

〈論 文〉

경계구속 및 내부결함을 고려한 이차원 패턴의 최적배치를 위한 다단계 배치전략

한국찬* · 나석주**

(1994년 6월 13일 접수)

A New Multi-Stage Layout Approach for Optimal Nesting of 2-Dimensional Patterns With Boundary Constraints and Internal Defects

G. C. Han and S. J. Na

Key Words : Nesting Algorithm(배치알고리즘), Self-Organization Assisted Layout(SOAL), Simulated Annealing(모의 어닐링), Cost Function(가격함수), Heuristics(발견론적 방법)

Abstract

The nesting of two-dimensional patterns into a given raw sheet has applications in a number of industries. It is a common problem often faced by designers in the shipbuilding, garment making, blanking die design, glass and wood industries. This paper presents a multi-stage layout approach for nesting two-dimensional patterns by using artificial intelligence techniques with a relatively short computation time. The raw material with irregular boundaries and internal defects which must be considered in various cases of nesting was also investigated in this study. The proposed nesting approach consists of two stages: initial layout stage and layout improvement stage. The initial layout configuration is achieved by the self-organizing assisted layout(SOAL) algorithm, while in the layout improvement stage, the simulated annealing(SA) is adopted for a finer optimization.

1. 서 론

주로 2차원 패턴의 절단응용 분야에서 직면하는 배치(layout)문제는 조선, 의류, 블랭킹 금형, 종이, 유리, 목재 등의 제조업에서 필수적인 작업공정이라 할 수 있는데, 최근 이러한 배치공정을 위한 CAD/CAM시스템이 일부 생산현장에 적용되고 있다. 넓은 의미에서 배치문제는 회로설계(circuit design)분야에서의 셀 배치(cell placement)로부터

프랜트 레이아웃(plant layout)과 창고, 컨테이너 혹은 화물선 등에서의 물품적재(loading, shipping and packing) 등 3차원 배치에 이르기까지 넓은 응용분야를 가지며, 특히 2차원 절단분야에서의 패턴 배치 알고리즘을 네스팅 알고리즘(nesting algorithm)이라 부른다. 판재소재를 다루는 생산분야에서의 네스팅작업은 설계가 끝난 각 부품의 절단작업의 전단계로서 수행되며, 일반적으로 네스팅공정이 완료되면 절단경로를 결정하고 가공조건과 함께 수치제어공작기계의 제어에 필요한 NC코드를 생성하게 된다.

네스팅작업은 적용되는 분야에 따라 요구사항과

*정회원, 한국과학기술원 대학원 정밀공학과

**정회원, 한국과학기술원 정밀공학과

구속조건이 달라지며 이로 인해 알고리즘과 자료구조도 달라지게 되나 공통적인 목표는 주어진 영역 안에서 겹침 없이 배치하면서 요구되는 면적을 최소화하는 것이다. 지난 10여 년간 여러 산업의 응용분야에서는 네스팅시스템의 도입이 활발하게 이루어지고 있는데 수동에서 반자동 및 자동에 이르기까지 다양하다. 그러나 자동 네스팅시스템의 경우 배치효율의 신뢰성이 비교적 부족하기 때문에 아직까지는 생산현장에서 기피하는 실정이다. 배치 알고리즘의 관점에서 볼때 이러한 문제들은 NP-complete 문제로 분류하며 제한된 시간안에 최적의 해를 구하기가 불가능한 조합 최적화문제로 알려져 있다.

2차원 패턴의 네스팅은 일반적으로 직사각형 패턴과 불규칙패턴의 배치문제로 나누어 볼 수 있으며 지금까지 발표된 연구논문들은 주로 직사각형 패턴이나 임의의 패턴들로 구성된 직사각형 모듈의 배치에 치중되어 왔다.^(1~6) 직사각형 배치에 관한 연구는 1960년대의 Gilmore^(1,2) 등에 의해 논문이 발표되면서부터 본격적으로 시작되었는데 이들은 선형계획법(linear programming technique)과 낭색 함수(knapsack function)를 사용하여 1차원, 2차원 및 3차원 절단문제를 풀었다. Adamowicz와 Albano⁽³⁾는 2단계 접근법을 도입한 휴리스틱 알고리즘을 제안하였다. 이는 직사각형들을 스트립(strip)들로 형성시키는 단계와 동적계획법(dynamic programming algorithm)을 사용하여 스트립들의 최적배치에 들어가는 2단계로 구성되어 있다. Christofides 등⁽⁴⁾은 동적계획에 기반을 둔 트리(tree) 탐색알고리즘을 사용하여 2차원 절단문제를 풀었다.

불규칙패턴의 배치 문제에 관한 연구는 Adamowicz와 Albano⁽⁵⁾에 의해 처음 발표되었다. 이 논문은 2단계 접근법에 관한 것으로 임의의 패턴 혹은 패턴들의 집단을 둘러싸는 최소크기 직사각형 모듈의 생성단계와 이 모듈들의 최적배치 단계를 통해 해를 구하였다. Albano⁽⁶⁾는 휴리스틱방법과 동적계획법을 조합해 대략적인 배치를 자동으로 구한 다음 사용자에게 의해 대화식으로 개선하는 방법을 제안하였다. Nee⁽⁷⁾는 블랭크 배치문제에서 동일패턴의 배치문제를 다루었으며 Nee 등⁽⁸⁾은 경계의 구속이 있는 경우의 임의의 패턴배치에 관한 논문을 발표하였다. 이러한 알고리즘들은 배치해를 탐색하는데 있어서 휴리스틱정보를 적용하여 탐색

속도를 증가시켰으나 탐색절차가 순차적(sequential)이며 결정론적(deterministic)이라는데 그 한계가 있다. 이는 집단성장(clustering)에 의한 배치방식을 채택하기 때문에 이미 배치된 패턴에 대해서는 재배치를 시도하지 않는 것이 일반적이다. 또한 원소재의 결합 및 패턴의 내부 홀(hole)을 고려한 효과적인 배치방법에 대해서도 아직까지 연구가 미진한 상태이다.

최근 조합 최적화문제와 관련하여 Kirkpatrick 등⁽⁹⁾에 의해 개발된 최적화기법 중 하나인 모의 어닐링(simulated annealing: SA)법은 그 적용성과 우수한 탐색능력때문에 여러 응용분야에서 적용하려는 시도가 이루어지고 있다. 특히 대규모 집적회로(VLSI) 설계분야에서 표준셀(cell) 배치에 성공적으로 적용되고 있으며,⁽¹⁰⁾ 네스팅분야에서도 이의 응용이 발표되고 있다. Jain 등⁽¹¹⁾은 블랭크(blank) 네스팅을 위하여 그리고 조경호⁽¹²⁾는 판재 가공에서의 네스팅문제를 해결하기 위해 배치될 임의의 패턴을 격자표현법을 사용하여 SA에 적용하였으며, 한국찬 등⁽¹³⁾은 직사각형 배치문제를 신경회로망과 SA를 결합한 다단계 접근법을 도입해 빠른 시간안에 해를 구하고자 하는 새로운 시도를 발표하였다.

본 논문은 직사각형 배치문제⁽¹³⁾의 다단계 접근법을 더욱 확장한 연구결과로써, 산소, 플라즈마, 레이저 등의 에너지를 이용한 절단응용분야의 CAD/CAM화를 위해 직사각형 원소재 및 경계구속을 갖거나 내부결합이 있는 원소재에 임의의 이차원패턴을 방향성없이 배치하면서 재료의 손실을 최소화시키는 방법에 대해 기술하였다. 또한 패턴 내부의 홀 및 유용한 배치공간까지도 배치에 활용될 수 있는 방법에 대해서도 기술하였다.

2. 접근방법 및 개요

2.1 다단계 접근법

제한된 네스팅 알고리즘은 빠른 시간안에 전체적인 배치윤곽을 생성하는 초기배치단계와 주어진 초기배치에서 출발하여 더 나은 상세배치를 시도하는 배치개선단계로 이루어진 2단계로 구성되어 있다. 초기배치는 자율조직 신경회로망중의 하나인 Kohonen모델의 학습규칙과 학습속도를 결정하기 위해 퍼지 c-means(FCM) 알고리즘의 소속함수를 결합한 SOAL(self-organization assisted layout)이라는

배치알고리즘에 의해 구현된다. 배치개선단계에서는 조합최적화문제의 해를 구하는데 매우 강력하면서 강인성이 있는 방법인 모의 어닐링에 의한 배치기법을 이용하여 SOAL로부터 얻은 초기배치의 한계를 극복하고 최상에 가까운 해를 구하고자 한다. 즉, 최적의 배치해를 얻기 위해 본 논문에서 제안하는 배치전략은 SOAL의 빠른 학습능력과 SA의 우수한 탐색능력을 결합하는 다단계 접근법이다. (13)

2.2 패턴정의 및 겹침판별

임의의 패턴을 효율적으로 다루고 처리시간을 줄이기 위해 본 논문에서는 단순패턴(simple pattern)들로 이루어진 복합패턴(complex pattern)에 대해 정의하고 이를 다룬다. 기존의 연구논문들이 다각형근사법(5~8)과 격자표현법 등(11,12)에 의해 패턴을 정의하고 단순화시켰는데 비해 본 논문에서는 단순패턴에 의한 복합패턴의 표현이라는 새로운 방법도 도입하였다. 다각형근사법은 임의의 패턴을 직선선분(line segment)들로 근사화시켜 처리시간을 빠르게 하며, 격자표현법은 패턴간 겹침 판별을 효율적으로 하기 위함인데, 본 논문에서의 복합패턴은 다양한 크기의 단순패턴들을 조합하여 구성하기 때문에 필요로 하는 메모리의 크기가 작으며 겹침 판별 또한 손쉽게 계산할 수 있다는 장점이 있다.

단순패턴은 길이와 폭으로 정의되는 직사각형과 반경에 의해 정의되는 원으로 나누어지며, 기준점(reference point)은 그 중심에 위치한다. Fig. 1은 직사각형과 원의 단순패턴에 의해 근사적으로 모델링된 배치하고자 하는 2차원 복합패턴을 나타내고 있다. 하나의 복합패턴은 최소 포함직사각형(smallest rectangular enclosure), 내부홀(inner hole) 및 외부홀(outer hole) 그리고 다각형을 이루는 정점들(vertexes)로 표현된다. 최소 포함직사각형의 기준점은 배치영역의 원점에 대한 절대좌표를 가지며 내부홀의 기준점들은 최소 포함직사각형의 기준점에 대한 상대좌표를 가진다. 단순화된 복합패턴의 정밀도는 이를 이루고 있는 단순패턴들의 수가 늘어날수록 향상되나 이는 컴퓨터의 메모리 용량 및 알고리즘 복잡도(complexity)로 인해 제한된다.

단순패턴들에 대한 겹침판별은 두 패턴사이의 가능한 겹침조합들을 고려함으로써 가능하며 복합패턴의 경우에서도 동일한 방법을 적용해 구할 수가

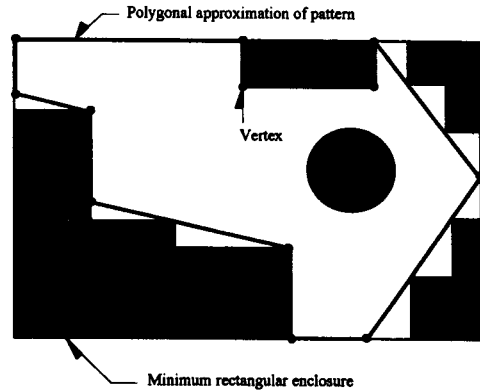


Fig. 1 An arbitrarily shaped pattern approximated by simple patterns of rectangle and circle

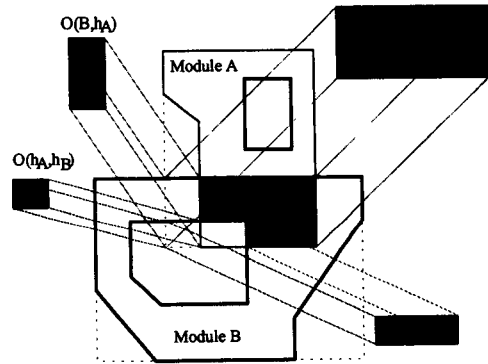


Fig. 2 Computation of net overlapping area between module A and B

있다. Fig. 2는 두 복합패턴 A와 B사이의 겹침을 보여주고 있으며, 두 패턴간 순수한 겹침면적 $O_{net}(A, B)$ 는 아래 식으로부터 구할 수 있다.

$$O_{net}(A, B) = O(A, B) - [\sum_i O(A, h_{Bi}) + \sum_j O(B, h_{Ai}) - \sum_{i \neq j} O(h_{Ai}, h_{Bj})] \quad (1)$$

여기서, $O(A, B)$ 는 패턴 A와 패턴 B의 최소 포함직사각형의 면적, $\sum_j O(A, h_{Bj})$ 는 패턴 A의 최소 포함직사각형과 패턴 B의 홀(hole)에 대한 겹침면적, $\sum_i O(B, h_{Ai})$ 는 패턴 B의 최소 포함직사각형과 패턴 A의 홀에 대한 겹침면적, $\sum_{i \neq j} O(h_{Ai}, h_{Bj})$ 는 패턴 A의 홀과 패턴 B의 홀에 대한 겹침면적이다.

2.3 경계구속 및 내부결합

네스팅 응용분야에 적용되는 원소재의 형상은 직사각형만이 아니라 불규칙한 경계를 갖는 경우도 자주 발생한다. 남은 부재를 재활용할 경우나 특히 피혁제품 제조업분야 등에서는 경계의 불규칙뿐만 아니라 원소재 내부를 이미 활용한 경우 혹은 결함이 있는 경우가 자주 발생하기 때문에 이를 고려한 효과적인 배치알고리즘의 개발이 필요하다. 이를 위해 본 논문에서는 배치에 적용되는 복합패턴의 정의와 데이터구조로부터 쉽게 적용할 수 있는 방법에 대해 기술한다. Fig. 3은 경계구속 및 내부결합에 대해 직사각형으로 단순화한 결과를 나타내고 있다. 패턴을 배치할 때에 이러한 직사각형으로 이루어진 영역은 피하고 나머지 영역에 채워지는 배치방법을 취하는 것이 기본적인 구상이며, 이는 다음의 4단계 절차에 의해 이루어진다.

- (1) 임의의 불규칙한 경계나 내부결합을 직선으로 이루어진 다각형으로 단순화시킨다.
 - (2) 각 직선은 몇 개의 직사각형들을 조합하여 근사화시킨다.
 - (3) 이러한 경계 혹은 내부 직사각형들은 먼저 배치시킨 패턴으로 간주하며 고정시킨다.
 - (4) 배치시 이미 고정된 직사각형들도 다른 패턴들과 함께 겹침판별에 고려된다.
- 이러한 직사각형들은 겹침판별에서 가격함수의 증가를 유발하기 때문에 알고리즘이 진행함에 따라 겹침이 제거되고 결국 경계구속과 내부결합을 피하는 배치결과를 얻을 수 있게 된다.

2.4 가격함수

네스팅에 있어서 가격함수는 배치의 목적과 여러 가지 구속조건들을 기술하고 평가하는데 사용되는

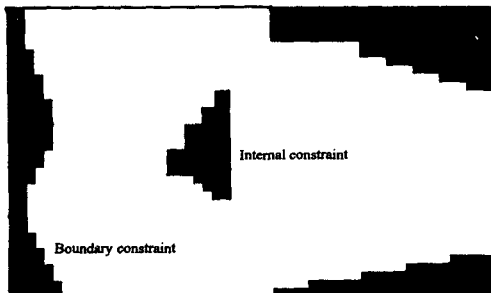


Fig. 3 Resource sheet with boundary and internal constraints

기준이 되기 때문에 주어진 알고리즘의 효율과 배치결과는 사용된 가격함수에 크게 의존한다. 본 논문에서 사용하는 가격함수에는 겹침면적 항과 면적 관성모멘트항으로 구성되어 있으며, 이는 패턴간 겹침면적의 제거와 좌측 우선순위배치에 관한 두 가지의 휴리스틱정보를 각각 표현한다.

최적배치문제는 배치공간(R), 배치공간내의 가능한 배치집합(r) 그리고 각 배치에 대한 가격함수($C(r)$)등에 의해 특징지워지는데, 다음과 같은 가격함수 C 가 작아질수록 상응하는 배치상태는 더 좋아진다.

$$C(r) = \alpha f_{ov}(r) + \beta f_{ma}(r) \tag{2}$$

여기서, f_{ov} 는 총 패턴간의 겹침면적이고, f_{ma} 는 면적 관성모멘트 항으로 아래와 같이 정의된다.

$$f_{ov} = \sum_{i \neq j} O(i, j)$$

$$f_{ma} = \sum_i x(i) A(i) \tag{3}$$

여기서, $O(i, j)$ 는 패턴 i 와 패턴 j 사이의 면적이고, $x(i)$ 는 패턴 i 의 좌측 끝단으로부터 기준점까지의 거리이며 $A(i)$ 는 그 패턴의 면적이다. Fig. 4(b)에서 두 패턴 P_2 와 P_3 의 겹침 면적을 식(3)에 의해 표현하면 $O(P_2, P_3)$ 이며, 패턴 P_1 에 대한 면적 관성모멘트는 $w_{11}A(P_1)$ 과 같다. 면적 관성모멘트 항 f_{ma} 이 줄어들수록 모든 패턴들은 좌측으로 접근하게 되며 패턴간의 간격이 줄어들게 된다. α 와 β 는 f_{ov} 와 f_{ma} 에 대한 가중치이며 패턴간 겹침을 완전히 제거하기 위해서는 α 값을 크게 하여야 하나 필요이상으로 크게 할 경우에는 겹침에 의한 가격함수의 증가를 상쇄시켜 패턴의 이동을 제한하게 되어 지역 최소점에 빠드릴 위험이 있다. 반대로 f_{ma} 를 증가 시키게 되면 패턴의 이동은 매우 활발하게 이루어지게 되나 최종배치 결과에서 겹침이 발생할 수가 있게 되므로 실험에 의한 적절한 선정이 필요하다.

3. 배치알고리즘

3.1 초기배치단계

초기배치단계의 목적은 빠른 시간안에 비교적 거칠기는 하나 합리적인 배치윤곽을 얻어내는 것이다. 이를 위해 본 논문에서는 SOAL이라하는 배치알고리즘을 제안하였다. SOAL은 Kohonen의 자율조직화형상지도(self-organizing feature maps: SO

M)⁽¹⁴⁾와 Bezdek 등에 의해 개발된 FCM⁽¹⁵⁾에 기반을 두고 있다. 이 알고리즘은 학습속도분포의 자동 조절을 가능케 하는 특성을 갖고 있는데, 이 네트워크는 비지도 학습구조(unsupervised learning architecture)이며, 2개의 층(layer)으로 구성된 단 하나의 전방패스를 사용하기 때문에 상당히 빠른 학습속도를 갖는 것이 특징이다.

Kohonen모델과 배치모델의 상관관계가 Fig. 4에 묘사되어 있는데, (a)에서 출력노드(output node)는 (b)에서 배치할 패턴에 대응된다. Kohonen모델의 입력노드(input node)의 개수는 패턴 배치영역의 차원에 대응하며 연결강도 W_{ij} 는 패턴 P_i 의 위치의 i -번째 좌표값을 나타내며 2차원 배치영역에서 각 패턴 P_i 의 위치는 벡터 $W_i = \{w_{1i}, w_{2i}\}$ 로 표시된다. 여기서, w_{1i} 와 w_{2i} 는 각각 패턴 i 의 위치의 x 와 y 좌표값을 나타낸다. 신경회로망에서 입력벡터는 규정된 배치영역에서 균등분포로 랜덤하게 선택된다. Kohonen모델의 출력노드는 각 패턴에 대한 최소 포함직사각형의 기준점에 해당하며, 경쟁층에서의 연결은 배치모델에서 두 패턴간의 겹침에 상응한다.

Kohonen 네트워크는 원래 클러스터링 알고리즘으로 잘 알려져 있다. 이러한 부류의 알고리즘이 겪게 되는 몇몇 중요한 문제점들은 그 성능에 영향을 주게 되며 결국 알고리즘은 최적으로의 수렴성을 보장할 수 없게 된다. 대표적인 문제점들을 살펴보면 다음과 같다. 첫째 알고리즘의 종결이 과정이나 데이터의 최적화에 기반을 두지 않고 강제적이라는 점과 둘째 최종 연결강도가 일반적으로 데이터의 입력순서에 의존한다는 점과 셋째 초기조건이 달라짐에 의해 다른 결과를 낳게 되며 마지막으로 데이터가 바뀔 때 따라 유용한 결과를 얻기 위해서는 학습속도와 이웃함수의 변화규칙이 반드시 바뀌어야 한다는 것이다.

이와 유사한 클러스터링 알고리즘으로 c-means (CM) 알고리즘이 있으며 Kohonen 네트워크와 유사한 관계가 있다는 것은 잘 알려져 있다. 일반적으로 CM 알고리즘은 최적화에 기반을 둔 절차이나 Kohonen 네트워크는 그렇지 못하기 때문에 이 두 알고리즘의 결합은 Kohonen 알고리즘의 단점을 극복하는 실마리를 제공할 수 있으리라 기대할 수 있다. Huntsberger와 Ajijmarangsee⁽¹⁶⁾는 처음으로 이러한 접근을 시도하였으며, Bezdek 등은 이들의 아이디어를 확장하여 퍼지 Kohonen 클러스터링 네트워크(FKCN)이라는 새로운 알고리즘을 제안하였다.⁽¹⁷⁾

Bezdek의 FKCN의 개념은 Kohonen 네트워크의 구조에 FCM 알고리즘의 소속함수(membership function)를 도입하여 학습중에 경쟁층에서의 이웃의 크기와 학습속도가 자동적으로 조절되도록 하는 것이다. 그러나 학습되는 이웃의 크기가 Kohonen 네트워크와 같이 줄어드는 개념이 아니라 각 클러스터에 대한 학습속도가 변화하게 되어 이웃의 효과가 시간에 따라 변화하도록 한 것이다.

본 논문에서는 학습속도를 증가시키기 위해 이웃의 크기를 다루며 이웃의 크기에 따른 학습속도의 변화와 시간에 따른 변화를 다루기 위해 소속함수를 도입하고 이를 Kohonen 네트워크에 적용한 자율조직 배치알고리즘을 개발하였다. SOAL의 위치 개선공식은 아래의 식과 같으며 Kohonen 네트워크의 학습규칙과 유사하다.

$$\begin{aligned} w_{1j,t} &= w_{1j,t-1} + \gamma_k \eta_{i,t} (x_{1j,t-1}) \\ w_{2j,t} &= w_{2j,t-1} + \gamma_k \eta_{i,t} (x_{2j,t-1}) \end{aligned} \quad (4)$$

여기서, 각 패턴 j 의 위치는 벡터 $W_j = \{w_{1j}, w_{2j}\}$ 로 표시되고, γ_k 는 0과 1사이의 유한한 값들 중에서 가역 함수가 최소가 되는 값을 선택한다. 또한 $\eta_{i,t}$ 이 바로 본 논문에서 개발한 자율조직 배치알고리즘의 학습속도를 제어하는 항이며 이를 결정하기 위해 다음과 같이 소속함수를 도입하였다. 배치할 각 패턴의 기준점은 Kohonen 네트워크의 출력 노드에 대응하는 동시에 FCM에 의해 소속값을 구하기 위한 클러스터의 중심이기도 하다. 하나의 랜덤한 입력이 들어왔을 때 이 입력과 각 패턴간의 기하학적인 거리의 유사도에 따라 소속값이 결정되어지며 이 소속값으로부터 Kohonen 네트워크의 학습속도가 결정된다. 이때 소속값은 아래같이 표현된다.

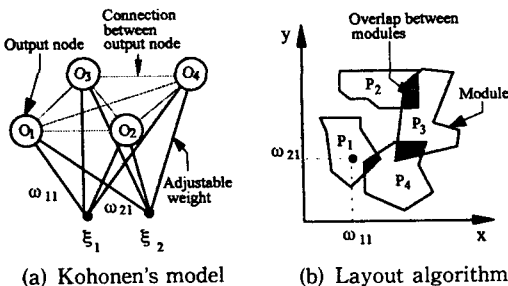


Fig. 4 Correspondence of items between

$$u_{i,t} = \frac{1}{\sum_{j=1}^c \left\{ \frac{(x_1 - w_{1j})^2 + (x_2 - w_{2j})^2}{(x_1 - w_{1i})^2 + (x_2 - w_{2i})^2} \right\}^{\frac{1}{m_i - 1}}} \quad (5)$$

여기서, $u_{i,t}$ 는 FCM알고리즘에 의해 결정되는 입력벡터와 패턴간의 퍼지소속함수이며 이는 위상학적인 정보에 의해 결정되는 유사도이다. x_1 과 x_2 는 각각 임의의 입력벡터의 x , y 좌표이며, w_{1i} 과 w_{2i} 는 i 번째 클러스터의 중심 혹은 i 번째 패턴의 기준점 좌표이다. 이 소속함수는 이웃의 크기가 감소함에 따라 전체적으로 커지는 경향이 있다. 그러나 우리가 기대하는 것은 학습이 진행됨에 따라 학습속도가 감소하고 또한 학습대상도 줄어들어 최종상태의 오차가 0에 수렴하는 것이다. 따라서 시간에 따른 학습속도 $\eta_{i,t}$ 를 다음과 같이 수정하여 사용하였다.

$$\eta_{i,t} = (u_{i,t})^{m_i}, \quad m_t = m_i - (m_i - m_f) \frac{t}{t_{max}} \quad (6)$$

여기서, 초기 m_i 와 최종 $m_f (> m_i)$ 는 실험에 의해 결정하는데, t 가 0일 때 m_t 는 m_i 가 되고 t 가 t_{max} 일때 m_t 는 m_f 가 되어 시간이 진행함에 따라 m_t 의 값은 m_i 에서 m_f 로 증가하여 학습속도를 감소시키게 된다. m_t 의 값이 클수록 학습속도는 저하되기 때문에 m_i 의 값을 작게 하고 최종 m_f 의 값을 크게 하여 해가 수렴하도록 결정한다. 실험에 의하면 m_i 는 1.0, m_f 는 3.0으로 했을 때 수렴특성이 좋았다. 배치알고리즘은 아래와 같다.

- Step 1 Initialization : $t=0$
Modules are randomly distributed near the center of region
- Step 2 Rotation of randomly selected pattern for reduction of overlap
- Step 3 Pairwise interchange for reduction of moment of area
- Step 4 Present a new input
- Step 5 Select the pattern which is nearest from the input
- Step 6 Update the position of the winner pattern and its neighbors
- Step 7 Increase t
If $t=t_{max}$ then stop
else go to step 2

초기에 모든 패턴은 배치영역의 중심에 임의로 모아 놓으며, 학습을 위한 입력을 주어진 배치영역안

에서 랜덤하게 발생시키고 입력의 위치로부터 가장 가까이 있는 패턴을 하나 선택한다. 선택된 패턴과 그 이웃패턴들은 과정 6에서 위치개선공식을 적용하여 새로운 위치를 지정해 준다. 이러한 과정이 끝나면 또 다른 입력벡터가 들어오게 되고 위에서 기술한 과정을 되풀이한다. 목적하는 배치전략을 수행하기 위해서는 어느 정도의 휴리스틱정보가 필요하게 되는데, 적용된 휴리스틱정보는 과정 2에서 겹침 면적을 쉽게 줄이고 총적률을 효과적으로 높이기 위해 임의로 선택된 패턴에 대해서 회전을 시도하는 것이다. 회전은 90°단위로 이루어지며 겹침이 줄어들 경우에만 허용된다. 또 하나는 과정 3으로 임의로 선택된 두 패턴에 대해 상호교환(pairwise interchange)을 시도하여 배치영역의 좌측 기준축에 대해서 전체패턴의 회전관성모멘트를 줄임으로써 목적하는 배치전략에 상응하는 결과를 얻고자 하는데, 이 결과로 좌측 우선순위의 배치운곽을 얻게 된다. 과정 6에서 위치개선공식에 의해 계산된 위치는 단지패턴의 이동방향만 결정한다. 이동거리는 패턴 간겹침 면적과 좌측 우선순위배치를 고려한 가격 함수(cost function)가 최소가 되는 위치로 결정된다. 이웃함수(neighborhood function) σ_i 는 다음과 같은 함수를 사용하였다.

$$\sigma_i = \sigma_i \left(\frac{\sigma_f}{\sigma_i} \right)^{\frac{t-t}{t_{max}}} \quad (7)$$

여기서, t 는 시간(반복회수), σ_i 는 초기이웃의 크기, σ_f , 최종이웃의 크기이고 t_{max} 는 전체 반복회수이다. 시간 t 에서 정의되는 식(7)의 이웃반경내에 있는 모든 패턴들은 식(5)에 의해 소속함수와 식(6)에 의해 학습속도가 각각의 패턴들에 대해 개별적으로 결정되는데, 이때 각 패턴들과 임의의 입력에 대한 학습속도는 승자패턴이 가장 크게 되고 상대적인 위상학적인 거리가 멀수록 작아지게 된다. 그러나 시간이 지날수록 전체적인 학습속도는 식(6)에서와 같이 감소하는 경향을 갖고 있다. 그러므로 알고리즘이 수행되기 시작하여 얼마동안은 하나의 입력에 대해 많은 패턴들이 영향을 받고 또 그 이동거리도 커지며, 시간이 흘러 이웃의 크기가 감소하게 되면 주어진 입력에 의해 영향을 받는 패턴이 적어지게 되고 원하는 배치운곽에 수렴하는 특징을 갖게 된다.

3.2 배치개선단계

전통적인 반복개선방식의 최적화 알고리즘은 가

격 함수가 감소하는 방향으로만 진행하기 때문에 지역 최소점(local minima)에 빠져버려 더 이상의 개선이 진전되지 못할때가 흔하다. 이를 극복하기 위해서는 임의의 초기점에 대해 수많은 시도가 필요하나 전역적인 최적해는 여전히 보장되기가 어렵다는 것이 일반적인 견해이다. SA(simulated annealing)는 이러한 반복개선방식의 알고리즘과 유사하나 가격함수가 증가하는 방향으로의 진행이 가능하다는 점에서 큰 차이점을 보이고 있다. 이 때문에 전보다 더 나쁜 상황으로의 상태전이가 발생함으로써 지역적 최소점을 탈출하고 결국에는 전역적 최소점에 도달할 수 있다는 잠재적인 가능성을 내포한다. (18~20)

SA는 초기온도 T_0 와 임의의 초기배치 X_0 에서 출발하여 매 온도 단계에서 새로운 배치 X' 를 생성한다. 또한 이때의 가격함수의 변화 $\Delta C = C(X') - C(X)$ 를 계산하고 새로운 배치에 대한 허용여부를 결정하게 된다. 만약 $\Delta C \leq 0$ 이면 확률 1로써 새로운 배치를 받아들이며, $\Delta C > 0$ 인 경우 새로운 배치상태는 볼츠만분포의 제한적인 확률로서 허용된다. 이 제한적인 확률은 온도 $T = \infty$ 에서는 1이며, 온도가 $T = 0$ 으로 감소함에 따라 확률도 0으로 감소하여 결국에는 응고하게 된다. 실제 응용에 있어서 SA는 유한한 회수의 상태전이와 온도 단계로 인해 각 온도단계에서 평형상태를 만족하면서 최적의 상태로 수렴한다는 보장은 없으나 냉각 스케줄이 충분히 완만하게 한다면 매우 훌륭한 결과를 얻을 수가 있다.

현 상태의 배치로부터 새로운 배치상태를 생성하기 위해 다음의 두가지 교란(perturbation)을 야기시킨다.

(1) 단일 패턴에 대한 병진이동과 회전 그리고 이의 조합

(2) 임의로 선택된 두 패턴간의 상호교환

본 알고리즘에서 단일패턴의 병진이동에 대해서는 범위제한함수를 적용하였다. 원소재 전체영역에 대한 전역적인 교란과 작은 영역에 대한 미소교란이 그것인데, 이는 전역적인 교란을 통해 지역적인 최소점에서 탈출하고 미소교란을 통해 좌측방향으로의 방향성을 주고 패턴간의 간격을 효과적으로 줄이기 위함이다.

이미 언급한 바와 같이 최적의 해를 보장해 주기 위해서는 냉각 스케줄이 매우 중요한 인자중의 하나이다. 이는 고체가 결함이 없는 완전한 결정체로

만들어질려면 용융 상태에서 무한한 시간의 냉각 스케줄을 통해서만이 가능하다는 원리와 유사하다. 그러나 현실적으로 유한한 시간안에 알고리즘을 구현해야하며 가능한 빠른 시간안에 좋은 결과를 얻는 것을 목표로 하지만 이를 만족시켜 주는 냉각 스케줄은 실제 존재하지 않는다. 본 논문에서 사용한 냉각 스케줄은 다음과 같이 고정된 2단계의 스케줄을 통해 온도가 감소하도록 하였다.

$$T_{i+1} = \alpha(T) T_i \quad (8)$$

이때 $\alpha(T)$ 는 온도감소율이며, 시행착오법으로 결정하였다. 초기의 높은 온도구간에서는 $\alpha(T) = 0.98$ 로 비교적 완만한 감소를 시도하며, 낮은 온도구간에서는 $\alpha(T) = 0.85$ 로써 좀더 빠른 냉각 스케줄을 채택하였다.

4. 실험결과

제안된 배치알고리즘은 IBM PC 호환기종에서 C언어로 구현하였으며, 2차원 패턴의 최적배치를 위한 네스팅시스템으로 개발하였다. Fig. 5는 SOAL에 의한 시간에 따른 배치과정을 보여주고 있다. 초기패턴들은 원소재의 중심부근에 모여있다가 (Fig. 5(a)) 시간이 지남에 따라 점점 전체영역으로 분산되면서 좌측으로 이동하고 있는 것을 볼 수 있다. 초기 패턴들의 움직임은 매우 크며 시간이 지남에 따라 점점 감소하여 원하는 배치윤곽에 수렴해 나가게 되는데 (Fig. 5(d)), 이는 학습에 참여하는 이웃의 크기가 학습속도가 점점 감소하기 때문이다. SOAL의 학습이 확률적이며 반복개선의 특성을 갖고 있기 때문에 최종의 배치결과는 반복횟수에 크게 의존한다. 그러나 SOAL에서 얻고자 하는 바가 최종의 배치결과가 아니라 대략적인 배치윤곽이기 때문에 반복횟수를 2000으로 제한하였으며, 임의의 패턴배치시 SOAL에서 계산시간을 단축시키기 위해서 내부 홀에 대한 결침계산은 수행하지 않았다. IBM PC/486 DX II (66 MHz)에서 수행한 결과 65초가 소요되었다.

SOAL에서 SA로의 스위칭이 완만하게 이루어지기 위해서는 SA의 적절한 초기온도 설정이 필요하다. 최근 SA의 초기온도 설정을 위한 이론적인 접근이 발표되고 있으나 큰 효과를 거두지 못하고 있는 실정이며 통상실험에 의존하는 경우가 일반적이다. 본 논문에서도 실험을 통해 초기온도를 결정하

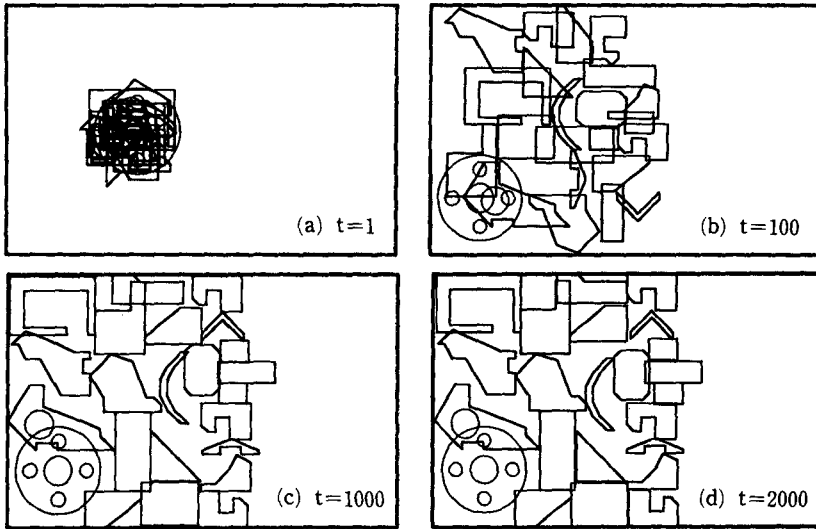
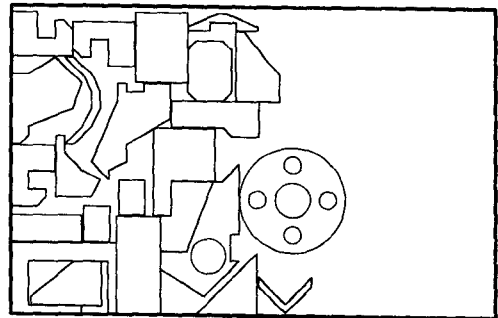


Fig. 5 SOAL optimization process : intermediate results as time t is increased

었는데 ($T_0=30$), 데이터의 크기가 바뀌면 초기온도의 설정이 달라져야하나 이 정도의 패턴 개수에서는 약간의 차이가 있을지라도 크게 영향이 없었다. Fig. 6은 SOAL로부터 얻은 초기 배치(Fig. 5(d))에 SA를 적용하여 얻은 최종 배치결과이다. 수행시간은 299초이다. Fig. 7과 Fig. 8는 경계구속 및 내부결함이 있고 내부홀들을 포함하는 임의의 패턴에 대해 SOAL과 SA를 수행하여 얻은 네스팅 결과들로 수행시간은 각각 98초와 371초이며, 원소



CPU time : SOAL=65, SA=299

Fig. 6 Nesting result for approximated arbitrary patterns

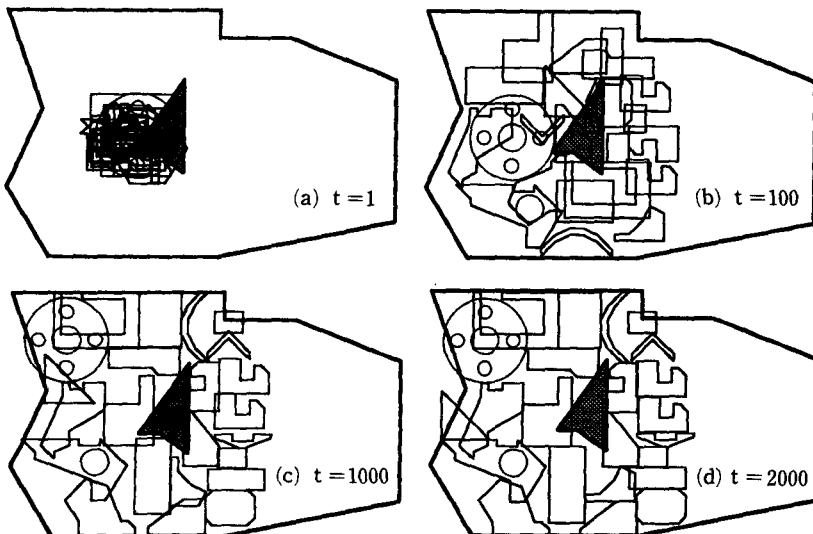
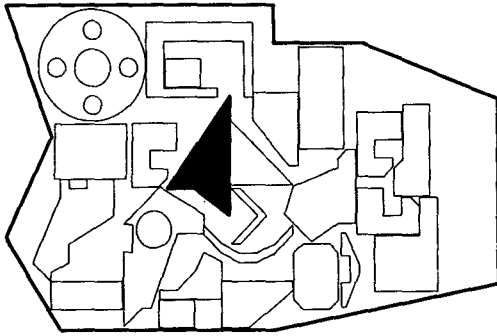


Fig. 7 Another SOAL optimization process with boundary and internal constraints



CPU time : SOAL=98, SA=371

Fig. 8 Nesting result with boundary and internal constraints

재의 내부결합과 임의의 경계구속이 있는 경우에도 효과적인 배치가 수행되었음을 볼 수 있다.

5. 결 론

제조업 분야에서 원가절감을 위한 한 방편으로 재료의 손실을 최소화 시키는 노력들을 들 수 있다. 판재 소재를 다루는 분야에서의 필수적인 작업 공정인 네스팅은 바로 이러한 관점에서 매우 중요한 공정이라 할 수 있다. 이 분야에 대한 기존의 연구를 비롯하여 근래에 상용화되고 있는 네스팅시스템의 공통적인 특징은 일반적으로 휴리스틱알고리즘에 치중되어 있거나 배치효율을 높이기 위해 대화형 시스템에 의존해야 한다는 아쉬움이 있다. 본 논문에서 제시한 네스팅알고리즘은 인공지능의 한 분야인 Kohonene의 자율조직 신경회로망과 FCM을 결합한 SOAL알고리즘과 VLSI 설계 등 조합 최적화문제에 성공적으로 적용되고 있는 SA알고리즘의 2단계 스위칭 구조이다. SOAL가 초기 배치 단계에서 빠른 시간안에 대략적인 배치운곽을 얻기 위한 전략이라면 SA는 배치개선단계에서의 상세배치를 위한 전략이다. 적용대상은 불규칙적이며 내부홀을 갖는 임의의 패턴들과 경계구속과 내부결합을 갖는 원소재에 대한 배치 등이며, 실제 산업현장에서 요구하는 다양한 배치전략에 대한 그 적용성과 확장성을 제시하였다. 본 연구를 통해 얻은 결과의 주된 성과로는 첫째, 현재까지 발표된 대부분의 네스팅알고리즘들이 순차적이며 결정론적 탐색방법인 집단 성장식의 배치방법을 도입하기 때문에 지역적 최소해(local solution)에서 빠져 나오지 못하는 한계가 있는 반면, 제안된 배치 전략은

확률적인 반복개선 방법들을 도입하기 때문에 전역해(global solution)를 구할 수 있다는 점이다. 둘째, 많은 연구 논문들이 최소 포함직사각형 패킹에 관한 것이라면 제안된 배치알고리즘은 임의의 복합 패턴을 단순패턴들의 조합에 의해 근사화하기 때문에 원소재의 내부결합과 경계구속을 비롯해 다양한 패턴들을 쉽게 표현하고 다룰 수가 있다는 것이다.

참고문헌

- (1) Gilmore, P. C. and Gemory, R. E., 1965, "Multi-stage Cutting Stock Problems of Two and more Dimensions," *Opns. Res.*, Vol. 13, pp. 94~120.
- (2) Gilmore, P. C. and Gemory, R. E., 1966, "The Theory and Computation of Knapsack Functions," *Opns. Res.*, Vol. 14, No.61, pp. 1045~1074.
- (3) Adamowicz, M. and Albano, A., 1976, "A Solution of the Rectangular Cutting-Stock Problem," *IEEE Trans. on System, Man and Cybernetics*, Vol. SMC-6, No. 4, pp. 302~210.
- (4) Christofides, N. and Whitlock, C., 1977, "An Algorithm for Two-Dimensional Cutting Problems," *Opns. Res.*, Vol. 25, No. 1, pp. 30~44.
- (5) Adamowicz, M. and Albano, A., 1976, "Nesting Two-Dimensional Shapes in Rectangular Modules," *Computer-Aided Design*, Vol. 8, No. 2, pp. 27~33.
- (6) Albano, A., 1977, "A Method to Improve Two-Dimensional Layout," *Computer-Aided Design*, Vol. 9, No. 1, pp. 48~52.
- (7) Nee, A. Y. C., 1984, "A Heuristic Algorithm for Optimum Layout of Metal Stamping Blanks," *Annals of the CIRP*, pp. 317~320.
- (8) Nee, A. Y. C., Seow, K. W. and Long, S. L., 1986, "Designing Algorithm for Nesting Irregular Shapes with and Without Boundary Constraints," *Annals of the CIRP*, Vol. 35, No. 1, pp. 107~110.
- (9) Kirkpatrick, S., Gelatt, C. D. and Vecchi, M. P., 1983, "Optimization by Simulated Annealing," *Science*, Vol. 220, No. 4598, pp. 671. 143~164.
- (10) Shahooar, K. and Mazumder, P., 1991, "VLSI Cell Placement Techniques," *ACM*

- Computing Surveys*, Vol. 23, No. 2, pp. 143~164.
- (11) Jain, P., Fenyves, P. and Richter, R., 1992, "Optimal Blank Nesting Using Simulated Annealing," *J. of Mech. Desn. of the ASME*, Vol. 114, pp. 160~165.
- (12) 조경호, 1993, "판매 부품의 가공 자동화를 위한 CAD/CAM 통합 시스템," 박사학위논문, 서울대학교.
- (13) 한국찬, 나석주, 1993, "신경회로망을 이용한 직사각형의 최적배치에 관한 연구," 대한기계학회논문집, 제17권, 제12호, pp. 3063~3072.
- (14) Kohonen, T., 1990, "The Self-Organizing Map," *Proceedings of the IEEE*, Vol. 78, No. 9, pp. 1464~1480.
- (15) Bezdek, J. C. and Castelar, P. F., 1977, "Prototype Classification and Feature Selection with Fuzzy Sets," *IEEE Trans. on System, Man and Cybernetics*, Vol. 7, No. 2, pp. 87~92.
- (16) Huntsberger, T. and Ajjimarangsee, P., 1989, "Parallel Self-Organizing Feature Maps for Unsupervised Pattern Recognition," *Int. J. General Systems*, Vol. 16, pp. 357~372.
- (17) Bexdek, J. C., Tsao, E. C. K. and Pal, N. R., 1992, "Fuzzy Kohonen Clustering Networks," *IEEE Int. Conference on Fuzzy Systems*, pp. 1035~1043.
- (18) Rutenbar, R. A., 1989, "Simulated Annealing Algorithms: An Overview," *IEEE Circuits and Devices Magazine*, pp. 19~26.
- (19) Aarts, E. H. L. and Van Laarhoven, P. J. M., 1985, "A New Polynomial-Time Cooling Schedule," in *Proc. Int. Conference on Computer-Aided Design*, pp. 206~208.
- (20) Aarts, E. H. L., Korst, J. H. M. and Van Laarhoven, P. J. M., 1988, "A Quantive Analysis of the Simulated Annealing: A Case Study for the Traveling Salesman Problem," *J. of Statistical Physics*, Vol. 50, No. 1/2, pp. 187~206.