

▣ 연구논문

2차원 길로틴 절단문제를 위한 새로운 상한
A New Upper Bound for
Two-Dimensional Guillotine Cutting Problem

윤 기 섭*
Yoon, Ki Seop
지 영 근*
G, Young Gun
강 맹 규*
Kang, Maing Kyu

Abstract

The two-dimensional guillotine cutting problem is to maximize sum of piece profits that cut from one stock rectangle and widely applied in the industry. The branch-and-bound method for this problem uses complementarily several upper bounds(the Gilmore and Gomory[8]'s two-dimensional knapsack function and the Hifi and Zissimopoulos[10]'s method using one-dimensional knapsack problem, etc) to reduce the number of searched nodes. These upper bounds has a shortcoming that does not consider the bound and layout of pieces simultaneously. In this paper, we propose an efficient upper bound which can complement the shortcoming of existing upper bounds. The proposed upper bound needs less memory spaces and computing time. Computational results show that the proposed upper bound significantly contribute to reduce the computational amount of time and number of searched nodes in tree.

1. 서론

2차원 절단문제(cutting problem)는 길이와 너비가 각각 L , W 인 하나의 직사각형 자재(stock)로부터 길이, 너비, 가치가 각각 l_i , w_i , v_i 인 n 종류의 직사각형 제품(piece)을 가치 총합이 최대가 되도록 절단하는 문제이다[7]. 절단문제는 유리회사, 철강회사, 제지회사, 가구회사 등에서 각종 자재를 절단하는데 응용될 수 있다.

2차원 절단문제는 절단 방법에 따라 일단 절단이 시작되면 도중에 멈추지 않고 끝까지 절단하는 길로틴(guillotine) 절단문제와 그렇지 않은 비길로틴(non-guillotine) 절단문제로 나눌 수 있는데[2], 현실적으로 많은 경우절단 기계의 특성상 절단도중 방향을 바꾸지 못하기 때문에 주로 길로틴 절단 문제를 대상으로 연구가 집중되어 왔다. 2차원 절단문제의 또 다른 구분은 제품 개수 제한의 유무에 따라 제품 개수 제한이 있는(constrained) 문제와 없는 문제(unconstrained)로 나누는 것이다[3]. 절단문제는 자재의 개수가 무한히 있다고 가정한 자재절

† 본 논문은 한국과학기술부, 한국과학재단에서 지원한 첨단도료연구센터의 연구수행결과임

* 한양대학교 산업공학과

단문제(cutting stock problem)의 부문제로 주로 이용된다[7]. 자재절단문제를 풀기 위한 방법으로는 열생성기법[7]과 발견적 기법이 있는데, 대부분의 발견적 기법은 문제에 따라 다양하지만 절단패턴(cutting pattern) 생성 과정에서 개수 제한이 있는 2차원 절단문제를 풀어야 한다[9]. 여기서 절단패턴이란 하나의 자재로부터 절단되는 제품의 조합을 의미한다.

본 연구에서는 제품 개수 제한 있는 2차원 절단문제를 다룬다. 이 문제에 대한 최적 해법 중 우수하다고 알려진 것들은 분지한계법을 이용한 것들인데, 대표적인 것들로 Christofides와 Whitlock[4]의 해법(이하 CW)과 Viswanathan과 Bagchi[13]의 해법(이하 VB)이 있다. CW는 자재를 제품의 크기로 절단해 나가는 하향식(top-down) 방법을, VB는 제품들을 수직/수평으로 결합하여 나가는 상향식(bottom-up) 방법을 각각 채택하고 있다. Hifi와 Zissimopoulos[10]는 하한(lower bound)과 개선된 상한(upper bound), 새로운 분지기법 등을 적용하여 CW를 개선한 해법 MCW(Modified CW)[11]를 개발하였다. Cung 등[5]은 VB에 개선된 하한과 상한을 적용하고, 중복되는 분지를 방지하기 위해 절단패턴들에 코드를 부여함으로써 계산속도를 크게 향상시켰다.

분지한계법의 효율을 향상시키기 위해서는 최적해에 가까운 하한(lower bound) 또는 상한(upper bound)을 구하는 방법과 가능한 한 빨리 최적해에 도달하도록 효과적인 분지기법을 사용하여 탐색되는 노드의 수를 줄이는 방법이 있다. 절단문제를 위한 기존의 최적해법에서는 한 가지 상한만을 이용하지 않고 여러 가지 상한을 사용하여 그 중에서 가장 작은 값을 이용한다.

본 연구에서는 제품 개수 제한이 있는 2차원 절단문제를 위한 새로운 상한을 제안하고, 실험을 통해 제안하는 상한이 기존의 상한들을 보완하여 최적해법에 적용했을 때 개선효과가 있음을 보인다. 먼저 제품 개수 제한 있는 2차원 절단문제를 위한 기존 상한들의 단점을 알아본 후, 제안하는 상한의 내용과 효과를 입증하기 위해 MCW에 새로운 상한을 적용하여 계산시간과 분지 되는 노드의 개수를 비교함으로써 제안하는 새로운 상한의 우수함을 보인다.

2. 절단문제를 위한 기존의 상한

절단문제를 위한 분지한계법에서 사용하는 상한은 두 가지 방법을 기본으로 하고 있다. 첫 번째는 Gilmore와 Gomory[8]의 2차원 배낭문제의 해법을 이용하는 것이다. 즉, 제품 개수 제한 없는 2차원 배낭문제(knapsack problem)의 최적해를 구하여 이를 절단문제의 상한으로 이용한다. 길이 x , 너비 y 인 직사각형을 (x, y) 라 하면 길이 L , 너비 W 인 자재 (L, W) 에 대해 이 해법을 적용하면 $x \leq L$, $y \leq W$ 인 임의의 (x, y) 에 대한 해도 계산된다. 이 방법은 개수 제한 없는 문제의 최적해로서 직사각형의 크기가 작을수록 제품 개수 제약을 만족할 가능성이 높으므로 상한이 최적해가 되는 경우가 많으며, 또한 여기서 얻은 정보를 분지한계법에서 이용할 수 있다는 장점이 있다. 반면에 계산량과 메모리 요구량이 각각 $O((L+W) \times LW)$ 와 $3 \times LW$ 로 크고, 제품 개수 제한을 고려하지 않으므로 직사각형의 크기가 커질수록 상한의 질이 떨어진다는 단점을 가지고 있다.

두 번째 방법은 Hifi와 Zissimopoulos[10]가 제안한 방법으로 제품의 레이아웃을 고려하지 않고 면적만을 고려한 1차원 배낭문제의 해법을 이용한 것이다. 즉, 배낭의 최대 적재 무게와 적재하는 제품의 무게를 각각 자재의 면적과 제품의 면적으로 하는 1차원 배낭문제의 최적해를 상한으로 이용한다. 이 방법 역시 Gilmore와 Gomory[8]와 마찬가지로 동적계획법을 이용한 것이므로 직사각형 (L, W) 에 대한 상한을 계산하면 $x \leq L$, $y \leq W$ 인 임의의 직사각형 (x, y) 에 대한 상한도 계산된다. 이 상한은 제품 개수 제한을 고려하기 때문에 최적해법에 Gilmore와 Gomory[8]의 상한과 같이 적용할 경우 효과를 볼 수 있다. 계산량은 $O(n \times L^2 \times W^2)$ 이지만 제품 개수 제한으로 인해 Gilmore와 Gomory[8]보다 더 빠르고 메모리 요구량도 LW 로 Gilmore

와 Gomory[8]의 방법보다 작다. 단점은 제품 레이아웃을 고려하지 않기 때문에 직사각형에 포함된 제품들의 면적합이 직사각형의 면적을 넘지 않더라도 기하학적으로 불가능한 결과가 나올 수 있다.

3. 제안하는 상한

먼저 알고리즘의 효율을 위해 다음을 제안한다. 자재 (X, Y) 에 대해 $S(X)$ 과 $S(Y)$ 을 식 (1)과 같이 정의하자.

$$\begin{aligned} S(X) &= \max \{x : x = \sum l_i a_i, x \leq X, a_i \text{는 양의 정수}\} \\ S(Y) &= \max \{y : y = \sum w_i a_i, y \leq Y, a_i \text{는 양의 정수}\} \end{aligned} \quad (1)$$

즉, $S(X)$ 는 자재의 길이 X 를 넘지 않으면서 낭비되는 부분을 최소로 하는 제품 길이의 조합이다. 자재를 정규절단을 하게 되면 낭비되는 부분이 자재의 오른쪽과 위쪽 모서리에 발생하게 되므로[4], 정규절단을 할 경우 $(S(X), S(Y))$ 에 대한 상한은 (X, Y) 에 대한 상한이 될 수 있다.

$$U(S(X), S(Y)) \leq U(X, Y) \quad (2)$$

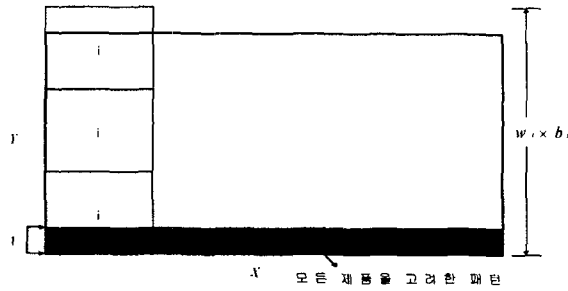
따라서 본 논문에서는 이후로 자재 (X, Y) 를 $(S(X), S(Y))$ 로 간주하고 문제를 풀기로 한다.

자재 (X, Y) 에 대한 상한을 구할 때, 그림 1의 (a)와 같이 길이 X , 너비 1이 되도록 자재를 절단하고 이때 나온 조각을 '수평띠'라 하자. 수평띠를 만들고, X 을 배낭의 최대적재무게, $l_i (i=1, \dots, n)$ 을 배낭에 넣을 제품의 무게, v_i/w_i 를 가치로 하는 제품 개수 제한이 있는 1차원 배낭문제를 푼다. 즉, 수평띠의 너비가 1이므로 너비는 고려하지 않고 길이만을 고려한 배낭문제를 풀어 수평띠의 최대가치를 구한다. 이 배낭문제의 최적해를 $f(X)$ 라 하면 $f(X)$ 는 식 (3)과 같이 정의할 수 있다.

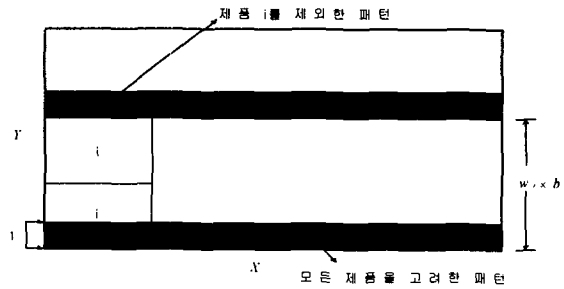
$$f(X) = \max \left\{ \sum x_i \left(\frac{v_i}{w_i} \right) : \sum x_i l_i \leq X, x_i \text{는 양의 정수}, i=1, 2, \dots, n \right\} \quad (3)$$

즉, $f(X)$ 는 모든 제품의 수평방향의 레이아웃과 제품 개수 제한을 고려한 수평띠의 최대가치가 된다. 이와 같이 구한 $f(X)$ 를 이용하여 직사각형 (X, Y) 의 수평띠를 이용한 상한 H 를 식 (4)와 같이 정의한다($\lfloor x \rfloor$ 는 x 보다 크지 않은 최대 정수).

$$H(X, Y) = \lfloor f(X) \times Y \rfloor \quad (4)$$



(a) 제품의 너비합이 직사각형의 너비보다 큰 경우



(b) 제품의 너비합이 직사각형의 너비보다 작은 경우

그림 1. 수평띠를 이용한 개선된 상한

임의의 제품 i 에 대해 제품 i 의 개수 제한을 b_i 라 하면 직사각형 (X, Y) 내에 있는 임의의 수평 띠에서 제품 i 를 포함한 수평띠의 개수는 수평 띠의 너비가 1이므로 $w_i \times b_i$ 를 넘을 수 없다. 이 사실로부터 $H_i(X, Y)$ 를 식 (5)와 같이 정의할 수 있다(그림 1 참조).

$$H_i(X, Y) = \begin{cases} \lfloor f(X) \times Y \rfloor & \text{if } (Y \leq w_i b_i) \\ \lfloor f(X) \times w_i b_i \rfloor + \lfloor f_i(X) \times (Y - w_i b_i) \rfloor & \text{if } (Y > w_i b_i) \end{cases} \quad (5)$$

$$f_i(X) = \max \left\{ \sum x_j \left(\frac{v_j}{w_j} \right) : \sum x_j l_j \leq X, x_j \text{는 양의 정수}, j=1, 2, \dots, n, j \neq i \right\} \quad (6)$$

여기서 $f_i(X)$ 는 제품 i 를 제외한 경우의 수평띠의 최대 가치를 나타낸다. 그림 1의 (a)는 제품 i 를 최대로 포함하였을 때 제품 i 의 너비합이 직사각형의 너비보다 큰 경우이다. 이 경우에는 상한을 구할 때 $f(X)$ 만을 이용한다. 그림 1의 (b)는 제품 i 를 최대로 포함하였을 때, 제품 i 의 너비합이 직사각형의 너비보다 작은 경우이다. 이 경우에는 제품 i 의 최대 너비합 $w_i \times b_i$ 까지는 $f(X)$ 를 이용하고, 나머지 $Y - w_i b_i$ 부분에 대해서는 제품 i 가 더 이상 포함될 수 없으므로 $f_i(X)$ 를 이용한다. $f_i(X) \leq f(X)$ 이므로 $H_i(X, Y) \leq H(X, Y)$ 이다. 즉, 배낭문제에서 모든 제품을 고려했을 때의 해가 제품을 하나 제외한 경우의 해보다 크므로 $H_i(X, Y)$ 는 항상 $H(X, Y)$ 보다 작거나 같다. 위의 방법을 모든 제품에 대해 적용하면 수평띠를 이용한 새로운 상한 $H'(X, Y)$ 를 식 (7)과 같이 정의할 수 있다.

$$H'(X, Y) = \min \{ H_i(X, Y), \quad i=1, 2, \dots, n \} \quad (7)$$

수직 방향에 대해서도 같은 방법으로 다음과 같은 상한을 구할 수 있다.

$$V'(X, Y) = \min \{ V_i(X, Y), \quad i=1, 2, \dots, n \} \tag{8}$$

$$V_i(X, Y) = \begin{cases} |g(Y) \times X| & \text{if } (X \leq l_i b_i) \\ |g(Y) \times l_i b_i| + |g_i(Y) \times (X - l_i b_i)| & \text{if } (X > l_i b_i) \end{cases} \tag{9}$$

$$g(Y) = \max \left\{ \sum y_i \left(\frac{v_i}{l_i} \right) : \sum y_i w_i \leq Y, y_i \text{는 양의 정수}, i=1, 2, \dots, n \right\} \tag{10}$$

$$g_i(Y) = \max \left\{ \sum y_j \left(\frac{v_j}{l_j} \right) : \sum y_j w_j \leq Y, y_j \text{는 양의 정수}, j=1, 2, \dots, n, j \neq i \right\} \tag{11}$$

식(7)와 식(8)로부터 식(12)와 같은 새로운 상한 U 를 구할 수 있다.

$$U(X, Y) = \min(H'(X, Y), V'(X, Y)) \tag{12}$$

제안하는 상한은 상한을 구하기 위한 배낭문제를 풀 때, 제품의 한쪽 방향에 대한 레이아웃만을 고려하는 1차원 배낭문제를 풀기 때문에, 양쪽 방향의 레이아웃을 고려하여 2차원 배낭문제의 최적해를 구하는 Gilmore와 Gomory[8], 면적을 고려하는 Hifi와 Zissimopolous[10]와 비교하여 계산량과 메모리 요구량이 작다. 제안하는 상한의 계산량과 메모리 요구량은 각각 $O(n^2 \times W + n^2 \times L + k)$ 와 $nW + nL$ 이다. 여기서 k 는 상한 계산때마다 1차원 배낭문제의 해 중에서 가장 작은 값을 찾는데 소요되는 부가적인 계산량이다. 또한 제품 레이아웃과 제품 개수 제한을 고려하므로, 제품 레이아웃을 고려하지 않는 Hifi와 Zissimopolous[10]의 단점과 제품 개수 제한을 고려하지 않는 Gilmore와 Gomory[8]의 단점을 보완할 수 있다.

4. 제안하는 상한의 증명

i 를 제품, j 를 자재의 아래쪽 모서리에서 $j-1$ 만큼 떨어진 수평띠, x_{ij} 를 수평띠 j 에 제품 i 가 출현한 횟수라고 할 때, 절단문제는 아래의 모형으로 변형할 수 있다.

$$\begin{aligned} \max \quad z &= \sum_{j=1}^W \sum_{i=1}^n x_{ij} (v_i / w_i) \\ \text{s.t.} \quad & \sum_{i=1}^n x_{ij} l_i \leq L \\ & \sum_{j=1}^W x_{ij} \leq w_i b_i \\ & \dots \\ & x_{ij} \leq b_i \quad x_{ij} \text{는 정수}, \quad \forall i, \forall j \end{aligned} \tag{13}$$

임의의 절단패턴에 대해 제품 a 가 포함된 수평 띠의 집합을 J_a , 포함되지 않은 집합을 \bar{J}_a 라 하자.

$$\begin{aligned} J_a &= \{j : x_{aj} \geq 1\} \\ \bar{J}_a &= \{j : x_{aj} = 0\} \end{aligned} \tag{14}$$

J_a 와 \bar{J}_a 에 대해 식 (15)와 식 (16)이 성립한다.

$$\begin{aligned} w_a b_a &\geq |J_a| \\ W - w_a b_a &\leq |\bar{J}_a| \end{aligned} \quad (\text{if } W > w_a b_a) \tag{15}$$

$$|J_a| + |\bar{J}_a| = W \tag{16}$$

그리고 $f(L)$, $f_a(L)$ 에 대해 식 (17)이 성립한다.

$$f(L) \geq f_a(L) \tag{17}$$

z_j 를 절단 문제의 임의의 절단패턴에서 수평띠 j 의 가치라고 식 (18)이 성립한다.

$$\begin{aligned} z_j &\leq f(L), \quad j \in J_a \\ z_j &\leq f_a(L), \quad j \in \bar{J}_a \end{aligned} \tag{18}$$

식 (18)에 의해 식 (19)가 성립한다.

$$\begin{aligned} |J_a| f(L) &\geq \sum_{j \in J_a} z_j \\ |\bar{J}_a| f_a(L) &\geq \sum_{j \in \bar{J}_a} z_j \end{aligned} \tag{19}$$

식 (19)의 우변의 합 $\sum_{j \in J_a} z_j + \sum_{j \in \bar{J}_a} z_j$ 은 원 문제의 목적식 값이므로 좌변과 우변을 각각 더하면 식 (20)이 성립한다.

$$|J_a| f(L) + |\bar{J}_a| f_a(L) \geq \sum_{j \in J_a} z_j + \sum_{j \in \bar{J}_a} z_j = z \tag{20}$$

식 (20)의 좌변 $|\bar{J}_a| f_a(L) + |J_a| f(L)$ 에 대해 식 (15), (16), (17)에 의해서 식 (21)이 성립한다.

$$|J_a| f(L) + |\bar{J}_a| f_a(L) \leq \begin{cases} f(L) \times w_a b_a + f_a(L) \times (W - w_a b_a) & \text{if } (W > w_a b_a) \\ f(L) \times W & \text{if } (W \leq w_a b_a) \end{cases} \tag{21}$$

식 (21)의 우변은 식 (5)에서 정의한 $H_a(L, W)$ 이므로 식 (20)에 의해서 식 (22)가 성립한다.

$$H_a(L, W) \geq z \tag{22}$$

따라서 식 (23)도 성립한다.

$$H'(L, W) = \min\{H_a(L, W), \quad a=1, 2, \dots, n\} \geq z \quad (23)$$

$V(L, W)$ 에 대해서도 같은 방법으로 전개하면 식 (24)가 만족되므로 본 연구에서 제안하는 상한은 항상 최적해보다 크거나 같다.

$$z \leq U(L, W) \quad (24)$$

5. 수치 예제

길이 3, 너비 5인 자재와 (길이, 너비, 가치, 개수제한)으로 주어지는 세 개의 제품 a(1, 3, 15, 1), b(2, 3, 10, 5), c(1, 4, 4, 3)이 있다고 가정하자(그림 2 참조).

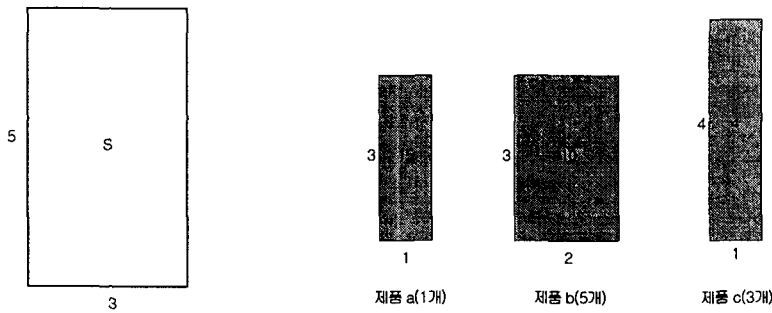


그림 2. 절단 문제의 예

그림 3은 제안하는 방법으로 상한을 구하는 예를 보여준다. 이 문제의 경우 모든 제품을 고려한 수직띠의 최대가치와 제품 a를 제외했을 때의 수직띠의 최대가치를 이용하여 상한을 구한다. 세 가지의 상한(Gilmore와 Gomory[8]의 방법, Hifi와 Zissimopoulos[10]의 방법, 제안하는 방법)을 비교하면 표 1과 같다.

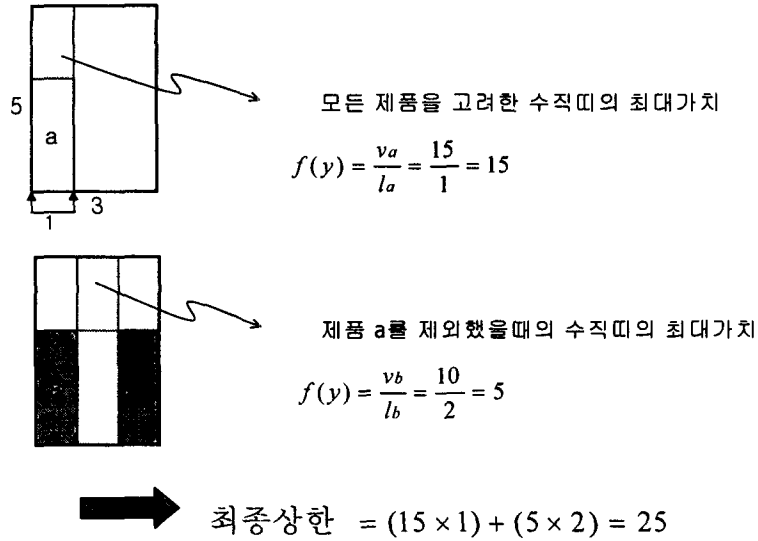
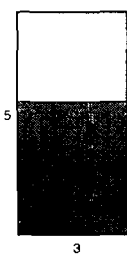


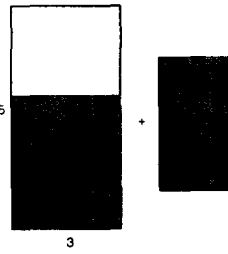
그림 3. 제안하는 상한을 구하는 예

표 1. 상한 값의 비교

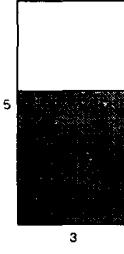
	Gilmore와 Gomory의 방법	Hifi와 Zissimopoulos의 방법	제안하는 방법
결과값	45	35	25
패턴	제품 a(1, 3, 15, 1)*3	제품 a(1, 3, 15, 1)*1 제품 b(2, 3, 10, 5)*2	제품 a(1, 3, 15, 1)*1 제품 b(1, 4, 4, 3)*1
설명	제품 a의 개수 제한 조건을 만족하지 않음	패턴에 포함된 제품의 면적 합이 자재의 면적을 넘지 않으나 불가능한 패턴임	제품 너비에 대한 1차원 배낭문제의 해(a:1개, b:1개)를 이용하여 산출



(a) Gilmore와 Gomory



(b) Hifi와 Zissimopoulos



(c) 제안하는 방법

그림 4. 상한 값의 비교

위의 결과에서 보듯이 제안하는 상한은 기존 두 가지 상한의 단점이 나타나는 경우, 즉 제품의 크기가 개수 제한을 고려하더라도 자재에 비해 많이 작은 경우와 제품의 면적만을 고려하여 상한을 구할 때 기하학적으로 불가능한 결과가 나오는 경우에 큰 효과를 발휘한다.

6. 실험 결과

본 연구에서 제안하는 상한의 우수성을 보이기 위해 세 가지 상한(Gilmore와 Gomory[8], Hifi와 Zissimopolous[10], 제안하는 방법)을 비교 실험하였다. 또한 제안하는 상한을 절단문제의 해법에 적용하였을 경우의 효과를 알아보기 위해 MCW를 구현하여 실험한다. 실험에 사용한 문제는 Cung 등[5]에서 사용한 문제를 이용하고, Pentium III 667MHz, RAM 128M의 IBM 호환 컴퓨터에서 실험을 수행하였다.

표 2은 각 문제에서 자재 (L, W)가 주어질 때, $x \leq L, y \leq W$ 인 임의의 직사각형 (x, y)에 대해 제안하는 방법을 이용하여 구한 상한값과 Gilmore와 Gomory[8], Hifi와 Zissimopolous[10]의 두 가지 방법을 이용해 구한 상한값중 더 낮은 상한값을 비교하여 제안하는 상한값이 작은 경우, 같은 경우, 큰 경우의 횟수를 나타낸 것이다. 문제 3, A1, A2의 경우 기존의 두 가지 방법을 같이 이용했을 때보다 제안하는 상한이 더 낮은 상한값을 구하는 횟수가 더 많았고, 각 문제에 대해서 낮은 상한값을 구한 횟수가 평균적으로 9.15% 나타났다. 표에서 보듯이 모든 문제에 대해 제안하는 상한이 더 낮은 상한값을 구하는 경우가 발생하는데, 여러 상한들 중 가장 작은 값을 이용하는 방법을 이용할 경우 제안하는 상한으로 인해 해법의 성능이 향상될 수 있음을 의미한다.

표 3은 MCW를 구현하여 기존의 두 가지 상한을 이용한 경우와 제안하는 상한을 추가로 적용한 경우로 나누어 실험한 결과이다. 실험 결과 새로운 상한을 적용한 경우, 1시간이내에 푼 문제들에 대해 분지 되는 노드의 개수는 평균 26.9%가 감소했고, 수행시간면에서는 평균적으로 31.2%의 개선 효과가 있었다. 기존 상한에 추가로 적용하기 때문에 탐색되는 노드 개수는 기존 상한을 이용했을 때보다 항상 작거나 같음을 알 수 있다.

표 2에서 낮은 상한값을 구하는 비율이 가장 높은 A1, 3, A2의 경우 표 3의 MCW 수행결과에서도 가장 개선 효과가 큰 것으로 나타났고, 이 문제들의 특징을 살펴보면 모두 제품의 가치와 면적의 상관관계가 약한 문제(weighted)였고, 각 제품에 대해 개수 제한이 자재의 면적을 고려할 때 비교적 큰 문제들이었다.

7. 결론

제품 개수에 제한이 있는 2차원 길로틴 절단문제를 위한 기존의 최적 해법은 자재를 제품의 크기로 절단해 나가는 하향식 접근 방법과 제품들을 수직/수평으로 결합하여 나가는 상향식 접근 방법으로 연구가 진행되어 왔다. 두 방식에서 상한을 구할 때, Gilmore와 Gomory[8]의 제품 개수 제한 없는 2차원 배낭문제의 해법과 Hifi와 Zissimopolous[10]의 1차원 배낭문제를 이용한 해법을 사용한다. 특히, Gilmore와 Gomory[8]의 해법은 직사각형에 포함되는 제품의 개수를 알아내는 데도 이용되기 때문에 최적해법에 반드시 필요하다. 두 가지 방법 각기 단점을 가지고 있는데, Gilmore와 Gomory[8]의 해법은 제품 개수 제한을 고려하지 않고, Hifi와 Zissimopolous[10]의 방법은 제품 레이아웃을 고려하지 않는다.

본 연구에서는 기존 해법에서 사용된 상한들을 보완하는 새로운 상한을 제안한다. 이 상한은 제품의 레이아웃과 제품 개수 제약을 동시에 고려함으로써 기존 상한들의 단점을 보완한다. 제안하는 상한은 기존 상한들보다 메모리 요구량이 적으며, 최적 해법에 기존 상한들과 함께 적용할 경우 더 낮은 상한을 구하여 알고리즘의 성능을 향상시킨다.

표 2. 제안하는 상한과 기존 두 가지 상한의 비교

문제	길이×너비	제품 종류수	제안하는 상한 vs. min(HZ, GG)			
			제안하는 상한값이 더 낮은 경우 (횟수)	상한값이 같은 경우 (횟수)	제안하는 상한값이 더 큰 경우 (횟수)	낮은 경우의 비율*
2	40×70	10	427	958	1415	15.25
3	40×70	20	1034	1020	746	36.93
A1	50×60	20	1111	1047	842	37.03
A2	60×60	20	1095	1417	1088	30.42
STS2	55×85	30	552	1350	2773	11.81
STS4	99×99	20	693	2775	6333	7.07
CHL2	62×55	10	148	1098	2164	4.34
CHL3	157×21	15	1901	1853	15243	10.01
CHL4	207×231	15	6501	4620	36696	13.60
2s	40×70	10	179	1194	1427	6.39
3s	40×70	20	267	1071	1462	9.54
A1s	50×60	20	197	1106	1697	6.57
A2s	60×60	20	98	1472	2030	2.72
STS2S	55×85	30	81	1908	2686	1.73
STS4S	99×99	20	335	3646	5820	3.42
CHL1S	132×100	30	32	4032	9136	0.24
CHL2S	62×55	10	103	1089	2218	3.02
CHL3S	157×121	15	556	7639	10802	2.93
CHL4S	207×231	15	3519	9621	34677	7.36
A3	70×80	20	254	2210	3136	4.54
A4	90×70	20	514	1892	3894	8.16
A5	132×100	20	23	3448	9729	0.17
HH	127×98	5	998	3637	7811	8.02

* : GG는 Gilmore와 Gomory[8]의 방법임
 * : HZ는 Hifi와 Zissimopoulos[10]의 방법임
 * : $\frac{\text{제안하는 상한이 낮은 경우의 횟수}}{\text{전체 횟수}} \times 100(\%)$

표 3. 제안하는 상한을 추가로 적용한 경우와 기존 상한을 이용한 경우의 MCW 수행결과 비교

문제	길이×너비	제품종류수	노드 개수			계산 시간		
			기존 방법 (개수)	제안하는 상한 추가 적용 (개수)	개선 효과 (%)	기존 방법 (초)	제안하는 상한 추가 적용 (초)	개선 효과 (%)
2	40×70	10	12833	11231	12.48	13.098	11.606	11.39
3	40×70	20	29372	14724	49.87	73.345	32.456	55.74
A1	50×60	20	45902	12111	73.61	151.027	29.542	80.43
A2	60×60	20	163230	73060	55.24	614.443	293.902	52.16
STS2	55×85	30	58591	36279	38.08	116.287	64.492	44.54
STS4	99×99	20	86685	53640	38.12	142.525	66.776	53.14
CHL2	62×55	10	3690	3637	1.40	4.676	4.085	12.63
CHL3	157×21	15	0	0		0	0	
CHL4	207×231	15	0	0		0	0	
2s	40×70	10	120453	83381	30.77	180.890	124.329	31.26
3s	40×70	20	496	484	2.41	0.641	0.63	1.71
A1s	50×60	20	0	0		0	0	
A2s	60×60	20	339	230	32.15	0.320	0.200	37.50
STS2S	55×85	30	2663	1805	32.21	3.635	2.353	35.26
STS4S	99×99	20	65059	48837	24.93	167.040	122.596	26.06
CHL1S	132×100	30	0	0		0	0	
CHL2S	62×55	10	6401	6110	4.50	7.600	7.140	6.05
CHL3S	157×121	15	0	0		0	0	
CHL4S	207×231	15	0	0		0	0	
A3	70×80	20	4858	3829	21.18	7.450	4.646	37.63
A4	90×70	20	*	300685		*	597.318	
A5	132×100	20	218800	214871	1.79	508.110	505.687	0.47
HH	127×98	5	282451	245753	13.00	533.527	459.250	13.92

- : 하한과 상한이 최적해와 같은 경우

참 고 문 헌

1. 강맹규, *네트워크와 알고리즘*, 박영사, 서울, 1991.
2. Beasley, J. E., "An exact two-dimensional non-guillotine cutting tree search procedure," *Operations Research*, Vol. 33, pp. 49-64, 1985.
3. Beasley, J. E., "Algorithms for unconstrained two-dimensional guillotine cutting," *Journal of the Operational Research Society*, Vol. 36, pp. 297-306, 1985.
4. Christofides, N. and Whitlock, C., "An algorithm for two-dimensional cutting problems," *Operations Research*, Vol. 26, pp. 30-44, 1976.
5. Cung, V. D., Hifi, M. and Cun, B. L., "Constrained two-dimensional cutting Stock problems a best-first branch-and-bound algorithm," *International Transactions in Operational Research*, Vol. 7, pp. 185-210, 2000.
6. Desler, J. F. and Hakimi, S. L., "A graph theoretic approach to a class of integer programming problems," *Operations Research*, Vol. 26, pp. 1017-1033, 1969.
7. Gilmore, P. C. and Gomory, R. E., "A linear programming approach to the cutting-stock problem," *Operations Research*, Vol. 9, pp. 849-859, 1961.
8. Gilmore, P. C. and Gomory, R. E., "The theory and computation of knapsack functions," *Operations Research*, Vol. 14, pp. 1045-1074, 1969.
9. Haessler, R. W. and Sweeney, P. E., "Cutting stock problems and solution procedures," *European Journal of Operational Research*, Vol. 54, pp. 141-150, 1991.
10. Hifi, M., "An improvement of Viswanathan and Bagchi's exact algorithm for constrained two-dimensional cutting stock," *Computers and Operations Research*, Vol. 14, pp. 727-736, 1997.
11. Hifi, M. and Zissimopoulos, V., "Constrained two-dimensional cutting: An improvement of Christofides and Whitlock's exact algorithm," *Journal of the Operational Research Society*, Vol. 48, pp. 324-331, 1997.
12. Pisinger, D., "Algorithms for knapsack problems," Ph.D. thesis, DIKU, University of Copenhagen, 1995.
13. Viswanathan, K. V. and Bagchi, A., "Best-first search methods for constrained two-dimensional cutting stock problems," *Operations Research*, Vol. 41, pp. 768-776, 1993.
14. Wang, P. Y., "Two algorithms for constrained two-dimensional cutting stock problems," *Operations Research*, Vol. 31, pp. 573-586, 1983.
15. Zissimopoulos, V., "Heuristic methods for solving (un)constrained two-dimensional cutting stock problems," *Methods of Operations Research*, Vol. 49, pp. 345-357, 1984.