Journal of the Korean Institute of Industrial Engineers
Vol. 30, No. 3, pp. 175-180, September 2004.

그리드 컴퓨팅을 이용한 기계-부품 그룹 형성

이종섭[1†] · 강맹규[2]

[1]우송대학교 IT(경영정보)학과 / [2]한양대학교 산업공학과

Machine-Part Grouping Formation Using Grid Computing

Jong-Sub Lee[1] · Maing-Kyu Kang[2]

[1]Department of Internet Science Technology, Woosong University, Daejeon, 300-718
[2]Department of Industrial Engineering, Hanyang University, Ansan, 425-791

The machine-part group formation is to group the sets of parts having similar processing requirements into part families, and the sets of machines needed to process a particular part family into machine cells using grid computing. It forms machine cells from the machine-part incidence matrix by means of Self-Organizing Maps(SOM) whose output layer is one-dimension and the number of output nodes is the twice as many as the number of input nodes in order to spread out the machine vectors. It generates machine-part group which are assigned to machine cells by means of the number of bottleneck machine with processing part. The proposed algorithm was tested on well-known machine-part grouping problems. The results of this computational study demonstrate the superiority of the proposed algorithm

***Keywords*:** grid computing, machine cell, part family, clustering, self-organizing maps

1. 서 론

전기와 같이 필요할 때에는 어디서나 사용이 가능하듯이 컴퓨터의 CPU, RAM, 그리고 저장장치 등의 컴퓨팅 자원이 필요할 때 필요한 만큼 사용하는 그리드 컴퓨팅이 전세계적으로 관심을 모으고 있다. 그리드 컴퓨팅 환경은 이전에는 많은 컴퓨팅 자원이나 계산시간의 소요로 인하여 도전하기 어려운 문제들을 해결하는 데 거의 무한의 컴퓨팅 자원을 제공하고 있다. 제조분야에서 기계-부품 그룹 형성문제도 그룹을 형성해야 할 기계의 수와 부품의 수가 증가하는 경우에 기하급수의 컴퓨팅 자원을 요구하는 문제이다.

셀 생산방식은 기계-부품 그룹을 형성하는 문제와 이렇게 형성된 기계-부품 그룹을 공장 내에 배치하는 문제, 그리고 기계-부품 그룹 내의 기계를 배치하는 문제 등3단계로 이루어 진다. 이 중에서 첫번째 단계로서 기계-부품 그룹형성(Machine-Part Group Formation)은 부품들의 이동비용을 최소화하기 위해 한 기계 그룹에서 특정 부품들만을 생산하도록 m 대 기계와 p개 부품을 k개 그룹으로 묶는다. 먼저 기계들을 k개 그룹으로 형성하고, 그 다음에 부품들을 기계 그룹에 할당하여 k개 기계-부품 그룹을 형성한다.

기계별로 처리할 수 있는 부품과 그렇지 않은 부품을 나타내기 위하여 $m \times p$ 기계-부품행렬(Machine-Part Incidence Matrix)을 사용하는데 <Figure 1>과 같이 i번째 기계가 j번째 부품을 처리하는 데 사용되면1의 값을 가지고, 그렇지 않으면 0의 값을 가진다(<Figure 1>에서 0은 생략함). <Figure 1>의 기계-부품 그룹은 <Figure 2>와 같이 대각선을 따라 표현할 수 있는데 이것을 블록대각(Diagonal Blocks)이라고 한다.

그룹이 얼마나 잘 형성되었는지를 측정하는 척도에는 여러 가지가 있는데 Kumar *et al.*(1990)가 제시한 그룹화 효율(GF; Grouping Efficacy)은 식(1)과 같이 계산한다.

$$GF = \frac{\text{블록대각구조 안 1의 수}}{\text{블록대각구조의 면적} + \text{블록대각구조밖 1의 수}} \times 100\% \quad (1)$$

GF는 0과 100 사이의 실수값을 가지며, 이 값이 클수록 그룹 형성이 잘된 것이다. GF는 Grouping Efficiency, Grouping Index,

† 연락저자 : 이종섭 교수, 300-718 대전시 동구 자양동 17-2 우송대학교 IT(경영정보)학과, Fax : 042-630-9859, E-mail : ljs@woosong.ac.kr
2003년 11월 접수; 2004년 2월 수정본 접수; 2004년 5월 게재 확정.

Grouping Measure 등의 척도와 비교하여 1의 밀집도가 높은 블록대각 구조를 더 잘 나타낼 수 있는 바람직한 성질을 가지고 있다(Sandbothe, 1998; Sarker, 2001).

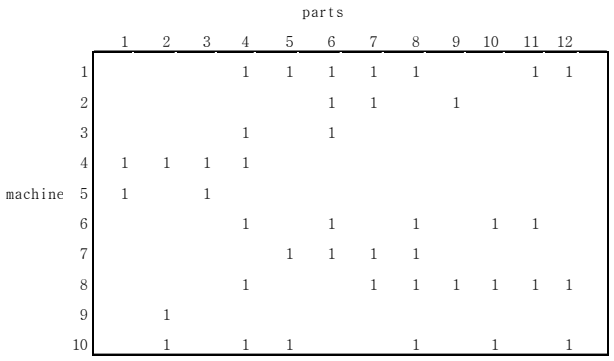

Figure 1. Illustration of Machine-Part Incidence Matrix (Ben Arieh *et al*., 1994).

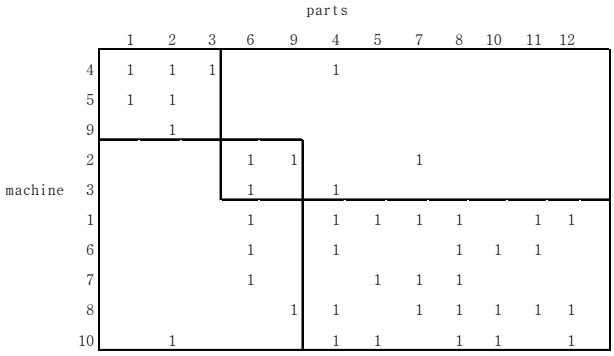

Figure 2. Result of Machine-Part Grouping(Viswanathan, 1996).

기계-부품 그룹 형성 문제는NP-Complete이므로(Kusiak, 1990) 그룹을 형성해야 할 기계의 수와 부품의 수가 많은 경우에는 많은 계산량이 소요된다. 따라서, 최적화 해법보다는 발견적 해법(Heuristic Algorithm)을 많이 사용하고 있는데 King(1980)의 배열을 사용한 방법, McAuley(1972)의 유사계수법, Gupta *et al*.(1995)의 유전해법, Kaparthi *et al*.(1993)의 신경망을 사용한 방법 등 여러 가지가 발표되어 있다.

최근에는 신경망을 기계-부품 그룹 형성 문제에 적용하는 연구가 활발히 이루어지고 있다. Kaparthi *et al*.(1993)는 Carpenter *et al*.(1988)가 제안한 ART 신경망(Adaptive Reasonance Theory)을 사용했고, Kulkarni *et al*.(1995)는 Kohonen(1984)이 제안한 자기조직화 신경망(SOM; Self-Organizing Maps)을 사용하였다.

Kulkarni *et al*.(1995)는 부품을 입력벡터로 2차원 출력노드에 출력하였으나 이를 다시 k개 부품 그룹으로 형성하는 명확한 방법을 제시하지 못하고, 그 가능성만 언급하였다. 일반적으로, SOM은 입력벡터를 1차원 출력노드 또는 2차원 출력노드에 출력시키지만 이를 원하는 k개 그룹으로 형성하지는 못한다. 따라서, SOM의 출력결과를 다시 k개 그룹으로 형성하는 단계가 추가로 필요하다.

본 연구에서는 그리드 컴퓨팅을 이용한 기계부품 그룹 형성 알고리듬을 제안한다. 그리드 컴퓨팅을 이용하여 학습률, 입력벡터 순서 등의 파라미터 연구를 수행하였다. 이 연구의 결과로 형성된 SOM에 의하여 1차원 출력노드에 기계를 일렬로 펼친다. 이 기계들 사이에 유클리디언 거리가 최대가 되는 지점을 분리하여 기계 그룹을 형성한다. 블록대각 밖에 있는 1인 예외요소의 수가 최소가 되도록 부품들을 기계대수에 따라 오름차 순으로 정렬된 기계 그룹에 할당하여 기계부품 그룹을 형성한다.

2. 자기조직화 신경망의 일반적 고찰

SOM은 Kohonen(1984)에 의해 제시된 신경망(Neural Network) 모델이다. SOM의 구조는 <Figure 3>과 같이 m개 입력노드로 이루어진 입력층과 n개 출력노드로 이루어진 출력층의 2개 층으로 구성된다.

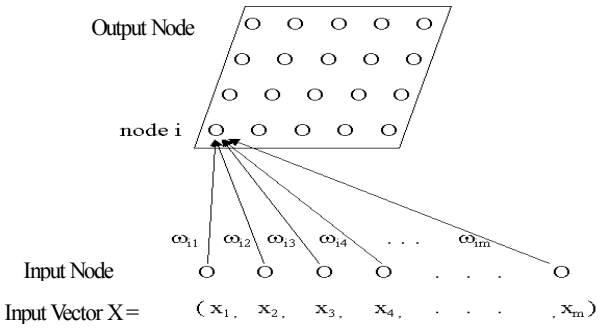

Figure 3. General structure of SOM.

입력층은 입력벡터를 입력 받아 이들을 출력층에 사상(Mapping)한다. 출력층은 1차원 또는 2차원 구조를 사용한다. 출력층에 있는 출력노드의 수는 각 입력벡터가 서로 다른 출력노드에 펼쳐질 수 있도록 입력벡터의 수보다 충분히 큰 수로 하거나 원하는 그룹의 수로 정할 수 있다 출력층의 각 노드에는 입력벡터가 사상된다.

입력층의 모든 노드와 출력층의 모든 노드는 연결되어 있고, 출력노드 i와 입력노드 j 사이의 연결선은 연결강도(Weight) w_{ij}를 가진다. 연결강도는 초기에는 임의로 주어지지만 입력벡터에 따라 조절된다. 각 입력벡터 X에 대하여 이와 가장 유사한 출력노드인 승자노드 i*를 결정하는데, 이는 식 (2)와 같이 연결강도벡터와 입력벡터 사이의 거리 D_i를 계산하여 가장 작은 출력노드로 정한다.

$$D_i = \sqrt{(x_1 - w_{i1})^2 + (x_2 - w_{i2})^2 + \cdots + (x_m - w_{im})^2} \quad i = 1, \cdots, n \qquad (2)$$

$$w(new)_{ij} = w(old)_{ij} + a(x_i - w(old)_{ij}) \quad i \in N_{i^*}(\delta)$$
$$j = 1, \cdots, m \qquad (3)$$

여기에서 $N_{i^*}(\delta)$는 승자노드 N_{i^*}로부터 δ만큼 떨어진 이웃 노드 집합이다.

승자노드의 앞과 뒤에 위치한 노드를 이웃(neighbor)이라 하는데 각 입력벡터 X에 대하여 이웃의 범위 δ와 학습률 $\alpha(0< \alpha \leq 1)$를 감소시키면서 이웃의 범위가 승자노드 자신이 될 때까지 승자노드와 그 이웃노드의 연결강도를 식 (3)과 같이 조절한다. 여기에서 $w(old)_{ij}$는 조절되기 전의 연결강도이고, $w(new)_{ij}$는 조절된 후의 연결강도이다.

학습률 α는 입력벡터와 기존의 연결강도벡터와의 차이를 조정하는 비율로서 학습이 진행됨에 따라 줄여나간다. 일반적으로 학습률이 너무 크면 제대로 학습이 되지 않고, 너무 작으면 학습시간이 오래 걸린다(Kohonen, 1984). SOM의 학습 알고리듬은 다음과 같다.

절차 1: 연결강도를 초기화한다. 학습률과 이웃의 범위를 정한다.
절차 2: 입력층에 하나의 입력벡터를 입력한다.
절차 3: 입력벡터와 연결강도벡터 사이의 거리 D_i를 식 (2)와 같이 계산한다.
절차 4: 하나의 승자노드를 결정한다.
절차 5: 학습규칙에 따라 연결강도를 식 (3)과 같이 조정한다.
절차 6: 이웃의 범위와 학습률을 감소시키면서 이웃의 범위가 승자노드 자신($\delta = 0$)이 될 때까지 절차 2에서 절차 5까지 반복한다.

3. 제안하는 알고리듬

본 연구에서는 SOM의 출력층은 1차원으로 단순화시키고 출력노드 수는 입력벡터가 충분히 펼쳐질 수 있도록 입력벡터 수 m의 두 배인 $2m$으로 한다. 이와 같은 출력노드 수는 기존 해법과 비교하는 문제들을 모두 실험한 결과를 바탕으로 결정하였다. 입력벡터 수가 m, $1.5m$인 경우에 입력벡터가 출력노드에 대응하는 입력벡터의 순서가 실험에 따라 변경되는 것을 발견하였으나 입력벡터 수가 $2m$, $2.5m$, $3m$ 이상인 경우는 이와 같은 현상을 발견하지 못하였다. 따라서 출력노드 수는 입력벡터 수가 최소인 $2m$으로 결정하였다. SOM에서 학습률과 이웃의 범위는 학습의 결과에 상당한 영향을 미치기 때문에 실험을 통해 결정하였다. 학습률은 초기에는 0.1 ~ 0.4로 하고, 학습이 진행됨에 따라 $(1-r/R)$비율로 줄여나간다(여기에서 r는 반복수이고, R는 총 반복수로서 기계대수와 같음). 이웃의 범위는 초기에는 출력층의 모든 노드를 포함하고 학습이 진행됨에 따라 이웃의 범위를 하나씩 줄여나가 마지막에는 승자노드만을 포함하도록 한다. 본 연구에서 사용하는 SOM의 학습 방법은 2장에서 서술한 일반적인 절차를 따른다.

본 연구에서는 기계를 1차원 출력노드에 출력하는 SOM을 구성하고, 그 결과를 이웃한 기계들 사이의 유클리디언 거리를 최대로 하는 지점을 분리하여 원하는 k개 기계 그룹을 형성한다. 예외요소의 수가 최소가 되도록 부품들을 형성된 k개 기계 그룹에 할당함으로써 k개 기계-부품 그룹을 형성할 수 있다.

본 논문에서 제안하는 알고리듬의 절차는 다음과 같다.

절차 1: SOM의 입력노드 수는 부품개수 p와 같고, 입력벡터는 기계로 한다. 출력노드 수는 기계대수 m의 두 배인 $2m$으로 한다.
절차 2: 두 인접한 기계 l과 l+1 사이의 유클리디언 거리 D_l를 계산한다.
절차 3: 계산된 유클리디언 거리중에서 가장 큰 지점을 분리하여 기계 그룹을 형성한다. 만약 유클리디언 거리가 같은 경우에는 모두 분리하여 두 개 이상의 기계 그룹을 형성한다. 여기에서 기계 그룹 번호는 기계대수에 따라 오름차순으로 번호를 정하였다.
절차 4: 하나의 기계 그룹에서 완전히 처리되는 부품은 그 기계 그룹에 할당한다.
절차 5: 두 개 이상의 기계 그룹에서 처리되는 부품들은 각각 예외요소의 수가 가장 적은 기계 그룹에 할당하여 기계-부품 그룹을 형성한다.
절차 6: 형성된 기계-부품 그룹의 GF를 계산한다.
절차 7: 그룹 수 g를 하나씩 증가시키면서 원하는 그룹 수 k에 도달하거나 더 이상 GF를 개선하지 못할 때까지 절차 3에서 6까지를 반복한다.

4. 수치예제

Ben Arieh *et al.*(1994)가 제시한 10대의 기계와 12개의 부품이 있는 <Figure 1>의 예제에 대하여 제안하는 알고리듬을 적용하면 다음과 같다.

절차 1: 본 예제에서 사용된 SOM 구조는 <Figure 4>와 같이 입력노드와 출력노드의 수가 각각 12, 20이다. 출력층에 사상된 기계의 순서는 <Figure 5>의 왼쪽에 있는 순서(위에서 아래로)와 같이 4, 5, 9, 6, 8, 1, 10, 7, 2, 3이다.

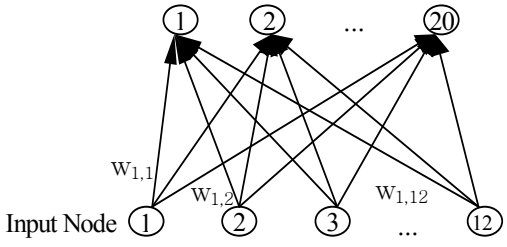

Figure 4. Suggesting structure of SOM.

절차 2: 기계 l 과 l+1 사이의 유클리디언 거리는 <Figure 5>의 오른쪽에 있는 D_l이다.

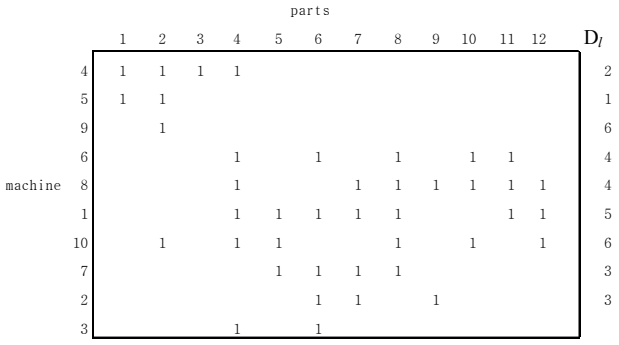

Figure 5. Order of machine on output node.

절차 3: 각 출력노드에 있는 기계l 과 l+1 사이의 유클리디언 거리가 가장 큰(D_3=6) 기계 9와 6 사이, (D_7=6) 기계 10과 7 사이를 분할하여 <Figure 6>의 실선과 같이 나누어 (4,5, 9), (7,2,3), (6,8,1,10)을 각각 기계 그룹1, 2, 3이라 한다.

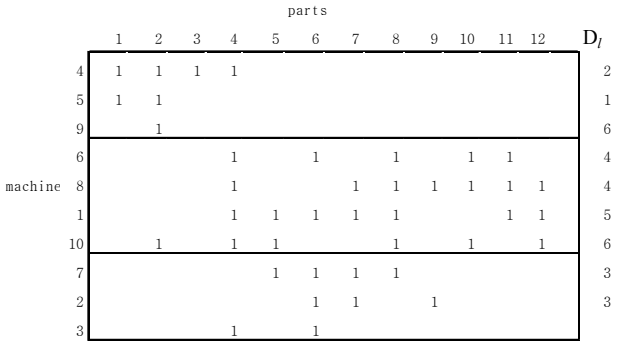

Figure 6. Result of three machine group divided by Euclidian Distan.

절차 4: 하나의 기계 그룹만으로 처리되는 부품1, 3, 10, 11, 12를 우선적으로 기계 그룹에 할당한다. 두 개 이상의 기계 그룹에서 처리되는 부품2, 4, 5, 6, 7, 8, 9는 기계 그룹에 할당하는 것을 유보하고, 부품의 위치를 오른쪽 끝으로 이동시킨다(<Figure 7> 참조).

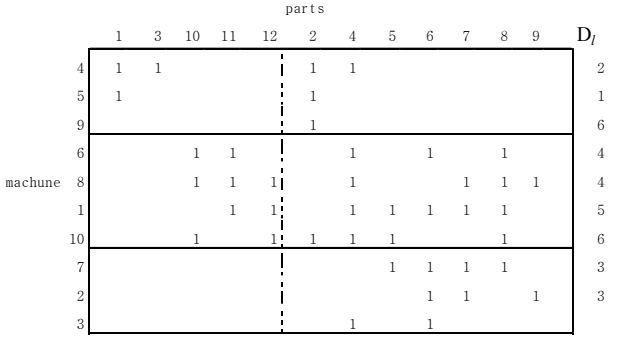

Figure 7. Parts assigned to machine cell(left side of the dotted line) and reserved parts (right side of the dotted line).

절차 5: 단계 4에서 할당이 유보된 부품2, 4, 5, 6, 7, 8, 9은 예외요소의 수가 최소가 되도록 기계 그룹1, 2, 3 에 할당한다. 부품 7과 같이 예외요소의 수가 같은 경우에는 순서에 따라 그 룹3이 아니라 그룹 2에 할당한다.

절차 6: <Figure 2>는 Viswanathan에 의하여 제시된 기계-부품 그룹 형성 결과로서 GF를 계산하면 53%이다. 제안하는 알고리듬에 의하여 형성된 <Figure 8>의 기계-부품 그룹 GF는 60%이다.

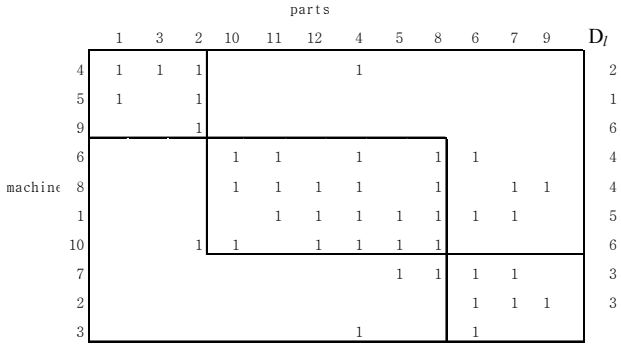

Figure 8. Result of Machine-Parts Grouping with three machine groups.

절차 7: 기계-부품 그룹의 수를 하나 증가시켜서4개일 때 기계 - 부품 그룹의 GF는 52%이므로 기계-부품 그룹 수를 증가시키지 않고 알고리듬을 정지한다.

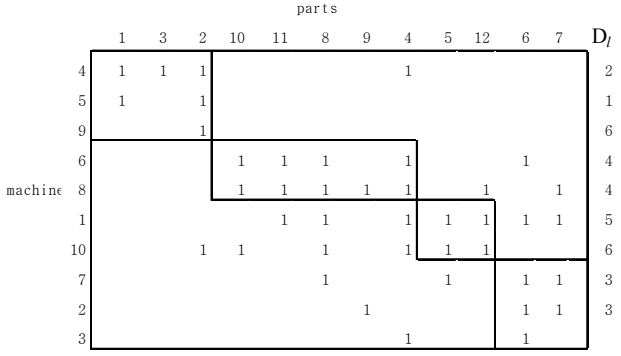

Figure 9. Result of Machine-Parts Grouping with four machine groups.

5. 실험 결과와 분석

K＊Grid 테스트베드(testbed)는 국내에 산재해 있는 Linux Cluster 시스템인 KISTI의 80노드, 포항공과대학교의 60노드, 부산대학교의 25노드를 초고속 네트워크로 연결하였다. Linux Cluster 시스템을 연동시키는 미들웨어 소프트웨어는 ANL (Argonne National Laboratory)에서 개발한 Globus Toolkit 2.4를 설치하였다. 본 연구에서는 제조업 분야의 기계-부품 그룹 형성 문제를 레드햇 리눅스 기반의K＊Grid 테스트베드에 처

음으로 적용하였다.

본 연구에서 제안하는 알고리듬은 국가 그리드 기반 구축 사업의 일부로 구축된 리눅스 기반의 K＊ Grid 테스트베드(KISTI의 80노드, 포항공과대학교의 60노드, 부산대학교의 25노드를 초고속 네트워크로 연결)뿐만 아니라 윈도우 기반의 단일 시스템에서도 동일한 파라미터(학습률, 입력벡터의 순서)로 실험을 수행하였다. 해는 레드햇 리눅스 기반의 K＊ Grid 테스트베드나 윈도우의 단일 시스템에서 모두 <Table 1>과 같았다.

실험에 사용된 문제는 기존 논문에서 많이 인용되는 예제로서 <Table 1>의 14개 문제이다. 입력은 <Figure 3>과 같이 기계로 하고, 출력노드 수 n과 초기 학습률은 각각 실험적으로 가장 좋은 해를 제공하는 값인 기계대수의 두 배인 $2m$과 0.1～0.4로 한다. 이웃의 범위는 처음에는 출력층의 모든 노드를 포함하도록 하다가 학습이 진행됨에 따라 하나씩 줄여나가서 마지막에는 승자노드만을 포함하도록 하였다.

<Table 1>은 기존 연구와 비교하여 14개의 문제 중에서 3개에 대해 본 연구에서 제안하는 해법이 우수한 해를, 나머지 11개에 대해서는 동일한 해를 보였다. 특히 Ben Arieh *et al.*(1994)에 의하여 제시된 10×12 문제에 대하여 기존의 해법보다 우수하다.

6. 결 론

본 연구에서는 기계 그룹을 형성하기 위하여 그리드 컴퓨팅을 이용하여 기계를 입력하는 SOM을 구성하였다. SOM의 구성은 그리드 컴퓨팅에 의하여 학습률, 입력벡터 순서 등의 파라미터 연구를 바탕으로 결정하였다. SOM의 수행결과로 일렬로 펼쳐진 n개 출력노드에서 기계를 사상(mapping)한다. 기계는 출력노드에 사상되지 않거나 하나가 사상된다. 하나의 기계벡터들이 사상된 출력노드는 사상된 기계들이 처리하는 부품들의 가공공정이 서로 유사한 정도에 따라 다시 k개 기계 그룹을 형성하는데, 이를 위하여 출력노드의 기계들 사이의 유클리디언 거리와 예외요소의 수를 최소화하도록 부품을 기계대수에 의하여 오름차순으로 정렬된 기계 그룹에 할당하여 기계-부품 그룹을 형성한다.

제안하는 알고리듬은 Ben Arieh *et al.*(1994)에 의해 제시된 10×12 문제를 포함한 14개의 문제 중에서 3개에 대해 기존의 해법보다 우수한 해를, 나머지 11개에 대해서는 동일한 해를 보였다.

제안하는 알고리듬은 기존의 방법과 비교하여 복잡한 연산을 사용하지 않기 때문에 규모가 큰 문제에 적용할 수 있다. SOM은 학습시간이 많이 소요되는 다른 신경망과는 달리 학습이 매우 빨리 수행되기 때문에 실시간 처리에 적합하다. 특히 그리드 컴퓨팅에서는 학습률, 초기 연결강도, 그리고 입력벡터의 순서와 같은 파라미터 연구를 진행할 수 있어서 더욱 더 좋은 해를 구할 수 있다.

참고문헌

Arvindi, B. and Irani, S. A.(1994), Principle component analysis for

Table 1. *GF* Comparison without machine duplication

Size (machine (parts)	Source of Problems	Existing Algorithms	Number of group	GF(%) of Existing Algorithms	GF(%) of Suggesting Algorithm
5×7	King *et al.* (1982)	King *et al.* (1982)	2	82.0	82.0
6×8	Kusiak *et al.* (1992)	Kusiak *et al.* (1992)	2	77.0	77.0
7×8	Kusiak *et al.* (1987)	Kusiak *et al.* (1987	3	85.0	85.0
7×11	Kusiak *et al.* (1987)	Kusiak *et al.* (1987	2	46.3	46.3
〃	〃	-	3	-	54.8*
8×20	Chandraseharan *et al.* (1986)	Crama *et al.* (1996)	3	85.0	85.0
10×8	Ham *et al.* (1985)	Arvindi *et al.* (1994)	3	74.0	77.0*
10×12	Ben Arieh *et al.* (1994)	Viswanathan (1990)	3	53.0	60.0*
15×10	Chan *et al.* (1982)	Chan *et al.* (1982)	3	92.0	92.0
10×20	Srinivasan *et al.* (1990)	Srinivasan *et al.* (1990)	4	100.0	100.0
16×30	Srinivasan *et al.* (1990)	Srinivasan *et al.* (1990)	4	68.0	68.0
20×35	Carrie (1973)	Kaparthi *et al.* (1992)	4	75.0	75.0
24×40	Chandraseharan *et al.* (1989)	Kaparthi *et al.* (1993)	7	100.0	100.0
40×100	Chandraseharan *et al.* (1987)	Kaparthi *et al.* (1993)	10	84.0	84.0

(－ : None produced from existing experiments, * : Superior to the result of existing thesis)

evaluating the feasibility of cellular manufacturing without initial machine-part matrix clustering, *International Journal of Production Research*, **32**(8), 1909-1938.

Ben Arieh, D. and Chang, P. T.(1994), An extension to p-median group technology algorithm, *Computers and Operations Research*, **21**(1), 101-114.

Carpenter, G. A. and Grossberg, S.(1988), The ART of adaptive pattern recognition by a self-organizing neural network, *IEEE Computer*, **21**(3), 77- 88.

Carrie, A. S.(1973), Numerical taxonomy applied to group technology and plant layout, *International Journal of Production Research*, **11**(1), 399-416.

Chan, H. M. and Milner, D. A.(1982), Direct clustering algorithm for group formation in cellular manufacturing, *Journal of Manufacturing Systems*, **1**(1), 65-75.

Chandrasekharan, M. P. and Rajagopalan, R.(1986), MODROC: An extension of rank order clustering for group technology, *International Journal of Production Research*, **24**(5), 1221-1233.

Chandrasekharan, M. P. and Rajagopalan, R.(1987), ZODIAC: An algorithm for concurrent formation of part-families and machine-cells, *International Journal of Production Research*, **25**(6), 835-850.

Chandrasekharan, M. P. and Rajagopalan, R.(1989), Groupability: An analysis of the properties of binary data matrices for group technology, *International Journal of Production Research*, **27**(6), 1035-1052.

Crama, Y. and Oosten, M.(1996), Models for machine-part grouping in cellular manufacturing, *International Journal of Production Research*, **34**(6), 1693-1713.

Gupta, Y. P., Gupta, M. C., Kumar, A., and Sundram, C.(1995), Minimizing total intercell and intracell moves in cellular manufacturing: A genetic algorithm approach, *International Journal of Production Research*, **8**(2), 92-101.

Ham, I., Hitomi, K., and Yoshida, T.(1985), *Group Technology: Production Methods in Manufacture*, Kluwer-Nijhoff, Boston, MA.

Kaparthi, S. and Suresh, N. C.(1992), Machine-component cell formation in group technology: A neural network approach, *International Journal of Production Research*, **30**(6), 1353-1367.

Kaparthi, S., Suresh, N. C., and Cervany, R. P.(1993), An improved neural network leader algorithm for part-machine grouping in group technology, *European Journal of Operational Research*, **69**(3), 342-356.

King, J. R.(1980), Machine-component group formation in production flow analysis: An approach using a rank order clustering algorithm, *International Journal of Production Research*, **18**(2), 213-232.

Kohonen, T.(1984), *Self-organization and association memory*, Springer, Berlin.

Kulkarni, U. R. and Kiang, M. Y.(1995), Dynamic grouping of parts in flexible manufacturing systems: A self organizing neural networks approach, *European Journal of Operational Research*, **84**(2), 192-212.

Kumar, C. S. and Chandrasekharan, M. P.(1990), Grouping efficacy: A quantitative criterion for goodness of block diagonal forms of binary matrices in group technology, *International Journal of Production Research*, **28**(2), 233-243.

Kusiak, A.(1990), *Intelligent Manufacturing Systems*, Prentice Hall, Englewood Cliffs, New Jersey.

Kusiak, A. and Cho, M.(1992), Similarity coefficient algorithms for solving the group technology problem, *International Journal of Production Research*, **30**(11), 2633-2646.

Kusiak, A. and Chow, W. S.(1987), Efficient solving of the group technology problem, *Journal of Manufacturing Systems*, **6**(2), 117-124.

McAuley, J.(1972), Machine grouping for efficient production, *The Production Engineer*, **51**(2), 53-57.

Sandbothe, R. A.(1998), Two observations on the grouping efficacy measure for goodness of block diagonal forms, *International Journal of Production Research*, **36**(11), 3217-3222.

Sarker, B. R.(2001), Measure of grouping efficiency in cellular manufacturing systems, *European Journal of Operational Research*, **130**(4), 588-611.

Srinivasan, G., Narendran, T. T., and Mahadevan, B.(1990), An assignment model for the part families problem in group technology, *International Journal of Production Research*, **28**(1), 145-152.

Viswanathan, S.(1996), A new approach for solving the p-median problem in group technology, *International Journal of Production Research*, **34**(10), 2691-2700.