\ w_{n1} & w_{n2} & \dots & w_{nn} \end{bmatrix} \times \begin{bmatrix} v_1(t-1) \\ v_2(t-1) \\ \dots \\ v_n(t-1) \end{bmatrix} + \begin{bmatrix} g_{i1}(T, t) \\ g_{i2}(T, t) \\ \dots \\ g_{in}(T, t) \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} U_1(s) \\ U_2(s) \\ \dots \\ U_n(s) \end{bmatrix} = (s + \frac{1}{\tau}) \times \begin{bmatrix} net_1(t-1) \\ net_2(t-1) \\ \dots \\ net_n(t-1) \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} v_1(t) \\ v_2(t) \\ \dots \\ v_n(t) \end{bmatrix} = \begin{bmatrix} f(u_1(t), \lambda, g_{o1}(T, t)) \\ f(u_1(t), \lambda, g_{o2}(T, t)) \\ \dots \\ f(u_1(t), \lambda, g_{on}(T, t)) \end{bmatrix} \quad (6)$$

이다. 신경회로망은 각 신경세포들 간에 완전결합을 이루고 있으므로 식(4)와 같이 n개의 신경세포들의 가중된 합 $net_i(t-1)$ 을 구하는데 $n \times n$ 회의 행렬 벡터 적과 n회의 가산이 필요하다. 또한 그 입력을 구

하기 위해서 식(5)와 같이 미분방정식의 해를 n회 구해야 하고, 출력을 구하기 위해 식(6)과 같이 출력함수 값을 n회 계산해야 하므로 매번 갱신시마다 많은 량의 계산이 요구된다.

신경회로망을 분할하여 각 신경세포들을 다른 노드 컴퓨터에 할당하고 신경회로망의 최적화과정을 병렬로 구현할 때는 노드 컴퓨터들 사이의 신경세포 출력값 통신과정과 각 신경세포들의 출력을 구하기 위한 전산 처리과정들이 포함된다. 본 연구에서는 식(4), (5) 및 (6)과 같이 일반화 시킨 최적화 신경회로망 모델에서 공간적 및 시간적 병렬성을 추출하여 병렬처리 모델을 구성한 다음, 이에 따라 다중 컴퓨터 시스템에 이용하여 최적화 신경회로망을 병렬구현 하였다.

2. 다중컴퓨터 시스템을 이용한 병렬구현

최적화 신경회로망의 병렬구현을 위해 식(4), (5) 및 (6)으로 표현되는 신경회로망의 일반화 모델을 그 고유의 공간적 및 시간적 병렬성에 따라 병렬분할하여, 데이터 의존 그래프를 작성하고 이를 다중컴퓨터 시스템에 할당하는 방식으로 신경회로망을 병렬 구현 하였다.

먼저 신경회로망의 공간적 병렬성에 따라 신경세포 및 시냅스 가중행렬을 병렬분할하는 방안을 마련하고, 다음 이를 이용하여 시간적 병렬성을 반영시킨 데이터 의존 그래프를 작성하였다.

4개의 신경 세포들이 1차원 단층 배열된 신경회로망을 그림 3(a)에 도시하였다.

그림 3(a)의 신경회로망은 논리적으로 그림 3(b)와 같이 도시할 수 있다. 그림에서 신경세포들을 두 개의 영역으로 병렬분할하여 그림 3(c)와 같이 다시 그릴 수 있다. 그림 3(c)에서 GN_i 는 i번째 영역에 포함된 신경세포들을 의미하고 GV_i 는 해당 신경세포들의 출력을 표시하며, G_{ij} 는 영역 i와 j사이의 시냅스 가중행렬을 의미한다. 따라서 t번째 갱신에서 그림 3(c)의 각 영역에 속한 각 신경세포들에 입력되는 다른 신경세포들의 가중된 출력 $Gn_i(t-1)$ 는 다음과 같이 표시될 수 있다. 즉,

$$\begin{bmatrix} Gn_1(t-1) \\ Gn_2(t-1) \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \times \begin{bmatrix} Gv_1(t-1) \\ Gv_2(t-1) \end{bmatrix} \quad (7)$$

이다. 그림4는 2차원 배열된 신경회로망을 4개의 영역으로 분할한 것으로서 편의상 영역3에 속한 신경세포들 GN_3 과 다른 영역에 속한 신경세포들간의 시냅스 연결행렬 G_{31}, G_{32}, G_{33} 및 G_{34} 만을 도시한 것이다. 그림에서 t번째 갱신에서 i번째 신경세포들의 입

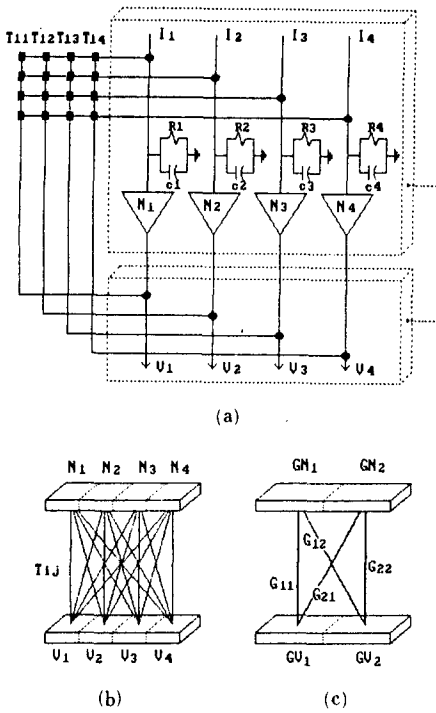


그림 3. 1차원 단층 신경회로망(a), 그 논리 구성도(b) 및 병렬 처리 분할 모델(c)
 Fig. 3. One dimensional single layer neural network (a), it's logical configuration (b), and it's parallel decomposition model(c).

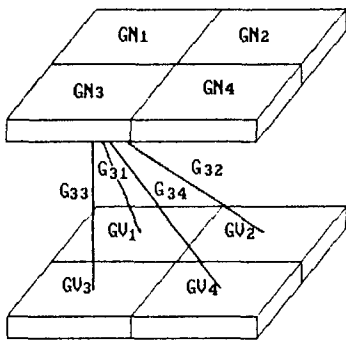


그림 4. 4개의 영역으로 분할된 2차원 신경회로망 모델 (편의상 GN₃의 시냅스 연결행렬만 표시함)
 Fig. 4. 2-dimensional neural network model which is decomposed into 4 parts (Only the synaptic interconnection for GN₃ are shown for convinience).

력 $Gn_i(t-1)$ 은 다음과 같이 다시 병렬 분할될 수 있다. 즉,

$$\begin{bmatrix} Gn_1(t-1) \\ Gn_2(t-1) \\ Gn_3(t-1) \\ Gn_4(t-1) \\ Gn_{13}(t-1) + Gn_{14}(t-1) \\ Gn_{23}(t-1) + Gn_{24}(t-1) \\ Gn_{33}(t-1) + Gn_{34}(t-1) \\ Gn_{43}(t-1) + Gn_{44}(t-1) \end{bmatrix} = \begin{bmatrix} Gn_{11}(t-1) + Gn_{12}(t-1) + \\ Gn_{21}(t-1) + Gn_{22}(t-1) + \\ Gn_{31}(t-1) + Gn_{32}(t-1) + \\ Gn_{41}(t-1) + Gn_{42}(t-1) + \\ Gn_{13}(t-1) + Gn_{14}(t-1) \\ Gn_{23}(t-1) + Gn_{24}(t-1) \\ Gn_{33}(t-1) + Gn_{34}(t-1) \\ Gn_{43}(t-1) + Gn_{44}(t-1) \end{bmatrix} \times \begin{bmatrix} Gv_1(t-1) \\ Gv_2(t-1) \\ Gv_3(t-1) \\ Gv_4(t-1) \end{bmatrix} \quad (8)$$

이다. 따라서 신경회로망의 신경세포들을 동일한 크기로 병렬분할하고, 분할된 신경세포들 간의 시냅스 가중행렬을 다시 병렬분할하는 과정을 신경회로망의 공간적 병렬성을 추출하는 과정으로 볼 수 있다.

제시된 방법에 따라 신경세포들간의 시냅스 가중행렬을 2차원으로 공간적 병렬분할하고, 각 신경세포들의 시간적 병렬성을 반영하여 4개의 부태스크로 구성된 데이터 의존 그래프를 작성하면 그림 5와 같다.

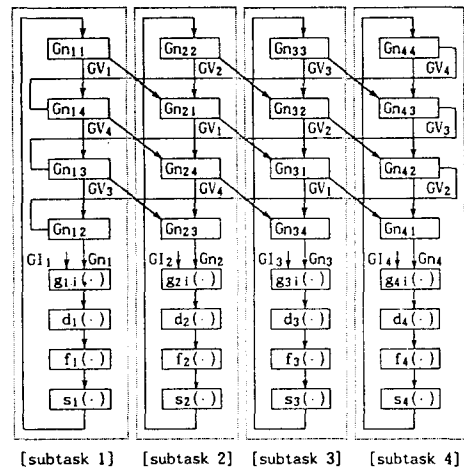


그림 5. 2차원 분할최적화 신경회로망의 데이터 의존 그래프
 Fig. 5. Data dependence graph of the 2 dimnesi dimensionally decomposed optimization neural network.

그림5에서 G_{ni} 는 부태스크 $i-1$ 에 속한 신경세포들의 출력 GV_i 를 받아서 부태스크 i 의 신경세포들의 j 번째 가중된 합을 구하는 것이다. 따라서, 신경회로망을 4개의 영역으로 2차원 분할했을 때 전체 신경세포들을 1회 갱신하기 위해서는 해당 영역의 신경세포출력 GV_i 를 인접한 영역으로 3회 쉬프트시켜야 한다. 일단 각 영역에 속한 신경세포들의 가중된 합 G_{ni} 를 구하고 나면 해당 영역의 신경세포들의 출력 GV_i 를 구할 때까지는 영역 상호간의 데이터 의존성이 존재하지 않음을 알 수 있다. 그림에서 $g_i(\cdot)$ 에서 잡음 및 입력 전류를 계산하고, $d(\cdot)$ 에서는 신경세포의 입력 전류를 구하기 위해서 미분방정식의 해를 구한다. 그리고, $f(\cdot)$ 에서는 신경세포의 출력을 계산하고, $s(\cdot)$ 에서는 파라미터 T 와 λ 를 기획한다.

$g_i(\cdot)$, $d(\cdot)$, $f(\cdot)$ 및 $s(\cdot)$ 는 최적화 신경회로망의 종류에 따라 달라질 수 있다. N 개의 신경세포들을 갖는 Gaussian machine 알고리즘을 P 개의 부태스크로 분할했을 경우의 병렬처리 모델을 그림6에 도시하였다.

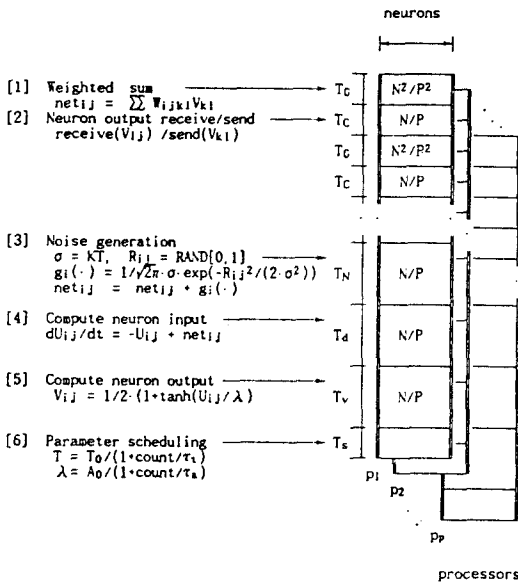


그림 6. N 개의 신경세포들을 갖는 Gaussian machine 알고리즘을 P 개의 부태스크로 분할했을 경우의 병렬처리 모델
Fig. 6. Parallel processing model for gaussian machine algorithm with N neurals neurons decomposed into P subtasks.

그림에서 보는 바와 같이 각 부태스크에 할당된 신경세포들간의 데이터 의존성은 각 신경세포들의 출력을 부태스크들 상호간에 전송할 때 존재하게 된다. 즉, 한 부태스크 i 의 신경세포들 G_{ni} 가 부태스크 $i-1$ 로 부터 신경세포들의 출력 GV_i 를 전송받아서 가중된 출력값 G_{ni} 를 계산하고, 해당 출력 GV_i 를 부태스크 $i+1$ 의 신경세포들에게 전송해야 한다.

그림5의 데이터 의존 그래프는 데이터 전송과 가중합 과정을 반복하는 환형구조의 파이프라인 처리 그래프로 볼 수 있다. 따라서 CSP로 구성된 환형구조의 트랜스퓨터 시스템에 태스크를 할당하여 병렬처리하면 라우팅(routing)에 의한 부담을 가급적 줄일 수 있기 때문에 통신 부담을 최소화할 수 있을 것이다. 다음은 4개의 부태스크가 할당된 트랜스퓨터 시스템을 도시한 것이다.

그림7에서 T_{root} 트랜스퓨터는 각 T_{slavei} 트랜스퓨터로 초기 데이터를 전송하고 T_{slavei} 트랜스퓨터들의 병렬처리 과정을 감시(monitoring) 한다. 그리고 T_{slavei} 트랜스퓨터들은 할당받은 태스크 및 데이터들을 파이프라인 구조에 따라 처리 및 통신하면서 최적화를 수행한다. 따라서 제안된 신경세포 및 가중행렬 분할방법, 데이터 의존 그래프 작성방법 그리고 태스크 할당방법등을 이용하면 환형 다중컴퓨터 시스템에서 routing 부담을 최소화하면서 최대 속도로 동작하는 최적화 신경회로망을 병렬 구현할 수 있다.

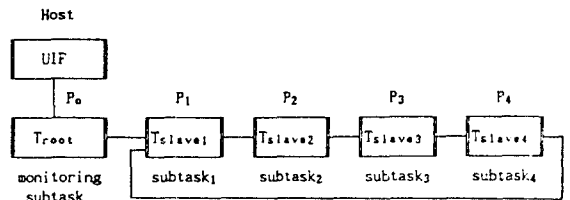


그림 7. 최적 태스크 할당된 다중컴퓨터 시스템
Fig. 7. Optimally task allocated multicomputer system.

3. 성능 평가 모델

그림5와 같이 N 개의 신경세포들이 P 개의 영역으로 분할된 태스크를 최적 할당받은 다중컴퓨터 시스템에서, 한개의 컴퓨터 PE_i 가 해당 신경세포들을 1회 갱신을 하는데 소요되는 병렬처리시간 T_{PEi} 는 다음과 같다. 즉,

$$T_{PE1} = (N^2/P) \cdot T_c + (P-1) (N/P) T_c + (N/P) \cdot T_N + (N/P) \cdot T_d + (N/P) \cdot T_v + T_s \quad (9)$$

이다. 한편 전체 태스크를 하나의 컴퓨터가 처리할 때 소요되는 순서처리시간 T_{sing} 은 다음과 같다. 즉,

$$T_{sing} = N^2 \cdot T_c + N \cdot T_N + N \cdot T_d + N \cdot T_v + T_s \quad (10)$$

이다. 따라서 병렬처리시스템의 속도 제고율 Sp 는

$$\begin{aligned} Sp &= T_{sing}/T_{PE1} \\ &= \frac{N^2 T_c + N T_N + N T_d + N T_v + T_s}{(N^2/P) T_c + (P-1) (N/P) T_c + (N/P) T_N + (N/P) T_d + (N/P) T_v + T_s} \\ &= \frac{N^2 T_c + N \cdot T_N + N \cdot T_d + N \cdot T_v + T_s}{(N^2 \cdot T_c + N \cdot T_N + N \cdot T_d + N \cdot T_v + T_s) / P} + \frac{(P-1) \cdot N \cdot T_c}{P} \\ &= \frac{P\alpha}{\frac{P\alpha}{P} + \frac{C\alpha}{P}} \quad (11) \end{aligned}$$

이 되며, 여기서 $P\alpha$ 및 $C\alpha$ 는

$$P\alpha = N^2 \cdot T_c + N \cdot T_N + N \cdot T_d + N \cdot T_v + T_s \quad (12a)$$

$$C\alpha = (P-1) \cdot N \cdot T_c \quad (12b)$$

이 된다. 또한 병렬처리 효율 E_{eff} 는

$$E_{eff} = \frac{Sp}{P} \quad (13)$$

이 된다. 식(12)에서 N 이 증가하면 $P\alpha$ 의 증가율이 $C\alpha$ 의 증가율보다 크다. 따라서 노드 갯수 P 를 고정하고 신경세포 N 를 크게 하면 $P\alpha \gg C\alpha$ 이 되고 따라서 식(11)에서 $Sp \approx P$ 가 되어 병렬처리 효율이 1에 접근하게 된다. 한편 신경세포 갯수 N 을 고정하고 노드 갯수 P 를 증가시키면, $P\alpha > C\alpha$ 인 범위에서는 Sp 도 노드 갯수에 선형적으로 비례하여 증가하지만, P 를 계속 증가시키면 $P\alpha > C\alpha$ 의 관계가 더 이상 유지되지 않게 된다. 이때 N 이 큰 값이라면 P 가 증가해도 $P\alpha > C\alpha$ 의 관계가 계속 유지되어 식(13)으로 정의되는 효율이 1에 접근한다. 따라서 신경회로망이 대규모화 함에 따라 P 의 증가에 대해서 Sp 의 선형적인 증가를 얻을 수 있다.

III. 실험 및 고찰

제안된 설계 및 병렬구현방안의 타당성을 확인하기 위해, 제안된 방식을 TSP문제의 최적해를 구하는데 적용하고 그 결과를 검토 고찰하였다. 실험에

사용된 다중컴퓨터 시스템은 IBM/PC Mother Board IMS B008-1 2개와 1개의 Root 트랜스퓨터 Module IMSB404, 그리고 16개의 Slave 트랜스퓨터 Module IMS B401-2로 구성하였다. 병렬프로그램은 IMSD 701-6 TDS를 이용하여 Occam으로 작성하였고 환형구조는 IMS S708 MMS2를 이용하여 설정하였다. Stochastic Gaussian Machine을 제안된 방식대로 병렬 분할하여 트랜스퓨터시스템에 할당하고, 이를 이용하여 8개, 10개 및 12개 도시를 방문하는 TSP문제의 적해를 구하였다. 각 TSP 문제들을 1개, 2개, 4개, 8개 및 16개의 PE들로 구성된 트랜스퓨터시스템에서 수행시켜 본 결과, 각각의 속도 제고율은 그림8과 같았다.

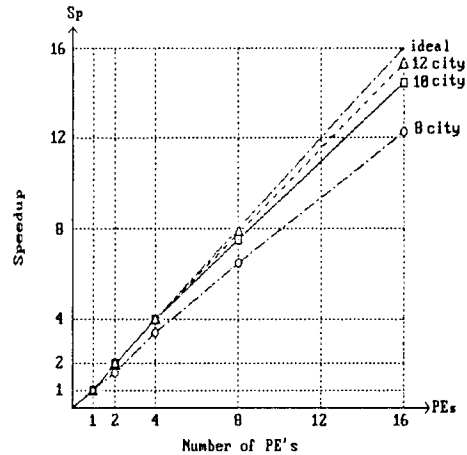


그림 8. 각 TSP 문제들에 대한 속도제고율
Fig. 8. Speedup for each of the TSPs.

그림에서 보면 도시수가 많아 문제의 규모가 커질수록, 즉 신경세포의 수가 증가할수록 Sp 의 증가가 노드수 증가에 대해 이상적인 선형증가에 가까워짐을 알 수 있다. 따라서, 최적화 문제의 규모가 커질수록 속도제고율이 노드수에 비례하여 선형적으로 증가하므로, 다중컴퓨터 시스템의 확장성이 용이함을 알 수 있다. 그림9는 각 TSP 문제에 대한 신경회로망의 병렬구현에 있어서의 다중컴퓨터 시스템의 병렬처리 효율 E_{eff} 를 도시한 것이다.

그림에서 보면 트랜스퓨터 16개로 구성된 다중컴퓨터 시스템은 12개 도시 TSP문제를 해를 약 96%의 효율로 구할 수 있다. 따라서 제안된 방식은 도시의 수가 어느 정도만 증가되더라도 최대 효율로서 그 해를 구할 수 있음을 확인할 수 있다. 제안된 방

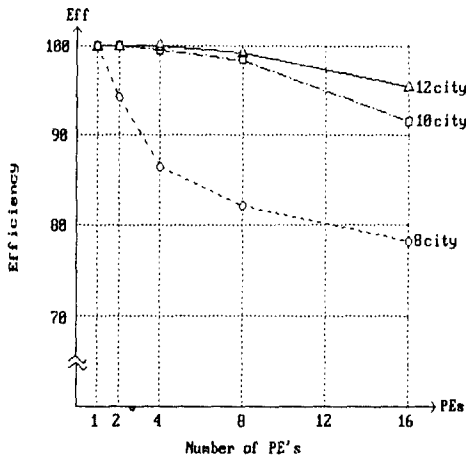


그림 9. 각 TSP 문제들에 대한 병렬처리 효율
Fig. 9. Efficiency for each of the TSPs.

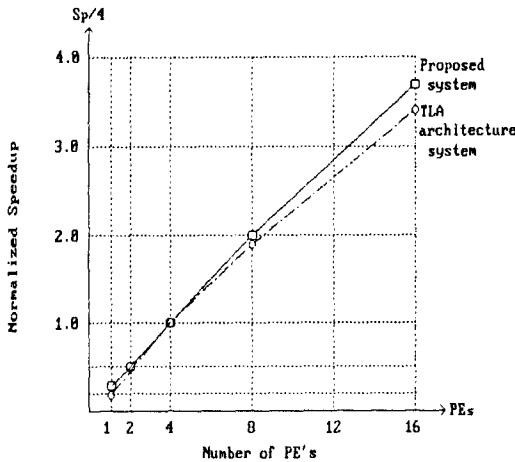


그림 10. 10개 도시 TSP에 대하여 제안된 방식과 TLA 구조의 표준화 속도제고율 Sp/4 비교
Fig. 10. Normalized speedup Sp/4 comparison between proposed method and TLA for 10-city TSP.

식과 Sp와 T800 트랜스퓨터를 16개 사용한 TLA 구조의 다중컴퓨터 시스템¹²⁾의 Sp를 비교하기 위해, 두 방식을 10개 도시의 TSP 문제에 각각 적용하고, 4개 트랜스퓨터를 사용한 병렬 처리 시스템의 Sp(=4)로 표준화 시켜 그 결과를 그림 10에 도시하였다.

그림에서 보면 제안된 방식이 TLA 보다 비교적 높은 Sp 증가를 가짐을 알 수 있다. 이것은 트랜스퓨터 T800의 전산처리 능력이 T414보다 우수하기

때문에 TLA에서는 전산처리 능력이 통신능력 보다 크고 또한 실제의 트랜스퓨터 시스템은 CSP로 구성되어 있으므로 TLA 구조가 환형 구조보다 routing 방식이 복잡하여 통신부담이 증가되었기 때문으로도 볼 수 있다.

궁극적으로 제안된 방식을 이용하면 12개 도시 TSP에서 보는 바와 같이 신경세포의 갯수가 140여 개 이상이 될 때 그 Sp가 노드수에 거의 선형적으로 비례증가하고 그 병렬처리 효율도 96% 이상 되므로 매우 우수한 병렬처리 결과를 얻을 수 있다.

IV. 결 론

본 논문에서는 다중컴퓨터 시스템에서 노드수의 증가에 비례하여 속도제고율 Sp의 선형증가가 가능한 최적화 신경회로망의 최적 병렬설계 및 구현 방안을 제안하고, 그 성능평가 모델을 제시하였다. 즉, 최적화 신경회로망 고유의 공간적 및 시간적 병렬성을 추출한 다음, 이를 병렬파이프라인 처리 모델화 함으로써 CSP 시스템에서 최대 속도제고율 및 병렬처리 효율을 얻을 수 있는 병렬처리 방안을 마련하였다. 제안된 최적화 신경회로망의 설계 및 구현 방안의 타당성을 확인하기 위해 Gaussian machine을 트랜스퓨터 시스템에 병렬 구현한 다음, 이를 TSP 문제의 해를 구하는데 적용하여 그 결과를 검토 고찰하였다.

제안된 방식을 이용하면 12개 도시 TSP에서 보는 바와 같이 신경세포의 갯수가 140여개 이상될 때 그 Sp가 노드수에 거의 선형적으로 증가하고 그 병렬처리 효율이 96% 이상 되므로 매우 우수한 병렬처리 결과를 얻을 수 있다. 앞으로 노드의 전용 메모리를 확장하여 좀 더 대규모의 신경회로망의 병렬구현에 관하여 연구가 계속 되어야 하리라 본다.

參 考 文 獻

- [1] W. Li and N.M. Nasrabadi, "Object Recognition Based on Graph Matching Implemented by a Hopfield-Style Neural-Network," *IJCNN*, vol. 2, pp. 287-290, Jun., 1989.
- [2] N. Fukuda, Y. Fujimoto, and T. Akabane, "A Transputer Implementation of Toroidal Lattice Architecture for Parallel Neurocomputing," *IJCNN*, vol. 2, pp. 43-46, Jan., 1990.
- [3] N. Bagherzadeh, T. Kerola, B. Leddy, and R. Brice, "On Parallel Execution of the Traveling Salesman Problem on a Neuarl

- Network Model," *IEEE ICJNN*, vol. 3, pp. 317-324, Jun., 1987.
- [4] J. Ghosh and K. Hwang, "Mapping Neural Networks onto Message-Passing Multicomputers," *Journal of PDC.*, vol. 6, pp. 291-330, Apl., 1989.
- [5] M. Milksa, "A Development Tool for Neural Networks Simulations on Transputers," *Parallel Processing in Neural Systems and Computers*, North-Holland, pp. 291-294, 1990.
- [6] M. Migliore, G. Ayala, and S. Fornili, "Modeling of Neural Systems on Transputer Networks," *Parallel Processing in Neural Systems and Computers*, North-Holland, pp. 291-294, 1990.
- [7] H. Kato, H. Yoshizawa, H. Iciki, and K. Asakawa, "A Parallel Neurocomputer Architecture towards Billion Connection Updates Per Second," *IJCNN*, vol 2, pp. 47-50, Jan., 1990.
- [8] T. Kraft and S.A. Frostrom, "Concurrent ANS Architecture Using Communicating Concurrent Processes," *IJCNN*, vol. 2, pp. 51-54, Jan., 1990.
- [9] A. Iwata, Y. Yoshida, S. Matsuda, Y. Sato, and N. Suzumura, "An Artificial Neural Network Accelerator Using General Purpose 24 Bits Floating Point Digital Signal Processors," *IJCNN* vol. 2, pp. 171-175, Jun., 1989.
- [10] J. Oglesby and J.S. Mason, "Dynamic Scheduling For Feed-Forward Neural Nets Using Transputer," *First IEE Intl. Conf.*, on *ANN*, no. 313, pp. 257-260, Oct., 1989.
- [11] J.H. Kim and H.M. Choi, "Distortion-invariant Object Recognition by Optimization Neural Network," *JTC-CSCC*, Dec., pp. 370-374, 1990.

 著 者 紹 介



金 鏡 浩 (正會員)

1960年 6月 20日生. 1985年 7月
경북대학교 전자공학과 졸업.
1988年 2月 경북대학교 대학원
전자공학과 졸업(공학석사). 1988
年 3月~1991年 12月 현재 경
북대학교 대학원 전자공학과 박

사과정 수료. 주관심분야는 병렬분산처리, 컴퓨터 구
조, 컴퓨터네트워크 및 신경회로망 등임.



崔 興 文 (正會員)

1944年 5月 17日生. 1966年 2月
인하대학교 전기공학과(전자공학
전공) 졸업. 1974年 2月 인하대
학교 대학원 전기공학과(전자공
학전공) 졸업(공학석사). 1987年
8月 인하대학교 대학원 전자공

학과 졸업(공학박사). 1982年~1983年 뉴욕 주립대
학교 객원교수. 1974年 11月~1991年 현재 경북대학
교 전자공학과 교수. 주관심분야는 병렬분산처리 시
스템, 컴퓨터네트워크, 신경회로망 응용 및 퍼지응용
등임.