

임의의 형상의 원자재 위에서의 효과적인 배치 알고리즘의 제안

趙慶鎭*

A Proposal of an Effective Algorithm for the Nesting on Irregular Raw Sheets

*Kyung-Ho Cho**

Summary

By using the grid-based model and the well-known simulated annealing algorithm(SAA) as the shape representation method and optimization technique respectively, an effective nesting algorithm has been proposed to allocate the arbitrary shapes onto one and more raw sheets. 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, the simultaneous allocation onto more than one raw sheets have been tackled on successfully in this study. A couple of computational experiments are presented to verify the robustness of the proposed algorithm.

서론

배치 알고리즘에 대한 연구는 배치 대상물인 형상의 종류, 형상의 표현방법, 형상의 가공 혹은 절단방법 및 최적화 기법 등에 따라 여러 형태로 변화 발전되어 왔다. 배치문제를 사용 원자재의 구속조건에 따라 분류하면 다음과 같다.

- (1) 경제 제한이 없는 원자재에서의 최적배치
- (2) 경제 제한이 있는 원자재에서의 최적배치

(2-1) 일정 폭의 원자재에서의 최적배치

(2-2) 임의의 형상의 원자재에서의 최적배치

원자재 경제에 대한 제한이 없는 배치의 예는 동일 형상의 반복 배치 문제로 동일 형상을 주로 2~3개단위로 묶어 반복 가공하는 데 사용한다. 실제의 가공은 일정 폭의 원자재로부터 일정한 주기로 수행되지만, 원자재의 폭, 형상간의 배치, 반복 주기 등은 배치의 최적화를 통하여 결정된다(Jain 등, 1991). 이와는 달리 옷감 원단이나 철판의 재단 문제는 원자재의 폭이 이미 일정한 값으로 고정된 경우로서 많은

* 공과대학 에너지공학과(Dept. of Nuclear and Energy Engineering, Cheju Univ., Cheju-do, 690-756, Korea)

배치 알고리즘(Adamowicz와 Albano, 1976; Christofides와 Whitlock, 1977; Chow, 1979; Albano와 Sapuppo, 1980; Nee 등, 1986)은 이러한 문제에 대해서만 효과적인 작동이 보장된다. 반면에 피혁제품의 재단에서 볼 수 있는 경계가 불규칙하고 내부에 부분적인 결합이 존재하는 임의 형상의 원자재 위에서 배치문제는 앞에서 언급되었던 문제 해결 방식으로는 바람직한 결과를 기대할 수가 없다. 특히, 기존 배치 알고리즘에서는 두개 이상의 원자재 위에서의 배치문제를 전혀 고려하지 있지 않기 때문에 이를 위한 적절한 배치 알고리즘의 개발이 필요하다(Nee 등 1986). 여기서 형상의 적절한 배치라 함은 배치 형상이 원자재 경계를 벗어나지 않고, 원자재 내부 결합을 피하고, 형상간의 중첩이 없는 배치를 일컫는다. 이를 위해서 본 연구에서는 다음과 같은 내용을 주요 특징으로 하는 최적배치 알고리즘을 개발하였다.

- 배치될 형상은 임의 모양을 갖는다.
- 원자재의 경계는 불규칙하고, 내부 결합을 갖을 수도 있다.
- 두개 이상의 원자재가 동시에 사용될 수도 있다.

연구내용 및 방법

1. 최적화 기법

본 연구에서 해결하고자 하는 최적배치 문제와 같이 수많은 국지해를 갖는 조합 최적화 문제(combinatorial optimization problem)들은 그 최적해를 구하는데 있어서의 난이도에 따른 분류로 볼 때 NP complete(non-deterministic polynomial time complete) 문제에 해당된다(Garey와 Johnson, 1979). 이들 문제는 대부분 부정정제이면서 미분 불가능한 특성을 갖고 있기 때문에 종래의 목적함수 변화율에 근거한 최적화기법으로는 전역 최적해(global optima)에도 도달하기 어렵고 주로 국지적인 해에 빠지기 쉽다. 따라서, 수많은 국지해를 거쳐 전역 최적해에 도달하기 위해선 목적함수값이 증가하는 방향으로 이동할 수 있는 능력(hill climbing ability)과, 과거 어느 순간 방문되었던 해라 할지라도 필요에 따라선 다시 과거의 그 해로 되돌아 가서

새로운 탐색을 시도할 수 있는 능력(backtracking ability)이 필요하다. 이상의 능력들을 고루 갖춘 최적화 기법으로 최근 주목을 받고 있는 것으로 풀림모사 기법(simulated annealing algorithm, SAA), 타부써치(tabu search, Glove, 1986; Widmer, 1991) 및 유전적 방법(genetic algorithm, Goldberg, 1989; Koza, 1993) 등이 있다. 이중에서도 SAA는 그 구현 방법이 비교적 쉬워서 목적함수의 구성방법, 구속조건, 자유도 등의 변화와는 거의 독립적으로 알고리즘의 골격을 유지할 수 있어 새로운 문제로의 이식성이 뛰어나기 때문에, 본 연구에서는 SAA를 최적화 기법으로 도입하였다(조, 1993, 1994). SAA의 구체적인 작동원리 및 여러 적용 사례들은 여러 참고 문헌에서 찾을 수 있다(Laarhoven과 Aarts, 1988; Wong 등, 1989; Otten과 Ginneken, 1989; Rutenbar, 1989; Jain 등, 1991).

2. 원자재의 효과적인 정렬 방향

일정 폭의 원자재 위에서의 배치 문제와는 달리, 불규칙한 원자재를 사용할 경우 배치효율의 평가는 여러가지 방법으로 할 수 있겠으나 아직은 객관적이고 정량화된 기준은 마련되어 있지 못한 것으로 본다. 본 연구에서는 배치효율의 평가기준으로 잠재적 가용 원자재의 크기를 도입, 사용하였다. 잠재적 가용 원자재(potential waste)라 함은 배치형상의 가공 후 남는 찌투리 중 가장 큰 것을 일컫는 것으로서, 이것이 가능하면 크게 되는 방향으로 원자재들을 정렬시킨 후 이들 위에 배치하고자 하는 형상들을 배치한다. 일정 폭의 원자재를 사용하는 경우에는 이 문제가 너무나도 자명한 것이어서 문제화 되지 않았을 뿐이나 이러한 개념은 일정 폭의 원자재를 사용하는 경우에도 그대로 적용이 가능하다. Fig.1은 원자재의 정렬 방향에 따라 잠재적 가용 원자재의 크기가 변하는 것을 보여 준다. Fig.2 및 Fig.3은 원자재 크기가 작아서, 두개 이상의 여러 원자재에서 제품이 재단되어야 할 경우에 대한 예시로서, 이 경우에는 원자재의 효과적인 정렬의 필요성은 더욱 분명해진다. 따라서, 원자재의 효과적인 정렬 방향의 결정문제도 또 다른 최적화 문제이다.

이를 위하여 본 연구에서는 정렬된 전체 원자재를 둘러싸는 최소 크기의 직사각형(minimum bounding

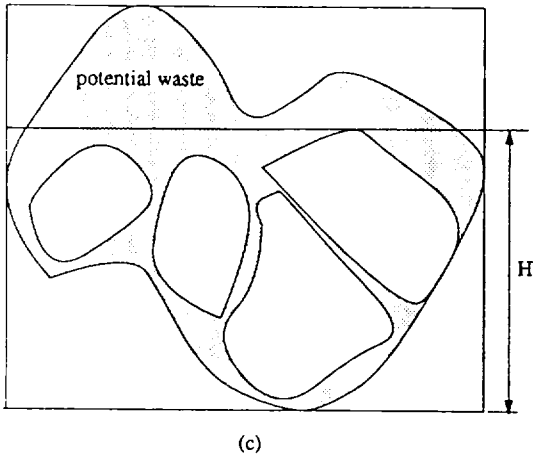
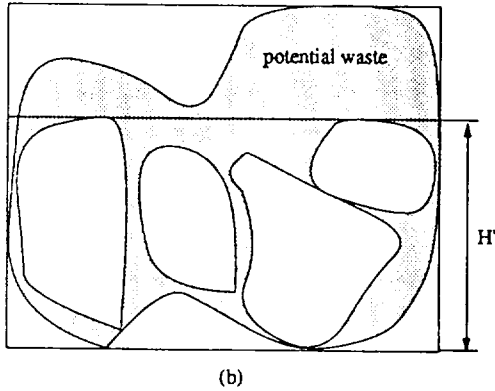
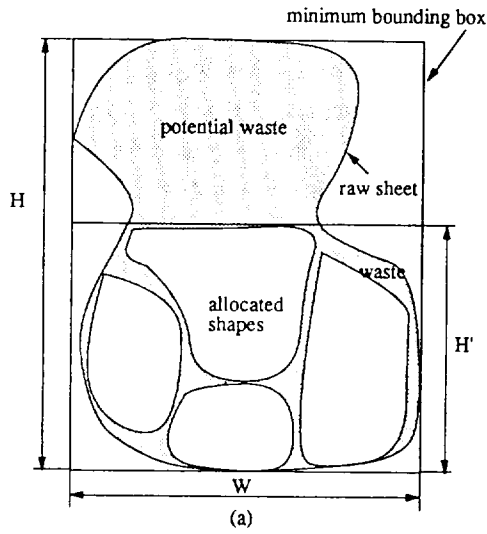


Fig. 1. Example of wastes potentially available for future usage depending upon the arrangement of the raw sheet materials.

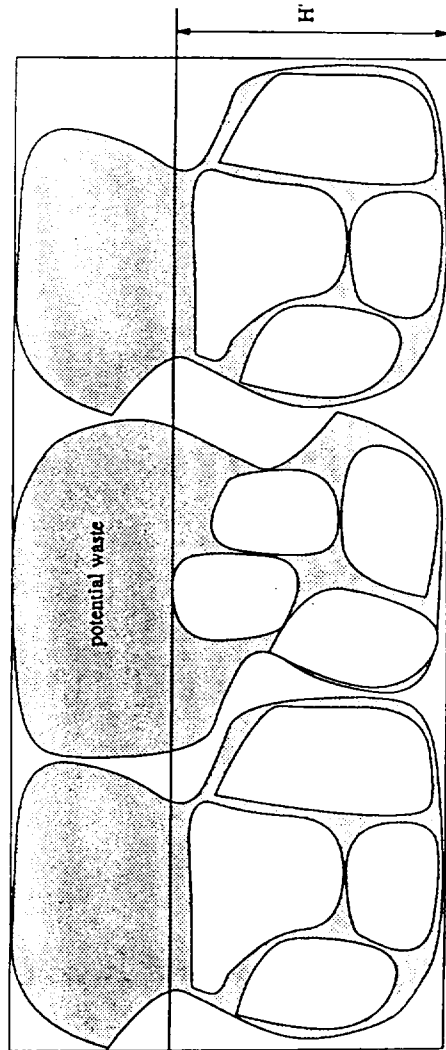


Fig. 2. Example of potential waste after the nesting on several raw sheet materials.

box, MBB)의 면적이 가장 작은 정렬 상태를 구하여 사용하였다. MBB에 대한 연구는 이미 여러 분야에서 있어 왔으나, 본 연구에서는 경제가 없는 무한 평판에서 임의의 형상의 효과적인 배치 문제로 처리하였다. 즉 원자재를 또 다른 배치 형상으로 보고 이들을 가능한 한 작은 직사각형 속에 겹치지 않고 배치한 결과를 원자재의 효과적인 정렬 방향으로 사용하였다. 이러한 접근방법은 원자재의 효과적인 정렬 방향 결정을 위하여 별도 알고리즘의 추가 없이, 본 연구에서 개발된 무한 평판에서 임의의 형상의 효과적인 배치 알고리즘을 그대로 사용할 수 있는 잇점 때문이다.

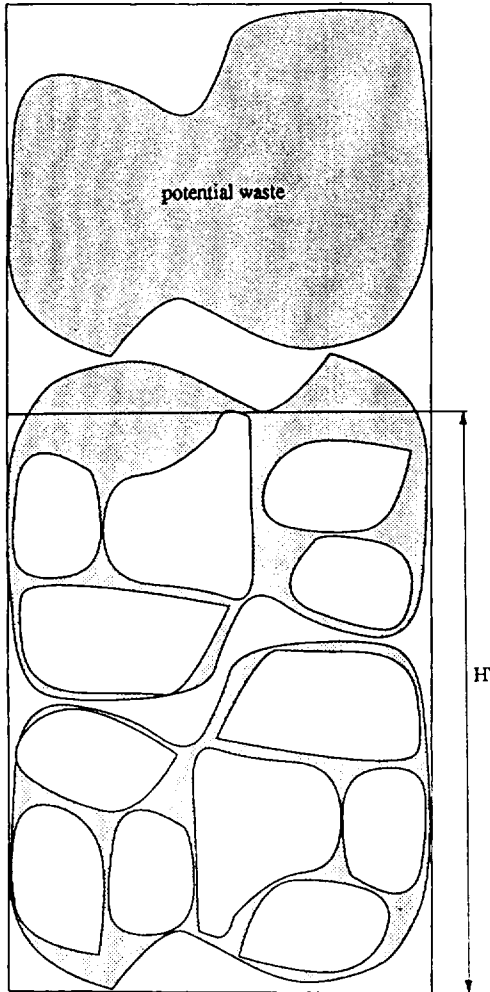


Fig. 3. Example of potential waste after the nesting on several raw sheet materials.

3. 격자모델 및 가상 원자재의 격자화

격자모델 : 배치 형상을 다루는 방법으로 다각형 근사 모델이 주로 사용되나 격자모델도 일부 이용된다. 다각형 근사 모델을 사용할 경우에는 배치 상황의 위치 정보 탐색 즉, 배치 형상간의 중첩 여부 검사 및 중첩 면적의 계산, 원자재 경계의 이탈 여부 검사 등을 위하여 교선 계산과 같은 해석적 기법이 이용된다. 이러한 계산작업은 수많은 배치상황의 개선을 필요로 하는 배치 알고리즘에서는 컴퓨터의 실행시간 면에서 과중한 부담이 된다. 따라서, 본 연구

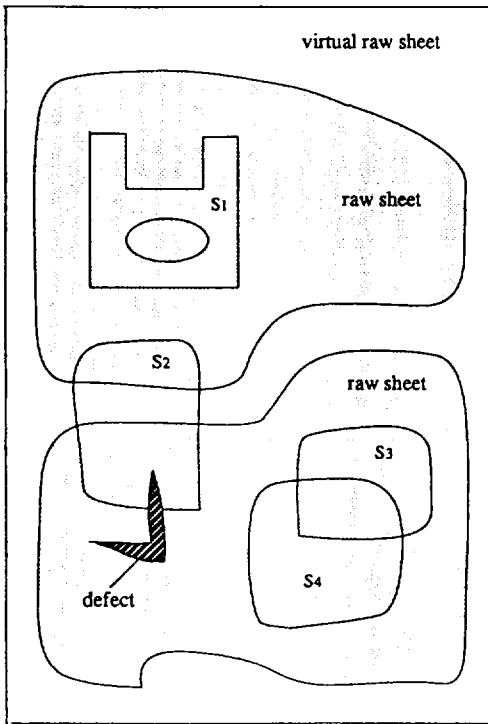
에서 처럼 최적화 기법으로 SAA를 사용할 경우는 다각형 근사 모델 보다 격자 모델이 훨씬 유리하다.

이에 따라, 본 연구에서는 원자재 및 그 위에 배치될 형상을 모두 격자화하여 사용하였고, 격자모델을 위한 자료구조는 (1) 형상 배치 및 제거작업의 신속한 수행, (2) 중첩 검사의 신속성 및 (3) 형상 저장 방법의 효율성 등을 고려하여 설계되었다(조, 1993, 1994). 또한 형상의 배치 방향에는 제한을 두어 사용자가 지정하는 각도 증분 만큼의 회전만 허용하였다. 그러나, 배치 방향이 바뀔 때마다 매번 회전격자 모델을 새로 생성하는 대신, 기본형상의 모든 회전 가능 방향에 대한 회전된 격자 모델을 한번만 생성시켜 이를 저장하고 있다가, 필요할 때마다 이를 참조하여 배치, 제거 및 중첩검사를 수행하도록 하였다.

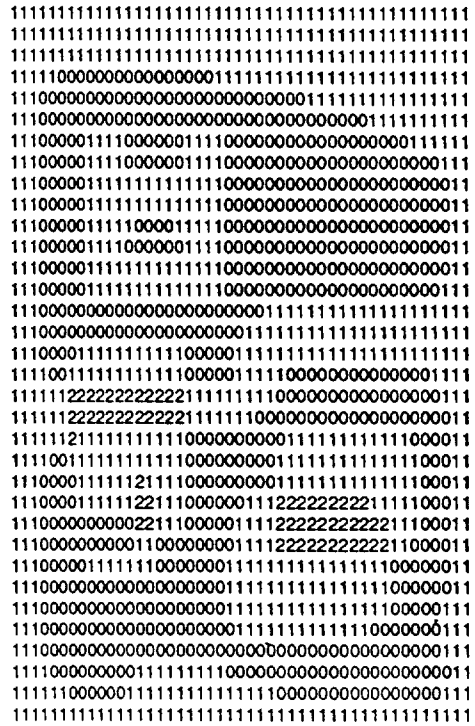
격자모델의 또 다른 잇점은 내부에 구멍이 있는 형상의 격자화 및 구멍내에 다른 작은 형상의 배치 등을 위해서 별도의 알고리즘을 필요로 하지 않는다는 것이다. 이에 반하여 Adamowicz와 Albano의 연구(Adamowicz와 Albano, 1976)에서는 내부에 구멍이 있는 형상에 대해서는 특별히 고안된 알고리즘을 통해 사전에 내부 구멍에 배치시킬 수 있는 형상을 찾아 이를 그 구멍안에 배치 시킨 후 이 형상들을 한데 묶어 하나의 형상(clustered shape)으로 취급하는 다단계 배치 방법을 필요로 하는 단점을 갖고 있다.

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

원자재의 정렬이 완료되면 이때의 MBB를 가상원자재라 한다. 가상원자재는 Fig. 4에서 보는 바와 같이 한개 혹은 그 이상의 실제 원자재에 대응하는 가용한 배치영역이 있고, 실제 원자재 경계 밖과 내부결합부에 대응하는 배치불가 영역이 있다. 격자 모델을 근간으로 하는 본 알고리즘에서는 이의 구현이 가상원자재의 다음과 같은 격자모델로 용이하게 이루어진다. 즉, m, n 을 각각 가상원자재의 가로 폭 및 세로 높이에 대응하는 격자수라 할 때 가장원자재 MBB는 일정크기의 이차원 격자 sheet(m)(n)로 정의된다. 각 격자는 0이상의 정수로 표현되는 격자값을 갖는다. sheet(m)(n)는 배치불가 영역은 1, 배치



(a)



(b)

Fig. 4. Example of (a) shape allocations on two irregular raw sheets and (b) its grid representation.

가능 영역은 0으로 초기화 된다. 형상들의 배치가 진 전됨에 따라 이들 격자값은 해당 위치에 배치된 형상의 수 만큼 증가한다. 따라서, 가상원자재의 어느 특정 격자의 격자값이 2이상이면 이는 그 위치에 2개 이상의 형상이 중첩되도록 배치된 것을 의미하거나, 배치불가 영역에 격자값보다 1개 작은 수에 해당하는 형상들이 배치되었음을 뜻한다. 중첩이 없고 가능한 원자재 내에서의 배치는 궁극적으로 sheet(m)(n)의 모든 값이 0과 1만의 값을 갖는 상태이다.

4. 초기배치, 목적함수 및 배치상황의 갱신

초기배치: 배치되어야 할 모든 형상에 대하여 앞 절에서와 같이 기준 격자모델을 완성한 후, 이미 격자화되어 있는 가상 원자재에 모든 형상을 임의 위치와 방향을 갖도록 한꺼번에 배치한다. 이때 형상과

형상 사이에는 중첩이 일어날 수도 있다. 다만, 각 형상들에게는 가상 원자재 경계를 벗어나지 않는 범위에서 무작위로 배치 위치 및 방향이 부여된다. 이렇게 각 형상의 초기 위치 및 방향이 결정되어 원자재에 배치된 상황을 초기 배치상황이라 한다. 프로그램상에서 형상의 배치, 제거는 격자 모델을 배치, 제거하고자 하는 위치에 이동시켰을 때 격자 형상에 대응하는 원자재의 각 격자의 격자값을 1씩 증가, 감소시키는 일에 해당한다.

목적함수: 최적화 작업은 앞의 초기 배치로부터 목적함수를 개선(감소)시키는 새로운 배치상태를 연속적으로 찾아 가는 과정이다. 이를 위해서 사용된 목적함수는 다음과 같다.

$$C(S) = Scrap + w * Overlap-area$$

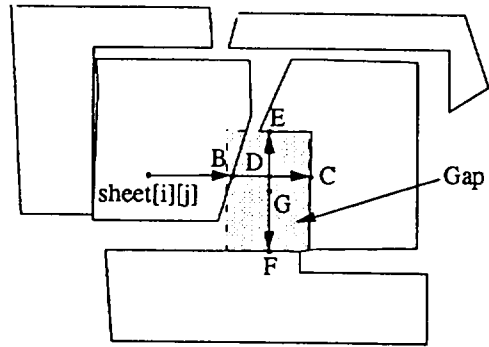
여기서, S는 현재의 배치 상황을 나타내고, C(S)는

이때의 목적함수 값이다. Scrap은 지정된 원자재로부터 모든 배치 형상을 가공해낸 후 남게되는 원자재의 찌투리양을 나타낸다. 이는 기본적으로 원자재의 격자값이 0인 모든 격자의 수에 해당한다. Overlap-area는 중첩부위의 면적으로서 원자재의 격자값이 2 이상인 모든 격자의 수에 해당한다. 본 연구에서는 최적 배치를 위한 탐색 도중에는 형상간의 중첩을 허용한다. 그러나, 최종배치 결과에서는 중첩이 나타나지 않도록 하기 위하여 가중치(weighting factor, w)를 도입하였다.

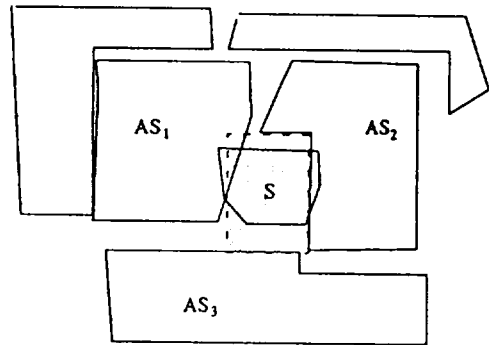
배치상황의 갱신 : 현재의 배치상태 S에 매우 작은 교란 ΔS 를 무작위로 부여하여 얻을 수 있는 모든 새로운 배치상태 S'를 S의 이웃해라 한다. 이를 위하여 여러 형상중에서 임의로 하나를 선택하여 그 형상의 기존 배치 정보, 즉 배치위치 및 방향을 기준으로 이에 작은 교란을 임의로 부여하여 새로운 배치 상태를 생성하여 이를 새로운 이웃해로 정의, 사용하였다. 배치상황의 갱신은 S를 새로 생성된 이웃해 S'로 대체하는 작업이다. 목적함수가 감소하면, 즉 $\Delta C = C(S') - C(S) < 0$ 이면 S'는 S보다 최적해에 더 가까와 졌다고 보고 S'를 새로운 현재의 상태 S로 받아들이고, 이로부터 최적해에 보다 더 가까와지기 위한 또다른 S'를 계속 탐색한다. 특기할 사항은 $\Delta C > 0$ 인 경우에도 S'는 항상 거부되는 것이 아니고, SAA가 갖는 힐 클라이밍 특성에 따라 목적함수가 증가하는 방향으로의 배치상황의 갱신이 확률적으로 허용된다는 것이다.

5. 국부 풀림을 통한 간극 채워넣기

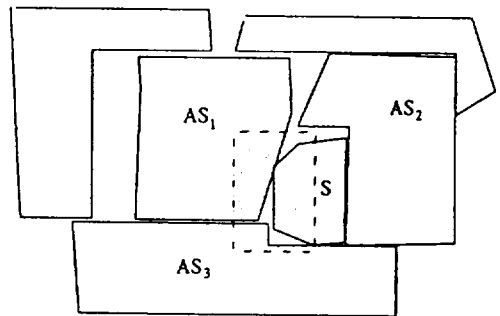
배치효율의 향상 방안으로 본 연구에서는 특히, 작은 형상들의 배치에서는 "국부 풀림을 통한 간극 채워넣기(gap filling with local annealing)"라는 새로운 개념을 고안하여 사용하였다. 즉 찌투리 영역의 일부를 나타내는 직사각형을 간극(gap)이라 정의하고, 작은 형상을 이 간극에 끼워넣기식으로 배치하는 방법으로써, 이는 여러가지 크기의 물건을 큰 상자 속에 넣을 때 큰 것들을 먼저 넣은 뒤 작은 물건들을 큰 것들 틈바귀에 적절히 끼워 넣고 상자를 조금씩 흔들어 주면 작은 것들이 큰 것들 틈바귀에서 스스로 제자리를 찾아가는 현상을 모사한 것이다. 이의 구체적인 작동원리는 다음과 같다.



(a)



(b)



(c)

Fig. 5. Gap filling with local annealing (a) Gap searching (b) Gap fill by positioning the small shape at gap center (c) Final result after local annealing.

● 1단계) 크기 순 정렬 : 가장 작은 그룹에 속한 형상들을 큰 것 부터 작은 순으로 정렬시킨 후 이들 중에서 큰 것부터 다음 단계에 따라 간극 채워넣기를

수행한다.

● 2단계) 간극탐색 (gap search) : Fig. 5 (a)처럼 시스템에 의해 임의로 선택된 원자재 격자점 (i, j) 로부터 $sheet(i)(j)$ 의 격자값이 1이면 i 만 증가시켜 짜투리의 경계점 B를 찾는다. 만약 그 격자값이 0이면 i 만 감소시켜 경계점 B를 찾는다. (참고로, 중간 그룹의 형상들을 배치하다가 본 단계의 간극 채워넣기를 시작할 시점에서는 전체 목적함수값은 상당히 낮아져 있는 상태로서 기 배치 형상 간에는 중첩이 거의 해소된 상태에 도달해 있다. 따라서 격자점 $sheet(i)(j)$ 의 격자값은 대부분 0 아니면 1이다). B로부터 i 만 증가시켜 짜투리의 경계점 C를 찾는다. B, C의 중점 D를 짜투리의 좌우 경계의 중점이라 한다. D로부터 j 값만 변화시켜 짜투리의 상하 경계 E, F를 찾는다. E, F의 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 경계점의 중점을 $D \rightarrow G \rightarrow D \rightarrow G \rightarrow \dots$ 처럼 순환적으로 반복하여 구하면 적절한 모양의 짜투리 영역에서는 이들 두 중점은 서로 가까워진다. 따라서 지정된 횟수 이내의 반복으로 두 중점 사이의 거리가 허용치내로 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 포함하는 가장 작은 직사각형을 간극이라 칭하고 그 크기는 이 직사각형의 가로 세로 격자의 갯수로 나타낸다. 간극의 크기가 현재 배치되어야 할 형상의 크기보다 작을 경우 격자점 (i, j) 를 임의로 새로 선정하여 위의 과정을 간극이 찾아질 때까지 지정횟수만큼 반복한다. 이렇게 하여서도 현재 배치되어야 할 형상 S의 크기 보다 큰 간극을 찾을 수 없으면 형상 S을 위한 간극탐색은 실패한 것으로 보고, 형상 S보다 작은 다음 형상에 대해 2단계)부터 새로 시작한다.

● 3단계) 간극 채워넣기 (gap filling) : 위에서 찾아진 간극의 중심과 현재의 간극 채워넣기 대상 형상인 S의 중심을 일치시켜 Fig. 5(b)와 같이 S를 간극 중심에 배치한다. 이때 S가 간극 주위에 이미 배치되어 있는 형상 $(AS_j, j=1, 2, \dots, j_n)$ 들과 중첩을 일으키면 다음 단계를 수행한다.

● 4단계) 국부 풀림 (local annealing) : Fig. 5(b)처럼 형상 S와 기존의 배치 형상 AS_j 만으로 국부적인 재배치를 SAA로 수행한다. 이는 Fig. 5(c)와 같이 형상 S와 기존의 배치 형상 AS_j 간의 중첩 뿐만 아니라 형상 AS_j 들과 이들 둘러싸는 기 배치 형상들 사이에서도 중첩이 일어나지 않을 때까지 계속된다. 이때 AS_j 들은 병진 이동만이 허용되고 S는 회전이동까지 할 수 있도록 하였다. 이는 간극 채우기 단계에 이르면 형상 사이의 간격이 매우 가까워져서 큰 형상의 조그만 회전도 쉽게 주위의 다른 형상과 중첩을 유발할 수 있기 때문이다. 만약 지정된 횟수 이상으로 재배치가 시도되어도 중첩이 해소되지 않으면, 그 위치에 대한 형상 S의 간극 채워넣기는 취소된다. 이 경우 기배치 형상 AS_j 들은 본래의 배치 상태 (Fig. 5(a))로 되돌아가고, 형상 S를 위한 새로운 간극탐색이 2단계)부터 다시 시도된다.

모든 간극 채워넣기 대상 형상에 대해서 위의 2~4 단계를 수행하고 나면 전체 간극 채워넣기 작업은 종료되고 모든 형상을 대상으로 한 본래의 SAA를 수렴조건이 만족될 때까지 수행한다.

결과 및 고찰

Fig. 6은 최적배치를 위해서 구현된 프로그램의 기본적인 흐름도이다. 그림에서와 같이 임의의 온도 T_k 에서 이웃해를 탐색하다가 이 온도에서 목적함수가 평형 상태에 도달하면 온도 T_k 를 새로운 온도 T_{k+1} 로 변화시켜 이 온도에서 다시 목적함수가 평형 상태에 도달할 때까지 새로운 탐색을 반복한다. 온도 변화는 $T_{k+1}/T_k=0.8\sim 0.95$ 로 하였고, 임의의 온도에서 대략 $200 * nshape$ 회의 이웃해에 대한 탐색이 시도되었으면 평형상태에 도달한 것으로 처리하였다. 또한 (1) 지정된 탐색 횟수 동안 목적함수의 최소값이 전혀 개선되지 않는 경우, (2) 온도가 지정된 최저온도 T_{min} 보다 작아졌을 경우, 및 (3) 이웃해에 대한 $200 * nshape$ 회의 탐색 동안에 한번도 이웃해가 새로운 해로 받아들여지지 못한 경우 중의 어느 한 경우에 도달하면 전체 최적화 과정은 종료되도록 하였다.

Fig. 7은 36개형상을 내부 결합이 있고 경계가 불규칙한 원자재 위에 배치한 결과이다. 실제 형상에 대

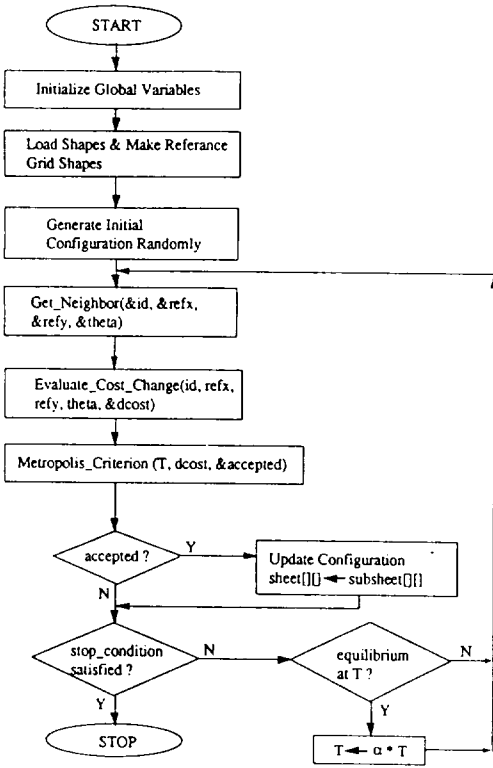
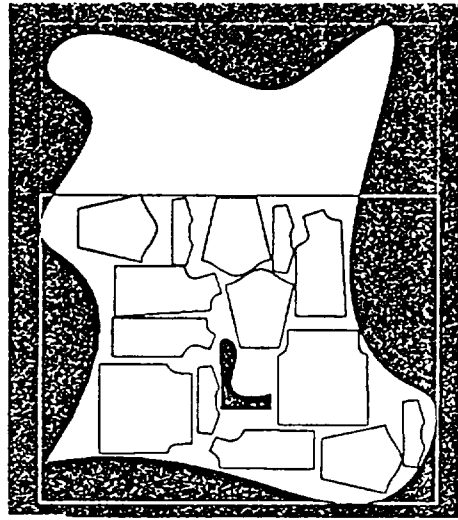
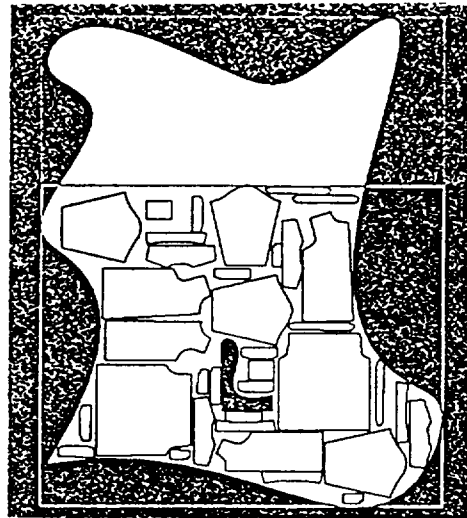


Fig. 6. Flow chart for the optimal nesting using SAA.

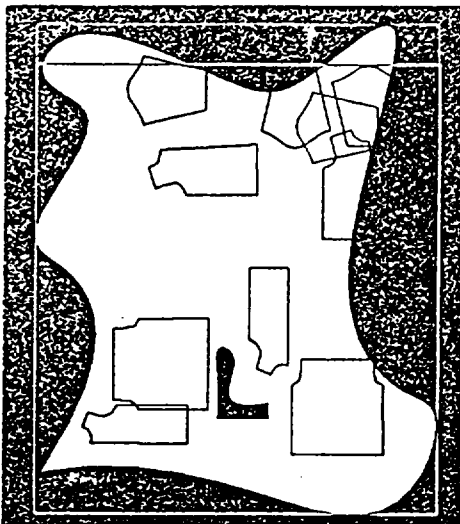


(b)



(c)

Fig. 7. Nesting of 36 patterns on the irregular sheet having interior defect (a) initial configuration generated randomly (b) near-optimal result of large shapes (c) final result after the allocation of small shapes by gap-filling with local annealing.



(a)

한 데이터는 삼성물산(주)에서 36개 옷감 모형의 배치에서 사용되었던 것이다. 본 예제의 배치 형상들을

동일 피혁제품으로 가공하는 것은 실제와는 거리가 있으나, 여기서는 단지 본 연구에서 개발된 알고리즘의 작동성능을 평가하기 위해 사용하였다.

배치효율의 평가를 위해 본 연구에서는 원자재의 버림률(waste ratio, wr)을 다음과 같이 정의하여 사용하였다.

$$wr (\%) = \left(\frac{\text{원자재 총 면적} - \text{잠재가용 원자재 면적}}{\text{배치형상 총 면적}} - 1 \right) \times 100$$

Fig. 7에서의 원자재의 정렬방향은 각 형상들의 배치에 앞서 또다른 최적화 과정을 통하여 이미 결정된 것이다. Fig. 7(a)는 비교적 크기가 큰 형상들의 초기배치 상황으로서 각 형상들의 배치위치 및 방향은 가상원자재 내에서 무작위로 부여된 것이다. Fig. 7(b)는 최적화 과정을 통하여 이들 형상의 배치상황이 갱신을 거듭하다가 수렴된 상태로서 더이상의 획기적인 갱신이 어려운 상황이다. Fig. 7(c)는 Fig. 7(b)의 상황에서 작은 형상들을 국부 플립을 통한 간극 채워넣기 방식으로 배치한 결과이다. 최종 결과에서 볼 수 있듯이 원자재 내부에 결함이 있는 경우 이를 피하여 배치가 이루어지고 있음을 알 수 있다.

Fig. 8(b)은 36개 형상을 한개의 원자재에서 재단해 낼 수 없어서 3개의 원자재가 사용된 예이다. 여기서도 앞의 예제에서와 같이 형상들의 배치에 앞서 3개 원자재의 정렬방향이 최적화 과정을 통하여 이미 결정된 것을 사용하고 있다. 이 예제에서도 크기별 배치 및 간극 채워넣기가 수행되었다. 특히 본 예제는 형상의 내부에 가용한 배치영역이 있으면 그곳에 다른 작은 형상을 별도의 추가 알고리즘 없이 배치할 수 있음을 보인다.

적 요

본 연구에서는 1) 형상의 내부에 유용한 배치공간이 있을 경우 이곳에 작은 형상을 배치하는 방법과,

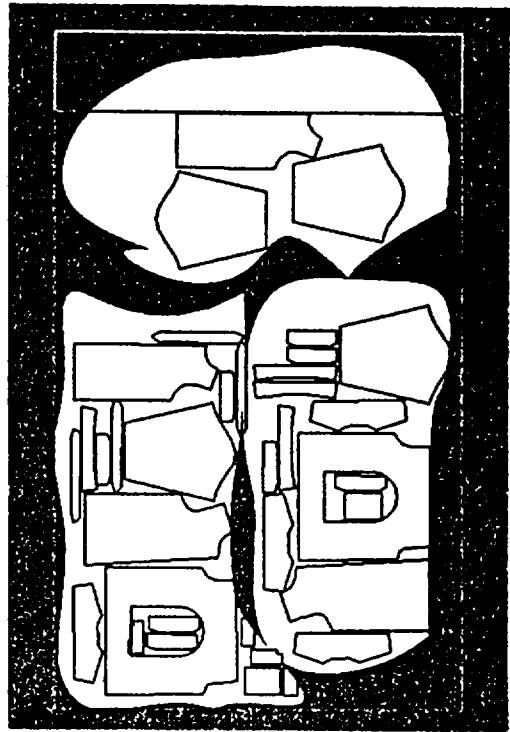


Fig. 8. Nesting of 36 patterns on 3 raw sheets.

2) 피혁제품의 배치에서 볼 수 있는 경계가 불규칙하고 내부에 결함이 있는 경우 및 3) 두개 이상의 원자재 위에서의 효과적인 배치문제물 해결할 수 있는 알고리즘을 개발하였다. 이의 구현을 위해 배치 형상을 격자화하여 사용하였고, 주요 최적화 기법으로는 플립모사 기법(SAA)을 이용하였다. 또한 "국부 플립을 통한 간극 채워넣기"라는 개념을 고안하여 배치효율면에서 괄목할 만한 개선 효과를 얻을 수 있었다.

앞으로의 연구에서는 배치 시간 및 효율의 향상에 대한 연구가 필요하고, 배치형상들을 스캐너 혹은 비디오의 영상정보로 직접 입력받아 처리하는 인터페이스 개발에 대한 연구가 남아 있다.

참 고 문 헌

Adamowicz, M. and A. Albano, 1976, Nesting Two-dimensional Shapes in Rectangular

Modules, *Computer Aided Design*, Vol. 8, No.1, 27-33.

- Albano, A. and G. Sapuppo, 1980, Optimal Allocation of Two-dimensional Irregular Shapes Using Heuristic Search Methods, *IEEE Trans. on Syst. Man & Cyber.* Vol. SMC-10, No. 5, 242-248.
- 조경호, 1993, 플립모사 기법을 이용한 최적배치 알고리즘 개발에 관한 연구, 제주대학교 논문집, 제 37권, 237-246.
- 조경호, 1994, 플립모사 기법을 이용한 임의의 2차원 형상의 최적배치에 관한 연구, 대한기계학회 춘계 학술대회 논문집, Vol. I, 208-212.
- Chow, W. W., 1979, Nesting of a Single Shape on a Strip, *Int. J. Prod. Res.* Vol. 17, No. 4, 305-322.
- Christofides, N. and C. Whitlock, 1977, An Algorithm for Two-dimensional Cutting Problems, *Oper. Res.* Vol. 25, No. 1, 30-44.
- Garry, M. R. and D. S. Johnson, 1979, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Co., U.S.A.
- Glover, F., 1986, Future Paths for integer programming and links to artificial intelligence, *Comput. Operations Res.* Vol. 13, 533-549.
- Goldberg, D. E., 1989, *Genetic Algorithm in Search, Optimization, and Machine Learning*, Addison-Wesley, U.S.A.
- Jain, P., P. Fenyas and R. Richter, 1990, Optimal Blank Nesting Using Simulated Annealing, *Advances in Design Automation*, DE-Vol. 23-2, ASME, 109-116.
- Koza, R. J., 1993, *Genetic Programming: on the programming of computers by means of natural selection*, MIT Press, U.S.A.
- Laarhoven, P. J. M. and E. H. L. Aarts, 1988, *Simulated Annealing: Theory and Application*, Reidel Publishing Co.
- Nee, A. Y. C., K. W. Seow and S. L. Long, 1986, Designing Algorithm for Nesting Irregular Shaps with and without Boundary Constraints, *Annals of the CIRP*, Vol. 35, No. 1, 107-110.
- Otten, R. H. J. M. and L. P. P. P. van Ginneken, 1989, *The Annealing Algorithm*, Kluwer Academic Publishers.
- Rutenbar, R. A., 1989, Simulated Annealing Algorithms: An Overview, *IEEE Circuits & Devices Mag.* 19-26.
- Widmer, M., 1991, Job Shop Scheduling with Tooling Constraints: a Tabu Search Approach, *J. Opt. Res. Soc.* Vol. 42, No. 1, 75-82.
- Wong, D. F., H. W. Leong and C. L. Liu, 1989, *Simulated Annealing for VLSI Design*, Kluwer Academic Publishers.