

금속 그레이팅 생산을 위한 A* 알고리즘의 2차원 절단 문제에의 적용

김진명*, 조대호*

*성균관대학교 컴퓨터공학과
e-mail:kjm77@ece.skku.ac.kr

Application of A* Algorithm to Two-Dimensional Cutting Problem for Producing Metal Grating

Jin-Myoung Kim*, Tae-Ho Cho*

*Dept of Computer Engineering, Sungkyunkwan University

요 약

그레이팅(grating) 생산 공정에서 절단공정은 패널(panel)을 원하는 형태의 그레이팅으로 절단하여 제품을 생산한다. 실제 절단이 이루어지기 전에 절단 계획을 생성하는데, 이것은 절단 대상이 되는 그레이팅들의 배치 방법을 결정한다. 이 계획은 제품의 생산비용의 증감에 중요한 역할을 한다. 절단 계획의 생성은 2차원 절단 문제와 유사하다. 본 논문에서는 금속 그레이팅 생산 공정의 절단 계획에 대하여 연구하고 인공지능 탐색 방법 중 하나인 A* 알고리즘을 계획 생성에 적용한다. 또한 시뮬레이션을 통하여 자재의 손실을 기준으로 제안된 방법과 기존의 방법을 비교 평가 하겠다.

1. 서론

NP-hard로 잘 알려진 절단 문제는 컴퓨터 과학, 산업 공학, 업무 계획, 제조업, 경영 그리고 생산 공정 등 많은 실세계 에서 통용된다[1]. 절단 문제는 요구되는 크기의 금속 각형을 절단하거나, 가구를 만들기 위하여 나무 시트(sheet)를 절단할 때 발생한다.

절단 문제는 Kantorovich[2]가 가장 처음으로 다루었고, 이 후에 Gilmore와 Gomory에 의하여 연구되었다[3,4]. Christofides 와 Whitlock는 트리 탐색 방법을 사용하여 2차원 절단 문제를 최적으로 해결하였으며[5], Beasley 또한 같은 방법을 사용하였다[6]. Beasley는 dynamic programming을 이용하여 최적 알고리즘과 휴리스틱 알고리즘을 비교하였다[7]. Hifi 와 Zissimopolous는 Christofides와 Whitlock에 의하여 사용된 해결방법을 개선하여 정확한 알고리즘으로 표현하였다[8].

금속 제품을 생산하는 S 회사는 다양한 형태의 그레이팅을 제작한다. 그레이팅(Grating)은 [그림 1]

과 같이 길가의 배수로 덮게, 공장이나 선박의 내부 바닥을 구성하는 등, 여러 가지 형태로 사용된다. 금속 그레이팅 생산 공정에서 절단 계획은 패널(panel or sheet)을 효율적으로 사용하기 위하여 원하는 형태의 그레이팅들을 절단하기 위한 배치 방법을 생성한다.



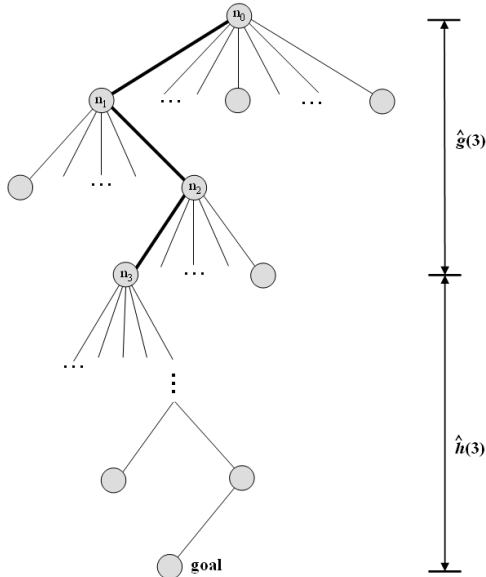
(그림 1 그레이팅의 사용)

절단 계획에서 그레이팅의 배치 방법을 생성 하는 문제는 2차원 절단 문제와 유사하다. 이 논문에서는 가장 좋은 절단 계획의 선택을 하기 위하여 인공지

능 탐색 방법 중 하나인 A* 알고리즘을 절단 계획에 적용한다.

2. A* 알고리즘

그래프의 탐색에서, A* 알고리즘은 휴리스틱 평가 함수를 사용하여 탐색한다[9].



(그림 2 가상적인 탐색 공간)

트리에서 임의의 노드 n 에 대하여, $h(n)$ 을 노드 n 과 목표 노드 사이의 최단 경로(n 에서 모든 목표 노드까지의 모든 가능한 경로 중에서)의 실제 값이라고 하자.

$g(n)$ 을 시작 노드 n_0 에서 노드 n 까지의 최단 경로의 값이라고 하자. 그림 수식 (1)은 n_0 에서 목표 노드까지 노드 n 을 통하여 갈 수 있는 모든 가능한 경로 중에서 최단 경로의 값이 된다.

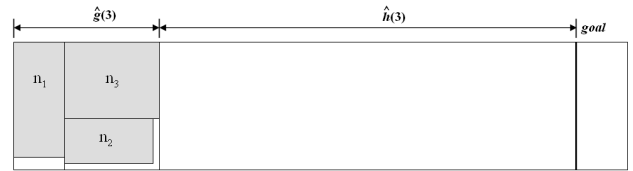
$$f(n) = g(n) + h(n) \tag{1}$$

각 노드 n 에 대하여 $\hat{h}(n)$ (휴리스틱 요소)을 $h(n)$ 의 추정값이라고 하고, $\hat{g}(n)$ (깊이 요소)을 $g(n)$ 이 A* 알고리즘에 의하여 지금까지 발견된 노드 n 까지의 경로 중에서 최단 경로의 값이라고 하면, A* 알고리즘에서는 수식 (2)를 평가함수로 사용한다.

$$\hat{f}(n) = \hat{g}(n) + \hat{h}(n) \tag{2}$$

[그림 2]은 식 (2)에 의한 가상적인 탐색 공간을 보

여준다. 노드 n_1, n_2 그리고 n_3 은 휴리스틱 평가 함수 수식 (2)에 의하여 선택된 것이다. [그림 2]에서 노드는 그레이팅을 의미하고, 노드와 노드를 연결하는 간선은 절단 계획에서의 절단 비용과 그레이팅의 배치 정보를 포함하고 있다. A* 알고리즘에 의하여 확장된 굵은 선은 시작 노드부터 목표 노드까지의 최소 비용을 의미한다. [그림 3]은 패널에서의 가상적인 그레이팅의 배치를 보여준다.

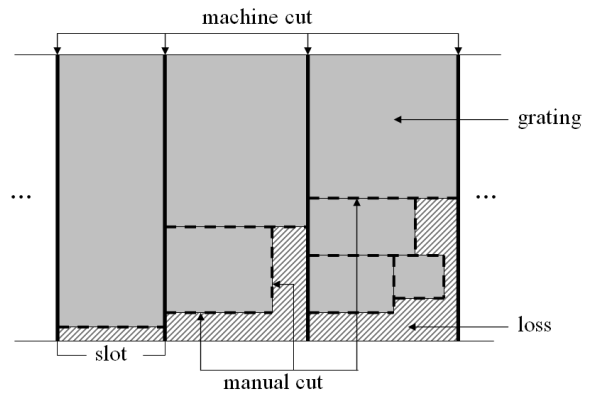


(그림 3 패널에서의 가상적인 배치)

[그림 3]에서 그레이팅 n_1, n_2 그리고 n_3 은 A* 알고리즘에 의해서 선택된 것이고, 이 그레이팅들은 [그림 2]에서의 노드 n_1, n_2 그리고 n_3 과 같다.

3. 그레이팅 절단 공정

금속 그레이팅 생산 공정에서 절단 공정은 원하는 형태의 크기로 패널을 절단 하여 그레이팅을 생산한다. 절단 계획은 절단 공정에서 실제 패널을 절단하기 이전에 그레이팅을 배치하여 절단하는 방법을 생성한다. 따라서 그레이팅 생산 공정에서 발생하는 자재의 손실은 절단 계획 단계에서 결정된다.



(그림 4 Machine and manual cut)

[그림 4]는 패널에서 그레이팅의 배치, 절단, 슬롯의 생성 그리고 자재의 손실을 보여준다. 절단 작업은 machine cut와 manual cut로 나뉘어진다. [그림 4]에서 Machine cut은 슬롯(slot) 단위로 이루어지며 굵은선(수직 절단)이 이것을 의미하고 점선이 Manual cut을 의미한다. [그림 4]에서 그레이팅이 패널에 배치되지 않고 남은 빗금친 부분이 자재 손

실(loss)이라 한다.

4. A* 휴리스틱 평가 함수

4.1 깊이 요소의 추정

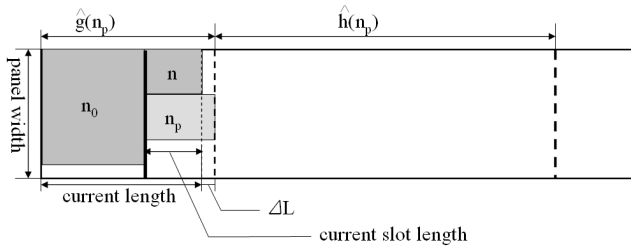
A* 알고리즘에서 휴리스틱 평가 함수는 수식 (2)에서 정의 되었다. 탐색 대상이 되는 트리에서, p 번째 노드 n_p 에 대하여 $\hat{g}(n_p)$ 는 다음과 같이 정의된다.

$$\begin{aligned} \hat{g}(n_p) &= 0 && \text{if } n_p = n_0 \\ &= (n_{parent}) + cost(n_p) && \text{otherwise} \\ cost(n_p) &= \Delta L \\ \Delta L &= 0 && \text{if } cslength \geq n_{p.length} \\ &= n_{p.length} - cslength && \text{otherwise} \end{aligned} \quad (3)$$

where

$cslength$: Current slot length

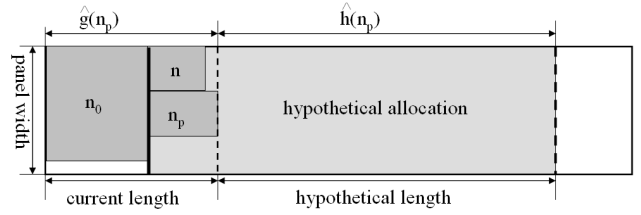
n_0 은 초기노드로써 어떠한 그레이팅도 배치되지 않은 상태이고, n_{parent} 는 n_p 노드 이전의 모든 노드에 대한 탐색 비용의 누적을 의미한다. 수식 (5)에서 $cslength$ 와 ΔL 은 [그림 5]에서 current slot length와 노드 n_p 의 배치로 인한 증가된 길이 즉 노드 n_p 의 길이에서 current slot length를 뺀 것이다.



(그림 5 그레이팅의 배치)

5.2 휴리스틱 요소의 추정

배치하지 않은 그레이팅의 집합을 M 이라 하고, M 의 그레이팅 들이 [그림 6]과 같이 자재의 손실 없이 배치된다고 가정하자.



(그림 6 가상적 배치)

자재의 손실이 전혀 발생하지 않고 배치되는 경우 모든 슬롯의 길이의 합은 다음과 같다.

$$\text{current length} + \text{hypothetical length} \quad (7)$$

그레이팅의 배치로 인하여 생성되는 자재의 손실은 $0 \leq loss$ 이므로, 가상적인 배치로 인한 길이 수식 (7)의 값은 항상 실제 배치에 의하여 발생하는 길이보다 작거나 같다. 따라서 실제 그레이팅이 배치되어 생성된 모든 슬롯의 합을 TSL (Total Slot Length)이라하면

$$\text{current length} + \text{hypothetical length} \leq TSL \quad (8)$$

이다. [그림 6]에서 패널의 너비(panel width)는 고정되어 있으므로 가상적인 길이(hypothetical length)를 수식 (9)과 같이 구할 수 있다.

$$\hat{h}(n) = \left\{ \sum_{k=1}^p n_{k.width} \times n_{k.length} \right\} / \text{panelwidth} \quad (9)$$

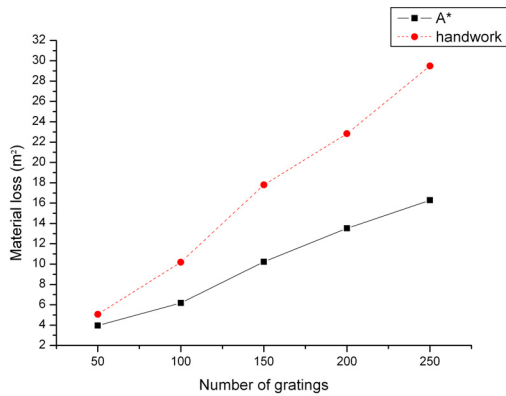
where

$$M = \{n_k | 1 \leq k \leq p\}$$

가상적인 배치는 자재의 손실이 없이 진행된다고 가정하였기 때문에 수식 (9)에서 배치해야 하는 그레이팅의 집합의 모든 원소의 넓이의 합을 패널의 길이로 나누게 되면 가상적인 길이(hypothetical length)를 구할 수 있게 된다.

6. 시뮬레이션 결과

절단 계획에서 우리는 시뮬레이션을 통하여 기존의 방법(수작업)과 A* 알고리즘이 적용된 방법을 비교 평가 하겠다.



(그림 7 기존의 방법과 제안된 방법의 비교)

[그림 7]은 S사에서 제품을 생산하는데 사용되었던 실제 데이터를 기반으로 제안된 방법과 기존의 방법을 사용하여 절단 계획을 생성한 것이다. [그림 7]을 통하여 A* 알고리즘을 적용한 방법이 기존의 것보다 자재의 손실이 작은 것을 알 수 있다. 또한 그레이팅의 개수가 증가 할수록 자재 손실의 차이가 커진다. 이는 그레이팅의 개수가 증가할수록 절단 계획에 대한 복잡도가 증가하기 때문이다.

6. 결론

이 논문에서 우리는 그레이팅 생산 공정의 절단 계획에서 발생하는 절단 문제와 이에 대한 적절한 해결방법에 대하여 연구하였다. 우리는 절단 계획에 A* 알고리즘을 적용하였으며 시뮬레이션을 통하여 제안한 알고리즘이 기존의 방법보다 생산 효율성이 높게 나타난 것을 보였다. 따라서 이것은 생산비용의 절감효과를 가져온다. 또한 수작업으로 절단 계획을 생성하는 기존의 방법에 비하여 컴퓨터를 이용한 빠른 계획의 생성은 제품의 생산 속도를 증가시킬 수 있다.

참고문헌

- [1] Van-Dat Cung, Mhand Hifi, Bertrand Le Cun, "Constrained two dimensional cutting stock problems best first branch and bound algorithm", Op Res. 7, pp.185-210, 2000.
- [2] L. V. Kantorovich, "Mathematical method of organizing and planning production", Mgmt Sci., pp363-442, 1960.
- [3] P. C. Gilmore, R.E. Gomory, "Multistage cutting problems of two and more dimensions", Op Res., pp94-119, 1965.

[4] P. C. Gilmore, "Cutting stock, linear programming, knapsacking, dynamic programming and integer programming, some interconnections", Ann. Discrete Math., pp217-235, 1979.

[5] N. Christofides, C. Whitlock, "An algorithm for two dimensional cutting problems", Operations Research, pp30-44, 1977.

[6] J. E. Beasley, "An exact two dimensional non-guillotine cutting tree search procedure", Oper. Res., pp49 - 64, 1985.

[7] J. E. Beasley, "Algorithms for unconstrained two-dimensional guillotine cutting", J. Oper. Res. Soc. 36(4) 297 - 306 1985.

[8] M. Hifi, V. Zissimopoulos, "Constrained two dimensional cutting: An improvement of Christofides and Whitlock's exact algorithm", J. Oper. Res. Soc, pp324 - 331, 1997.

[9] Nils J. Nilsson, *Artificial Intelligence: A new synthesis*, Morgan Kaufmann Publishers, 1998