

〈論 文〉

## 임의 형상의 여러 원자재 위에서의 효과적인 배치방안

조경호\* · 이건우\*\*

(1994년 11월 10일 접수)

### An Effective Method for the Nesting on Several Irregular Raw Sheets

Kyung-Ho Cho and Kun-Woo Lee

**Key Words** : Optimal Nesting(최적배치), Pattern Marking(옷감모형배치), Grid Model(격자모델), Simulated Annealing Algorithm(풀림모사기법), Combinatorial Optimization(조합최적화)

#### Abstract

An effective nesting algorithm has been proposed to allocate the arbitrary shapes on one or several raw sheets by applying the well-known simulated annealing algorithm as the optimization technique. In this approach, both the shapes to be allocated and the raw sheets are represented as the grid-based models. This algorithm can accommodate every possible situations encountered in cutting apparel parts from the raw leather sheets. In other words, the usage of the internal hole of a shape for other small shapes, handling of the irregular boundaries and the interior defects of the raw sheets, and the simultaneous allocation on more than one raw sheets have been tackled on successfully in this study. Several computational experiments are presented to verify the robustness of the proposed algorithm.

#### 1. 서 론

최적화기법의 관점에서 볼 때 배치문제의 최적해를 구하는 방법으로 비교적 최근에 등장한 것은 풀림모사기법(simulated annealing algorithm, SAA)을 이용한 연구가 있다. SAA<sup>(1)</sup>는 1983년 최적화기법으로 처음 소개된 이래, VLSI의 최적설계와 같은 조합 최적화문제(combinatorial optimization problem)에 적용되기 시작하였으며,<sup>(2-4)</sup> 최적배치 분야에선 경계제한이 없는 원자재 위에서 동일 형상의 반복배치에 대한 연구<sup>(5)</sup>에 처음으로 적용되어 좋은 평가를 얻은 바 있다.

한편, 최적배치에 대한 최근의 국내 연구는 참

고문헌<sup>(6-12)</sup>에서 찾아볼 수 있다. 이 중에서 참고문헌<sup>(7-9)</sup>의 연구는 SAA를 임의 형상들의 최적배치 문제에까지 확장시켰고, 참고문헌<sup>(10,11)</sup>의 연구는 최적 배치문제를 신경회로망과 SAA를 혼합하여 해결하고자 노력하였으며, 배치형상의 분류와 다단계 배치전략을 제시한 바 있다.

최적배치에서의 고전적인 주요 목표는 원자재의 버림률(waste ratio,  $wr$ )을 최소화하는 것이나, 최근의 연구 동향으로는 여러가지 구속조건을 고려한 배치, 즉 피혁원단과 같은 원자재의 경계가 불규칙한 경우 및 원자재 내부의 결함을 고려한 배치에 관한 연구가 있다. 특히, 피혁제품의 재단 특성으로는 원자재 경계의 불규칙함과 원자재 내부의 결함 가능성뿐만 아니라, 일정 폭의 strip 형태의 옷감원단, 철판 등에 비하여 원자재 크기가 비교적 작고 한정되어 있다는 점이다. 그러나, 최적배치에

\*정회원, 제주대학교 에너지공학과

\*\*정회원, 서울대학교 기계설계학과

대한 기존의 대부분의 연구결과는 이러한 특수상황을 고려한 배치문제에는 적합하지 못하다.

한편, Nee<sup>(13)</sup>의 연구는 원자재의 경계가 불규칙한 경우 및 원자재 내부의 결함을 고려한 배치문제를 다각형 근사화방법을 이용하여 처음 시도하였으나, 가용 원자재 영역의 표현 및 배치효율 면에 있어서 개선의 여지가 많이 있다.

본 연구에서는 피혁제품의 재단에서와 같은 특수상황에서도 효과적으로 작동할 수 있는 배치 알고리즘을 제안하고 몇가지 실험 예를 통해 제안된 배치 알고리즘의 작동 성능과 배치효율 향상방안을 검토 하였다.

## 2. 본 론

### 2.1 문제정의 및 최적화기법

배치문제를 사용 원자재의 구속조건 유무에 따라 다음과 같이 분류할 수 있다.

첫째, 경계 제한이 없는 원자재에서의 최적배치  
 둘째, 경계 제한이 있는 원자재에서의 최적배치

- ① 일정 폭의 원자재에서의 최적배치
- ② 임의 형상의 원자재에서의 최적배치

원자재 경계에 대한 제한이 없는 배치의 예는 동일 형상의 반복배치문제로 동일 형상을 주로 2~3개단위로 묶어 반복가공하는 데 사용된다. 실제의 가공은 일정 폭의 원자재로부터 일정한 주기로 수행되지만, 원자재의 폭, 형상간의 배치, 반복주기 등은 배치의 최적화를 통하여 결정된다.<sup>(6,14,15)</sup> 이와는 달리 옷감원단이나 철판의 재단문제는 원자재의 폭이 이미 일정한 값으로 고정된 경우로서 대부분의 배치 알고리즘은 이러한 문제에 대해서만 효과적인 작동이 보장된다.

반면에 피혁제품의 재단에서 볼 수 있는 경계가 불규칙하고 내부에 부분적인 결함이 존재하는 임의 형상의 원자재 위에서의 배치문제는 기존배치 알고리즘으로는 바람직한 결과를 기대할 수가 없다. 특히, 기존배치 알고리즘에서는 두개 이상의 원자재 위에서의 배치문제를 전혀 고려하고 있지 않기 때문에 이를 위한 새로운 배치 알고리즘의 개발이 필요하다.

이를 위해서 본 연구에서는 다음과 같은 내용을 주요 특징으로 하는 최적배치 알고리즘을 개발하였다.

-원자재의 경계는 불규칙하고, 내부결함을 갖을

수도 있다.

-여러 원자재가 동시에 사용될 수도 있다.

-배치될 형상은 임의 모양을 갖는다.

-별도의 추가 알고리즘없이도 형상의 내부 구멍에도 작은 형상을 배치할 수 있다.

본 연구에서 해결하고자 하는 최적배치문제와 같이 수많은 국지해를 갖는 조합 최적화문제들은 그 최적해를 구하는데 있어서의 난이도에 따른 분류로 볼때 NP complete (non-deterministic polynomial time complete<sup>(16)</sup>) 문제에 해당된다. 이들 문제는 대부분 부정정계이면서 미분 불가능한 특성을 갖고 있기 때문에 종래의 목적함수 변화율에 근거한 최적화기법으로는 전역 최적해(global optima)에 도달하기 어렵고 주로 국지적인 해에 빠지기 쉽다. 따라서, 수많은 국지해를 거쳐 전역 최적해에 도달하기 위해선 목적함수 값이 증가하는 방향으로 이동할 수 있는 능력(hill climbing ability)과 과거 어느 순간 방문되었던 해라 할지라도 필요에 따라선 다시 과거의 해로 되돌아 가서 새로운 탐색을 시도할 수 있는 능력(backtracking ability)이 필요하다. 이상의 두 능력을 고루 갖춘 최적화기법으로 최근 주목을 받고 있는 것으로 SAA, 타부써치(tabu search<sup>(17,18)</sup>) 및 유전적 방법(genetic algorithm<sup>(19,20)</sup>) 등이 있다.

이중에서도 SAA는 그 구현방법이 비교적 용이하고, 목적함수의 구성방법, 구속조건, 자유도 등의 변화와는 거의 독립적으로 알고리즘의 골격을 유지할 수 있어 새로운 문제로의 이식성이 뛰어나기 때문에, 본 연구에서는 SAA를 최적화기법으로 도입하였다. SAA의 구체적인 작동원리 및 적용사례들은 여러 참고문헌<sup>(1-5,7-11)</sup>에서 찾을 수 있다.

### 2.2 원자재의 효과적인 정렬방향

일정 폭의 원자재 위에서의 배치문제와는 달리, 불규칙한 원자재를 사용할 경우 배치효율의 평가는 여러가지 방법으로 할 수 있겠으나 아직은 객관적이고 정량화된 기준은 마련되어 있지 못한 것으로 본다. 참고로 참고문헌<sup>(12)</sup>에서는 일정 폭의 원자재 위에서의 배치효율평가를 위해 면적비 버림률 및 길이비 버림률을 정의하여 사용하였으나, 객관적인 비교를 위한 버림률의 정의로는 부적절함을 보였다.

본 연구에서는 '잠재적 가용짜투리(potential

waste)’의 개념을 도입하여, 배치효율의 평가를 위한 원자재의 버림률(*wr*)을 다음과 같이 정의하여 사용하였다.

$$wr(\%) = \left[ \frac{Scrap}{\sum AS_i} \right] \times 100$$

$$= \left[ \frac{\sum RS_j - \sum AS_i - A_{pw}}{\sum AS_i} \right] \times 100 \quad (1)$$

여기서,  $RS_j$ ,  $AS_i$ 는 각각 원자재  $j$  및 형상  $i$ 의 넓이이다. 또한,  $A_{pw}$ 는 잠재적 가용짜투리의 넓이이다. 이때 잠재적 가용짜투리란 함은, Figs. 1~3에서 보는 바와 같이, 배치형상의 가공후 남은 짜투리 중에서 형상들이 배치된 영역(직사각형  $W \times H$ )밖의 잔여 원자재를 일컫는 것으로서, 배치효율의 극대화는 곧 잠재적 가용짜투리의 극대화에 해당한다. 이러한 개념은 일정 폭의 원자재를 사용하는 경우에도 그대로 적용이 가능하다.

Fig. 1은 원자재의 정렬방향에 따라 잠재적 가용짜투리의 크기가 변하는 것을 보인다. Fig. 2와 Fig. 3은 원자재 크기가 작아서, 두개 이상의 여러 원자재에서 제품이 재단되어야 할 경우에 대한 예시로서 사용된 원자재와 배치될 형상이 동일한 조

건임에도 불구하고 원자재의 정렬방향에 따라 가공 후 남은 잔여 원자재의 ‘잠재적 사용가치(potential value of waste, PVW)’가 달라짐을 보인다. 즉, Fig. 2의 여러 조각의 작은 짜투리보다는 Fig. 3과 같은 한개의 커다란 짜투리의 잠재적 사용가치를 선호하는 보편적 성향에 비추어 볼 때, 원자재의 효과적인 정렬의 필요성은 더욱 분명해진다.

일반적으로 (1) 잔여 원자재의 총 면적이 클수록, (2) 동일 면적이더라도 잔여 원자재의 둘레길이가 작을수록 PVW는 크다는 관점에서, 본 연구에서는 PVW를 다음과 같이 정량화하였다.

$$PVW = A_{pw} / c = 2\sqrt{\pi} A_{pw}^{1.5} / L_{pw} \quad (2)$$

여기서,  $A_{pw}$  및  $L_{pw}$ 은 각각 잠재 가용짜투리의 총 면적과 총 둘레길이이고,  $c$ 는  $A_{pw}$ 와 같은 넓이인 원의 원주길이에 대한  $L_{pw}$ 의 비이다. 한편 식 (1)로부터  $A_{pw}$ 는 다음과 같이 표현된다.

$$A_{pw} = \sum RS_j - \left[ 1 + \frac{wr}{100} \right] \cdot \sum AS_i \quad (3)$$

따라서, 원자재의 효과적인 정렬방향을 결정하기 위해서 원자재의 여러 정렬방향( $\phi$ )에 대해 배치될

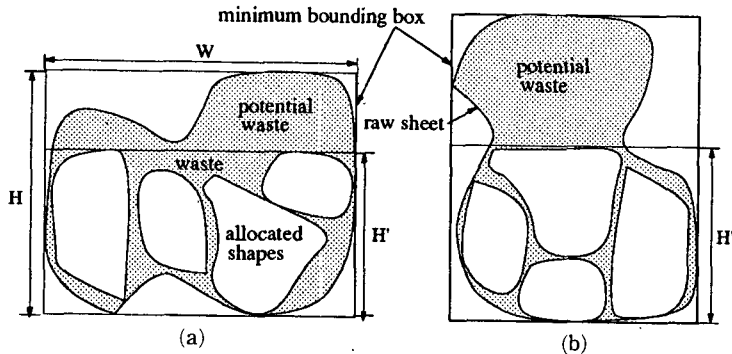


Fig. 1 Example of potentially usable wastes for two different arrangements of the raw sheet

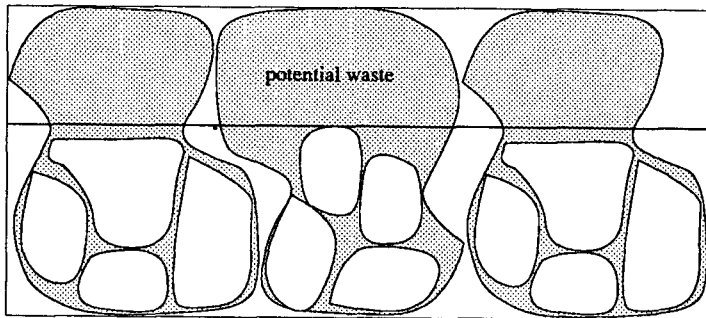


Fig. 2 Example of potentially usable waste for three raw sheets arranged horizontally

형상들을 Fig. 5와 같이 실제로 배치해야 하는 수고 없이, 원자재의 형상과 이에 배치될 형상들이 주어지면 식 (2), (3)으로부터 원자재의 정렬방향  $\phi$ 에 대해서  $wr$  변화에 따른 PVW의 변화추이를

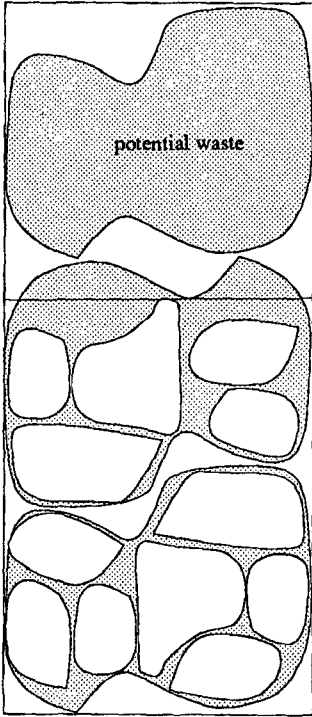


Fig. 3 Example of potentially usable waste for three raw sheets arranged vertically

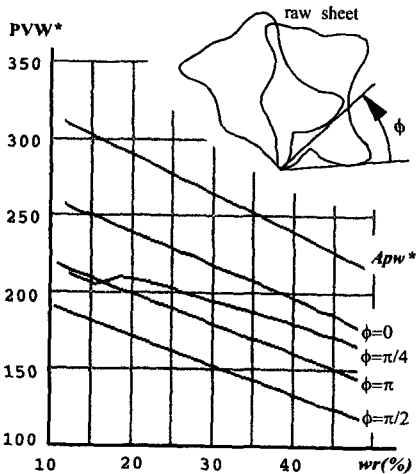


Fig. 4 Pre-estimated PVW-wr diagram for 26 shapes allocation on four different arrangements of a given raw sheet

Fig. 4와 같이 예측할 수 있다. Table 1에서 보는 바와 같이, 동일한 배치 알고리즘으로도 원자재의 서로 다른 정렬방향에 대해서는 최종 배치효율은 일반적으로 다를 수 밖에 없다. 그럼에도 불구하고 Fig. 4와 같은 예측자료는 여러 정렬방향 중 과연 어떤 방향으로 정렬한 원자재 위에서의 배치가 가장 효과적인 것인가에 대한 중요한 정보를 제공함을 알 수 있다. 본 예에서는 4가지 정렬방향중  $\phi = 0$  경우가 주어진 형상들의 배치에 가장 효과적임을 보인다.

이상의 고찰을 바탕으로, 본 연구에서는 정렬된 전체 원자재를 둘러싸는 최소 크기의 직사각형을 MBB(minimum bounding box)라 할때, (1) 원자재 사이의 중첩이 없는 한 MBB의 면적은 가능한 한 작고, (2) MBB의 상변 쪽에 가능한 한 적은 수의 원자재가 정렬되고, (3) MBB의 두변중에서 작은 것을 밑변으로 하는 정렬상태가 되도록 원자재를 정렬하여 사용하였다. 동시에 사용되는 원자재의 개수가 1~5개 정도인 실제 문제에 있어서는 원자재의 효과적인 정렬이 위와 같은 기준에 의거한 대화식 작업(interactive work)으로 빠른 시간 내에 결정된다.

### 2.3 격자모델 및 가상 원자재의 격자화

#### 2.3.1 격자모델

배치형상의 표현방법으로는 다각형 근사모델과 격자모델이 있다. 다각형 근사모델을 사용할 경우에는 배치상황의 위치 정보탐색, 즉 배치형상간의 중첩여부 검사 및 중첩면적의 계산, 원자재 경계의 이탈여부 검사 등을 위하여 교선계산과 같은 해석적 기법이 이용된다. 이러한 계산작업은 수많은 배

Table 1 Summary of the allocation results shown in Fig. 5 (\* : Dimensionless values)

Fig. 5	(a)	(b)	(c)	(d)
PVW *	224.0	194.9	158.7	190.1
$A_{pw}$ *	270.7	264.5	272.9	278.1
$L_{pw}$ *	70.5	78.3	100.8	86.5
wr (%)	27.7	30.6	25.8	24.0
cpu(sec)	443.7	460.9	529.1	508.7

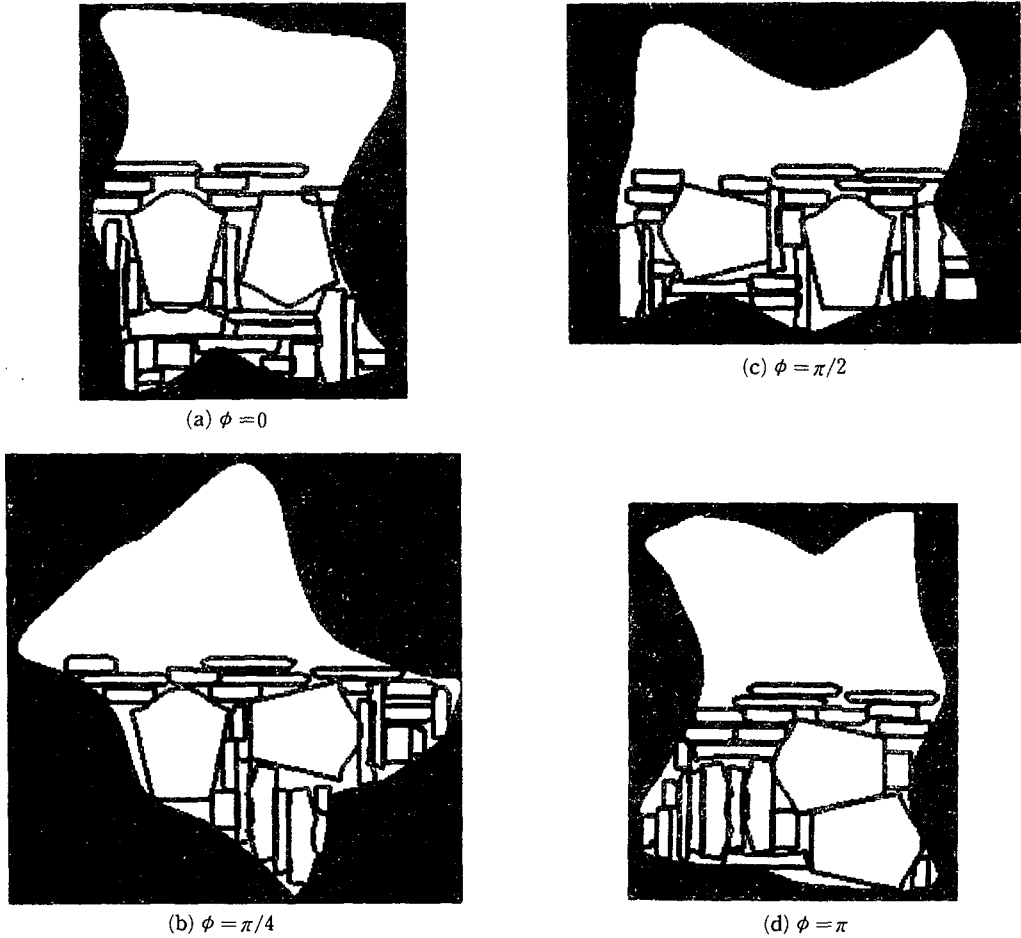


Fig. 5 Actual allocations of 26 shapes on the raw sheet shown in Fig. 4 arranged

치상황의 개선을 필요로 하는 배치 알고리즘에서는 컴퓨터의 실행시간 면에서 과중한 부담이 된다. 반면에 격자모델을 사용할 경우에는 위와 같은 배치 상황에 대한 위치정보탐색작업이 간단한 격자정보 탐색작업으로 전환될 수 있다. 따라서, 본 연구에서 처럼 많은 탐색작업을 요구하는 SAA를 사용할 경우는 다각형 근사모델보다 격자모델이 훨씬 유리하다.

이에 따라 본 연구에서는 원자재 및 그 위에 배치될 형상을 모두 격자화하여 사용하였고, 격자모델을 위한 자료구조는 (1) 형상의 배치 및 제거작업과 중첩검사의 신속성, (2) 형상 저장방법의 효율성 등을 고려하여 설계되었다.<sup>(7,8)</sup> 또한 형상의 배치방향에는 제한을 두어 사용자가 지정하는 각도 증분만큼의 회전만 허용하였다. 그러나, 배치 방향이 바뀔 때마다 매번 회전 격자모델을 새로 생성<sup>(6)</sup>

하는 대신, 배치될 형상의 모든 회전가능방향에 대하여 회전된 기준 격자모델을 한번만 생성시켜 이를 저장하고 있다가, 필요할 때마다 이를 참조하여 배치, 제거 및 중첩검사를 수행하도록 하였다.

본 연구에서 개발된 격자모델의 부가적 이점은, 별도 알고리즘의 추가없이, 내부에 구멍이 있는 형상의 격자화 및 그 구멍속에 다른 작은 형상을 배치하는 문제를 일관되게 처리할 수 있는데 있다. 이에 반하여, 현재 대부분의 상용 시스템의 근간이 되고 있는 Adamowicz와 Albano의 연구<sup>(21)</sup>에서는 내부에 구멍이 있는 형상에 대해서는 특별히 고안된 전처리 알고리즘을 통해 내부 구멍에 배치 시킬 수 있는 형상을 찾아 이를 그 구멍 안에 배치 시킨 후 이 형상들을 한데 묶어 하나의 형상(clustered shape)으로 취급하는 다단계 배치방법을 필요로 한다.

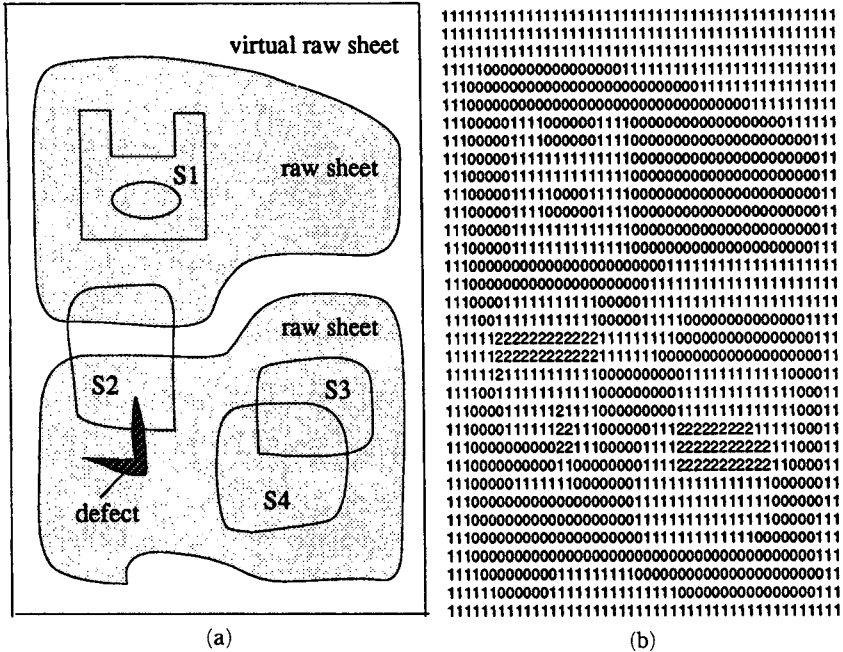


Fig. 6 Example of (a) allocating shapes on two irregular raw sheets and (b) resulting grid representation

2.3.2 가상 원자재의 격자화

원자재의 효과적인 정렬이 결정된 후 원자재의 내부결합, 경계의 불규칙성 및 2개 이상의 원자재 사용 등을 효과적으로 다루기 위하여 다음과 같은 가상 원자재(virtual raw sheet)의 개념을 도입, 사용하였다.

원자재의 정렬이 완료되면 이때의 MBB를 가상 원자재라 한다. 가상 원자재에는 Fig. 6(a)에서 보는 바와 같이 한개 혹은 그 이상의 실제 원자재에 대응하는 가용한 배치영역이 있고, 실제 원자재 경계밖과 내부결합부에 대응하는 배치불가 영역이 있다. 이는 가상 원자재를 다음과 같은 격자모델로 나타냄으로써 용이하게 정의된다. 즉,  $m, n(m < n)$ 을 각각 가상 원자재의 가로 및 세로에 대응하는 격자수라 할 때 가상 원자재 MBB는 일정 크기의 이차원 격자 sheet  $[m][n]$ 로 정의되고, 각 격자는 Fig. 6(b)에서와 같이 0 이상의 정수로 표현되는 격자값을 갖는다. Sheet  $[m][n]$ 는 배치불가 영역은 1, 배치가능 영역은 0으로 초기화된다. 프로그램상에서 형상의 배치 또는 제거는 그 형상의 격자 모델을 배치 혹은 제거하고자 하는 위치에 이동시켰을 때 이에 대응하는 가상 원자재의 각 격자의 격자값을 1씩 증가 혹은 감소시키는 일에 해

당한다. 따라서, 형상들의 배치가 진전됨에 따라 이들 격자값은 해당 위치에 배치된 형상의 수만큼 증가한다. 가상 원자재의 어느 특정격자의 격자값이 2이상이면 이는 그 위치에 격자값 만큼의 형상들이 중첩되어 배치된 것을 의미하거나, 배치불가 영역에 격자값보다 1개 작은 수에 해당하는 형상들이 배치되었음을 뜻한다.

2.4 SAA를 이용한 최적배치

Fig. 7은 본 연구에서 구현되고 있는 SAA를 이용한 최적배치의 개괄적인 흐름을 나타낸 것이다. Fig. 7(a)는 초기배치상황이고, 이로부터 SAA를 통한 확률통계론적 탐색방법에 의해 배치상황의 갱신이 무수히 시도되고, Fig. 7(b)는 이러한 최적화 과정에서 얻어질 수 있는 어느 한 배치상황을 나타낸 것이다. 그러나, 버림률이 줄어들어 따라 Fig. 7(c)와 같은 배치상황에 도달하면 최적화의 관점에서 볼때 커다란 국지해(local minima)에 빠져 배치상황의 효과적인 갱신을 더이상 기대할 수 없는 상황에 봉착할 수도 있다. Fig. 7(d), (e)는 이러한 국지해로부터 탈출하여 궁극적으로 도달할 수 있는 개선된 여러 배치상황 중의 일례를 나타낸 것으로서, 본 연구에서는 이를 위하여 특별히 '국부

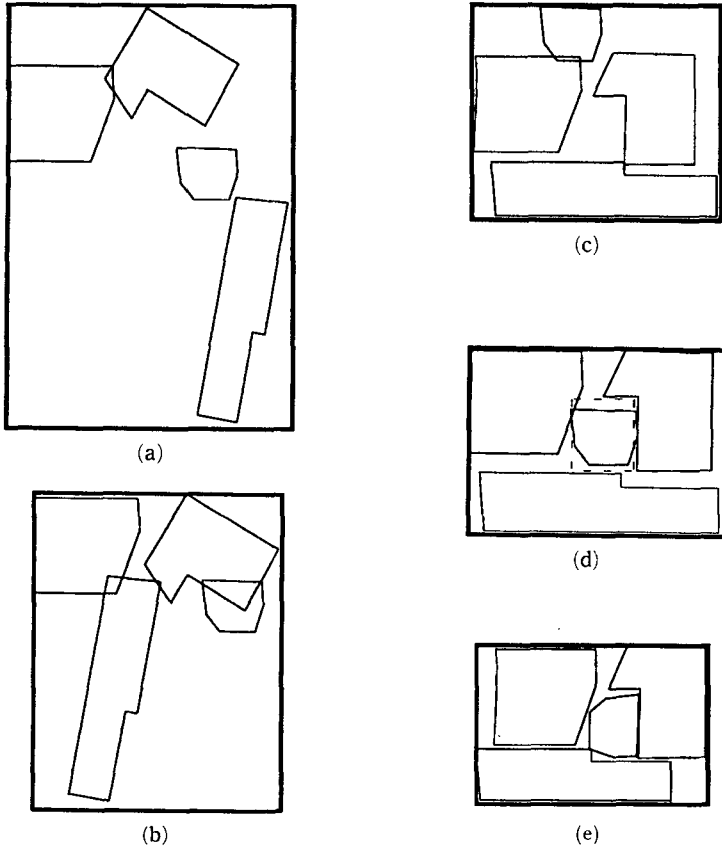


Fig. 7 Schematic view of the proposed nesting algorithm

플림을 통한 간극채워넣기(gap-filling with local annealing, GFLA)'라는 새로운 개념을 시도하였다. 위에서 개괄적으로 소개된 최적배치방법을 구현하기 위한 기본적인 구성은 (1) 초기배치, (2) 목적함수의 정의, (3) 이웃해의 생성 및 배치상황의 갱신, (4) 온도변화 방법 및 종료조건, (5) GFLA 등으로 나눌 수 있다. 본 절에서는 위의 (1)~(4)항에 대해서 구체적으로 소개하고, 제 (5)항에 대해서는 다음의 2.5절에서 상세히 다루고자 한다.

2.4.1 초기배치

배치되어야 할 모든 형상에 대해 앞 절에서와 같이 기준 격자모델을 완성한 후, 이미 격자화되어 있는 가상 원자재에 모든 형상을 Fig. 7(a)와 같이 임의의 위치와 방향을 갖도록 한꺼번에 배치한다. 이때 형상과 형상사이에는 중첩이 일어날 수도 있다. 다만, 각 형상들에는 가상 원자재 경계를 벗어나지

않는 범위에서 무작위로 배치위치 및 방향이 부여된다. 이렇게 각 형상의 초기위치 및 방향이 결정되어 원자재에 배치된 상황을 초기배치상황이라 한다.

2.4.2 목적함수

최적화작업은 앞의 초기배치로부터 목적함수를 개선 또는 감소시키는 새로운 배치상태를 반복하여 탐색하는 과정이다. 이를 위해 사용된 목적함수는 다음과 같다.

$$C(S) = Scrap + w * Overlaparea \tag{4}$$

여기서, S는 현재의 배치상황을 나타내고, C(S)는 이때의 목적함수 값이다. Scrap은 S의 배치영역(직사각형 W\*H)속의 원자재로부터 모든 배치형상을 가공해낸 후 남게되는 원자재의 찌투리 양을 나타낸다. 이는 기본적으로 위의 직사각형 속의 격자값이 0인 모든 격자의 수에 해당한다.

Overlaparea는 중첩부위의 면적, 원자재 경계바깥으로 배치된 면적 및 내부결함부에 배치된 면적의 총합으로서 가상 원자재의 격자값이 2이상인 모든 격자수에 해당한다. 본 연구에서는 최적배치를 위한 탐색도중에는 형상간의 중첩과 배치불가 영역에의 배치 등과 같은 부적절한 배치상황을 허용한다. 그러나, 최종배치 결과에서는 이러한 부적절한 배치상태가 나타나지 않도록 하기위하여 가중치  $w$ 를 도입하였다.

2.4.3 배치상황의 갱신

Fig. 8은 본 연구에서 사용하는 이웃해 생성법으로서, 현재의 배치상태  $S$ 에서 임의로 한 형상을 선택하여 그 형상의 기존 배치정보(배치위치 및 방향)을 기준으로 이에 작은 교란을 임의로 부여하여

Procedure Get\_Neighbor()

```

id := (int) r[0,nshape] /* one shape selected randomly */
Δx := r[-Δx,Δx] /* perturbation on x */
Δy := r[-Δy,Δy] /* perturbation on y */
Δθ := r[-δθ,δθ] /* perturbation on θ */

x := x0 + Δx /* r[a,b]: random no. between a and b */
y := y0 + Δy /* Δx, Δy, Δθ: perturbation upper bounds */
θ := θ0 + Δθ
    
```

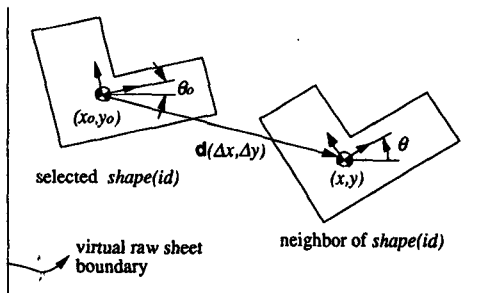


Fig. 8 Neighborhood generation mechanism

```

begin
  Initialize system parameters ;
  S := Initial random configuration S0 ;
  k := 1 ;
  Tk := Initial temperature T0 ;
  while (stop condition is not satisfied) do
    begin
      while (not yet in equilibrium) do
        begin
          S' := Get neighboring configuration of S ;
          ΔC := C(S') - C(S) ;
          Prob := min(1, exp(-ΔCTk)) ;
          if random(0,1) < Prob then S := S' ;
        end ;
        Tk+1 := α * Tk ;
        k := k+1 ;
      end ;
      Print out best configuration ;
    end ;
  end ;
    
```

Fig. 9 Pseudo code of SAA for the optimal nesting

생성된 새로운 배치상태  $S'$ 를  $S$ 의 이웃해라 한다. 배치상황의 갱신은  $S$ 를 새로 생성된 이웃해  $S'$ 로 대체하는 작업이다. 즉, 목적함수가 감소하면 ( $\Delta C = C(S') - C(S) < 0$ ),  $S'$ 를 새로운 현재의 상태  $S$ 로 받아들이고, 이로부터 최적해에 보다 더 가까와지기 위한 또 다른  $S'$ 를 계속 탐색한다. 특기할 사항은  $\Delta C > 0$ 인 경우에도  $S'$ 는 항상 거부되는 것이 아니고, Fig. 9에서 보는 바와 같이 SAA가 갖는 힐 클라이밍 특성에 따라 목적함수가 증가하는 방향으로 배치상황의 갱신이 확률적으로 허용된다는 것이다.<sup>(22)</sup>

2.4.4 온도변화 방법 및 종료조건

Fig. 9는 최적배치문제를 SAA를 이용하여 구현할 경우의 일반적인 개념도이다. 그림에서와 같이 온도  $T_k$ 에서 반복되는 배치상황의 갱신으로 목적함수가 평형상태에 도달하면 온도  $T_k$ 를 새로운 온도  $T_{k+1}$ 로 변화시켜, 목적함수가 새로운 평형상태에 도달할 때까지 배치상황의 갱신을 반복한다. 이때, 이웃해에 대한 탐색이 대략 1000회 이상 시도 되었으면 평형상태에 도달한 것으로 처리하였고, 온도변화는  $T_{k+1}/T_k = 0.8 \sim 0.95$ 로 하였다. 또한 (1) 일정한 탐색회수 동안 목적함수의 최소값이 전혀 개선되지 않았거나, (2) 온도가 지정된 최저온도  $T_{min}$ 보다 작아지면 전체 최적화과정은 종료되도록 하였다.

2.5 크기별 배치 및 국부풀림을 통한 간극채워 넣기

배치효율 향상을 위한 발견론적 방법(heuristic method)으로써, 본 연구에서 검토된 것으로는

- 상호위치교환(pairwise exchange)
- 격자크기의 이원화(dual grid size)
- 냉각시 재열과정의 도입(reheating)
- 크기별 배치(groupwise allocation)
- 국부풀림을 통한 간극채워넣기(gap-filling with local annealing, GFLA)

등이 있다. 이웃해 생성방법의 일종인 상호위치교환 방법은 임의로 선택된 두 형상의 현재의 배치위치를 서로 바꾸는 것으로써 최적화 초기에는 다소 효과가 있으나 최적화가 상당히 진전되어 형상간의 거리가 매우 가까워지면 상호위치교환이 이루어질 확률은 극히 작아져 이 방법을 통한 배치효율 향상은 더 이상 기대할 수 없었다. 격자크기의 이원화



는 최적화의 초기에는 비교적 큰 격자로 표현된 격자모델을 사용하여 빠른 배치를 수행하고, 최적화 후반부에서는 작은 격자모델로 형상과 원자재 경계 등을 보다 정확히 기술하여 배치의 세부 조정을 얻고자 하는 방법이다. 그러나, 본 연구에서는 여러 번의 실험을 통해 이 방법에 의해 초기단계에서 단축된 실행시간은 전체 실행시간과 비교할 때 큰 비중을 차지하지 않음을 확인하였다. 재열과정의 도입은 SAA의 냉각방법의 변형으로서 최적화과정중 커다란 국지해에 빠졌을 경우 이를 탈출하여 더 나은 해를 탐색하기 위하여 보다 큰 교란도 허용할 수 있도록 현재의 온도를 상향 조정하는 방법이다. 그러나, 이 방법의 구현에는 실행시간의 증가 뿐만 아니라 과연 어느 단계에서 어떻게 국지해에 빠졌나를 판단할 것인가하는 어려움이 있다.

이상의 실험과 고찰을 통해 본 연구에서는 새로운 배치효율향상 방안으로써 크기별 배치 및 GFLA 방법을 개발 사용하여 좋은 효과를 보였다. 특히, GFLA 개념은 본 연구에서 처음 고안된 것으로서 그 기본적인 원리는 다음과 같다.

먼저 여러가지 크기의 물건을 큰 상자 속에 넣는 경우를 생각해 보자. 이때 큰 것들을 먼저 넣은 뒤 작은 물건들을 큰 것들 위에 올려놓거나 큰 것들 틈바구니에 적절히 끼워넣고, 상자를 조금씩 흔들어 주면 작은 것들이 큰 것들 틈바구니에서 스스로 제자리를 찾아가는 현상을 경험하거나 이해할 수 있을 것이다. 이런 현상의 예로는 도목공사에서의 토사 매립 후의 흩다짐작업이나 콘크리트 타설시의 진동자(vibrator)의 사용에서도 볼 수 있다.

이상에서 고찰된 기본개념의 구현을 위해 본 연구에서는 배치형상들을 크기별로 큰 형상그룹 g1 과 작은 형상그룹 g2로 나누어 g1 그룹을 일반적인 SAA 기법으로 먼저 배치한 후, 이들 배치형상 사이의 찌투리영역에 g2 그룹의 형상들을 GFLA를 이용하여 끼워넣기 식으로 배치한다. 다음에 형상의 분류방법과 GFLA의 구체적인 구현 방법이 소개된다.

### 2.5.1 끼워넣기 대상 형상의 분류

큰 형상그룹 g1에 속한 형상만으로 배치가 수행되어 이들의 배치영역(직사각형  $W*H'$ ) 속에 남은 찌투리영역에 작은 형상그룹 g2가 모두 배치되어 찌투리가 하나도 남지 않는 경우가 가장 이상적인 배치가 될 것이다. 본 연구에서는 위와 같은 이상

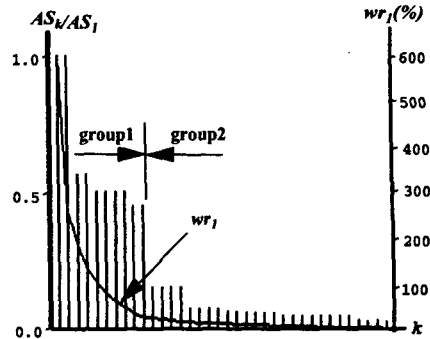


Fig. 10 Distribution of shapes' size and variation of  $wr_1$  calculated by equation (5) for  $n_1=1, 2, \dots, n$  for an example of 36 patterns allocation

적인 배치상황을 상정하여, 다음과 같은 방법으로  $g_1, g_2$ 에 속하는 형상의 개수  $n_1, n_2$  를 결정하였다.

$$Scrap = \frac{wr_1}{100} \cdot \sum_{i=1}^{n_1} AS_i = \sum_{j=n_1+1}^n AS_j$$

$$n_1 + n_2 = n \text{ with integer } n_1, n_2 \geq 0 \quad (5)$$

여기서,  $wr_1$ 은 그룹  $g_1$ 의 버림률이고,  $n$ 은 전체 형상의 개수이다.  $AS_i$ 는  $i$ 번째 형상의 면적으로 모든 형상들은  $AS_i \geq AS_{i+1}$ 을 만족하도록 크기 순으로 정렬하여 사용한다. 위의 식(5)는 부정정계로 식(5)만으로는  $n_1, n_2$  및  $wr_1$ 을 유일하게 결정할 수 없으나, 일반적인 최적배치 알고리즘으로 무리없이 도달할 수 있는 배치효율은  $wr_1=0.25 \sim 0.35$  정도로 보고, 이러한 범위에서 식(5)를 이용하여  $n_1, n_2$ 를 결정할 수 있다. Fig. 10은 제 3 장의 실험 예에서 다룬 36개 형상의 크기별 분포와 식(5)에서  $n_1$ 의 변화에 따른  $wr_1$ 의 변화를 나타낸 것이다. 이 실험 예에서는 36개 형상들을  $n_1=10, n_2=26$ 의 두 그룹으로 나누어 사용하였고, 이때 식(5)에 의한  $wr_1$ 은 27.9% 였다.

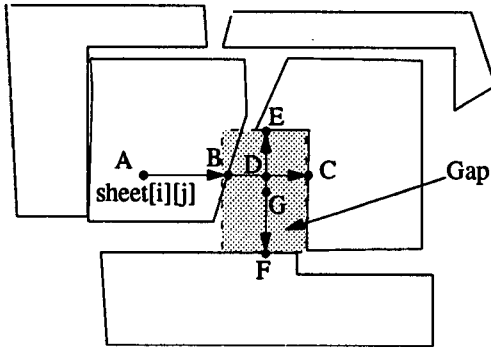
배치형상들의 개수가 많고, 각 형상간의 크기의 변화가 현저할 경우에는 위의 그룹별 배치전략과 작은 형상들의 끼워넣기식 배치가 효과적인 전략이 된다. 이 경우 식(5)는  $g_1, g_2$  구분을 위한 유용한 기준으로 사용될 수 있다.

### 2.5.2 국부풀림을 통한 간극채워넣기

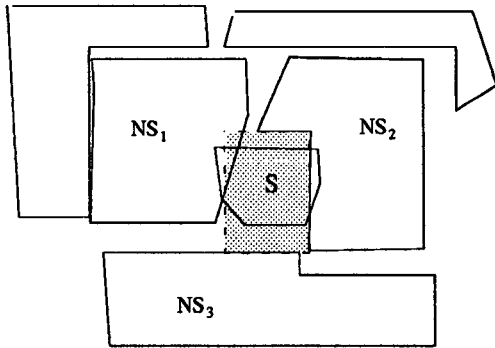
GFLA의 기본개념은 위에서 소개된 개념을 해석적으로 접근하기 보다는 있는 그대로를 모사(simulation)하는 데 있다. 즉, GFLA는 이미 배

치되어 있는 큰 형상사이의 짜투리영역 중 직사각형의 잠재적 가용영역을 간극(gap)이라 정의하고, 작은 형상을 이 간극에 끼워넣기식으로 배치하는 방법으로써 그 구체적인 구현방법은 다음과 같다.

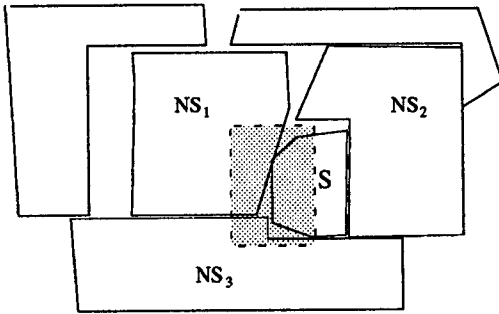
작은 형상그룹  $g_2$ 로 분류된 모든 끼워넣기 대상 형상들을 큰 형상부터 다음 단계에 따라 간극 채워넣기를 수행한다.



(a)



(b)



(c)

**Fig. 11** Gap-filling with local annealing (a) gap searching (b) gap-filling by positioning the small shape at the gap center (c) final result after local annealing

1단계 : 간극 탐색(gap-search) : Fig. 11(a)처럼 임의로 선택된 원자재 위의 격자점  $sheet[i][j]$ , ( $i = 1, 2, \dots, m, j = 1, 2, \dots, n$ )로부터 이의 격자값이 0이면  $i$ 만 감소시켜 (격자값이 0이 아니면  $i$ 만 증가시켜) 짜투리의 경계 B를 찾는다. B로부터  $i$ 만 증가시켜 짜투리의 또 다른 경계 C를 찾는다. B, C의 중점 D를 짜투리의 좌우경계의 중점이라 한다. D로부터  $j$ 값만 변화시켜 짜투리의 상하 경계 E, F를 찾고, 이들의 중점 G를 짜투리의 상하 경계의 중점이라 한다. G로부터 다시  $i$  값만 변화시켜 새로운 D를 구하고 이로부터  $j$  값만 변화시켜 새로운 G를 구한다. 이렇게 짜투리의 좌우 및 상하 경계의 중점을  $D \rightarrow G \rightarrow D \rightarrow G \rightarrow \dots$ 처럼 순환적으로 반복하여 구하면 적절한 모양의 짜투리 영역에서는 이들 두 중점 D, G는 서로 가까워진다. 따라서 지정된 회수이내의 반복으로 D, G 사이의 거리가 허용치 이내로 수렴하면 하나의 간극이 찾아진 것으로 본다. (실제의 프로그램 수행에서는 대략 5번 이내의 반복으로 두 중점간의 거리가 격자단위 1 이하로 수렴되는 간극을 쉽게 탐색할 수 있었다.) 이때의 새로운 경계점 B, C, E, F를 각 변의 중점으로 하는 직사각형을 간극이라 한다. 이렇게 탐색된 간극이 현재의 끼워넣기 대상 형상 S를 수용할 수 있을 정도로 크면 다음의 2단계) 작업을 수행한다. 탐색된 간극이 S를 수용할 수 없을 경우 이 간극은 취소되고 앞의 1단계)부터 새로운 간극 탐색이 반복된다. 지정회수 만큼의 반복탐색으로도 S를 수용할 수 있는 간극을 찾을 수 없으면 형상 S을 위한 간극탐색은 실패한 것으로 보고, S보다 작은 다음 형상에 대해 앞의 1단계)부터 새로 시작한다. Fig. 12(a), (b)는 각각 간극탐색 직전의 어느 배치상황과 이 상황에서 S를 수용할 수 있는 간극들이 빗금친 부분처럼 찾아졌음을 보인다. 그림에서도 알 수 있듯이 가용한 간극이 거의 모두 찾아졌고, 결과도 양호함을 알 수 있다.

2 단계 : 간극채워넣기(gap-filling) : 위에서 찾아진 간극의 도심과 현재의 간극채워넣기 대상형상인 S의 도심을 일치시켜 Fig. 11(b)와 같이 S를 간극에 배치한다. 이때 S가 간극주위에 이미 배치되어 있는 주변형상(neighboring shapes,  $NS_j, (j=1, 2, \dots, j_n)$ )들과 중첩을 일으키면 다음 단계를 수행한다.

3단계 : 국부풀림(local annealing) : Fig. 11(b)처럼 형상 S와 이와 관계하는 기존의 배치형상  $NS_j$

만으로 국부적인 배치조정을 SAA로 반복 수행하여 중첩을 해소하는 것을 국부풀림이라 한다. 이때 g1 그룹에 속한 형상들은 병진이동만이 허용되고 g2 그룹에 속한 형상들은 회전이동까지 할 수 있도록 한다. 이는 간극채워넣기 단계에 이르면 형상 사이의 간격이 매우 가까워져서 큰 형상의 조그만 회전도 쉽게 주위의 다른 형상과 중첩을 유발할 수 있기 때문이다. 만약 국부풀림 후에도 중첩이 해소되지 않으면, 그 위치에 대한 S의 간극채워넣기는 취소된다. 이 경우 형상 NS<sub>i</sub> 들은 본래의 배치상태인 Fig. 11(a)로 되돌아가고, S를 위한 새로운

간극탐색이 1단계)부터 다시 시도된다.

작은 형상그룹 g2의 모든 형상에 대해서 위의 1~3단계를 수행하고 나면 전체 GFLA 작업은 종료되고 전체 형상을 대상으로 섬세한 배치조정 작업이 본래의 SAA를 이용하여 수렴조건이 만족될 때까지 수행된다.

### 3. 적용 예 및 고찰

Fig. 13은 본 연구에서 구현된 최적배치 프로그램의 전체적인 흐름도로서, 크기별 배치가 고려되

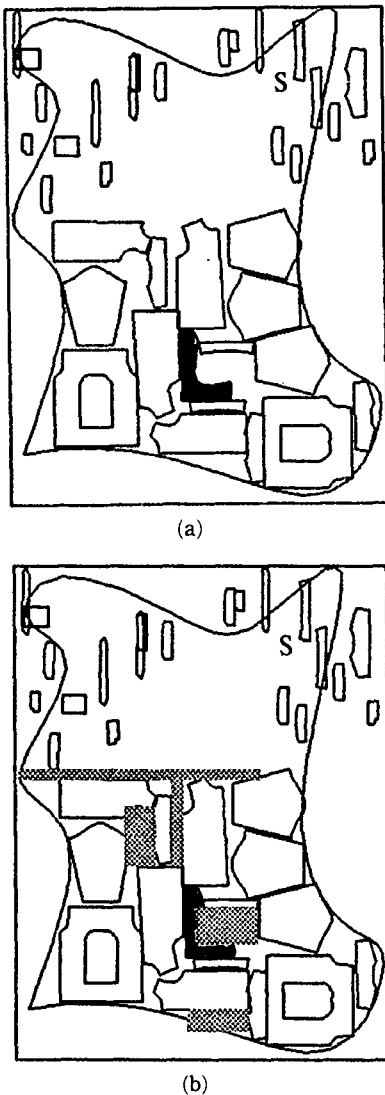


Fig. 12 Example of gap-searching results

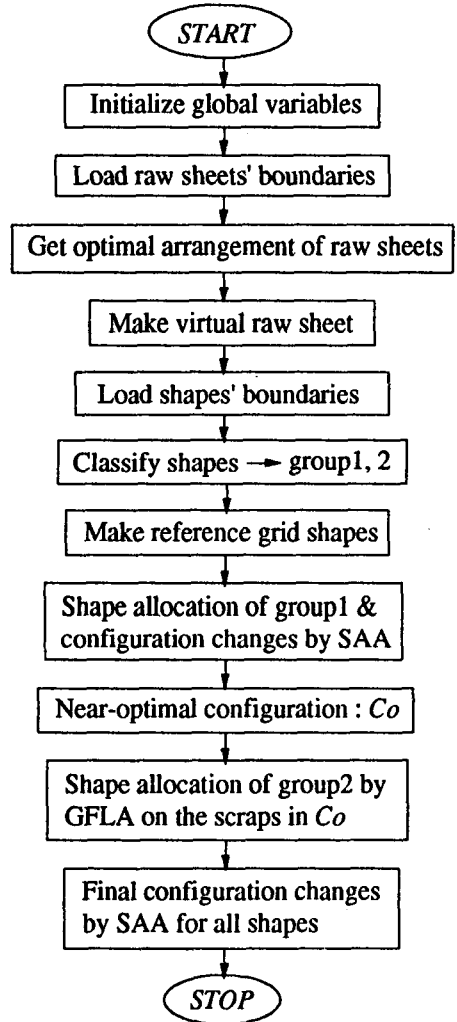


Fig. 13 Overall flow chart for the nesting using groupwise allocation & GFLA

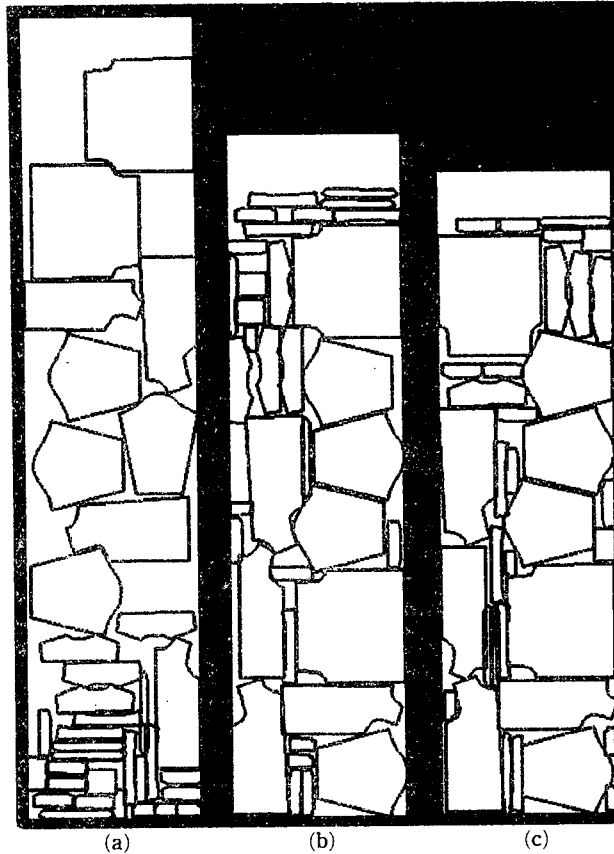


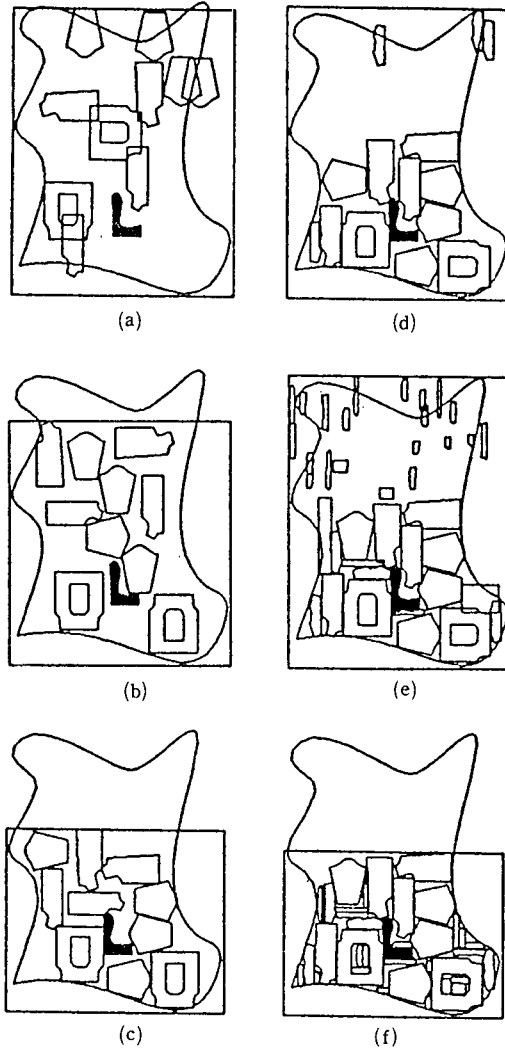
Fig. 14 Nesting of 36 patterns on the raw strip of constant width (a) without groupwise allocation (b) groupwise allocation followed by gap-filling without local annealing (c) final result by groupwise allocation & GFLA (wr= 10.5%, cpu=1957 sec on 16 Mips EWS)

어 큰 형상그룹이 SAA를 이용하여 먼저 배치되고, 이들의 배치가 수렴되면 그 위에 작은 형상 그룹이 GFLA에 의해 배치된다.

Fig. 14는 36개 형상을 일정 폭의 원자재 위에 배치한 비교 예이다. 실제 형상에 대한 데이터는 삼성물산(주)에서 36개 옷감모형의 배치(pattern marking)에 사용되었던 것이다. 본 실험 예에서는 이들을 1cm×1cm 크기의 격자를 기준으로 격자 모델로 변환하여 사용하였다. Fig. 14(a)는 36개 형상 모두를 동시에 고려하여 최적배치 해를 찾아가는 도중 일종의 커다란 국지해에 빠져 목적함수의 효과적인 개선이 크게 기대되지 않는 상황을 보이는 것이다. 그림에서도 나타나듯이 크기가 작은 형상들이 원자재 하단에 몰려있는 특징을 보인다. 이런 현상은 큰 형상들의 움직임은 짜투리가 줄어들수록, 즉 최적해에 가까워질수록 주위의 다른 형

상과 중첩을 유발하기 쉬운 반면에, 작은형상들의 이동확률은 상대적으로 커서 보다 많은 위치 이동이 수행될 수 있었기 때문으로 보인다. 그러나, 결과적으로는 작은 형상들이 많이 모인 하단의 기준선 부분의 국부적 짜투리 비율은 줄어드나 커다란 형상들만 남은 상단부에는 상대적으로 큰 짜투리 비율이 발생하는 문제점을 보인다.

이러한 문제점을 해결하기 위하여 Fig. 14(b)에서는 크기별 배치 및 작은 형상에 대한 간극채워넣기 개념이 도입되어 Fig. 14(a)보다 개선된 배치 결과를 보인다. 그러나, Fig. 14(b)에서는 중첩이 발생하더라도 이를 해소하기 위한 국부풀림이 적용되지 않은 경우로서 간극채워넣기가 성공될 확률이 저조하였다. 이에 비해, Fig. 14(c)는 간극채워넣기때 발생할 수 있는 중첩문제를 해소하기 위해 국부풀림까지 적용시킨 것으로써 가장 좋은 결과를



**Fig. 15** Nesting of 36 patterns on the irregular sheet having interior defect (a) initial random allocation of large shapes (b) ~ (c) interim configurations by SAA (d) ~ (e) GFLA are applied for small shapes when large shapes are in the near-optimal configuration (f) final result ( $wr = 27.2\%$ ,  $cpu = 869$  sec on 16 Mips EWS)

보인다. 이와같이 최적해에 가까워짐에 따라 GFLA가 배치효율 향상에 상당한 효과를 보임을 알 수 있다.

Fig. 15는 36개 형상을 내부결함(검게 칠해진 부분)이 있고 경계가 불규칙한 원자재 위에 배치한 결과이다. 이때의 원자재의 정렬방향은 각 형상들의 배치에 앞서 또다른 최적화과정을 통하여 이미 결정된 것이다. 큰 형상들의 초기배치상황 Fig. 15 (a)로부터 SAA를 통하여 Fig. 15(b), (c)와 같이 큰 형상들의 배치상황의 갱신이 거듭되어 Fig. 15

(d)와 같이 큰 형상들의 배치상황이 수렴되면 작은 형상들에 대한 GFLA가 Fig. 15(d), (e)에서와 같이 시작된다. Fig. 15(f)는 GFLA가 종료되고 모든 형상에 대한 SAA가 수행되어 얻은 최종 해이다. 참고로 Fig. 15(f)의 배치효율  $wr$ 은 그림의 직사각형 좌측 상단의 커다란 짜투리 면적(이는 형상 사이의 실제 짜투리 면적의 총합과 거의 비슷함)을 포함하여 식(1)로부터 계산된 것으로서, 이점을 고려해 볼때 임의의 형상의 원자재 위에서의 최적배치를 위해 제시된 본 연구의 배치 알고리즘은 배치효



**Fig. 16** Nesting of 56 patterns on three irregular sheets one of which has an interior defect (wr =26.3%, cpu=1198 sec on 16 Mips EWS)

을 면에서 매우 만족스런 결과를 보인 것으로 사료된다.

Fig. 16은 56개 형상을 한개의 원자재에서 재단해 낼 수 없어서 3개의 원자재가 사용된 예이다. 여기서도 앞의 예제에서와 같이 형상들의 배치에 앞서 3개 원자재의 정렬방향이 최적화과정을 통하여 이미 결정된 것을 사용하고 있다. 이 예제에서도 크기별 배치 및 GFLA가 수행되었다.

위의 두 예제에서 볼 수 있듯이 원자재 내부에 결함이 있는 경우 이를 피하여 배치가 이루어지고, 형상의 내부에 가용한 배치영역이 있으면 그곳에 다른 작은 형상을 별도의 추가 알고리즘없이 배치할 수 있음을 보인다.

C-언어로 구현된 본 배치 알고리즘은 프로그램의 실행시간 면에서 특별히 최적화되어 있지는 않다. 일례로, 위의 실험 예에서의 실행시간은 최적화과정에서 나타나는 수 많은 새로운 배치상태를 매 10번 단위로 새로 그리도록 하는 시간을 포함하고 있다.

#### 4. 결론 및 향후 과제

본 연구에서는 임의의 2차원 형상의 효과적인 배치를 위하여 배치형상을 격자화하여 사용하였고, 주요 최적화기법으로는 조합 최적화문제 해결에 있

어 최근 주목받고 있는 폴림모사기법(SAA)을 이용하였다. 격자화모델을 사용함으로써 최적화 과정 중 수없이 검토해야 할 배치형상간의 중첩검사 문제를 용이하게 처리할 수 있었고, 형상의 내부에 유용한 배치공간이 있을 경우 이곳에 작은 형상을 배치하는 방법과, 피혁제품의 배치에서 볼 수 있는 경계가 불규칙하고 내부에 결함이 있는 경우 및 두 개 이상의 원자재 위에서의 효과적인 배치에 있어 본 연구에서 개발된 배치 알고리즘의 유용성을 확인하였다.

한편, SAA를 최적배치문제에 그대로 이용할 경우 배치문제의 일반적인 특성때문에 극지해에 쉽게 빠져 배치상황의 개선이 매우 어렵다는 것을 여러 번의 실험을 통해 알 수 있었다. 따라서, SAA를 최적배치문제에 효과적으로 적용하기 위해선 커다란 극지해로부터의 적절한 탈출방안이 모색되어야 한다. 본 연구의 크기별 배치방법이나 '국부풀림을 통한 간극채워넣기' 개념의 도입은 이러한 문제점의 대처 방안으로 고려된 것으로써 배치효율면에서 괄목할 만한 개선효과를 얻을 수 있었다. 특히, 국부풀림을 통한 간극채워넣기 방법은 형상의 개수가 많고 형상크기의 변화가 현저 할때 유용한 배치전략임을 확인 하였다.

앞으로의 연구에서는 SAA의 효과적인 냉각스케줄에 대한 연구와, 격자크기의 다변화 혹은 신경회로망의 도입 등을 통한 배치시간 및 효율향상에 대한 연구가 필요하다. 또한, 배치형상들을 스캐너 혹은 비디오카메라의 영상정보로 직접 입력받아 처리하는 인터페이스 개발에 대한 연구가 남아 있다. 이들 연구에서는 격자모델이 갖는 특성이 크게 도움이 될 것으로 기대된다. 또한, 본 연구결과를 생산현장에서 바로 이용하기 위해서는, 재단시간 및 가공여유를 고려한 최적의 재단경로를 산출하는 문제가 남아 있다.

#### 후 기

본 연구는 한국과학재단 지원 '94년 핵심전문연구(과제번호 941-1000-061-2)의 일부로 수행되었으며, 이에 관계자 제위께 감사드립니다.

#### 참고문헌

- (1) Kirkpatrick, S., Gelatt Jr., C. D. and Vecchi,

- M. P., 1983, "Optimization by Simulated Annealing," *Science*, Vol. 220, No. 4598, pp. 671~680
- (2) Laarhoven, P. J. M and Aarts, E. H. L., 1988, *Simulated Annealing : Theory and Application*, Reidel Publishing Co.
- (3) Wong, D. F., Leong, H. W. and Liu, C. L., 1989, *Simulated Annealing for VLSI Design*, Kluwer Academic Publishers
- (4) Otten, R. H. J. M. and van Ginneken, L. P. P., 1989, *The Annealing Algorithm*, Kluwer Academic Publishers
- (5) Jain, P. Fenyas, P. and Richter, R., 1991, "Optimal Blank Nesting Using Simulated Annealing," *Advances in Design Automation, ASME, DE-Vol. 23-2*, pp. 109~116.
- (6) 방기범, 1990, "판재소모를 최소화하는 이차원 형상의 최적배치," 서울대학교 석사학위 논문.
- (7) 조경호, 1993, "판재부품의 가공 자동화를 위한 CAD/CAM 통합시스템," 박사학위논문, 서울대학교, 서울.
- (8) 조경호, 이진우, 1994, "폴림모사기법을 이용한 임의의 2차원 형상의 최적배치에 관한 연구," 대한기계학회, 춘계학술대회논문집(I), pp. 208~212.
- (9) 조경호, 이진우, 1994, "불규칙한 여러 원자재 위치에서의 효과적인 배치방법의 제안," 대한기계학회, 추계학술대회 논문집(I), pp. 694~698.
- (10) 한국찬, 나석주, 1993, "신경회로망을 이용한 직사각형의 최적배치에 관한 연구," 대한기계학회논문집, 제17권, 제12호, pp. 3063~3072.
- (11) 한국찬, 나석주, 1994, "이차원 패턴배치의 최적화를 위한 패턴분류 및 다단계 배치전략," 대한기계학회, 춘계학술대회논문집(I), pp. 200~203.
- (12) 백창현, 이춘만, 1994, "2 차원 임의 형상의 최적 자동배치에 관한 연구," 대한기계학회, 춘계학술대회논문집(I), pp. 196~199.
- (13) 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.
- (14) Chow, W. W., 1979, "Nesting of a Single Shape on a Strip," *Int. J. Prod. Res.*, Vol. 17, No. 4, pp. 305~322.
- (15) Nee, A. Y. C., 1983, "Computer-Aided Nesting of Similar Blanks," *Manuf. Engin., Technical Paper No. MS 83-925*.
- (16) Garey, M. R. and Johnson, D. S., 1979, *Computers and Intractability : A Guide to the Theory of NP-Completeness*, W. H. Freeman and Co.
- (17) Glover, F., 1986, "Future Paths for Integer Programming and Links to Artificial Intelligence," *Comput. Operations Res.*, Vol. 13, pp. 533~549.
- (18) Bland, J. A. and Dawson, G. P., 1991, "Tabu Search and Design Optimization," *Computer-Aided Design*, Vol. 23, No. 3, pp. 195~201.
- (19) Goldberg, D., E., 1989, *Genetic Algorithm in Search, Optimization, and Machine Learning*, Addison-Wesley.
- (20) Koza, J., R., 1992, *Genetic Programming : on the Programming of Computers by Means of Natural Selection*, MIT Press.
- (21) Adamowicz, M. and Albano, A., 1976, "Nesting Two-Dimensional Shapes in Rectangular Modules," *Computer-Aided Design*, Vol. 8, No. 1, pp. 27~33.
- (22) Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. and Teller, E., 1953, "Equation of State Calculations by Fast Computing Machines," *J. of Chem. Physics*, Vol. 21, pp. 1087~1092.