

유전자 알고리즘을 이용한 부재 자동배치에 관한 연구

조민철* · 박제웅

*조선대학교 대학원 선박해양공학과, 조선대학교 선박해양공학과 교수

A Study on the Piece Auto-Nesting Using Genetic Algorithm

Min-Cheol Cho*, Je-Woong Park

*Graduate school of Naval Architecture and Ocean Engineering, Chosun University

Professor, School of Naval Architecture and Ocean Engineering, Chosun University

KEY WORDS: Genetic Algorithm 유전자 알고리즘, Nesting 부재배치, Row Plate 부재배치용 강판.

ABSTRACT: In this paper, consider the three cases of decide for applying point a general Simple Genetic Algorithm about heuristic method(Bottom and Left Sliding) at the piece auto-nesting on the row plate. The 1st case, about only using the Simple Genetic Algorithm. The 2nd case, applied the heuristic method to the genetic operating of the Simple Genetic Algorithm. The 3rd case, applied the heuristic method to the final result of the Simple Genetic Algorithm. The estimation of final result were proceed to developed simulation program in this research.

1. 서 론

조선에서 사용하는 모든 부재들은 네스팅(nesting)이라는 일련의 부재 배치 및 절단 공정을 통해 생산된다. 이런 부재(piece)들은 크기와 모양이 각각 다르기 때문에 생산하는데 소요되는 시간과 공수 그리고 비용은 조선생산에 적지 않은 영향을 미친다.

네스팅에 있어서 자동화의 의미는, 첫째, 부재의 빠른 생산과 재고관리를 병행함으로써 생산성 향상과 쉬운 공정관리(부재생산)를 하는데 있으며, 둘째, 부재를 최적으로 배치함으로써 강판(row plate)을 효율적으로 사용하여 부재 생산에 드는 비용을 절감하는데 있다.

본 연구에서는 휴리스틱(heuristic) 방법(Bottom & Left Sliding)의 적용 시기를 결정하기 위해 일반적인 SGA(Simple Genetic Algorithm)와 휴리스틱 방법을 유전연산 작용 중에 적용한 GA, 그리고 유전연산 작용 후 도출된 결과 값에 적용한 GA의 결과 값을 비교하였다.

본 연구를 통해 결과 값이 좋은 방법을 채택하여 부재들의 최적 자동 배치를 시뮬레이션하고, 그에 따른 결론을 도출하였다.

2. 시뮬레이션 방법

2.1 Piece와 Row plate의 단순화

본 연구는 유전자 알고리즘에서 휴리스틱한 방법을 적용시키는 시점을 고찰하는 것이기 때문에, 부재나 강판 등을 단순

화하여 적용하였다.

부재는 크게 모양이 다른 것(①)과 같은 것으로 나누었고, 같은 것 중에서 면적이 큰 것(②)으로 구분하였다. ①은 가로와 세로를 각각 (100×100) 1개, (100×50) 2개, (100×25) 2개, (50×50) 2개, (50×25) 4개로 모두 11개, ②는 가로와 세로를 각각 (100×50) 10개로 각각 시뮬레이션을 하였으며, 강판은 ①과 ② 모두 511×511로 정의하였다.

2.2 유전자 모델링

유전자 모델링이란 유전자 알고리즘을 적용하고자 하는 모델의 결과 값을 최적화하기 위해 염색체의 배열이나 유전연산자들을 어떠한 방법으로 변형, 적용시킬 것인가를 결정하고, 이를 토대로 염색체의 구성요소를 생성하는 일련의 작업을 의미한다.

본 연구에서는 휴리스틱 방법의 적용 시기와 부재의 최적 자동배치를 목적으로 두고 있으므로, 부재와 강판에 대한 특성들을 염색체의 구성요소로 매핑(mapping)시키는 것을 의미한다. 내용은 아래와 같다.

2.3 염색체 정의

하나의 부재를 놓고 그것을 어떻게 문자열(string: '1'과 '0'으로 표현되는 일련의 문자배열)로 표현할 것인가는 유전자 알고리즘에서 가장 중요한 부분중의 하나이다. 예를 들어 사람의 눈을 염색체로 표현한다면, 눈에 해당하는 것을 phenotype이라고 하고, 이에 대응하는 염색체를 genotype이라 한다. 본 연구에서는 부재의 특성과 강판에서의 위치를 phenotype으로 하고, 이를 문자열로 표현한 것을 genotype이라 정의하였다.

부재의 특성은 다음과 같다.

| 부재의 회전 | 강판에서의 위치 |
|---|---|
| 부재 원본을 기준으로 CCW 방향으로 회전하면서 기준점 방향을 left-bottom으로 옮긴다. | 강판의 가로방향의 길이를 x 방향으로 하고, 세로방향을 y 방향으로 정의한다. |

Table. 1 Properties of the piece and row plate.

부재의 회전은 Fig.1 과 같이 4가지만을 고려하였고, 강판의 폭과 길이의 변화에 따른 부재의 배치위치 정보 x, y좌표를 결정하였다.

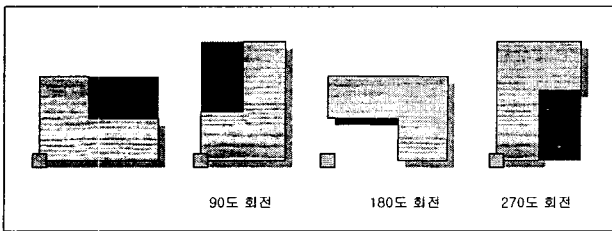


Fig. 1 Kind of simple rotation of piece.

부재의 회전과 강판의 크기를 고려하여 염색체를 정의하면 Fig. 2와 같이 나타낼 수 있다.

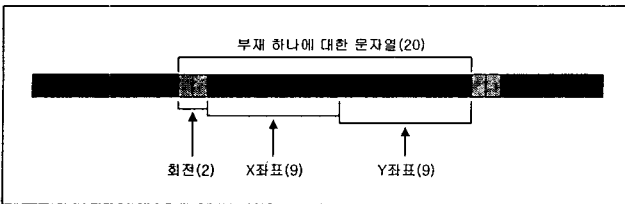


Fig. 2 Genotype of Properties of the piece.

부재가 회전할 수 있는 경우의 수(90도를 기준하여)는 4가지이므로, 2비트로 정의할 수 있게 된다. 강판에서의 부재의 위치는 강판의 가로와 세로의 길이(511×511로 고정)를 이진수로 표현하여 매핑 시켰다.

| Genotype | Phenotype | Genotype | Phenotype |
|----------|-----------|----------|-----------|
| 0 0 | 회전하지 않음 | 1 0 | 180도 회전 |
| 0 1 | 90도 회전 | 1 1 | 270도 회전 |

| Genotype | Phenotype |
|---------------------|---------------------|
| 00000000 ~ 11111111 | x좌표의 범위 (0 ~ 511) |
| 00000000 ~ 11111111 | y좌표의 범위 (0 ~ 511) |

Table. 2 Relation of the genotype and phenotype on the .

염색체의 길이는 부재의 개수에 한 부재가 가지는 문자열의 수(20)를 곱한 것이 된다.

예) 15개의 부재가 이루는 전체적인 염색체의 길이는 15×20으로 300이 된다.

2.4 Object Function 정의

한 세대에서 각각의 염색체에 대한 적응도 및 평가방법은 다음 세대를 생산하기 위해 중요한 요소로 사용된다. 본 연구에서는 적응도를 평가하는 함수를 강판에 대한 부재들이 이루는 면적으로 나타내었으며, 정식화 하면 다음과 같이 정의할 수 있다.

$$F_o = \left\{ (S_{row} - (S_{rect} - S_{piece})) + Reward - \frac{(S_{row} - S_{piece})}{Penalty} \right\} \times Life$$

There,

S_{row} = 강판의총면적(Height×Width).

S_{rect} = 배열된부재들이이루는최대사각형.

S_{piece} = 부재들의총면적($\sum_{i=0}^{P-1} A_i$). (P= 부재의갯수, A_i = 부재면적)

Reward = 면적이 큰 부재가 원점에 가까이 있을 때 주는 값.

Penalty = 부재끼리 겹치게 되면 주는 값.

Life = 염색체를 도태 시키기 위한 factor.

여기서, Reward는 면적이 큰 부재가 원점으로부터 가까이 있을 때 염색체가 다음 세대에 살아남을 확률을 높이기 위한 것으로, 다음과 같이 정식화 할 수 있다.

$$Reward = \sum_{i=0}^{P-1} (A_i \times (P-1))$$

There, P = 부재의 갯수.

A_i = 부재의 면적(큰 부재로 정렬된 배열).

Penalty는 부재들이 겹치는 것을 방지하게 위해 염색체의 적응도를 낮게 함으로써 염색체가 다음 세대에 살아남을 확률을 낮추기 위한 것으로, 다음과 같이 정식화 할 수 있다.

$$Penalty = \sum_{i=0}^{P-2} \left\{ \sum_{j=i+1}^{P-1} Overlap(i, j) ? value++ : continue \right\}$$

There, $Overlap(i, j)$ = 부재 i와 j의 겹침판단.

value = '0'으로 초기화된 정수.

위의 $Overlap(i, j)$ 는 부재 i와 j가 서로 겹치면 'value++'를 실행하고, 그렇지 않으면 계속해서 루프를 돈다. 최종적으로 Penalty의 값은 최소 0부터 최대 부재의 개수 만큼이 된다.

Life는 강판의 최대 가로, 세로를 벗어난 부재들을 도태시키기 위한 중요한 요소로써, 각 세대에서 생성된 각 부재들의 x, y정보들 중 하나라도 강판에서 벗어나게 되면 적응도를 '0'으로 초기화함으로써 다음 세대에서 살아남을 확률을 아주 적게 해줌으로써 부재들이 강판의 안쪽에 위치하며 진화하게 한다.

2.5 교배 및 돌연변이 정의

교배(crossover)는 보통 simple, multi, uniform으로 구분되는데, simple이나 multi point 교배는 부재들을 모델링한 염색체를 폭 넓게 교배시키지 못 한다는 단점이 있어, 본 연구에서는 uniform 형태의 교배연산을 채택하였다.

uniform 형태의 교배연산자는 mask라는 염색체가 필요한데, 이는 '1'과 '0'의 조합으로 이루어져 있으며 '1'이 나올 확률이 교배확률($P_{crossover}$)이 된다. 두 염색체간의 교배는 mask의 locus의 gene에 따라 이루어지는데, 만약 첫 번째 locus의 gene이 '1'이면 두 염색체의 첫 번째 locus의 gene을 서로 교환하는 형식으로 교배가 이루어진다. 물론 '0'이면 교배는 이루어지지 않는다.

돌연변이(mutation)는 한 세대의 염색체를 교배확률($P_{mutation}$)로 검색하면서 돌연변이 gene을 생성한다.

2.6 휴리스틱 방법 적용 시기

휴리스틱 방법(Bottom & Left sliding)의 적용 시기는 크게, 각 세대의 염색체들을 decoding하여 평가할 때 적용하는 방법과 GA의 루핑이 끝나고 최적의 결과값을 도출한 후에 적용하는 방법이 있다. 본 연구에서는 아래의 세 가지 경우를 고려하여 결과값을 비교하였다.

- ① 단순한 SGA(Simple Genetic Algorithm)만을 사용하는 방법. (이하 "CASE-1"로 정의)
- ② SGA를 수행 중 염색체를 평가하는 시기에 적용하는 방법. (이하 "CASE-2"로 정의)
- ③ SGA를 수행 후 얻은 결과 값에 적용하는 방법. (이하 "CASE-3"으로 정의)

3. 시뮬레이션 결과 및 고찰

3.1 유전자 알고리즘 주요 factor 결정

유전자 알고리즘에서 사용되는 중요한 factor들은 유전자 모델링에서 어느 정도 결정이 된다. 예를 들면 염색체의 수, 초기 염색체 생성에 사용되는 seed 등이 있다.

Generation은 보통 개체의 진화 횟수를 의미하기도 하는데, 이는 모델링된 객체의 특성에 따라 그 값이 유동적이다.

Pop Size는 개체군의 크기를 의미한다. 우성의 유전인자를 만들어낼 확률을 높이기 위해 대상 객체의 특성에 따라 유동적으로 변화시킨다.

Chrom Length는 문자열의 길이를 나타내며 본 연구에서는 부재의 개수에 따라 유동적으로 변화하게 작성하였다.

$P_{crossover}$ 는 염색체들 간에 교배할 확률을 나타낸다.

$P_{mutation}$ 는 염색체의 돌연변이 될 확률을 나타낸다. 보통 0.005 ~ 0.01 사이의 값들이 연산에 사용되지만, 시뮬레이션 결과 0.05 부근에서 가장 좋은 결과는 도출할 수 있었다.

Random Seed는 초기개체군을 생성할 때 염색체가 나타내는

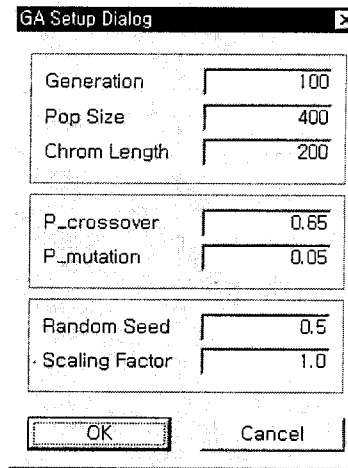


Fig. 3 GA setup dialog.

gene의 특성 중 '1'이 나올 확률이다. 초기개체군을 생성시킬 때 이 값이 0.5 이하의 값이면 부재들간의 겹침의 정도가 심하며, 0.5 이상의 값이면 부재가 강판의 위치하지 못하고 분산되어 도태되어버리는 개체가 많아지게 된다. 여러 차례의 시뮬레이션을 통해 알아본 결과 0.5부근에서 가장 좋은 값을 나타냄을 알 수 있었다.

Scaling Factor는 Object Function을 정의할 때 fitness value를 충분히 고

려하여 하였으므로, 결과 값에 큰 영향을 미치지 않는 것을 알 수 있었다. 그래서, 적응도에 영향을 미치지 않은 1을 적용시켰다.

본 연구를 통해 개발하게 된 프로그램의 단축아이콘들의 기능은 다음과 같다.

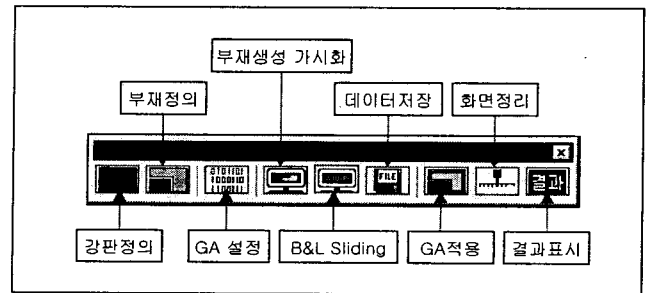


Fig. 4 Explain to a program icon.

3.2 모양과 면적이 동일한 경우

100×50의 직사각형 형태를 하고 있는 부재 10개를 강판에 배치했을 때 CASE-1, CASE-2, CASE-3에 대한 결과는 다음과 같다.

< CASE-1의 경우 >

[좌측] Gen = 0037, Pop = 258, Max = 200339

[우측] Gen = 0079, Pop = 342, Max = 201001

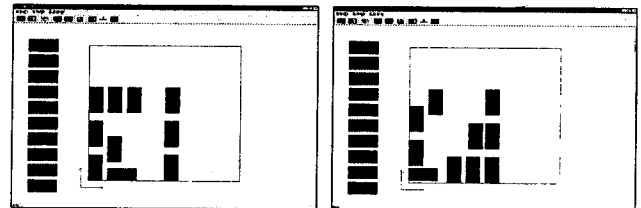


Fig. 5 Results of CASE-1 on the same shape

< CASE-2의 경우 >

[좌측] Gen = 0003, Pop = 009, Max = 260080
 [우측] Gen = 0003, Pop = 368, Max = 260080

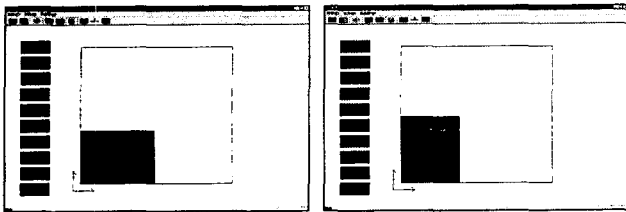


Fig. 6 Results of CASE-2 on the same shape

< CASE-3의 경우 >

[좌측] Gen = 0054, Pop = 399, Max = 216794
 [우측] Gen = 0087, Pop = 136, Max = 232047

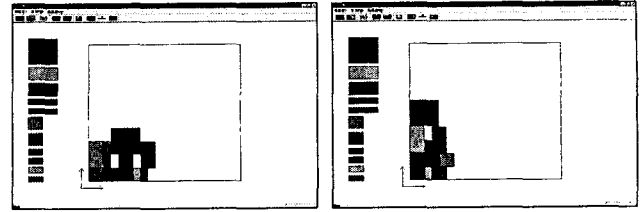


Fig. 10 Results of CASE-3 on the different shape

< CASE-3의 경우 >

[좌측] Gen = 0029, Pop = 056, Max = 212929
 [우측] Gen = 0044, Pop = 103, Max = 201356

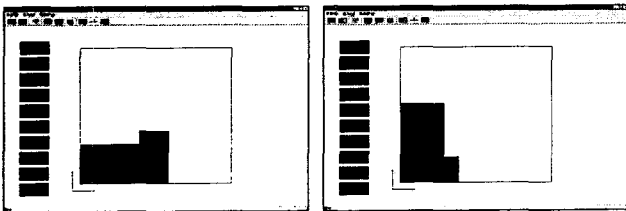


Fig. 7 Results of CASE-3 on the same shape

3.3 모양과 면적이 상이한 경우

(100×100) 1개, (100×50) 2개, (100×25) 2개, (50×50) 2개, (50×25) 4개의 직사각형 형태의 부재 11개를 강판에 배치했을 때 CASE-1, CASE-2, CASE-3에 대한 결과는 다음과 같다.

< CASE-1의 경우 >

[좌측] Gen = 0074, Pop = 358, Max = 224058
 [우측] Gen = 0079, Pop = 130, Max = 213942

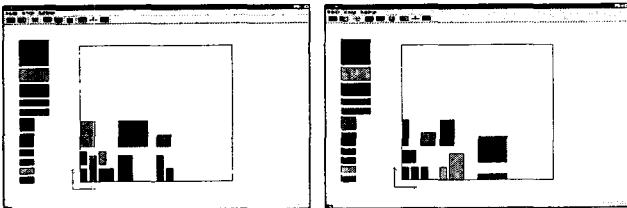


Fig. 8 Results of CASE-1 on the different shape

< CASE-2의 경우 >

[좌측] Gen = 0031, Pop = 087, Max = 260256
 [우측] Gen = 0034, Pop = 010, Max = 260256

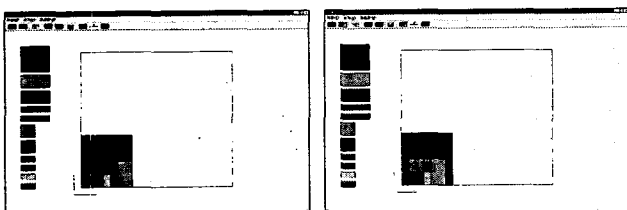


Fig. 9 Results of CASE-2 on the different shape

4. 결 론

유전자 알고리즘에 휴리스틱 방법을 접목하여 모양과 면적이 동일한 부재((100×50) 10개)와 상이한 부재((100×100) 1개, (100×50) 2개, (100×25) 2개, (50×50) 2개, (50×25) 4개)를 강판(511×511)에 자동 배치하는 시뮬레이션프로그램의 결과를 통해 다음과 같은 결론을 얻을 수 있었다.

(1) 휴리스틱 방법은 탐색체의 fitness를 결정하는 Object Function을 수행하기 전에 적용시킬 때 가장 좋은 결과 값을 도출해 낼 수 있다.

(2) 유전자 알고리즘에 휴리스틱 방법을 접목시킨 결과, SGA에 비해 최적배치에 수렴하는 시간(generation)이 짧게 걸림을 알 수 있다.

(3) 부재의 개수가 증가함에 따라 탐색체 보유정보가 많아져서 유전연산을 수행하는데 소요되는 시간이 길어짐을 알 수 있다.

(4) 부재의 위치정보의 크기는 강판의 크기(Height와 Width)에 따라 결정되므로, 강판의 클수록 탐색체에 매핑되는 부재정보의 길이가 길어짐을 알 수 있다. 이는 부재의 개수와 비례하여 커지게 되므로 유전연산에서 소요되는 시간이 많아지게 된다.

부재를 모델링 할 때 강판에서의 부재위치정보에 대한 탐색체로의 매핑은 차후 유전연산을 함에 있어 커다란 단점(소요시간)으로 지적될 것이므로 고려되어야 할 것이다.

참 고 문 헌

- 김여근, 윤복식, 이상복(1999), “메타 휴리스틱 유전알고리즘 · 시뮬레이티드 어닐링 · 타부서치”, 영지문화사, pp 3~174.
- 이현찬, 채수원, 최영(1996), “컴퓨터 그래픽스 및 형상 모델링”, 스그마프레스, pp 63~84.
- Goldberg David E(1989), “Genetic Algorithms in Search, Optimization, and Machine Learning”, ADDISON-WESLEY PUBLISHING COMPANY, pp 59~88.
- Jain, P., Fenyves, P. and Richter, R.(1992), “Optimal Blank Nesting Using Simulated Annealing”, General Motors Research Laboratories, 160 / Vol. 114
- Lance Chambers(1995), “GENETIC ALGORITHMS Applications Volume I”, CRC Press, pp 33~56.
- Pham, D. T. and Karaboga, D.(1998), “Intelligent Optimisation Technigues”, Springer, pp 51~141.