

트랜스포터의 공주행(空走行) 최소화를 고려한 블록 운반 계획 최적화

임선빈*, 노명일†**, 차주환*, 이규열***

서울대학교 조선해양공학과 대학원*/ 울산대학교 조선해양공학부**
서울대학교 조선해양공학과 및 해양시스템공학연구소***

Optimal Block Transportation Scheduling Considering the Minimization of
the Travel Distance without Overload of a Transporter

Sun-Bin Yim*, Myung-II Roh†**, Ju-Hwan Cha* and Kyu-Yeul Lee***

Department of the Naval Architecture and Ocean Engineering, Seoul National University*
School of Naval Architecture and Ocean Engineering, University of Ulsan†**

Department of the Naval Architecture and Ocean Engineering, and Research Institute of
Marine Systems Engineering, Seoul National University***

Abstract

A main issue about production management of shipyards is to efficiently manage the work in process and logistics. However, so far the management of a transporter for moving building blocks has not been efficiently performed. To solve the issues, optimal block transporting scheduling system is developed for minimizing of the travel distance without overload of a transporter. To implement the developed system, a hybrid optimization algorithm for an optimal block transportation scheduling is proposed by combining the genetic algorithm and the ant algorithm. Finally, to evaluate the applicability of the developed system, it is applied to a block transportation scheduling problem of shipyards. The result shows that the developed system can generate the optimal block transportation scheduling of a transporter which minimizes the travel distance without overload of the transporter.

※Keywords: Block transportation scheduling(블록 운반 계획), Transporter(트랜스포터), Block assignment(블록 할당), Path searching(경로 탐색), Genetic algorithm(유전 알고리즘), Ant algorithm(개미 알고리즘)

접수일: 2007년 7월 31일, 승인일: 2008년 11월 7일

† 교신저자: miroh@ulsan.ac.kr, 052-259-2165

1. 서론

1.1 연구 배경 및 필요성

선박은 계획된 일정에 따라 수백 개의 블록들(재화 중량 300,000 톤 유조선의 경우 보통 120 개의 블록으로 구성)이 합쳐져 한 척의 선박으로 건조된다. 또한, 하나의 블록은 강판으로부터 시작하여 여러 공정을 거쳐 수직에서 수백 톤에 달하는 블록으로 완성하게 된다. 즉, 한 척의 선박을 건조하기 위해서는 인도 일(delivery date)에 맞춰 계획된 일정에 따라 모든 작업들이 정확하게 이루어져야 한다. 이러한 생산 공정들 사이의 중요한 블록 재공품(work in process, 이하 간단히 블록이라 칭함)의 운반을 담당하는 것이 바로 트랜스포터(transporter)이다. 따라서 공정간의 효율적인 블록 운반을 위해서는 트랜스포터의 운영 관리가 무엇보다도 중요하다. 비효율적인 트랜스포터의 운영으로 인한 물류 비용의 증가와 물류 흐름의 저하는 전체적인 생산 공정의 지연을 초래 할 수 있기 때문이다. 지금까지 조선소 내의 트랜스포터를 체계적으로 관리하기가 어려웠다. 블록 운반을 위해서는 먼저 해당 블록의 작업을 담당하는 부서에서 블록 운반 작업을 신청하게 된다. 이때 각 부서가 자신의 일정에 따라 특정 트랜스포터를 서로 필요한 시각에 사용하려고 하는 문제가 발생하게 된다. 이를 해결하기 위해 트랜스포터의 운영 담당자는 각 블록의 공정 단계, 우선 순위 및 자신의 경험에 의하여 각 블록의 운반 순서, 운반 시각, 해당 블록을 운반할 트랜스포터를 적절히 할당하게 된다. 따라서 하루에 수백 건의 블록들이 운반되는 환경에서(국내 모 대형 조선소의 경우 하루 300 개 이상의 블록들을 운반함) 트랜스포터 운영 담당자가 효율적인 트랜스포터의 운영을 수작업으로 계획하는 데에는 어려움이 있게 된다. 이러한 트랜스포터의 블록 운반 계획은 일일 계획으로써 보통 하루 전날에 계획된다. 따라서 이러한 문제를 해결하기 위해 본 연구에서는 트랜스포터의 최적 블록 운반 계획 시스템을 개발하였다. 개발된 시스템은 먼저 해당 블록을 운반하기 위한 트랜스포터의 최적 운반 경로를 결정한다. 그 후

개미 알고리즘과 유전 알고리즘을 혼합한 하이브리드 알고리즘을 적용, 각 블록에 대해 공주행 거리를 최소로 하는 블록 할당 작업을 수행한다. 끝으로 개발된 시스템의 효용성을 검증하기 위해 실제 조선소의 블록 운반 계획 문제에 적용해 보았다.

1.2 관련 연구 현황

조선소내의 효율적인 트랜스포터 운영을 통해 생산성 향상과 생산비용을 절감 할 수 있음에도 불구하고 지금까지 관련 연구는 그리 많지 않았다. 최근에 Joo et al.(2005)은 조선소에서의 블록 운반을 위한 복합 트랜스포터의 일정 계획에 관한 연구를 수행하였다. 트랜스포터의 운반 시간을 최소화 하기 위하여 2 단계의 알고리즘을 제안하였다. 이 연구는 먼저, 비슷한 중량의 블록들을 그룹화한 뒤 각 그룹을 한 종류의 트랜스포터를 이용한 블록 운반 문제로 축소하였다. 그 후 1 단계로 Maxwell and Muckstadt(1982), Koo et al.(2004)이 제시한 최적화 모형을 이용하여 트랜스포터의 최소 사용 대수를 구하였다. 2 단계에서는 각 트랜스포터에 최단 거리에 있는 블록을 할당하여 운반시키는 휴리스틱 알고리즘을 적용하여 각 트랜스포터별로 블록을 할당 시키고 운반 순서를 결정하였다. 그리고 계획 기간 내 블록 운반 작업을 완료하지 못하면 작업할 트랜스포터를 추가시켜 블록 운반 작업을 끝마치는 계획을 세웠다. 그러나 앞서 언급하였듯이, 이 연구는 비슷한 중량을 가진 블록들을 그룹화하여 각 그룹을 한 종류의 트랜스포터로 운반하는 문제로 축소하였다. 따라서 적재 능력이 다양한 복수의 트랜스포터를 이용하여 각각의 중량을 갖는 블록들을 운반시켜야 하는 현실적인 문제에 적용하기에는 한계가 있다. 또한 각 블록을 현재 사용 가능한 트랜스포터 중 가장 가까이 있는 것에 할당시키는 것은 적절치 않을 수도 있다. 즉, 운반할 블록의 선후 관계 및 타 트랜스포터와의 상관 관계 등을 고려하지 않고 단순히 지금 상황의 최적해만을 찾는다면 전체적인 최적해를 얻기 어렵다.

한편, Yu(2005)는 트랜스포터를 이용한 블록 운

반 시 최적 경로를 찾는 연구를 수행하였다. 이 연구에서는 백 트래킹 알고리즘(back tracking algorithm)을 이용하여 블록 운반을 위한 트랜스포터의 운반 거리 및 경로에 따른 운반 시간을 최소화 하는 최적 경로를 결정하였다. 하지만 이 연구는 블록의 최적 운반 순서 결정은 다루지 않았다.

2 개미 알고리즘(Ant algorithm)과 유전 알고리즘(Genetic algorithm)

2.1 개미 알고리즘

Dorigo et al.(1991, 1996)은 개미들이 먹이를 찾을 때 처음에는 임의의 경로로 이동을 하다가 점차 최적 경로로 이동을 하여 먹이를 찾는 모습을 보고 개미 알고리즘을 개발하였다. 개미들은 이동을 할 때 자신이 지나간 경로에 페로몬(pheromone)이라는 화학 물질을 남긴다. 이 페로몬을 통하여 뒤에 오는 개미에게 먹이가 있는 곳을 알려 주게 된다. 뒤에 따라 오는 개미는 이 페로몬의 영향을 받아 이동을 하게 되며, 시간이 경과 함에 따라 개미가 이동한 경로에는 점차적으로 많은 페로몬이 누적되어, 더 많은 개미들이 그 경로로 이동하게 된다. 결국 이 경로가 먹이를 찾기 위한 최적 경로가 된다.

Fig. 1 은 개미를 통해 최적 경로를 찾는 예를 나타낸다. 매 단위 시간마다 A와 E에서 각각 30마리의 개미가 출발한다고 가정한다. 또한 개미는 일정 속력으로 1 시간마다 1만큼의 거리를 이동한다고 가정한다.

시간 $t = 1$ 이 되면 A와 E에서 각각 출발한 개미는 1만큼 이동하여 각각 B와 D에 도착한다. B와 D의 갈림길에 도착한 개미는 같은 확률을 가지고 양방향으로 이동하게 되며 그 결과 한 방향으로 15마리씩의 개미들이 이동한다. 시간 $t = 2$ 가 되면 B-C-D 경로로 이동한 개미들은 각각 B에서 D로, D에서 B로 도착한다. B-F-D 경로로 이동한 개미들은 1만큼 이동하여 F에 도착한다. 또한 A와 E에서 각각 출발한 30마리 개미는 각각 B와 D에 도착하여 어느 방향으로 이동할지를 선택하게 된다. 이때 B-C-D 경로의 양방향에는

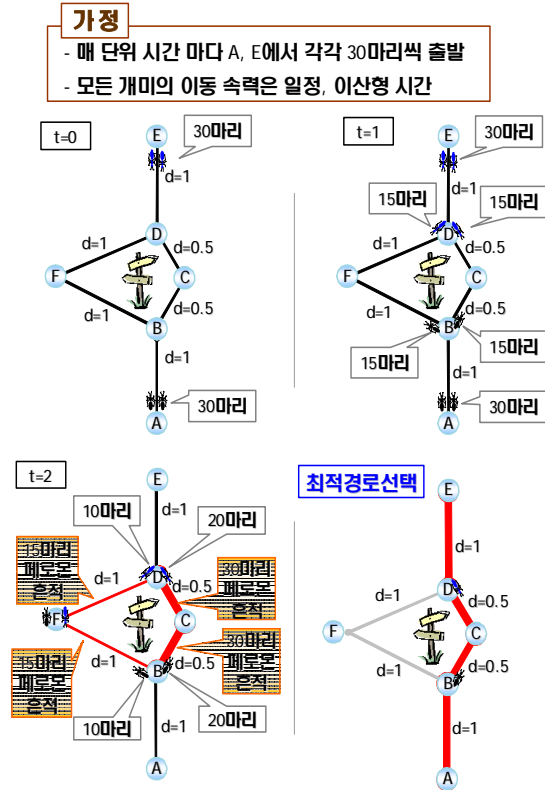


Fig. 1 An example of the moving pattern of artificial ants

이전에 각각 15 마리씩 개미가 이동했기에 총 30마리의 개미들에 의한 페로몬의 흔적이 남아 있다. 하지만 B-F-D 경로는 15마리의 개미들의 흔적만 남아 있다. 따라서 B와 D에 도착한 개미들은 확률적으로 B-F-D 경로보다 B-C-D 경로로 더 많이 이동하게 된다. 시간이 흐르면 흐를수록 페로몬의 양이 상대적으로 많은 B-C-D 경로로 개미들은 더 많이 이동하게 되어 A-B-C-D-E 경로가 A와 E 간의 최적 경로가 된다.

한편, 개미 알고리즘에서 개미는 기억력을 가지고 있어 한번 방문한 곳은 다시 방문하지 않는다고 가정한다. 또한 약간의 시력을 가지고 있어 거리가 가까운 곳으로 이동하려는 경향이 있다고 가정한다. 본 연구에서는 이러한 개미의 행동 패턴을 이용, 개미 알고리즘으로부터 얻어진 최적해의 집합을 유전 알고리즘의 초기해 집합(population)

으로 사용한다.

2.2 유전 알고리즘

유전 알고리즘은 생물의 진화 메커니즘을 이용하여 다양한 문제를 해결하고자 하는 방법이다. 적자 생존과 자연 도태의 진화 원리를 도입하여 초기해로부터 선택(selection) 연산, 교차(crossover) 연산, 돌연 변이(mutation) 연산과 같은 일련의 과정을 반복적으로 거쳐 최적해를 구하는 것이다. 선택 연산은 교차 연산을 위한 두 개의 부모해를 선택하는 과정이고, 교차 연산은 두 부모해를 조합하여 자식해를 생성하는 과정이며, 돌연 변이 연산은 교차 연산으로부터 얻은 자식해를 국부적으로 변형시켜 다양성(diversity)을 가진 자식해를 생성하는 과정이다(Moon 2001). 이와 같은 과정을 반복하면 전역 최적해(global optimum)를 찾을 수 있다.

3. 최적 블록 운반 계획 알고리즘

3.1 개요

본 연구에서 제안한, 트랜스포터의 공주행 거리를 고려한 최적 블록 운반 계획 알고리즘은 Fig. 2와 같다.

첫 번째 단계(Fig. 2의 (a))에서는, 각 지점간 최단 거리를 구하기 위해 Dijkstra 알고리즘을 사용하여 각 블록의 출발지에서부터 도착지까지의 최단 운반 거리 및 경로, 각 블록의 도착지에서 다른 모든 블록들까지의 최단 공주행 거리 및 경로를 구한다(Foulds 1992).

두 번째 단계(Fig. 2의 (b))에서는, 앞서 구한 블록간 최단 운반 거리 및 경로와 최단 공주행 거리, 경로 등의 정보를 기반으로 어떤 트랜스포터로 어느 블록을 운반하는 것이 효율적인지를 개미 알고리즘을 이용하여 결정하게 된다. 즉, 개미 알고리즘을 수행하게 되면 트랜스포터의 블록 운반 경로에 대한 페로몬 값을 얻을 수 있다. 페로몬의 양이 많다는 것은 그만큼 개미들이 해당 경로로 많이 이동했다는 것을 의미하고, 결국 그 경로가 블록을 운반하기 위한 트랜스포터의 최적 경로가

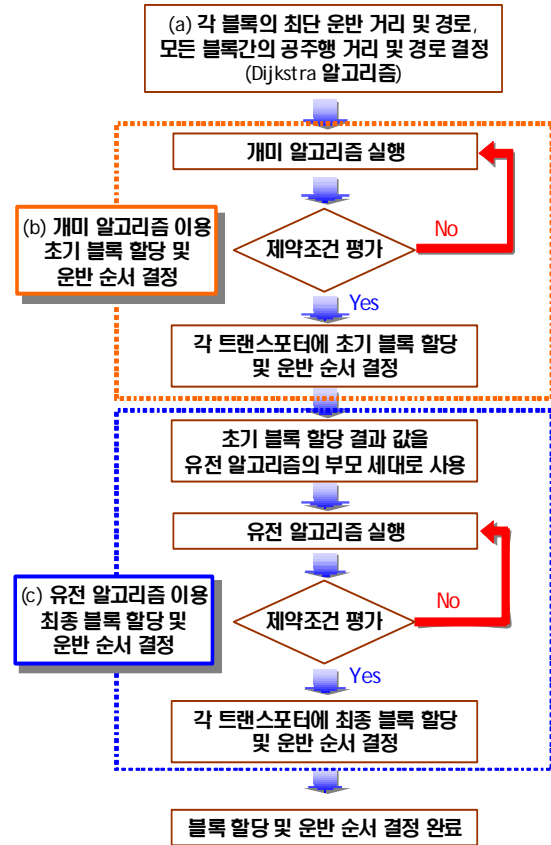


Fig. 2 Algorithm for optimal block transportation scheduling proposed in this study

된다. 따라서 페로몬의 양과 제약 조건을 평가하여 각 트랜스포터가 운반해야 할 블록들과 이들의 운반 순서를 결정하게 된다. 마지막 단계(Fig. 2의 (c))에서는 개미 알고리즘을 통해 얻은 블록 운반 계획을 초기해 집합으로 가정하고 유전 알고리즘을 수행한다. 그 결과 각 트랜스포터별로 운반해야 할 블록들과 이들의 운반 순서를 얻을 수 있다.

3.2 최적 블록 운반 계획 문제의 정식화

최적 블록 운반 계획 문제는 블록 할당 및 운반 순서를 결정하는 문제로서 다음과 같은 최적화 문제로 정식화될 수 있다.

Minimize

$$F = \alpha \sum_{h=0}^B \sum_{i=1}^B \sum_{k=1}^T x_{hi}^k (e_{hi}^k / V^k) + \beta \sum_{i=1}^B \sum_{k=1}^T x_i^k (d_i^k - s_i) \quad (1)$$

Subject to

$$w_i \leq c_k \quad (2)$$

$$r_i \leq p_i^k \quad (3)$$

$$d_i^k \leq s_i \quad (4)$$

$$p_j \leq p_i \quad (5)$$

$i, j = 1, \dots, B$ (단, $i \neq j$)

$h = O$ (각 트랜스포터의 최초 블록 운반을 위한 가장 블록), $1, 2, \dots, B$ (단, $h \neq i$)

$k = 1, \dots, T$

여기서,

e_{hi}^k : 트랜스포터 k 가 블록 h 를 운반 후, 블록 i 의 운반에 소요되는 주행거리 (공주행 거리 + 블록 운반 거리)

V^k : 트랜스포터 k 의 속력

d_i^k : 트랜스포터 k 가 블록 i 를 운반 완료한 시각

s_i : 블록 i 의 계획된 운반 완료 시각 (일정계획상 운반이 완료되어야 하는 시각)

$x_{hi}^k = 1$: 트랜스포터 k 가 블록 h 를 운반 후 블록 i 를 운반할 때

$x_{hi}^k = 0$: 트랜스포터 k 가 블록 h 를 운반 후 블록 i 를 운반하지 않을 때

$$\left(\sum_{k=1}^T x_{hi}^k = 1, \sum_{h=0}^B x_{hi}^k = 1, \sum_{i=1}^B x_{hi}^k = 1 \right)$$

$x_i^k = 1$: 트랜스포터 k 가 블록 i 를 운반할 때

$x_i^k = 0$: 트랜스포터 k 가 블록 i 를 운반하지 않을 때

α, β : 가중치(weight factor)

B : 운반해야 할 블록의 총 수

T : 사용 가능한 트랜스포터의 총 수

w_i : 블록 i 의 중량

c_k : 트랜스포터 k 가 운반할 수 있는 블록의 최대 중량

r_i : 블록 i 의 계획된 운반 가능 시각

(블록이 운반을 위해 준비되는 시각)

p_i^k : 트랜스포터 k 가 블록 i 를 운반하기 시작하는 시각

p_i : 우선 순위가 높은 블록 i 의 운반 시작 시각

p_j : 우선 순위가 낮은 블록 j 의 운반 시작 시각

식 (1)에 나타나 있듯이 이 문제의 목적 함수는 트랜스포터의 공주행 시간을 최소화 하는 동시에 블록들을 가능한 한 계획된 운반 완료 시각보다 일찍 운반하도록 하는 것이다. 공주행 시간을 최소화 함으로써 트랜스포터의 생산성을 최대한 증대시킬 수 있다. 또한 블록들을 가능한 한 계획된 운반 완료 시각보다 일찍 운반함으로써 공정간의 물류 흐름을 향상시킬 수 있다. 즉, 블록 적치장에 적치되어 있는 블록의 경우, 보다 일찍 후속 블록을 적치할 수 있도록 하며, 공장에서 공정이 완료된 블록의 경우, 보다 일찍 후속 블록을 작업할 수 있도록 한다. 한편, 이 문제의 제약 조건인 식 (2)는 운반할 블록의 중량이 트랜스포터의 적재 능력보다 작거나 같아야 함을 나타내고, 식 (3)은 트랜스포터가 운반할 블록을 계획 운반 가능 시각 이후에 운반해야 함을 나타낸다. 식 (4)는 트랜스포터가 운반하는 블록의 계획 운반 완료 시각 내에 운반을 완료해야 함을 나타내고, 식 (5)는 트랜스포터가 우선 순위에 맞춰 블록들을 운반해야 함을 나타낸다.

위와 같은 최적 블록 운반 계획 문제를 해결하기 위해, 본 연구에서 제안한 방법을 보다 상세히 설명하면 다음과 같다.

3.3 각 블록의 최단 운반 거리 및 경로, 모든 블록간의 최단 공주행 거리 및 경로 결정

본 연구에서는 블록의 최단 운반 거리 및 경로를 결정하기 위해, 먼저 Fig. 3 과 같이 블록의 모든 운반 가능 경로들을 그래프(graph)화 하였다. 그래프는 정점(vertex)과 정점들을 연결한 간선(edge)으로 구성된다. 생성된 그래프에 대해 Dijkstra 알고리즘을 적용하면, 임의의 정점에서

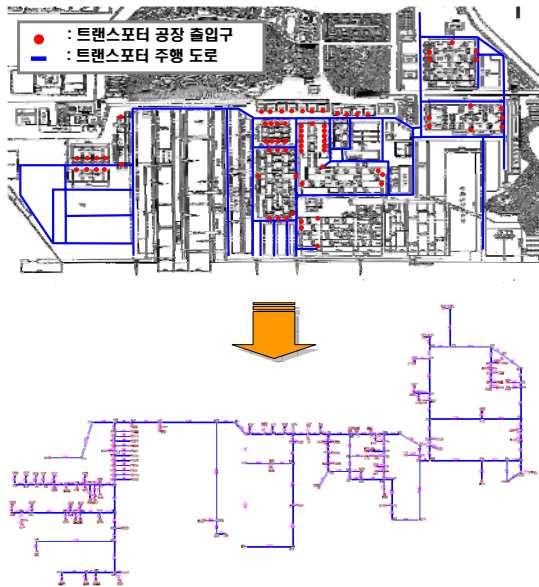


Fig. 3 Example of a graph for the block transportation in a shipyard

이동 가능한 모든 정점까지의 최단 거리 및 경로를 찾을 수 있다(Foulds 1992). 즉, 각 블록의 출발지로부터 도착지까지의 최단 거리와 경로를 찾을 수 있고, 각 블록의 도착지로부터 모든 블록의 출발지까지의 최단 공주행 거리 및 경로를 찾을 수 있다.

3.4 개미 알고리즘을 이용한 트랜스포터의 초기 블록 운반 순서 결정

앞서 구한 블록간 최단 운반 거리 및 경로를 기반으로 개미 알고리즘을 이용하여 트랜스포터의 초기 블록 운반 순서를 결정한다. 즉, 본 연구에서는 트랜스포터의 블록 운반 순서 결정을 위해, 개미 알고리즘에서의 경로 선택 방법으로 (1) STR(State Transition Rule)을 이용하였고, (2) 이동 거리에 반비례하는 페로몬의 분비 방법, (3) 페로몬의 증발과 갱신 방법으로 GPUR(Global Pheromone Update Rule)을 이용하였다(Dorigo et al. 1991, 1996).

3.4.1 STR(State Transition Rule)

본 연구에서 이용한 STR 은 다음과 같이 표현할 수 있다. STR 에 따르면 경로 탐색 시 개미들은 확률적으로 페로몬의 양이 많은 경로를 선호한다.

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta}, & \text{if } j \in J_i^k \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

여기서,

$P_{ij}^k(t)$: t 번째 반복에서 개미 k 가 vertex i 에서 vertex j 로 이동할 확률

τ_{ij} : vertex(i,j)간 페로몬 강도

η_{ij} : 개미의 시인성이며 vertex i 에서 vertex j 간의 거리(D_{ij})에 반비례($\eta_{ij} = 1/D_{ij}$)

J_i^k : vertex i 에 있는 k 번째 개미가 앞으로 방문할 vertex 들(이미 방문한 vertex 들은 제외)

α : 페로몬 흔적강도 관계를 결정하는 파라미터 ($\alpha > 0$)

β : 시인성($1/D_{ij}$)에 대한 관계를 결정하는 파라미터 ($\beta > 0$)

3.4.2 페로몬의 분비 방법

한편, 페로몬의 분비 방법에 대해서는 다음과 같이 이동 거리에 반비례 하는 방법을 이용하였다. 즉, t 번째 반복 수행에서 개미 k 가 vertex (i, j) 간에 페로몬을 분비하는 양(거리에 반비례)은 식(7)과 같다. vertex (i, j)간이라는 것은 블록 운반 순서 결정에 있어서 트랜스포터가 한 블록의 운반 완료 지점(도착지) i 에서 다음 운반을 위해 다른 블록의 운반 시작 지점(출발지) j 로 공주행하는 경로를 뜻한다.

$$\Delta \tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k}, & \text{if } (i, j) \in T^k(t) \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

여기서,

$T^k(t)$: t 번째 반복 수행에서 개미 k 에 의해 수행된 여행

L_k : t 번째 반복 수행에서 개미 k 에 의한 수행된 총 여행거리

Q : 상수(페로몬 양)

3.4.3 GPUR(Global Pheromone Update Rule)

마지막으로 페로몬의 증발과 갱신 방법으로 본 연구에서는 다음과 같은 GPUR 을 이용하였다. 즉, GPUR 에 따라 일정량의 페로몬을 증발시키고 페로몬을 갱신하여 개선된 경로 탐색을 한다.

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k \quad (8)$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t)$$

여기서,

ρ : 페로몬 증발 계수($0 \leq \rho < 1$)

m : 개미의 수

이상과 같이 STR, 페로몬의 분비 방법, GPUR 로 구성되는 개미 알고리즘을 수행하게 되면 트랜스포터의 블록 운반 경로에 대한 페로몬 값을 얻을 수 있다. 페로몬의 양이 많다는 것은 그만큼 개미들이 해당 경로로 많이 이동했다는 것을 의미하고, 결국 그 경로가 블록을 운반하기 위한 트랜스포터의 최적 경로 즉, 트랜스포터의 공주행 거리를 최소화 하는 경로가 된다.

따라서 각 트랜스포터별 블록 할당 방법은, 각 트랜스포터의 위치에서 페로몬 양이 많은 경로에 있는 블록부터 차례로 해당 트랜스포터에 할당하는 것이다. 이때 목적 함수(식 (1))와 제약 조건(식 (2)~(5))을 만족하는 블록들을 대상으로 순차적으로 각 트랜스포터에 할당한다.

3.5 유전 알고리즘을 이용한 트랜스포터의 최종 블록 운반 순서 결정

개미 알고리즘을 통해 트랜스포터의 초기 블록 운반 순서의 집합을 구한 후에는, Fig. 4 와 같이 이를 초기해 집합으로 가정하고 유전 알고리즘을 수행하게 된다.

먼저 블록에 번호를 붙여 운반 순서대로 염색체(chromosome)로 표현한다. 이때 염색체는 사용할 트랜스포터의 수만큼 segment 로 나뉘어진다.

Fig. 4 에서는 두 대의 트랜스포터를 사용한다고 가정하였고, 따라서 염색체는 두 개의 segment 로 구성됨을 알 수 있다. Segment1 은 트랜스포터 1

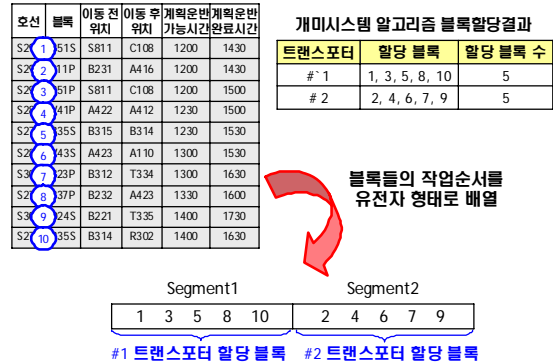


Fig. 4 Chromosome for block transportation scheduling proposed in this study

에 할당할 블록을 운반 순서대로 배열한 것이고, segment2 는 트랜스포터 2 에 할당할 블록을 운반 순서대로 배열한 것이다. 유전 알고리즘의 교차 연산, 돌연변이 연산 등을 통해 각 트랜스포터에 할당되는 블록의 수가 변하며 각 트랜스포터에 효율적인 블록 운반 순서가 결정된다.

3.5.1 선택(selection) 연산

교차 연산을 수행하기 위해서는 초기해 집합으로부터 두 개의 부모해를 선택해야 한다. 본 연구에서는 비례 선택(proportionate selection) 방법을 선택 연산으로 이용하였다. 이 방법은 적합성 값(목적 함수를 최소화 하는 문제의 경우, 목적 함수 값에 반비례)이 높은 즉, 목적 함수 값이 낮은 염색체를 부모해로서 더 높은 확률로 선택하는 방법이다.

3.5.2 교차(crossover) 연산

교차 연산은 선택 연산으로 선택된 두 개의 부모해로부터 새로운 두 개의 자식해를 생성하는 연산이다. 본 연구에서는 블록 운반 순서 결정을 위해 Fig. 5 와 같은 교차 연산을 제안하였다. 선택 연산을 통해 얻은 두 부모해 P₁, P₂ 의 상대적인 적합성 값이 각각 60%와 40%라고 가정하면 부모해 P₁ 으로부터는 염색체 중 60%의 유전 인자가 그리고 부모해 P₂ 로부터는 염색체 중 40%의 유전 인자가 자식해로 전달되는 것이다.

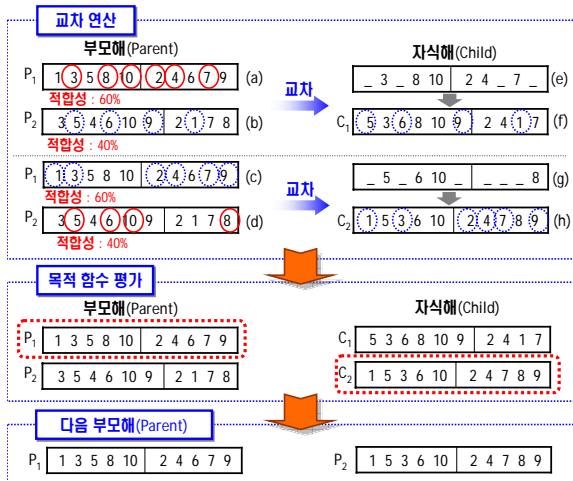


Fig. 5 Crossover operation proposed in this study

교차 연산은 먼저 부모해 P₁(Fig. 5 의 (a))에서 적합성(60%)에 따라 임의(random)로 유전 인자 10 개중 6 개를 선택한다. 선택된 6 개의 유전 인자(3, 8, 10, 2, 4, 7)는 자신의 위치와 순서를 그대로 자식해 C₁(Fig. 5 의 (e))에게 물려준다. 부모해 P₂(Fig. 5 의 (b))에서는 부모해 P₁(Fig. 5 의 (a))에서 선택하지 않은 4 개의 유전 인자(5, 6, 9, 1)를 선택하여 순서대로 자식해 C₁(Fig. 5 의 (f))에게 물려준다. 이때 비어있는 각 segment 의 해당 공간에 유전 인자들을 순서대로 하나씩 넣어준다. 즉, 부모해 P₂(Fig. 5 의 (b))에서 물려 받을 유전 인자는 segment1 에 ‘5, 6, 9’ 그리고 segment2 에 ‘1’ 이다. 자식해 C₁(Fig. 5 의 (e))의 segment1 에는 부모해 P₁(Fig. 5 의 (a))으로부터 유전 인자를 물려 받고 비어있는 공간이 2 자리이다. 이곳에 부모해 P₂(Fig. 5 의 (b))로부터 물려받을 유전 인자 ‘5, 6, 9’ 를 차례로 넣게 되면 ‘9’ 가 남게 된다. 남은 유전 인자는 해당 segment 의 마지막에 넣어 준다. 그리고 만약, 비어있는 공간은 있지만 넣어줄 유전 인자가 없다면 그 공간은 삭제한다. 따라서 자식해 C₁ 의 segment2 에 부모해 P₂로부터 물려 받은 ‘1’ 을 넣게 되면 ‘2, 4, 1, 7’ 이 되며 ‘7’ 다음에 남은 공간에 넣어줄 유전 인자가 없으므로 남은 공간은 삭제 된다(Fig. 5 의 (f)).

자식해 C₂(Fig. 5 의 (h))를 생성하는 과정도 위와 동일하다. 먼저 부모해 P₂(Fig. 5 의 (d))에서 적합성(40%)에 따라 임의로 4 개의 유전 인자(5, 6, 10, 8)를 선택하여 자식해 C₂(Fig. 5 의 (g))에게 자신의 위치와 순서를 그대로 물려주게 된다. 그리고 부모해 P₂(Fig. 5 의 (d))에서 선택하지 않은 부모해 P₁(Fig. 5 의 (c))에 있는 6 개의 인자들(1, 3, 2, 4, 7, 9)을 자식해 C₂(Fig. 5 의 (h))에 비어있는 공간에 차례로 넣어준다. 이때도 마찬가지로 물려 받을 해당 segment 에 유전 인자를 넣을 공간이 없을 경우 그 segment 의 맨 마지막에 넣어 주며, 빈 공간은 있지만 채워줄 유전 인자가 없으면 그 공간은 삭제된다. 일련의 과정을 거쳐 자식해가 만들어지면, 부모해와 자식해 4 개(P₁, P₂, C₁, C₂)에 대해 목적 함수 값을 평가하게 된다. 그 중 가장 목적 함수 값이 좋은 두 해(예: P₁, C₂)를 선택하여 다음 부모해로 만들어 또 다른 자식해를 생성하게 된다.

3.5.3 돌연 변이(mutation) 연산

돌연 변이 연산은 교차 연산으로부터 얻은 자식해를 국부적으로 변형시켜 다양성을 가진 자식해를 생성하는 과정이다. 본 연구에서는 블록 운반 순서 결정을 위해 Fig. 6 과 같은 돌연 변이 연산을 제안하였다. 이 그림에 나타나 있듯이, 교차 연산을 통해 생성된 자식해 중에서 하나를 선택한 후, 선택된 자식해의 염색체 중 유전 인자 하나를 임의로 선택하여 다른 임의의 위치에 넣어준다. Fig. 6 은 자식해 C₂ 를 선택한 후 segment 1 의

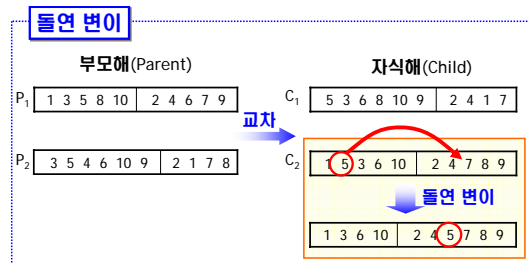


Fig. 6 Mutation operation proposed in this study

예를 나타낸다. 이와 같은 과정을 통해 새로운 자식해 C_2 를 생성하게 된다.

이상과 같은 일련의 선택 연산, 교차 연산, 돌연 변이 연산을 반복하게 되면 트랜스포터의 최종 블록 운반 순서를 결정할 수 있다.

4. 개발된 블록 운반 계획 시스템의 적용 예

본 연구에서 제안된 최적 블록 운반 계획 알고리즘의 효용성을 검증하기 위하여 제안된 알고리즘을 기반으로 한 블록 운반 계획 시스템을 개발하였고, 이를 실제 조선소의 블록 운반 계획 문제에 적용해 보았다. 본 장에서는 그 적용 결과를 소개한다.

4.1 트랜스포터의 블록 운반 계획 문제

Table 1 은 국내 모 대형 조선소의 특정 일에 오전 8시부터 24 시(자정)까지 4 대의 트랜스포터를 이용하여 총 97 개의 블록을 운반시킨 작업 데이터를 나타낸다. 이 데이터를 이용하여 본 연구에서 개발한 시스템의 효용성을 검증해 보았다. 이 표에 나타나 있듯이 블록의 운반 전·후의 위치, 블록의 계획 운반 가능 시각(블록이 운반 가능한 가장 빠른 시각), 계획 운반 완료 시각(블록이 운반 완료되어야 하는 가장 늦은 시각), 트랜스포터 이동 속력(공주행, 적재 주행), 블록을 트랜스포터에 싣고(loading) 내리는(unloading) 시간, 블록 운반 우선 순위(탑재 블록 순서, 부서 요청 등)가 주어져 있다.

4.2 최적화 수행 결과

Table 1 과 같이 주어진 문제에 대해 초기해 집

Table 1 Example of an operation data of transporters in a shipyard

트랜스포터	최대적재능력	운영시간	운반블록
#1	300	08:00~17:40	11
#2	500	08:00~21:20	34
#3	500	08:00~22:10	24
#4	550	08:00~24:00	28

합의 크기(population size)를 50, 최대 반복 수(generation number)를 10,000으로 하여 본 연구에서 제안한 알고리즘을 수행해 보았고, 그 결과를 Table 2 에 정리하였다. 블록운반시간은 블록을 트랜스포터에 싣는 작업을 시작하는 시각부터 블록을 운반하여 블록을 적치장에 내리는 작업을 완료하는 시각까지를 고려하였다. 그리고 그 이외의 트랜스포터가 운행되는 시간 즉, 블록을 적재하지 않은 상태로 운행되는 시간을 공주행 시간으로 하여 수행 하였다. 그러나 트랜스포터가 블록운반 또는 공주행시 운행에 장애를 주는 돌발적인 상황 같은 변수에 대해서는 고려하지 않았다. 이 표에 나타나 있듯이 조선소에서 실제로 작업했을 때 소요된 트랜스포터의 총 공주행 시간(Table 2 의 1 행)은 6 시간 10 분이였다. 반면 본 연구에서 제안한 알고리즘을 이용하여 최적해를 얻은 결과 트랜스포터의 총 공주행 시간(Table 2 의 2 행)이 4 시간으로서 약 35%의 공주행 시간 단축 효과를 얻을 수 있었다.

앞서 언급하였듯이 본 연구에서 제안한 블록 운반 계획 알고리즘은 개미 알고리즘을 이용, 초기 블록 운반 순서의 집합을 결정하고 유전 알고리즘을 이용, 최종 블록 운반 순서를 결정하는 혼합형(hybrid) 알고리즘이다. 따라서 제안된 알고리즘의 성능을 비교하기 위해, 주어진 문제를 유전 알고리즘만을 이용해 수행한 결과(Table 2 의 3 행)와도 비교하였다. 그 결과 유전 알고리즘만을 이용한 경우에 비해 본 연구에서 제안한 알고리즘이 보다 나은 결과를 산출함을 알 수 있다.

Table 2 Optimization result for the optimal block transportation problem

	적용	총 공주행 시간	CPU (sec)
1	조선소의 실제 작업 데이터	6 : 10	-
2	제안된 혼합형 알고리즘을 이용한 경우	4 : 00	350
3	유전 알고리즘을 이용한 경우	5 : 42	595

5. 결론 및 향후 연구 계획

본 연구에서는 트랜스포터의 생산성 향상을 위한 최적 블록 운반 계획 알고리즘을 제안하고 이를 기반으로 한 시스템을 개발하였다. 트랜스포터에 의한 각 블록의 최단 운반 거리 및 경로, 모든 블록간의 공주행 거리 및 경로를 결정하기 위해 그래프 이론의 Dijkstra 알고리즘을 이용하였고, 최적의 블록 할당 및 운반순서를 결정하기 위해 개미 알고리즘과 유전 알고리즘을 혼합한 혼합형 알고리즘을 제안하였다. 본 연구에서 제안한 알고리즘의 효용성을 검증하기 위해, 이를 실제 조선소의 블록 운반 계획 문제에 적용해 보았다. 그 결과 본 연구에서 개발한 알고리즘이 조선소 업무에 효과적으로 적용 가능함을 확인할 수 있었다.

향후에는 최적화 결과로부터 얻어진 운반 계획에 따라 트랜스포터들을 운용할 때 트랜스포터간의 충돌 문제가 발생하는지 등의 문제를 사전에 파악하고, 만약 충돌이 발생할 때 새로운 경로를 제시해 주는 기능을 추가할 예정이다. 또한 개발된 시스템의 완성도를 높이기 위해 사용자 인터페이스(GUI)를 보완할 것이다.

후 기

본 연구는

(a) 울산대학교 조선해양공학부 세계일류화사업단(SOTOP)

(b) 서울대학교 해양시스템공학연구소 및 BK 21 해양기술인력양성사업단

(c) 국방과학연구소 수중운동체기술특화센터 SM-11 과제 “수중 운동체의 체계/부체계 기능 및 성능 시뮬레이션을 위한 네트워크 기반의 가상(Virtual) 복합 시스템 모델 구조(Architecture) 연구”

의 지원과 2008년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음(KRF-2008-314-D00494)을 밝히며, 이에 감사 드립니다.

참 고 문 헌

- Dorigo, M., Maniezzo, V. and Colorni, A., 1991, Ant System: An Autocatalytic Optimizing Process, Technical Report No.91-016.
- Dorigo, M., Maniezzo, V. and Colorni, A., 1996, “ The Ant System : Optimization by a Colony of Cooperating Agents,” IEEE Transactions on Systems, Man, and Cybernetics-Part B, Vol. 26, No. 1, pp. 1-13.
- Foulds, L.R., 1992, Graph Theory Applications, Springer-Verlag.
- Joo, C.M., Lee, W.S. and Lee, K.B., 2005, “ Transporter Scheduling for Block Transportation in the Shipyard,” Annual Spring Meeting of Korean Operations Research and Management Science Society/Korean Institute of Industrial Engineers, pp. 348-352.
- Koo, P.H., Lee, W.S. and Jang, D.W., 2004, “ Fleet Sizing and Vehicle Routing for Container Transportation in a Static Environment,” OR Spectrum, Vol. 26, pp. 193-209.
- Maxwell, W.L. and Muckstadt, J.A., 1982, “ Design of Automatic Guided Vehicle Systems,” IIE Transactions, Vol. 14, No. 2, pp. 114-124.
- Moon, B.R., 2001, Genetic Algorithm, DasungPress.
- Yu, H.K., 2005, A Study on the Transportation Routing Optimization of Shipbuilding Blocks, M.Sc. Thesis, University of Ulsan.



< 임 선 빈 > < 노 명 일 > < 차 주 환 > < 이 규 열 >