

부품 조립 공정에서 경로의 최적화 알고리즘

오 제 휘*, 차 영 업**

Optimal Algorithm of Path in the Part-Matching Process

Oh, Je-Hui*, Cha, Young-Youp**

ABSTRACT

In this paper, we propose a Hopfield model for solving the part-matching in case that is the number of parts and positions are changed. The goal of this paper is to minimize part-connection in pairs and total path of part-connections. Therefore, this kind of problem is referred to as a combinatorial optimization problem. First of all, we review the theoretical basis for Hopfield model and present two optimal algorithms of part-matching. The first algorithm is Traveling Salesman Problem(TSP) which improved the original and the second algorithm is Weighted Matching Problem (WMP). Finally, we show demonstration through computer simulation and analyze the stability and feasibility of the generated solutions for the proposed connection methods. Therefore, we prove that the second algorithm is better than the first algorithm.

Key Words: part-matching (부품 조립), combinatorial optimization (결합 최적화), stability (안정성), feasibility (유용성)

1. 서 론

과거의 조립 공정은 작업자의 위치와 상태, 조립 부품의 위치와 개수, 조립 조건 그리고 조립 방법 등에 따라 외란이 다양하고 많아서 그 공정 효율이 심하게 변하였다. 이러한 단점을 극복하기 위한 현재의 자동화는 공정 효율을 극대화하기 위하여 작업자의 위치와 상태 그리고, 부품의 위치와 개수를 단일화하고, 생산 라인의 균일화로 작업자와 부품에 따른 외란을 극소화 시켰으며, 자동화를 위한 고가의 공장을 지어서 외란을 제거하였다. 그리고 조립 방법의 차이는 조립 공정에서 여러 라인을 만들어서 각 라인에 한가지 또는 여러 개의 조립 공정으로 공정을

분할하므로써 연속적인 조립이 가능하게 하였다. 하지만 이러한 문제의 해결책들은 각 공장에 따라 외란이 다르기 때문에 각 공장마다 그 공장 알맞은 치밀한 설계와 고가의 기자재 및 정밀한 기기 등을 요구하며, 만일 예견하지 못한 외란이 발생하면 그 공장의 효율은 떨어지고 그에 따른 또 다른 지출이 생기게 된다. 그러므로 공정을 단순화 또는 획일화 할 수 없는 복잡한 공정, 투자대 효율이 적은 조립 공정, 자금 사정이 어려운 기업의 공장에서의 자동화 공정 등에는 현재의 자동화를 쓸 수가 없게 되었다.

현재의 이러한 단점을 극복하기 위하여 본 논문은 조립 공정에서 다양한 조건에 만족하며 효율이 높은 시스템을

* 원광대학교 대학원.
** 원광대학교 기계설계공학과

구축하기 위한 알고리즘을 만들고자한다. 이를 위하여 조립 부품의 개수와 위치가 일정하지 않는 경우와 조립 조건이 변하는 경우를 능동적으로 만족시키는 이중(二種) 부품의 조립 공정을 생각해보겠다. 이 경우에 고려해야 할 사항으로는

- 1) 부품의 구별 및 개수 파악
- 2) 정확한 위치인식
- 3) 결합 부품간의 최적 연결 결정
- 4) 전체 부품간의 최적 연결 경로 결정

등 4가지로 간략화 할 수 있다. 첫 번째와 두 번째의 사항은 영상 인식에 의해 해결되며, 나머지 두 가지 사항은 결합 최적화 문제(Combinatorial Optimization Problem)로 취급하여 해결할 수 있다. 본 연구에서는 첫 번째와 두 번째의 사항은 패턴 인식에 따른 이미지 처리 과정의 문제로 보아 취급하지 않았고, 세 번째와 네 번째의 사항을 중점으로 연구하였다.

Fig.1은 이중 부품의 경우 발생하는 결합 최적화 문제의 4가지 경우이다. 각각 (a) 전체연결 경로와 결합부품 연결이 모두 나쁜 경우, (b) 전체연결 경로는 좋으나 결합부품 연결이 나쁜 경우, (c) 결합부품 연결은 좋으나 전체연결 경로가 나쁜 경우, (d) 전체연결 경로와 결합부품 연결이 모두 좋은 경우 등이다. 앞에서 언급한 4가지 경우에서 우리가 취득하고자 하는 최적 연결은 최소의 거리를 갖는 경우이므로 (d)와 같은 경우가 최적의 해라고 볼 수 있다. 이러한 결합 최적화 문제를 해결하는 모델로서는 여러 가지가 있으나 그 중 효율적인 것으로는 Hopfield Model의 Traveling Salesman Problem (이하

TSP)을 들 수 있다.

이 문제에 대하여 Hopfield와 Tank⁽¹⁾가 Hopfield Model을 적용한 TSP 해결 방법을 최초로 제시하였고, Wilson과 Pawley⁽²⁾가 위의 방법에 대한 안정성(Stability) 및 유용성(Feasibility) 문제에 대해 이의를 제기한 후로 많은 연구자들은 TSP의 유효한 값을 찾기 위해 노력하였다. 그 중에 Tagliarini와 Page⁽³⁾는 Stack Neurons라는 여분의 항을 이용하였고, Abe⁽⁴⁾는 고유치를 이용하여 거동 방정식의 수렴을 증명하였으며, Carvalho와 Barbosa⁽⁵⁾는 Boltzmann Machine과 Simulated Annealing을 이용하였고, Jappe와 Cardon⁽⁶⁾은 2개의 층(Layer)을 이용하여 유효한 식을 얻으려고 노력하였다. 특히 Aiyer, Niranjan, Fallside^(4,7)는 고유치를 이용하여 50개까지의 도시에서도 유효한 값을 얻을 수 있는 식을 얻어냈다. 하지만 위의 연구들은 결합 최적화 문제 해결 방법으로 TSP를 사용하였으나 거리 최적화 부분의 불안정성(Unstability)에 의해 전체 TSP의 안정성과 유용성이 낮아져 실질적인 결합 최적화 문제에는 적용이 어렵다고 볼 수 있다.

본 논문은 위의 어려움을 극복하기 위하여 우선적으로 Hopfield Model을 개괄적으로 설명한 후, TSP식에 대해 기존의 연구자들의 이론 중 고유치를 이용한 TSP식을 Algorithm I로 하며 설명하였다. 그리고 결합 부품간의 최적 연결에는 Weighted Matching Problem(이하 WMP)식을, 전체 부품간의 최적 연결에는 TSP식을 사용한 WM-TS Problem식을 유도하여 Algorithm II로 정의하였다. 그리고 이들을 이용하여 컴퓨터 시뮬레이션을 한 결과를 보임으로서, 이 논문에서 제안한 Algorithm II가 안정성과 유용성의 면에서 Algorithm I보다 더 우수함을 보이고자 한다.

2. Hopfield Model과 TSP

이 장에서는 Hopfield Model과 TSP의 기본 이론을 Hopfield와 Tank⁽¹⁾의 식을 통하여 설명한다. 고유의 TSP Algorithm은 도시들의 연결에 대하여 설명하고 해결하였으나, 부품 조립 공정에 이 Algorithm을 이용하기 위하여 '도시'라는 개념을 '부품'으로 고쳐 사용하였다.

2.1 Hopfield Model

Hopfield Model는 뉴런(Neuron)이라고 불리는 각각의 요소들이 서로 피드백(Feedback)으로 연결된 구조로

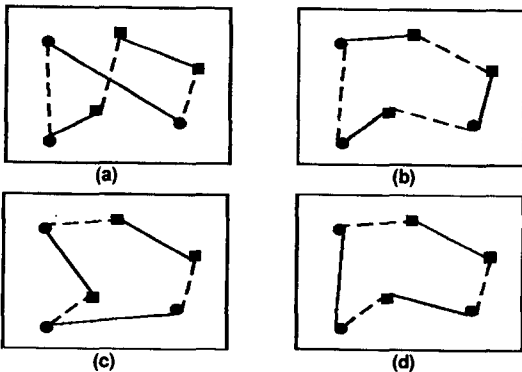


Fig.1 N=6인 경우의 이중(二種)부품에 대한 부품간, 전체 부품간의 연결 비교
 실선: 전체 부품 연결, 점선: 결합부품 연결, ● : A형 부품, ■ : B형 부품

서 Fig.2와 같이 나타낼 수 있다. 일반적으로 i 번째의 뉴런은 현재의 상태 또는 입력값 U_i , 그리고 그 출력값 V_i 등으로 표시가 가능하며, 출력값 V_i 는 비선형의 0~1(또는 -1~+1) 영역을 갖는 출력 함수 $g(U_i)$ 와 관계를 가지고 있다. 출력함수 $g(U_i)$ 는 여러 가지 형태가 있으나, 일반적으로 쌍곡선 탄젠트(Hyperbolic Tangent) 형태를 갖는다. i 번째 뉴런의 출력값 V_i 는 연결행렬(Connection Matrix) T_{ij} 와 곱해져서 j 번째 뉴런의 입력값 U_j 가 되며, 각각의 뉴런에는 Offset Bias라는 외부자극(External Excitation) I_i 를 받게 된다.

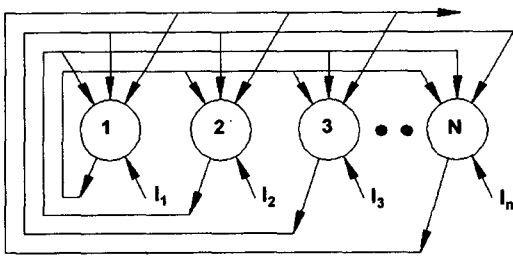


Fig. 2 Hopfield Model

i 번째 뉴런 U_i 는 이 뉴런의 총 입력값에 대한 함수로 값이 갱신(Update)되며, 이 함수는 사용하는 값이 연속적(Continuous)이나, 이산적(Discrete)이나에 따라 구별되어진다. 그러나 이산적인 경우에는 Network의 거동 방정식을 정의할 때, 미분 방정식의 적용이 불가능하므로 이 논문에서는 연속적인 경우라 설정하여 다룰 것이다^(2,4).

연속적인 경우의 미분 방정식으로 표시되는 Network의 거동 방정식은

$$\frac{dU_i}{dt} = \sum_{j=1}^N T_{i,j} V_j - \frac{U_i}{\tau} + I_i \quad (1)$$

이며, 식(1)에서 τ 는 회로의 시간 상수(Time Constant)이다. 또한 갱신을 위한 출력함수 $g(U_i)$ 는 쌍곡선 탄젠트로서 아래와 같다.

$$V_i = g(U_i) = \tanh\left(\frac{U_i}{\theta}\right) \quad (2)$$

2.2 TSP의 정의와 Hopfield - Tank식

TSP (Traveling Salesman Problem) 란 N 개의 부품을 오직 한 번에 한 부품씩 연결하는 최단의 연결 거리를

를 찾는 문제로서, NP complete문제로 분류되어 해결하기 어렵다. 여기서 발생하는 경우의 수는 부품이 N 개라면 $N! / 2N$ 가지 경우의 수가 존재한다. Hopfield와 Tank는 이 문제 해결을 위해 연결할 부품이 N 개라면, 뉴런의 수를 N^2 개로 설정하였다. N^2 뉴런을 N 뉴런에 대한 N 개 그룹으로 분류하고, N 뉴런에 대한 각각의 그룹은 그 순번에 대한 부품으로 나타내었다. 그리고 $V_{x,i}$ 는 x 번째 부품에 대한 i 번째 순번을 나타내며, 만일 임의의 x 번째 부품이 i 번째 순번을 갖는다면, 그 값이 1이고 (즉 뉴런이 켜짐), 그렇지 않다면 0(뉴런이 꺼짐)이 된다.

위의 표기법과 최적 설계를 문제 해결에 이용하여 최소 연결을 갖는 에너지 함수를 만들기 위해 제약 조건을 설정한다면

- C1) 오직 한 번만 그 부품을 연결하도록 한다;
- C2) 오직 한 개의 부품만이 그 순번에 존재한다;
- C3) 모든 N 개의 부품을 연결해야 한다;
- C4) 총 연결 거리는 최소가 되어야 한다.

등 네 가지로 나타낼 수 있다. 위의 네 가지 항을 모두 만족하여야 하지만, C1, C2, C3는 강한 제약조건(Strong Constraint)으로 유효한 값을 얻기 위하여 꼭 만족해야 할 조건들이며, C4는 그것의 조건과 비교하여 약한 제약조건(Weak Constraint)으로 분류할 수가 있다^(8,9). 그러므로 강한 제약조건은 약한 제약조건보다 선행되어야 하며, 이에 걸리는 하중(Load) 또한 더욱 커야 한다. 먼저 $V_{x,N+1} + V_{x,1}$, $V_{x,0} = V_{x,N}$ 로 정의한다면, 위의 조건에 만족하는 에너지 함수는

$$E = \frac{A}{2} \sum_{x=1}^N \sum_{i=1}^N \sum_{j=1, j \neq i}^N V_{x,i} V_{x,j} + \frac{B}{2} \sum_{i=1}^N \sum_{x=1}^N \sum_{y=1, y \neq x}^N V_{x,i} V_{y,i} + \frac{C}{2} \left(\sum_{x=1}^N \sum_{i=1}^N V_{x,i} - N \right)^2 + \frac{D}{2} \sum_{x=1}^N \sum_{y=1, y \neq x}^N \sum_{i=1}^N d_{x,y} V_{x,i} (V_{y,i+1} + V_{y,i-1}) \quad (3)$$

이다. 식(3)에서 $d_{x,y}$ 는 x 부품과 y 부품의 거리를 뜻하며, A, B, C 그리고 D 는 임의의 비례 상수이다. 그리고 A 항은 C1을, B 항은 C2를, C 항은 C3을, D 항은 C4를 각각 만족시킨다.

이에 대한 거동 미분 방정식을 식(1)과 같은 형태로 쓴다면

$$\frac{dU_{x,i}}{dt} = \sum_{y=1}^N \sum_{j=1}^N T_{x,i,y,j} V_{y,j} - \frac{U_{x,i}}{\tau} + I_{x,i} \quad (4)$$

$$(V_{x,i} = g(U_{x,i}) = \frac{1}{2}(1 + \tanh(\lambda U_{x,i})))$$

이고, λ 는 상수인 Gain 변수(Gain Parameter)라 한다.

식(3)와 식(4)에 의해 연결행렬 $T_{x,i,y,j}$ 와 거동 방정식에 양의 가중치를 주어 뉴런을 켜지게(Firing)하는 외부 자극 $I_{x,i}$ 는 각각

$$T_{x,i,y,j} = -A\delta_{x,y}(1 - \delta_{i,j}) - B\delta_{i,j}(1 - \delta_{x,y}) - C - Dd_{x,y}(\delta_{j,i+1} + \delta_{j,i-1}) \quad (5)$$

$$I_{x,i} = CN' \quad (6)$$

이다. 식(5)에서 $\delta_{i,j}$ 는 $i=j$ 인 경우에는 1, $i \neq j$ 인 경우에는 0과 같이 정의되며, 식(6)에서 N' 는 뉴런의 수 N 이 아닌 임의의 상수이다.

Hopfield와 Tank는 10개의 부품에 대해, 식(4), (5), (6)의 변수 값들을 $A=B=500, C=200, D=500, \tau=1, \lambda=50, N'=15$ 로 하여 풀었다. 이 식들은 10개의 부품까지는 안정되나 10개를 초과했을 경우에는 유효한 값을 갖지 못하는 심각한 불안정 상태가 나타난다⁽⁴⁾.

3. Algorithm I

이 장은 Hopfield와 Tank의 식을 근거로 하여 Aiyer, Niranjana, Fallside^(4,7)가 제안한 고유치를 이용한 TSP 식을 설명한다.

3.1 고유치를 이용한 TSP식

부품의 수가 10개 이상에서도 유효한 값을 얻기 위해, 고유치를 이용하여 Hopfield와 Tank식의 수렴성을 살피고, 안정된 수렴을 위해 변수들을 고치겠다. 고유치를 얻기 위하여 연결행렬에 대한 고유치의 형태를 D 항을 제거하여 계산하겠다. 그 이유로는 D 항의 값이 일정치 않고 식(3)의 처음 세 가지 항이 강한 제약조건(Strong Constraint)이어서 D 항보다 네트워크가 유효한 값에 정착(Settle down)할 수 있게 하는데 영향을 많이 미치기 때문이다. 또, C1과 C2가 같은 조건의 항으로 값이 같기 때문에 $A=B$ 라 보고, $N'=N$ 이라 가정하면, 연결행렬 $T_{x,i,y,j}$ 는

$$T_{x,i,y,j} = -A\delta_{x,y}(1 - \delta_{i,j}) - A\delta_{i,j}(1 - \delta_{x,y}) - C \quad (7)$$

이 되며, 이 연결행렬 $T_{x,i,y,j}$ 에 대한 고유치는 $\lambda_1 = -CN^2 - 2A(N-1), \lambda_2 = 2A, \lambda_3 = -A(N-2)$ 로 나타낼 수 있다. 이 세 가지 고유치의 각 요소들을 독립시켜, 최적 수렴을 위해 A, B, C 그리고 N 의 상호 관계 값들을 조절하여 유효한 값을 얻을 수가 있다. 첫 번째로 C 항을 독립시키기 위해, 여분의 상수 $2A_1$ 을 각각의 연결행렬에 넣는다. 그 다음 λ_1 이 $-CN^2$ 이 되게 하기 위해 각각의 연결행렬에 $2A(N-1)/N^2 + 2A_1/N^2$ 을 넣고, 그 고유치들을 다시 정리하면, $\lambda_1 = -CN^2, \lambda_2 = -2(A - A_1), \lambda_3 = -AN + 2(A - A_1)$ 이 된다.

식(7)의 연결행렬 $T_{x,i,y,j}$ 에 D 항을 포함시키고, 고유치에 의해 변형되고 생성된 변수로 재정리하여 연결행렬 $T_{x,i,y,j}$ 와 외부자극 $I_{x,i}$ 를 정리하면

$$T_{x,i,y,j} = -A\delta_{x,y}(1 - \delta_{i,j}) - B\delta_{i,j}(1 - \delta_{x,y}) - 2A_1\delta_{x,y}\delta_{i,j} - C + \frac{2(AN - A + A_1)}{N^2} \quad (8)$$

$$- Dd_{x,y}(\delta_{j,i+1} + \delta_{j,i-1})$$

$$I_{x,i} = CN \quad (9)$$

이 된다. 식(8)과 식(9)에서 N 은 실제 부품 수이며, B 는 항상 A 와 같다.

고유치를 이용한 식은 많은 부품을 연결할 때 유효한 연결 값을 100%낼 수 있다는 장점은 있지만, 위의 과정에서 보듯이 거리의 항(D 항)은 그 자신의 특수성-거리의 정도가 일정하지 않아 다루기 힘들다는 것 때문에 고려하지 않아서 일정치 않은 거리를 갖는 공정의 경우에는 최적의 연결을 하지 못하는 불안정성을 갖는다⁽⁸⁾. 실제로 Algorithm I은 이 연구에서 요구하는 최단의 연결을 완벽히 만족시켜 주지 못해 그 유용성에 문제가 있다.

4. Algorithm II

앞의 Algorithm I에서의 문제점을 해결하기 위하여 우리는 거리항에 걸리는 하중(Load)을 분할하는, 즉 결합 부품간의 최적 연결에는 거리항에 강한 제약조건을 주는 WMP식과 전체 부품간의 최적 연결에는 TSP식을 사용하는 Algorithm II를 제안하고 이를 WM-TS Problem (Weighted Matching-Traveling Salesman Problem)

이라 하겠다. 이 장에서는 WMP식을 결합 부품간의 연결에 사용할 수 있도록 새로이 제안하겠다. 그리고 TSP식을 전체 부품간의 연결에 사용할 수 있게 하여 WM-TS Problem에 대하여 설명하고자 한다.

4.1 Weighted Matching Problem

WMP란 한 부품이 다른 부품에 연결하여 이들의 최단 거리를 찾는 문제로서, TSP식보다 상대적으로 간단하며 거리의 항에 강한 제약조건을 줄 수 있는 알고리즘이다. WMP의 조건은 다음과 같다^(6,10)

- 1) 각 부품은 오직 다른 한 부품에만 연결되어야 한다;
- 2) 쌍으로 연결된 거리의 총합이 최소가 되어야 한다는 것이다.

이 조건을 만족하는 에너지 함수는

$$E = \sum_{x < y} d_{x,y} V_{x,y} + \frac{\gamma}{2} \sum_x (1 - \sum_y V_{x,y})^2 \quad (10)$$

이고, 여기서 x, y 는 각각의 부품들을 나타내며, γ 는 Penalty 상수라 정의한다. Hopfield model에서는 $V_{x,y} = V_{y,x}$ 가 성립되므로 $x < y$ 인 경우만 고려하여 계산을 줄일 수 있다. 식(10)의 우변 첫 번째 항은 거리에 관한 항이고, 두 번째 항은 각 행/열 중 오직 하나만이 연결 되어야 하는 제약조건을 나타내는 항이다.

결합 부품간의 최적 연결을 위해 식(10)를 미분하면

$$dE_{x,y} = (d_{x,y} - \gamma + \gamma \sum_{k \neq y} V_{x,k} + \gamma \sum_{k \neq x} V_{y,k}) \cdot dV_{x,y} \quad (11)$$

이 된다. 또, 각 뉴런의 거동에 관한 미분 방정식을 만든다면

$$\frac{dU_{x,y}}{dt} = \sum_{n=1}^N \sum_{m=1}^N T_{x,y,n,m} V_{n,m} - \frac{U_{x,y}}{\tau} + I_{x,y} \quad (12)$$

$$(V_{x,y} = g(U_{x,y}) = \frac{1}{2}(1 + \tanh(\lambda U_{x,y})))$$

와 같이 표현된다.

만일 2장의 Hopfield Model에서 U_i 가 비동기적 (Asynchronous : 뉴런들의 값이 무작위로(Random) 갱신되는 경우)으로 갱신되고, 연결행렬 $T_{i,j}$ 가 대칭적이면, 식(1)로부터 유도되는 에너지 함수 (Energy

Function)는

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N T_{i,j} V_i V_j - \sum_{i=1}^N I_i V_i \quad (13)$$

과 같이 나타낸다. 이 에너지 함수는 Liapunov 함수, 목적 함수(Objective Function)라고도 하며, 중요한 특성으로는 시스템은 언제나 감소(또는 일정한 상수로 수렴)하는 방향으로 나아간다는 것이다. 그러므로 초기의 입력값이 점점 감소하여 에너지 최소점(Energy Minima)에 도달하면 일정 상수가 되고, 이 에너지 값이 우리가 원하는 해가 될 수 있다. 이 에너지 함수를 이용하여 i 번째 뉴런에 대한 거동 방정식으로 쓰다면

$$\frac{dU_i}{dt} = -\frac{U_i}{\tau} = -\frac{\partial E}{\partial V_i} \quad (14)$$

과 같이 표현할 수 있으며 식(12)를 위와 같은 형태로 하면

$$\frac{dU_{x,y}}{dt} = -\frac{U_{x,y}}{\tau} - \frac{\partial E}{\partial V_{x,y}} \quad (15)$$

로 나타낼 수가 있다. 식(11)을 식(15)에 대입하면, 미분 방정식에 대한 연결행렬 $T_{x,y,n,m}$ 과 외부자극 $I_{x,y}$ 는 아래식들과 같다.

$$T_{x,y,n,m} = -\gamma(\delta_{m,x} + \delta_{x,n}) - \gamma(\delta_{y,n} + \delta_{y,m}) \quad (16)$$

$$I_{x,y} = \gamma - d_{x,y} \quad (17)$$

본 연구에서, 좀 더 빠르고 안정된 최적의 결합을 얻기 위하여 연결행렬에 각 점의 상호 연결 거리항 $\alpha(\gamma - d_{x,y,n,m})$ 을 새로 추가하여, 아래와 같은 연결행렬 $T_{x,y,n,m}$ 로 다시 정의하였다.

$$T_{x,y,n,m} = -\gamma(\delta_{m,x} + \delta_{x,n} + \delta_{y,n} + \delta_{y,m}) + \alpha(\gamma - d_{x,y,n,m}) \quad (18)$$

식(18)에서

$$d_{x,y,n,m} = d_{x,y} + d_{n,m}$$

$$\alpha = \begin{cases} 1 & (x \neq y, x \neq m, x \neq n, y \neq n, y \neq m) \\ 0 & (\text{그외의 경우}) \end{cases}$$

이다. α 는 상호 연결 변수라 정의한다.

4.2 WM-TS Problem

앞 절에서 보듯이 WMP식은 그 식의 간결성에 의해서 거리의 항에 강한 제약조건을 걸 수가 있으나 전체 연결 경로는 얻을 수 없다. TSP식은 식의 복잡성에 의하여 거리의 항에 대한 조건이 상대적으로 적어서 우리가 원하는 최적 연결은 어려우나, 전체 연결경로를 결정할 수 있는 장점을 가지고 있다. 그래서 WMP식과 TSP식의 장점을 이용하면, 우리가 원하는 알고리즘을 얻을 수 있다. 즉, 결합 부품간의 최적 연결에는 WMP식을 사용하고, 전체 부품간의 최적 연결에는 WMP에서 얻어진 값들을 이용한 TSP를 사용한다.

WMP식은 앞 절에서와 같이 사용하고, TSP식은 고유치를 이용한 방법과 기본적으로 같은 방법으로 구하나, 전체 부품간의 최적 연결에 알맞은 거리항을 만들기 위해 이론적 식(19), (21)과 식(8)에서 각 항들의 A, B, C 그리고 D 등 값에 여러 비례 상수들을 적용하는 실험에 의하여 얻어진 식(20)을 사용한다.

$$\frac{dU_{x,i}}{dt} = \sum_{y=1}^N \sum_{j=1}^N T_{x,i,y,j} V_{y,j} - \frac{U_{x,i}}{\tau} + I_{x,i} \quad (19)$$

$$(V_{x,i} = g(U_{x,i}) = \frac{1}{2}(1 + \tanh(\lambda U_{x,i})))$$

$$T_{x,i,y,j} = -2A\delta_{x,y}(1 - \delta_{i,j}) - 2B\delta_{i,j}(1 - \delta_{x,y}) \quad (20)$$

$$- A_1\delta_{x,y}\delta_{i,j} - C + \frac{4(AN - A + A_1)}{N^2}$$

$$- Dd_{x,y}(\delta_{j,i+1} + \delta_{j,i-1})$$

$$I_{x,i} = CN \quad (21)$$

5. 시뮬레이션

Algorithm II에서 언급한 WM-TS Problem식을 이용하여 부품 조립문제를 해결하고자 한다. Hopfield와 Tank 식은 여러 연구자들에 의해 시뮬레이션 되었으며^{1,4,11)} 여기서는 거론하지는 않겠다. 그 대신 이론적 고유치를 이용한 Algorithm I과 WM-TS Problem식을 이용한 Algorithm II의 시뮬레이션을 비교 분석하여 Algorithm II의 우수성을 보이겠다.

5.1 Algorithm I의 시뮬레이션

Algorithm I은 $-U_{x,i}/\tau$ 항을 소거한(왜냐하면, A, A_1, C, D 값이 상당히 크다면 이 항의 값을 보상해 준다.) 미분 방정식은 식(4)을 이용하였으며, 출력함수 $g(U_{x,i})$ 로 쌍곡선 탄젠트를 쓰는 대신 더욱 안정적이고 빠른 수렴을 위해 아래와 같은 계단 선형함수를 새로 만들어 사용하였다.

$$g(U_{x,i}) = \begin{cases} 0 & U_{x,i} \leq -0.5 \\ U_{x,i} + 0.5 & -0.5 < U_{x,i} < 0.5 \\ 1 & U_{x,i} \geq +0.5 \end{cases} \quad (22)$$

그 외의 조건으로, $A=8, A_1 = \frac{31}{32}A, C = \frac{A}{30}, D = \frac{AN}{80}$ 을 사용했고, $A, A_1 \dots$ 등에 의해서 dt 는 작은 값을 갖는다(10개의 도시 : 0.02s, 50개의 도시 : 0.005s). 초기 값 $V_{x,i}$ 는 무작위로 얻었지만, $V_{x,i} \cdot e^1 \approx N$ 을 유지시켰다. 연결행렬 $T_{x,i,y,j}$ 와 외부자극 $I_{x,i}$ 의 식은 식(8)과 식(9)을 이용한 경우와 식(20)과 식(21)을 이용한 경우로 구분하여 시뮬레이션을 하였고, 그 결과는 Fig.3과 같다.

5.2 Algorithm II의 시뮬레이션

Algorithm II의 시뮬레이션 순서와 조건은 다음과 같다.

1) WMP 이용한 Data 획득 : 미분 방정식은 식(12)에서 $-U_{x,y}/\tau$ 항을 소거하였고, 출력함수 $g(U_{x,y})$ 로 식(22)을 변형하여 사용했다. 그 외의 조건으로, $\gamma=1000, dt=0.02$ 그리고 초기값 $V_{x,y}=0.5$ 로 하여 뉴런의 갱신을 무작위로 하였다.

2) WMP에서 얻은 값을 Algorithm II의 TSP식에 적용

: Algorithm II의 TSP식(식(19), (20), (21))을 이용하였고 그 식에 대한 변수값들은 $D=0.01 \sim 0.02$, 초기 값 $V_{x,i} = 0$ 으로 한 것을 제외하고 Algorithm I의 변수 값 정의와 같으며, 그 결과는 Fig.4와 같다.

5.3 결과

Fig. 3과 Fig. 4는 각각 부품의 개수가 12개일 경우의 시뮬레이션 결과이다. Fig.3은 Algorithm I을 이용한 식(8)과 Algorithm II의 식(20)을 이용한 결과를 나타낸 것이고, Fig.4는 Algorithm II을 이용한 시뮬레이션 결과를 나타낸 것이다. 이들 그림에서 X축은 실행 횟수

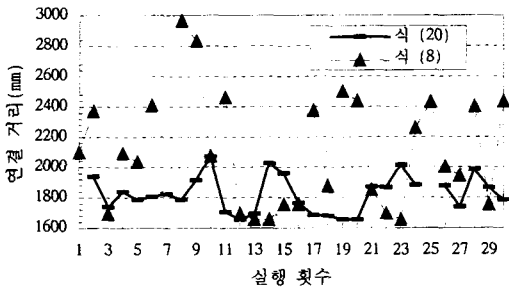


Fig. 3 식(8)과 식(20)의 결과값

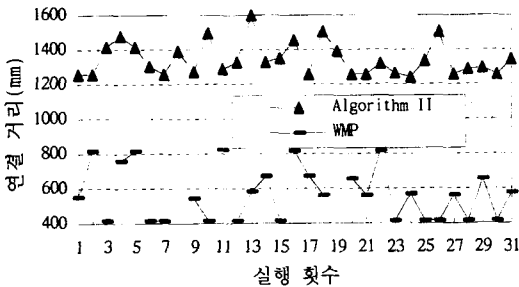


Fig. 4 Algorithm I의 결과값

Table. 1 30번 실행했을 때의 시뮬레이션 결과값

	Algorithm I		Algorithm II
	식 (10)	식 (20)	
최 적 거 리(mm)	1654	1654	1253
최적 거리를 나타낸 횟수(회)	5	7	9
에러 발생 횟수(회)	1	2	0
최 대 거 리(mm)	2963	2015	1603
총 평균 거 리(mm)	2107.69	1823.26	1344.73
연 산 시 간 (Sec)	20	16	12

(30회)이고, Y축은 연결 거리(mm)를 나타낸다. Fig. 3과 Fig. 4는 식(20)을 이용한 식이 식(8)을 이용한 식보다 연결거리의 변화 폭이 월등히 적으며, Algorithm II를 사용할 경우는 식(20)보다 더욱 폭이 적음을 보여준다. 여기에 대하여 Table. 1은 Algorithm I과 Algorithm II의 최적 거리, 최적 거리를 나타낸 횟수, 에러 발생 횟수, 최대 거리, 총 평균 거리 그리고 연산 시간을 비교한 것을 나타낸 것이다. Table. 1을 보면 Algorithm II를 이용한 방법이 시간, 최적 거리를 나타낸 횟수, 최적 거리와 평균 거리의 차이 등에서 Algorithm I

보다 월등하게 좋음을 알 수 있다.

6. 결 론

본 논문에서는 일정하지 않은 부품 개수와 거리를 갖는 부품 조립 공정에서 결합의 최적 연결 경로를 찾기 위해 신경 회로망의 Hopfield Model을 이용한 WM-TS Problem식을 제안하였다.

제안된 식의 시뮬레이션을 통하여 다음과 같은 특징들을 얻어냈다.

- 1) 안정성 획득 : 거리항에 의해 최적의 해를 찾지 못하는 불안정성을 거리항을 분할하여 해소하였다.
- 2) 유용성 획득 : 최적 거리 횟수, 최적 거리와 평균 거리의 차이 등 여러 가지 다양한 조건에서도 아주 우수한 최적(Optimal) 또는 준 최적상태의 값을 가진다.
- 3) 수행 시간의 단축 : 연산 시간을 Algorithm I에 비해 1/3을 더 줄였다.

서론에서 언급했듯이 본 논문에서는 네가지 고려사항 중 첫 번째와 두 번째를 영상 인식 분야로 생각하여 논하지 않았다. 만약 영상 인식 부분과 본 논문의 연구를 결합한다면, 영상 인식 부분에서는 작업 조건과 부품의 특성(종류와 개수)을 자동적으로 인식하여 파악하고, 이를 이용하여 각 부품을 최적하게 연결하는 신경 회로망 알고리즘에 적용하므로써 복잡한 공정에서도 능동적으로 대처하는 부품 조립 시스템을 구축할 수 있을 것이다. 또한, 조립 공정은 아니더라도 경로 결정이 필요한 작업(예로써, PCB검사를 위한 카메라 경로의 결정)에서도 아주 효율적으로 사용될 수 있을 것이다.

참 고 문 헌

1. Freemam, J.A. and Skapura, D.M., "Neural Networks", Addison-Wesley Pub., pp. 141-168, 1992.
2. Ali, M.K.M. and Kamoun, F., "Neural Network for Shortest Path Computation and Routing in Computer Networks", IEEE Trans. N. N. vol. 4, no. 6, pp. 941-954, 1993.
3. Tagliarini, G.A., Crist, J.F. and Page,

- E.W., "Optimization Using Neural Networks", IEEE Trans. computers, vol. 40, no. 12, pp. 1347-1358, 1991.
4. Aiyer, S.V.B., Niranjan, M. and Fallside, F., "A Theoretical Investigation into the Performance of the Hopfield Model", IEEE Trans. N. N, vol. 1, no. 2, pp. 204-215, 1990.
 5. Carvalho, L.A.V. and Barbosa, V.C., "A TSP Objective Function that Ensures Feasibility at Stable Points", in Proc. INNC90, vol. 1, pp. 249-253, 1990.
 6. Kung, S.Y., "Digital Neural Networks", Prentice Hall, pp. 321-325, 1993.
 7. Aiyer, S.V.B., Niranjan, M. and Fallside, F., "On the Optimization Properties of the Hopfield Model", in Proc. INNC90, vol. 1, pp. 243-248, 1990.
 8. Freemam, J.A., "Simulating Neural Networks with mathematica", Addison-Wesley Pub., pp. 143-175, 1994.
 9. Wang, Jun, "A Deterministic Connectionist Machine the Traveling Salesman Problem", ICSMC90, vol. 1, pp. 373-375, 1990.
 10. Hertz, J., Krogh, A. and Palmer, R.G., "Introduction to the Theory of Neural Computation", Addison-Wesley Pub., pp. 71-87, 1991.
 11. Joppe, A., Cardon, H.R.A. and Bioch, J.A., "A Neural Network for Solving the Traveling Salesman Problem on the Basis of City Adjacency in the Tour", in Proc. INNC90, vol.1, pp. 254-257, 1990.