

타부탐색을 이용한 물류정보시스템의 선석계획 최적화 모듈 설계

홍 동 희[†] · 김 창 곤^{**}

요 약

항만 운영은 크게 게이트부문, 야드부문, 선석부문으로 구분되며, 이들 부문에 대한 운영전략과 효율적인 자원할당이 항만의 생산성향상에 중요한 역할을 한다. 특히, 선석부문에서의 자원할당 계획은 선석계획으로서 선박의 작업시간과 직결되기 때문에 최적화를 필요로 한다. 선석계획은 독립적인 자원할당이 아니라 야드부문의 장치 공간 할당과 상호 관련성을 가지고 있다. 따라서 기존 연구들이 선석계획에 대해 독립적인 자원할당을 연구한 것과 달리 야드의 장치 공간 할당과 상호 관련성에 중점을 두고 선석계획의 최적화 모듈을 설계하였다. 선석계획 최적화 모듈의 설계는 수리적기법으로 표현된 기존의 제약조건에 야드의 장치공간 할당과 관련된 제약조건을 제안하여, 해를 탐색하고, 찾은 해를 최적화하는데, 타부탐색을 사용하여 선석계획 최적화 모듈을 설계하였다. 그리고 선석계획 최적화 모듈의 성능 실험에서는 야드의 장치공간 할당계획을 적용한 선석계획이 그렇지 않은 경우에 비해 작업시간이 단축됨을 알 수 있었다.

Optimized Module Design for Berth Planning of Logistics Information System Using Tabu Search Algorithm

DongHee Hong[†] · ChangGon Kim^{**}

ABSTRACT

Port operation is largely divided into gate operation, yard operation and berth operation. Operation strategy and optimal resource allocation for three parts are important in the productivity of the port operation.. Especially the resource allocation planning in berth operation needs optimization, because it is directly connected with the processing time in shipping. Berth planning is not independent on recourse allocation but interrelated with yard stacking area allocation. Therefore, we design the optimized module of berth planning and give priority to interrelationship with yard space allocation, while existing studies design independent resource allocation in berth planning. We suggest constraints by mathematical method, and they are related to yard stacking area allocation with existing constraints. Then we look for solutions, use tabu search to optimize them, and design optimized the berth planning module. In the performance test of optimized module design of berth planning, we find that the berth planning with yard stacking area allocation takes less processing time than without yard stacking area allocation.

키워드 : 선석 계획(Berth Planning), 자원 할당(Resource Allocation), 컨테이너 항만(Container Port), 타부탐색(Tabu Search), 최적화 설계(Optimized Design), 컨테이너(Container)

1. 서 론

항만의 효율적인 운영을 위해서는 계획자의 자원할당 계획이 잘 이루어져야 한다[1]. 항만 운영은 크게 게이트 부문, 야드부문, 선석부문으로 구분되며, 이들 부문에 대한 운영전략과 효율적인 자원할당이 항만의 생산성향상에 중요한 역할을 한다. 특히, 선석부문에서의 자원할당 계획은 선석계획으로서 선박의 작업시간과 직결되기 때문에

최적화를 필요로 하는데, 선석계획은 독립적으로 자원할당이 이루어지는 것이 아니라 야드부문의 장치 공간 할당과 밀접한 관련이 있다. 그러나 기존 연구들은 선석계획이 선박과 컨테이너 크레인(CC : Container Crane)을 중심으로 한 독립적인 자원할당으로 이루어지고 있다. 따라서 본 논문에서는 기존 연구에서 제시된 선석계획을 위한 제약조건에 야드의 장치 공간 할당과 관련된 제약조건들을 추가로 제시하여, 해를 탐색하였다. 그리고 선석계획 최적화 모듈을 설계하는데에는 타부탐색을 이용하여 찾은 해를 최적화하였다.

† 정 회 원 : 동원대학 e-비즈니스과 교수
 ** 정 회 원 : 국립순천대학교 경영통상학부 교수
 논문접수 : 2003년 9월 6일, 심사완료 : 2004년 2월 12일

2. 관련 연구

선석계획과 관련된 연구들 중에는 적하계획에 대한 연구가 가장 많다. 1976년 Cojeen이 맨 처음으로 적하계획을 위한 컴퓨터 프로그램을 소개하였는데 그에 대한 구체적인 알고리즘은 알려져 있지 않다[2]. Cho는 적하계획을 수학적 모델로서 정형화하였는데 수학적모델은 문제의 크기가 커지면 계산량이 기하급수적으로 늘어나는 문제점이 있어 대규모의 선박에 적용하기에는 무리가 따른다[3]. Hayuth는 항만에서 점차 증가되는 요구(선박 도착 패턴과 항만 설계의 다양성)를 모델화하고, 그 효과를 검증하는 연구를 수행하였으며[4], Tolujev는 전통적인 항만 시뮬레이션의 상태를 항만운영, 훈련시스템, 장비관리, 항만 건설시스템 등으로 4개의 그룹으로 분류하여 실험하였다[5]. Wilson, Roach 및 Ware는 적하계획을 위한 정확한 해를 찾는 데에 계산적인 어려움을 극복하기 위해, 일반화된 컨테이너들을 블록화 된 장치장에 할당하고, 그 때 해치에 대응하는 슬롯들을 함께 그룹화하는 전략계획과정과 특정의 컨테이너들이 전략적 계획 단계동안 결정된 블록 안의 특정한 슬롯에 할당하는 전술계획과정으로 나누어, 전략계획과정에서는 분지한계법을 사용하고, 전술계획과정에서는 타부 탐색을 이용하였다[6]. Botter와 Brinati는 배의 안정성과 컨테이너크레인의 움직임, 컨테이너 적하 순서와 같이 적하계획과 관계된 다양한 특성을 고려한 정수계획법 모델을 제시하였고, 모델의 해를 구하기 위하여 명시적 계산방법을 사용하였다[7]. Avriel과 Penn은 직사각형의 해치가 없는 하나의 베이를 대상으로 하여 계산 시간이 오래 걸리는 적하계획과정을 간단한 휴리스틱 기법으로 빠른 시간에 답을 찾을 수 있게 했으며, 오직 재취급 횟수를 줄일 수 있는 방법을 제시하였다[8].

국내의 연구사례를 보면 김기영, 김갑환 등이 수학적 방법, 빔탐색법, 유전알고리즘등의 다양한 방법으로 구현을 하였다[9, 10]. 이들은 정수프로그래밍표를 사용하여 수학적 방법으로 구현을 하였는데 이 방법은 매우 우수한 적하계획을 수립하지만 수행 시간이 매우 길고 대상 문제의 규모가 커지게 되면 적용할 수 없는 단점이 있다. 또한 빔탐색법과 유전알고리즘으로 구현을 하여 수학적 방법으로 구한 해보다는 못 미치지만 그에 근접한 해를 찾아냈다. 그러나 이들은 선박에서의 작업순서를 고정시켜 두고 야드에서 TC의 작업순서만 결정하는 문제로서 보고, TC가 각 야드를 방문하는 순서를 결정하는 라우팅 문제로 변형하여 적하계획을 수립하였다. 또한 항만운영

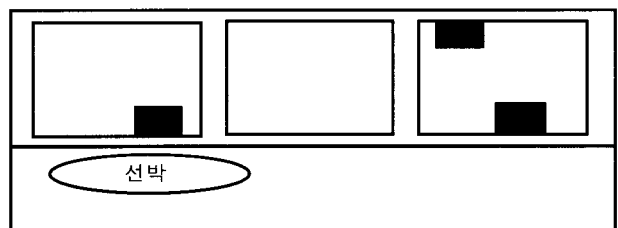
계획과 관련하여 수학적 방법, 빔 탐색법, 유전자 알고리즘 등의 다양한 방법으로 구현을 하였다.

강기중, 이철영은 컨테이너선의 적하계획 수립을 위하여 선박의 복원성과 야드 컨테이너의 기대 재취급횟수를 최소화하는 정수계획모형을 제시하였다. 분지한계법으로 초기해를 구하고, 동적계획법을 이용하여 제시된 모형의 최적해를 도출하였다[11]. 강준규, 김영대는 배의 안정성 유지를 하는 동시에 적하와 양하에 필요한 시간을 최소화하는 배의 화물배열을 찾아내기 위해, 해치에 컨테이너 그룹을 할당하는 방법과, 각 슬롯에 개별의 컨테이너를 할당하는 두 단계로 나누어 해를 찾았다[12].

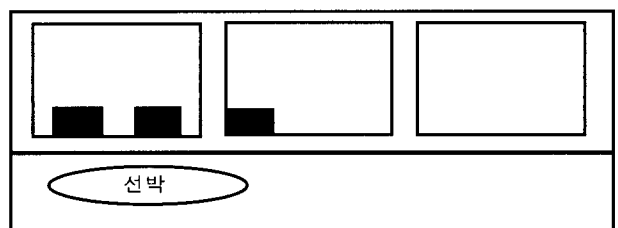
3. 선석계획의 최적화 모듈 설계

3.1 선석계획 수립

선석계획은 이용선사로부터 선석요청을 받아 선석배정을 합리적이고 효율적으로 처리하는 것이다. 이러한 선석계획은 항만에서 선박 개개의 요구조건을 수용하면서 작업을 할 때 발생하는 운영 비용을 줄일 수 있도록 계획하고 점안 위치도 결정되어야 한다. 선박의 요구사항은 각 선박의 도착예정시간 이후에 서비스를 받아 선박의 예정출항시간 이전에 출항하는 것이다. 만약 실제 선박의 출항시간이 요구 시간보다 늦어질 경우에는 다음 목적지에 도착도 늦어지기 때문에 그에 따른 추가비용이 발생된다. 그리고 항만 내에서의 운영 비용도 선박에 실을 컨테이너가 야드 어디에 장치되어 있으며, 선박이 어느 선석에 접안되어 있는냐에 따라 달라진다((그림 1)과 (그림 2)).



(그림 1) 컨테이너장치와 접안위치간의 고비용 발생



(그림 2) 컨테이너장치와 접안위치간의 저비용 발생

그래서 선석계획에 있어서 접안위치 결정 문제는 항만 측과 선박측의 추가 운영비용을 최소화할 수 있는 접안시각과 접안위치로 결정하게 된다.

3.2 선석계획 설계

선석계획의 설계 문제는 각종 제약조건을 만족시키는 많은 해들 중에서 가장 최적의 해를 찾아내는 탐색 문제이다[13]. 이를 위한 방법으로 기존의 제약조건에 추가적 제약조건을 제시하고 이를 만족하면서 최적의 해를 효율적으로 탐색해 낼 수 있는 휴리스틱기법을 활용하였다. 먼저 수리적방법으로 제약조건들을 나타내고, 이러한 제약조건들을 만족시키면서 탐색 해를 최적화하는데 타부탐색을 사용하였다.

3.2.1 선석계획의 제약조건

선석계획에 있어서 기존에 제시된 선박과 CC와 관련된 제약조건은 다음과 같다.

식 (1) 선수와 선미가 각각 선석끝에서 어느 정도의 여유가 있다.

$$(x_i \geq 10) \wedge (x_i + L_i \leq B - 30) \quad (1)$$

- x_i : 선수의비트 ($x_i = 0, 1, 2, \dots, B-1$)
- L_i : 선박의 길이
- B : 선석의 길이(비트)

식 (2) 입항시간은 입항예정시간(ETA)이다.

$$ETA(i) - BTS \leq y_i' \leq ETA(i) \quad (2)$$

- $ETA(i)$: Estimated Time of Arrival
- BTS : 선박의 입항시간 여유폭
- y_i' : 접안 시간, ($y_i' = 0, 1, 2, \dots, T-1$),
T는선석계획 총기간(시간)

식 (3) 출항시간은 출항예정시간(ETD)이다.

$$ETD(i) - DTS \leq y_i'' \leq ETD(i) \quad (3)$$

- $ETD(i)$: Estimated Time of Departure
- DTS : 선박의 출항시간 여유폭
- y_i'' : 출항 시간, ($y_i'' = 0, 1, 2, \dots, T-1$)

식 (4) 두 선박은 선석을 차지하는 공간과 시간이 겹치지 않는다.

$$(x_i + L_i + A \leq x_j) \vee (x_i + L_j + A \leq x_i) \vee (y_i'' + D \leq y_j') \vee (y_j'' + D \leq y_i') \quad (4)$$

- A : 선박사이의 허용 거리
- D : 선박사이의 시간 간격

식 (5) CC가 선박을 서비스하면 서비스 시작시간은 항 시간 이후이다.

$$\neg CS_{ki} \vee (b_{ki} \geq y_i') \quad (5)$$

- CS_{ki} : 선박 i 에 대한 CCk 의 서비스 여부
- b_{ki} : 선박 i 에 대한 CCk 의 서비스 시작시간
($b_{ki} = 0, 1, 2, \dots, T-1$)

식 (6) CC가 선박을 서비스하면 서비스 완료시간은 출항시간 이전이다.

$$\neg CS_{ki} \vee (e_{ki} \leq y_i'') \quad (6)$$

- e_{ki} : 선박 i 에 대한 CCk 의 서비스 완료시간
($e_{ki} = 0, 1, 2, \dots, T-1$)

식 (7) CC가 선박을 서비스하면 서비스 시작시간은 서비스 완료시간 이전이다.

$$\neg CS_{ki} \vee (b_{ki} \leq e_{ki}) \quad (7)$$

식 (8) CC가 선박을 서비스하지 않으면 서비스시간은 0이다.

$$\neg(\neg CS_{ki}) \vee (b_{ki} = 0 \wedge e_{ki} = 0) \quad (8)$$

식 (9) CC가 선박을 서비스하면 서비스시간은 2시간 이상이다.

$$\neg CS_{ki} \vee (e_{ki} - b_{ki} \geq 2) \quad (9)$$

식 (10) 출항시간은 CC 서비스종료시간 중 제일 늦은 시간 이후이다.

$$Max(e_{ki}) \leq y_i'' \quad (10)$$

식 (11) 하나의 CC는 서로 다른 선박에 대한 서비스를 겹치지 않게 한다.

$$(e_{ki} < b_{kj}) \vee (e_{kj} < b_{ki}) (i \neq j) \quad (11)$$

식 (12) CC 서비스시간 총합은 선박이 필요로 하는 CC 서비스시간 이상이다.

$$\sum_{k=1}^m (e_{ki} - b_{ki}) \geq S_i \quad (12)$$

- S_i : 선박 i 에 대한 CC 1대의 서비스 시간,
- m : CC의 총대수

식 (13) CC가 선박을 서비스하면 CC 이동 범위가 선박이 선석을 차지하는 공간과 일정범위 이상 겹친다.

$$\neg CS_{ki} \vee (bc_k < x_i + L_i - A \wedge ec_k > x_i + A) \quad (13)$$

bc_k : CC k 가 움직일 수 있는 범위(Lower bound)
 ec_k : CC k 가 움직일 수 있는 범위(Upper bound)

식 (14) 두 개의 CC가 서로 서비스시간이 겹치고 엇갈린 선박을 서비스한다면 동시에 서비스는 불가능하다.

$$\begin{aligned} &(\neg((x_i < x_j) \wedge (b_{ki} < e_{lj}) > b_{li})) \\ &\vee \neg(CS_{li} \wedge CS_{kj})(i \neq j, k1 < k2) \end{aligned} \quad (14)$$

식 (15) 한 선박의 서비스 가능한 CC의 수는 최소 몇대에서 최대 몇 대이다.

$$k_{\min} \leq \sum_{k=1}^m CS_{ki} \leq k_{\max} \quad (15)$$

k_{\min} : 한 선박의 서비스 가능한 최소 CC 대수
 k_{\max} : 한 선박의 서비스 가능한 최대 CC 대수

식 (16) 선수와 선미가 각각 적어도 한 CC의 이동범위와 겹친다.

$$\begin{aligned} &(\bigcup_{k=1}^m (\neg CS_{ki} \vee (bc_k \leq x_i \leq ec_k)) \wedge CS_{ki}) \\ &\wedge (\bigcup_{k=1}^m (\neg CS_{ki} \vee (bc_k \leq x_i + L_i \leq ec_k)) \wedge CS_{ki}) \end{aligned} \quad (16)$$

식 (17) 한 선석을 서비스하는 CC들 사이에 서비스를 하지 않는 CC가 있으면 안된다.

$$(\neg(CS_{k1} \wedge CS_{k2}) \vee CS_{k3})(k1 < k2 < k3) \quad (17)$$

3.2.2 선석계획 최적화를 위한 제약조건

야드의 장치 공간 할당과 관련된 제약조건들을 도출하면 다음과 같다.

식 (1) 선적작업이 일어나는 블록에는 장비간의 간섭이 발생하므로 공간을 할당할 수 없다.

$$\begin{aligned} &IF b(X_{ki}) = u \text{ and } t(s_{kl}) = i \text{ THEN } B(X_{ij}) \neq u \\ &\text{for } \forall k < i, l, u, j \end{aligned} \quad (18)$$

$t(j)$: 선박 j 의 적화작업

식 (2) 서비스하는 선석과 해당 블록사이의 거리는 y 미터 이하여야 한다.

$$d_1(b(x_{ij}), s_{ij}) \leq y \quad \text{for } \forall i, j \quad (19)$$

$d_1(b, s)$: 블록 b 와 선박 s 의 접안위치까지의 거리
 X_{ij} : i, j 위치에 할당되는 bay set
 s_{ij} : i, j 위치에 있는 선박

식 (3) 같은 선박에 실릴 컨테이너클러스터에 할당된 블록은 동일한 행이어야 한다.

$$\begin{aligned} &IF s_{ij} = s_{kl} \text{ THEN } r(b(X_{ij})) = r(b(X_{kl})) \\ &\text{for } \forall (i, j), (k, l) \end{aligned} \quad (20)$$

$r(b)$: 블록 b 의 행의 값

식 (4) 같은 선박에 실릴 컨테이너클러스터는 n 개 이상의 블록에 흩어져 있으면 안된다.

$$|(b(X_{ij}) \text{ for } \forall i, j | S_{ij} = s)| \leq n \text{ for } \forall s \in S \quad (21)$$

S : 선박의 집합

식 (5) 같은 선박에 실릴 컨테이너클러스터가 할당된 베이들은 그 베이에 소속된 블록간의 거리가 z 미터 이하여야 한다.

$$\begin{aligned} &IF s_{ij} = s_{kl} \text{ THEN } d_2((b(X_{ij}), b(X_{kl}))) \leq z \\ &\text{for } \forall (i, j), (k, l) \end{aligned} \quad (22)$$

$d_2(b_1, b_2)$: 블록 b_1 과 블록 b_2 사이의 거리

식 (6) 한 블록에는 선박에 실을 컨테이너 클러스터들을 m 개 이상 혼합해서 장치하면 안된다.

$$\begin{aligned} &|(s_{kj} | t(k) \geq i, b(X_{kj}) = b)| \leq m \\ &\text{or } \forall i \text{ and } b \in B \end{aligned} \quad (23)$$

B : 블록의 집합

3.2.3 휴리스틱기법 적용

제약조건을 모두 만족하는 해를 탐색하고, 그 해를 최적화 할 수 있는 휴리스틱기법을 사용한다. 휴리스틱기법에는 여러 가지가 있으나 여기에서는 Hill Climbing 탐색처럼 지역적인 이웃 해를 탐색해 나가면서 타부 리스트와 집중화, 다각화 전략을 이용, 지역적 최적해를 넘어서 전역적 최적해를 탐색할 수 있는 타부 탐색기법을 사용한다. 목적함수에 따라 해의 최적화작업을 수행하도록 하기 위해 다음과 같이 목적함수를 정의한다.

- ① 우선 순위를 고려한 선석계획 : 정기선이 부정기선보다 우수함
- ② 컨테이너 장치위치를 고려한 선석계획 : 각 선박마다 원하는 위치에 가까울수록 penalty를 적게 부여함
- ③ 선호하는 선석 고려
 - 화물이 벌크인 경우 1번 선석쪽
 - 작은 선박은 1, 2번 선석, 큰 선박은 3, 4번 선석
 - 선석에 선박이 1대만 있는 경우, 2번 또는 3번 선석

- ④ 선박이 계획대로 출항하지 못 할 경우를 고려한 선석계획

3.3 선석계획의 최적화

선석계획 최적화를 위한 타부탐색의 과정은 먼저 정해진 순서에 의해 초기해를 생성하고, 다음으로 전체 클러스터를 대상으로 후보해를 생성하여 평가한다. 최적화를 위해 해를 클러스터 튜플 리스트로 나타낸다. 클러스터 튜플은 <표 1>에 나타난 바와 같이 선박 클러스터 번호, 야드 클러스터 번호, 컨테이너 개수 등 3가지로 구성된다[14].

<표 1> 클러스터 튜플(해의 표현)

	작업 순서 →						
선박클러스터	0	0	1	2	2	3	...
야드클러스터	1	2	2	1	3	3	...
컨테이너 수	3	5	4	5	5	6	...

3.3.1 초기해 생성

선박 클러스터의 작업순서는 선후관계 행렬을 지키는 임의의 순서로 정하고, 선박과 야드의 작은 선박 클러스터의 작업순서에 따라 선박 클러스터에 최근접탐색(Nearest Neighbor Search)으로 야드 클러스터를 짝을 이어준다. 시작노드가 될 수 있는 모든 노드를 Seed로 시작하여 초기해를 만든 뒤 가장 좋은 평가값을 가지는 것을 초기해로 선정한다.

```

Algorithm Initialize()
  Y : Set of yclusters
  x : Current solution
  k : The number of yclusters to be selected
Begin
  x = ∅
  While |x|!k Do
    find j which has minimum Δ j where j ∈ Y
    x = x ∪ {j}
    Y = Y - {j}
  End While
  Return x
End Begin
    
```

초기에 해의 집합 x에는 선택된 해(짝지어진 클러스터)가 하나도 존재하지 않으며, 선박클러스터의 작업개수 제약조건에 지정된 개수 k가 선택될 때까지 하나씩 선택하는 과정을 반복적으로 수행한다.

Y는 대상 야드클러스터를 중 현재 x에 포함되지 않은 클러스터들의 집합을 의미하며, j는 현재 남아있는 야드

클러스터들 중에서 대상 선박클러스터에 가장 가까이 있는 클러스터를 의미한다. 단, 대상 선박클러스터에 속한 컨테이너 개수만큼 선택된 경우에는 작업순서에 의해 다른 선박클러스터를 대상으로 선택한다.

3.3.2 후보해 생성

전체 클러스터를 대상으로 가능한 모든 후보해를 만들어 탐색하는 경우 매우 많은 후보해가 생성되므로, 한번에 새로운 현재의 해를 위해 매우 많은 후보해를 평가해야 한다. 따라서 선박 클러스터의 작업순서를 기준으로 클러스터 튜플을 윈도우라는 단위로 구분하고, 이를 기반으로 후보해의 생성방법을 다음과 같이 두 단계로 구분하여, 한번에 매우 많은 후보해가 생성되는 것을 방지한다[14]. 하나의 윈도우는 선박 베이에서 홀드와 데크 단위로 만든다.

[1 단계]

- ① 한 윈도우내에서 동일한 종류(목적지항, 유형, 크기)의 컨테이너를 가지는 야드클러스터를 교환한다.

0	0	1	2	2	3
1	2	2	1	3	3
3	5	4	5	5	6



0	0	1	1	2	2	3
2	2	2	1	1	3	3
3	5	1	3	5	5	6

- ② 한 윈도우내에서 인접하면서 교환하였을 때 선후관계 행렬을 어기지 않는 클러스터 튜플을 서로 교환한다.

0	0	1	2	2	3
1	2	2	1	3	3
3	5	4	5	5	6



0	0	2	1	2	3
1	2	1	2	3	3
3	5	5	4	5	6

- ③ ②를 수행한 뒤 추가적으로 야드클러스터를 교환한다.

[2 단계]

서로 다른 윈도우간에 동일한 종류의 컨테이너를 가지는 야드클러스터를 교환한다. 단, 야드클러스터 교환 시

개수가 서로 다를 경우는 큰 쪽을 분할 한 뒤 교환한다.

0	0	1	2	2	3	100	101	102	200
1	2	2	1	3	3	5	10	2	6
3	5	4	5	5	6	5	2	4	3

←----- 윈도우(홀드) -----> <----- 윈도우(테크) ----->

0	0	1	2	2	3	100	100	101	102	200
5	2	2	1	3	3	1	5	10	2	6
3	5	4	5	5	6	3	2	2	4	3

두 단계의 탐색은 다음의 경우에 종료한다. 1단계에서 한 윈도우에서 다음 윈도우로 넘어가기 위한 조건은 주어진 횟수동안에 새로 취한 현재 해의 향상이 없는 경우이다. 모든 윈도우를 다 탐색하고 나서, 1단계에서 하나의 윈도우에서라도 향상이 있으면 2단계를 수행하고, 향상된 경우가 없으면 탐색이 종료된다. 2단계에서는 주어진 횟수동안 향상이 없으면 2단계는 종료되고, 2단계를 수행하는 동안 향상이 있으면, 1단계를 다시 수행하고, 그렇지 않으면 종료한다.

4. 최적화 모듈 평가

4.1 평가 요소

선석계획의 최적화 설계의 평가요소는 야드에서 컨테이너를 이동하는 이동장비인 AGV와 부두의 대형크레인인 CC, 그리고 컨테이너의 이동이라고 할 수 있다. 모델을 평가하는데 있어서는 이들 평가요소에 대하여 높은 수준의 접근 방법(즉, 자원의 실행 가능성 요구와 야드 클러스터 지역에서 CC까지의 거리에 비례한 시간에 대한 자원의 유지 등)을 사용한다.

따라서 가장 중요한 수행 평가 요소로서는 컨테이너 이동에 걸리는 시간(즉, 야드에서 선박에까지 컨테이너가 이동할 거리)으로서 다음의 요소들을 고려한다.

- 선적할 야드클러스터의 위치
- 집안된 선박클러스터의 위치
- 클러스터에 할당된 컨테이너의 수

최적화 모듈은 직접적으로 위의 요소들을 반영하여 수행하였고, 그래서 야드에서 집안된 선박에까지 이동할 컨테이너의 거리가 얼마나 잘 수행하였는지의 적절한 척도가 된다.

모델의 평가를 위한 실험 자료로는 2002년 부산항의 신선대부두의 2주간의 계획자료를 대상으로 하였으며, <표

2>와 같다.

<표 2> 실험 자료

항 목	값
CC 수	4
평균 선박 작업시간 :	
1번 라인(280개)	7.1 시간
2번 라인(612개)	16.2 시간
3번 라인(1230개)	30.6 시간
선박당 평균 클러스터	10
선박당 예약 크레인 수	1

4.2 검증 및 평가

여기에서는 모델을 검증하기 위해 다음과 같이 단계를 설정하여 수행하였다.

- ① 모델은 수행되는 동안 속성 값의 실험을 허용하는 범위내에서 서로 검사한다.
- ② 적당하고 합리적인 반응이 일어나는 지를 보기 위해, 내부 도착 비율을 증가시키는 것과 같은 꾸준한 검사를 수행한다.
- ③ 모든 개별적인 사건을 추적하기 위해 광범위한 추적 분석이 반복 수행하도록 한다.

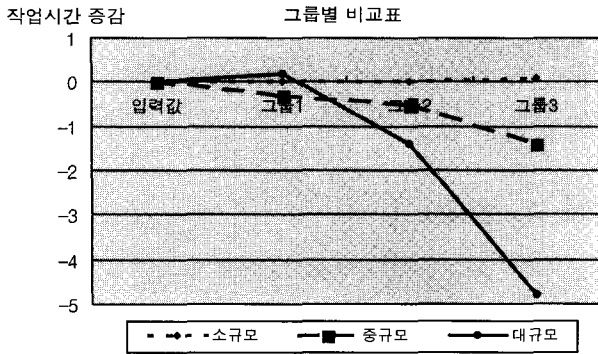
따라서 모델의 검증은 통계적인 수치로 그 결과를 서로 비교하는데 중점을 두었으며, 특히 각 선박 라인들의 도착율과 선박 정박지로부터 그 선박에 선적할 컨테이너의 야드클러스터까지의 거리에 대한 비교를 하였다.

여기에서의 성능 평가는 3그룹으로 실행하였다. 그룹 1은 현재 항만에서 일반적으로 적용하고 있는 FCFS 접근 방법으로, 선석 계획에 최적화 모듈을 적용하지 않은 것이다. 대신에 클러스터는 미리 준비하지 않고, 컨테이너가 도착했을 때 만든다. 그룹 2는 기존 연구에서 제안된 제약 조건(식 (1)~식 (17))을 만족하는 방법을 적용하였다. 그룹 3은 본 논문에서 제안한 야드 장치공간 할당의 제약조건들을 포함한 최적화 모듈을 적용했다. 그리고 이 그룹 3은 다시 문제의 규모를 컨테이너의 개수에 따라 280, 612, 1230개의 소규모, 중규모, 대규모로 나누어 실험을 하였다. 실험환경은 윈도우2000에서 MS Visual C++ 6.0으로 수행하였다.

<표 3> 그룹별 평균작업시간 비교표

(단위 : 시간)

항 목	입력값	그룹 1	그룹 2	그룹 3
1번라인(소규모)	Basic	7.1	7.1	7.2
2번라인(중규모)	Basic	15.9	15.7	14.8
3번라인(대규모)	Basic	30.8	29.2	25.8



(그림 3) 그룹별 작업시간 증감 비교도

결과는 <표 3>에 있는 것처럼, 그룹 1인 FCFS 접근 방법은 항만에서 적용하고 있는 현재의 방법으로 입력값과 거의 차이가 없음은 당연한 결과이다. 그룹 2의 결과는 선박의 작업시간이 다소 단축되었음을 알 수 있다. 이것은 선박의 도착 예정시간과 CC의 할당과 관련된 제약조건 등을 만족하는 선석계획을 적용하였기 때문이다. 그룹 3은 컨테이너의 수가 소규모인 경우는 선석과 야드 장치 공간할당과의 관계가 작업시간에 별 영향을 주지 못하고, 오히려 FCFS보다 약간 지연되는 결과를 보여준다. 그러나 중규모와 대규모인 경우는 그룹 2보다도 작업시간이 단축되었으며, 컨테이너 수가 대규모인 경우에는 작업시간에 상당한 개선을 보여준다. 이것은 타부탐색이 전역적인 탐색능력으로 선박과 야드와의 관계를 선박 클러스터의 작업순서에 따라 선박 클러스터에 최인접한 야드 클러스터의 짝을 찾아냈음을 보여준다.

5. 결 론

이 연구의 결과에 대한 정리는 선석계획의 최적화 설계에 사용된 매개변수들에 있다는 것이다. 다른 알고리즘에서도 그렇듯이, 매개변수가 중요한 역할을 한다. <표 3>과 (그림 3)에 나타난 것처럼, 결과는 컨테이너 수인 규모의 매개 변수에 민감하다는 것을 알 수 있다. 또한 선석계획은 최적화 모듈을 적용하되 반드시 야드장치공간 할당계획에도 같이 적용하여 이루어져야만 한다는 것이다. 그리고 위의 결과에서 알 수 있듯이 그룹 3의 소규모에는 작업시간이 오히려 그룹 2보다 좋지 않게 나타난다. 야드의 컨테이너 장치 상황과의 연계가 오히려 좋지 않은 결과를 보여준다. 그러나 앞으로 항만의 규모가 커지고 기능이 확대될수록 대규모 컨테이너 처리가 발생할 수 밖에 없기 때문에 그룹 3의 대규모와 같이 최적화 모듈의 적용으로 항만의 생산성을 향상시킬 수 있을 것이다.

참 고 문 헌

- [1] 박강태, "모델변경의 용이성을 고려한 컨테이너 터미널의 공간할당 계획", 석사학위 논문, 부산대학교, 1997.
- [2] Cojeen, H. P. and Dyke, P. V., "The Automatic Planning and Sequencing of Containers for Containership Loading and Unloading," Ship Operational Automation, Eds. Pitkin, Roche and Williams, North-Holland Publishing Co., 1976.
- [3] Cho, D. W., "Development of a Methodology for Containership Load Planning," 0, Ph. D. Thesis, Oregon State University, 1982.
- [4] Hayuth, Y., Pollatschek, M. A. and Roll, Y., "Building a Port Simulator," SIMULATION, Vol.63, No.3, pp. 179-189, 1997.
- [5] Tolujev, J., Merkurjev, Y., Blumel, E. and Nikitins, M. "Port Terminal Simulations : State of the Art-A Survey," ANCAI, 1998.
- [6] I. D. Wilson and P. A. Roach, "Container stowage planning : a methodology for generating computerized solutions," Journal of Operation Research Society, Vol.51, pp.1248-1255, 2000.
- [7] R. C. Botter and M. A. Brinati, "Stowage container planning : a model for getting an optimal solution," IFIP Trans. B (App. in Tech.) B-5, pp.217-229, 1992.
- [8] M. Avriel, M. Penn, N. Shpirer and S. Witteboon, "Stowage planning for container ships to reduce the number of shifts," Annals of Operations Research, 76, pp.55-71, 1998.
- [9] 김기영, "Mathematical Models and Search Techniques for Containership Load Planning," 박사학위논문, 부산대학교, 1998.
- [10] Kim, K. H and Kim, K. Y., "An Optimal Routing Algorithm for a Transfer Crane in Port Container Terminals," Institute for Operational Research and the Management Science, Vol.33, No.1, pp.17-33, 1999.
- [11] 강기중, 이철영, "컨테이너선의 적재계획에 관한 연구", 한국항만학회지, Vol.14-4, pp.1-15, 1990.
- [12] J. G. Kang, Y. D. Kim, "Stowage Planning in Maritime Container Transportation," Master Thesis, Kaist, 2000.
- [13] 이종술, "제약만족기법을 이용한 컨테이너터미널의 수출장치장 공간할당 계획", 석사논문, 부산대학교, 2000.
- [14] 이용환, "메타 휴리스틱 기법을 이용한 컨테이너 적하계획 시스템", 석사논문, 부산대학교, 2001.



홍 동 희

e-mail : sonbal2000@dreamwiz.com

1981년 홍익대학교 전자계산학과(이학사)

1987년 연세대학교 산업대학원 전자계산
전공(공학석사)

2004년 경희대학교 대학원 전자계산공학과
(공학박사)

1987년~1999년 한국해양수산개발원 책임연구원

2000년~현재 동원대학 e-비즈니스과 겸임 조교수

관심분야 : planning, 물류정보시스템, 시스템설계 방법론 등



김 창 곤

e-mail : cgkim@sunchon.ac.kr

1985년 경희대학교 산업공학과(공학사)

1988년 서울대학교 대학원 산업공학과
(공학석사)

2000년 경희대학교 대학원 산업공학과
(공학박사)

1987년~1996년 한국해양연구소 선임연구원

1996년~2002년 한국해양수산개발원 부연구위원

2002년~현재 국립순천대학교 경영통상학부 전임강사

관심분야 : 대기이론, 물류정보시스템, 시뮬레이션 등