

Print ISSN: 1738-3110 / Online ISSN 2093-7717  
<http://dx.doi.org/10.15722/jds.16.12.201812.95>

## Performance Analysis of Preprocessing Algorithm in Container Terminal and Suggestion for Optimum Selection

### 컨테이너 터미널의 선처리 알고리즘 성능분석과 최적선택 제안

Young-Kyu Park(박영규)\*

Received: November 02, 2018. Revised: November 15, 2018. Accepted: December 05, 2018.

## Abstract

**Purpose** – In order to gain the upper hand in competition between container terminals, efforts to improve container terminal productivity continue. Export containers arrive randomly in the container terminal and are carried in the container terminal yard according to the arrival order. On the other hand, containers are carried out of the container terminal yard in order based on container weight, not in order of arrival. Because the carry-in order and the carry-out order are different, rehandling may occur, which reduces the performance of the container terminals. In order to reduce rehandling number, containers can be moved in advance when they arrive, which is called preprocessing. This paper proposes an effective preprocessing algorithm and analyzes the factors that affect the productivity of the container terminals. It also provides a way to choose the best factors for preprocessing for a variety of situations.

**Research design, data, and methodology** – To analyze the impact of factors affecting the performance of preprocessing algorithms presented in this paper, simulations are performed. The simulations are performed for two types of bays, 12 stacks with 8 tiers, and 8 stacks with 6 tiers.

**Results** – The results of the factor analysis that affects the performance of the preprocessing algorithm were as follows. (1) As the LMF increased, preprocessing number increases and rehandling number decreased. (2) The LML effect was greatest when the LML changed from 0 to 1, and that the effect decreased when it changed above 1. (3) The sum of preprocessing number and rehandling number was then shown to be increased after decrease, as the LMF increased. (4) In the case of NCI, a decrease in NCI showed that the containers would become more grouped and thus the performance was improved. (5) There was a positive effect in the case of EFS.

**Conclusion** – In this paper, preprocessing algorithm was proposed and it was possible to choose the best factors for preprocessing for a variety of situations through simulations. Further research related to this study needs to be carried out in the following topic : a study on the improvement of container performance by connecting the preprocessing with remarshalling.

**Keywords:** Preprocessing Algorithm, Rehandling, Remarshalling, Performance Factor, Container Terminal Yard.

**JEL Classifications:** C61, C88, L91, R49.

## 1. 서론

유통 분야에서의 국가 간 경쟁은 컨테이너 터미널 간에도 치열하다. 세계의 각 컨테이너 터미널들은 생산성을 높임으로써 다른 터미널보다 높은 경쟁력을 확보하려는 노력을 계속하고 있다. 컨테이너 터미널의 생산성은 선박이 항만에 입항한 후 본선작업을 수행하고 이항할 때까지의 경과 시간 즉, 재항

시간에 의해 크게 영향을 받는다. 본선작업은 적하작업과 양하작업으로 이루어져 있는데, 이중 적하작업은 수출 컨테이너를 선박에 적재하는 작업으로 재항시간에 영향을 미치는 중요한 작업 중 하나이다. 양하작업은 선박에서 컨테이너를 내리는 작업으로 선박에서 컨테이너를 내리는 순서대로 장치장에 장치하므로 작업시간에 변화가 적어서 적하작업보다 상대적으로 터미널의 생산성에 영향을 적게 미친다. 장치장에서 컨테이너 반입순서와 컨테이너 반출 순서가 다른 이유는, 반입 순서는 컨테이너가 장치장에 무작위로 도착하는 순서를 따르고, 반출 순서는 선박의 안정성 때문에 무거운 컨테이너들을 선박의 바

\* First Author, Professor, Dept. of Port & Logistics, Kaya University, Korea, Tel: +82-55-330-1136. E-mail: ykpark@kaya.ac.kr

닥 쪽에 배치하는 기본원칙(Kang, Oh, Ru, & Kim, 2004)을 반영하기 때문이다. 그래서 컨테이너를 반출할 때 컨테이너가 적재되어 있는 상태에 따라, 작업시간에 변화가 많이 발생하여 작업시간이 많이 소요될 수 있다. 컨테이너를 반출할 때 반출할 컨테이너위에 다른 컨테이너들이 장치되어 있다면, 그 컨테이너를 먼저 이동시켜야 하는 작업이 필요한데, 이를 재취급(rehandling)이라 한다. 재취급이 발생하면 반출작업에 지연이 발생하게 되고, 이는 곧 선박에 적재하는 작업에 지연을 발생시켜 재항시간을 높이게 되므로 컨테이너 터미널의 생산성을 저해하게 되는 것이다. 재취급이 발생하는 비율은 베이의 크기나, 적재되어 있는 컨테이너의 수, 선박에 적재하기 위한 인덱스의 수에 따라 다양하다. 본 논문에서는 컨테이너를 반입할 때 재취급이 발생하지 않도록 장치하거나, 재취급 발생이 최소화 하도록 장치하는 알고리즘을 제시한다. 재취급을 발생하지 않도록 하는 방법으로 많이 사용하는 방법으로 재정돈 작업(remarshalling)이 있다. 재정돈 작업은 반출 작업을 할 때 재취급 작업이 발생하지 않도록 하기 위하여, 모든 반입 작업이 끝난 후 선박이 도착하기 전까지의 유휴 시간 동안 컨테이너들을 정리하는 작업을 말한다. 이는 모든 반입이 완료된 후에 실시한다는 점에서 후처리라고 볼 수 있는데 본 논문에서 제시하는 알고리즘은 컨테이너가 반입될 때 마다 필요에 따라 실시하는 방법으로 모든 반입 작업이 완료되기 전에 실시한다는 점에서 선처리(preprocessing)라고 할 수 있다. 선처리 작업은 각각의 컨테이너를 장치장에 반입하기 직전이나 반입한 직후에 실시하여 재취급을 줄이는 방안으로서 장치장의 성능개선을 이루는 것이 목적이다. 본 논문에서는 항만에서 일어나는 컨테이너 터미널 작업 중 장치장 반입시의 선처리 알고리즘을 제시하고 선처리 알고리즘의 성능에 영향을 미치는 요소들과 그들의 효과를 알아본다. 다음은 논문의 구성에 대한 부분이다. 2장은 본 논문과 관련한 선행 연구와 선처리 알고리즘에 대하여 알아본다. 3장에서는 연구 방법론에 대하여 다루며, 4장에서는 실험결과와 결과 분석에 대하여 언급한다. 5장에서는 연구와 관련한 시사점과 연구의 한계 및 향후 연구방향에 대하여 언급한다.

## 2. 이론적 고찰

### 2.1. 선행연구

컨테이너 터미널과 관련한 연구로 터미널에서 일어나는 다양한 작업들의 성능을 개선하려는 연구들이 있는데, 먼저 터미널 운영에 관한 여러 가지 의사결정문제를 다루는 연구가 있고(Kim, 2007), 컨테이너 터미널에 컨테이너가 도착할 때 장치장의 영역을 효율적으로 할당하는 방안에 관한 연구도 있다(Kim & Park, 2003; Zhang, Liu, Wan, Murty, & Linn, 2003; Jeong, Kim, Woo, & Seo, 2012; Mohammad, Nima, & Nikbakhsh, 2009). 장치장의 영역이 할당된 뒤에는 구체적으로 컨테이너를 장치할 순서와 위치가 정해져야 하는데 이를 효율적으로 하는 것도 중요한 작업이다(Kim, Park, & Ryu, 2000; Kang, Ryu, & Kim, 2006), 재취급을 줄이기 위해서, 컨테이너들이 장치된 한 선박에 신기 전의 시간을 활용하여 컨테이너를 이동하는 방안에 관한 연구와(Kim & Hong, 2006; Wan, Liu, & Tsai, 2009; Lee & Lee, 2010; Caserta, Voß, & Sniedovich, 2011). 선박에 컨테이너를 싣는 순서에 대한 연구

들도 장치장 성능에 영향을 미친다(Kim, Kim, & Ko, 1997; Imai, Sasaki, Nishimura, & Papadimitriou, 2006; Lee, 2011). 이밖에도 컨테이너 터미널과 관련한 연구로 선석배정과 안벽 크레인(Quay Crain)을 효율적으로 수행하는 것과 관련한 연구들도 있다(Han, Lu, & Xi, 2010). 본 논문과 직접적으로 관련 있는 재정돈과 재취급에 대한 연구에 대하여 자세히 알아보기 전에, 컨테이너 이동과 관련 있는 트랜스퍼 크레인(Transfer Crane)에 관한 연구를 먼저 알아본다. 위에서 언급한 작업 중 컨테이너를 이동하거나 선박에 싣는 작업들을 위해서는 트랜스퍼 크레인 작업을 하게 되는데 이 작업도 장치장의 성능에 영향을 미친다. Kim, Park, Park, and Ryu (2009)은 장치장의 상황 때문에 트랜스퍼 크레인을 사용하는데 제한이 있을 때, 작업 대상이 되는 컨테이너를 선택하는 방안에 관하여 연구를 하였다. 이 연구에서는 네 가지 선택 방안으로 작업할 컨테이너를 선택하였으며 각각의 선택 방안에 대한 성능을 시뮬레이션을 통해서 비교 분석하였다. 그리고 Kim, Yang, Bae, and Ryu (2014)은 재정돈을 수행하는 시점에서 트랜스퍼 크레인 할당에 대한 제약 없이 사용할 수 있을 경우, 트랜스퍼 크레인을 효과적으로 할당하는 방안에 대한 연구를 하였다. Kang, Oh, Ru, and Kim (2005)은 블록 내에 있는 여러 베이들로 흩어져 적재되어 있는 컨테이너들을 한 트랜스퍼 크레인으로 모아 재취급이 발생하지 않도록 하는 방안에 대하여 연구를 수행하였으며, Oh, Kwang, Yu, and Kim (2005)는 여러 블록들에 흩어져 있는 컨테이너들을 복수 트랜스퍼 크레인으로 한 곳에 모으는 휴리스틱에 대한 연구를 수행하였다. 이 연구에서 교차를 할 수 없는 복수 트랜스퍼 크레인을 대상으로 트랜스퍼 크레인들 간의 간섭을 최소화하면서 작업시간을 단축시키는 방안을 제시하였다. 마지막으로 재취급을 줄이기 위하여 수행하는 작업들로 반입 작업과 관련한 연구들을 살펴본다. 먼저 하나의 베이를 대상으로 재정돈 계획을 수립하는 연구가 있었는데, Hirashima, Ishikawa, and Takeda (2006)은 컨테이너를 재정돈하기 위하여 강화학습을 활용하는 방안에 대하여 연구하였고, Ha and Kim (2012)는 A\*알고리즘(Nilsson, 1998)을 이용한 재정돈 계획 수립 방안에 대하여 연구를 하였다. 이 연구에서는 재정돈 계획을 수립하는 방안과 함께 이적을 수행하는 횟수의 하한을 찾는 방법도 제안하였는데, 베이의 복잡도에 따라서 재정돈 계획을 수립하지 못하는 경우도 있었으나, 일반적인 복잡도에서는 허용될 수 있는 시간 내에 재정돈 계획을 효과적으로 수립하는 것을 시뮬레이션으로 증명하였다. Park (2016)은 하나의 베이를 대상으로 재정돈 계획을 수립하는데 있어서 기존의 방안들이 가지고 있는 제한 중 하나인 컨테이너 베이내 이동이라는 제한을 없앤 논문이다. 이 논문에서는 재정돈 대상이 되는 베이의 이웃한 베이에 있는 빈 슬롯을 활용하여 재정돈의 성능을 높이는 방안을 제시하였으며 복잡한 베이를 대상으로 제안한 알고리즘의 효과를 시뮬레이션 기법으로 입증하였다. 지금까지 모든 반입과정이 완료된 후의 작업인 재정돈에 대한 연구를 알아보았는데, 컨테이너가 반입되는 과정에 수행하는 작업들에 대한 연구들도 있다. 먼저 Kang et al. (2004)은 반입하는 단계에서 컨테이너 무게를 활용하여 장치할 위치를 결정함으로써 재취급을 줄이는 방안을 제안하였다. 컨테이너 무게를 고려하지 않고 무작위로 장치하는 경우와 비교하면, 재취급 감소 효과가 있다는 것을 시뮬레이션을 통하여 증명하였다. 본 논문에서도 장치장에 반입하는 시점에 컨테이너를 장치할 위치를 결정할 때 사용하는 방법으로 이 방안을 채택하였다. Park and Kwak (2011)은 컨테이너를 반입할 때마다 수행하는 선처리에 대한 연구를 하였다. 이 연구에서 선처리를 수행하

따라서 나타나는 재취급의 감소효과를 시뮬레이션을 통하여 증명하였다. 본 논문은 이 연구의 선처리 알고리즘의 기능을 확장한 논문으로 이 연구에서 제안한 두 가지 선처리 알고리즘들에 대하여 컨테이너 이동의 횟수제한을 없앤 후 하나로 통합하고, 인덱스 기법을 도입함으로써 알고리즘의 성능을 높였다. 그리고 선처리 알고리즘의 성능에 미치는 요소들을 분석하여 다양한 상황에 적합한 최적의 알고리즘을 수행할 수 있는 방안을 제시하였다. 재정돈 작업은 한 번에 전체 작업을 수행하여야 하므로 많은 시간을 필요로 한다는 단점이 있다. 현장에서는 재정돈 작업을 하는 동안 시간이 부족하여 베이 내의 컨테이너들 중 일부만 재정돈 작업을 하는 경우도 발생한다(Park, Park, & Ryu, 2010). 이에 비하여 선처리 알고리즘은 반출시마다 나누어서 수행하므로 재정돈에 비하여 시간적인 면에서 상대적인 장점이 있다고 할 수 있다.

2.2. 관련용어

장치장의 구성과 관련 용어를 먼저 알아본다. 장치장은 컨테이너를 적재하기 위한 공간으로, 하나의 컨테이너 위로 여러 개의 컨테이너를 쌓아놓을 수 있는데 그것을 스택이라고 한다 (Zhang et al., 2003). 즉 1개의 컨테이너 위에 다른 컨테이너가 4개 더 쌓여 있다면, 그 5개를 5단으로 된 컨테이너라고 한다. 여러 개의 스택이 모인 것을 베이라고 하며, 하나의 베이 내에서 순서를 나타내기 위하여 스택대신 열을 사용하기도 한다. 블록은 여러 개의 베이를 모아 놓은 것을 말하며, 장치장은 이런 블록이 여러 개 모여 있는 곳이다. <Figure 1>는 5개의 스택으로 이루어진 하나의 베이를 나타내었는데, 육면체들은 컨테이너들을 나타내며, 제일 좌측의 빗금 친 스택은 컨테이너가 3단으로 쌓여 있는 것을 나타내었다. <Figure 2>는 여러 베이들로 구성된 블록을 나타내었는데, 위에서 내려다 본 모습이다.

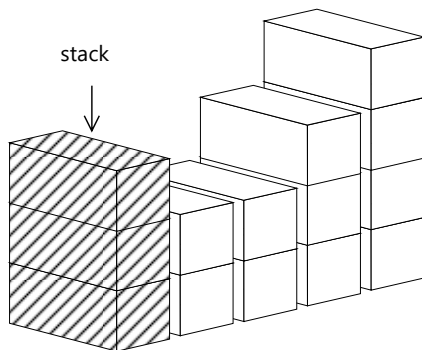


Figure 1: Example of a bay

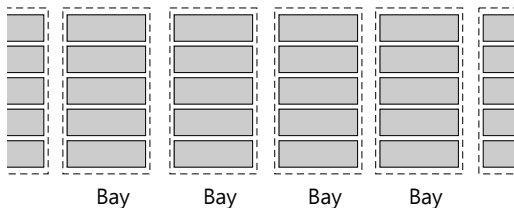


Figure 2: Example of a block

컨테이너들을 반출하는 순서는 인덱스라는 수로 나타내는 우선순위를 기준으로 정해진다. 무거운 무거운 컨테이너에 높은 우선순위를 부여하며, 우선순위가 높은 컨테이너를 우선순위가 낮은 컨테이너보다 먼저 반출하게 된다. 이는 선박의 안정성을 위한 것으로 선박의 하단에 무거운 컨테이너를 장치하고 상단에 상대적으로 가벼운 컨테이너를 장치하기 위한 것이다. 우선순위는 컨테이너들의 무게를 몇 개의 그룹으로 나누어서 각 그룹에 부과하는 인덱스라는 수로 나타내며, 모든 컨테이너들에게 서로 다른 인덱스를 부과할 수도 있다. 다음은 선처리 알고리즘과 관련요소들에 대한 설명이다.

2.3. 제안 알고리즘

컨테이너를 장치장에 반입할 때 적재할 위치인 스택을 선택하여 반입을 하게 되는데, 스택에 따라서 그리고 반입하는 컨테이너에 따라서, 향후 반출시에 재취급이 발생할 수도 있고, 발생하지 않을 수도 있다. 본 논문에서 제안하는 알고리즘은 컨테이너 반입시, 먼저 재취급이 발생하지 않을 스택을 선택하여 적재하고, 만약 재취급이 발생할 수밖에 없을 때는 선처리를 통하여 재취급의 발생을 최소화하는 알고리즘이다. 재취급이 발생하지 않을 스택을 선택하는 방안은 Kang et al. (2004)에서 제시한 방식으로, 먼저 반입 컨테이너의 인덱스  $X_c$ 와 각 스택의 최고인덱스(그 스택 중 가장 우선순위가 높은 인덱스)를 비교한다.  $X_c$ 보다 낮은 최고인덱스를 가진 스택에 적재하면 재취급이 발생하지 않으며, 그런 스택이 여러 개 있을 경우 그들 중 가장 높은 최고인덱스를 가진 스택에 적재하게 된다.  $X_c$ 보다 낮은 최고인덱스를 가진 스택이 없을 경우, 어느 스택에 적재하더라도 재취급이 발생하게 되는데, 이때 본 논문에서 제안하는 선처리 알고리즘을 수행함으로써 재취급의 발생을 최소화할 수 있게 된다. 선처리 알고리즘은 2부분으로 구성되어 있다. 첫 부분은 장치장에 반입하는 컨테이너가 정해진 후 그 컨테이너를 반입하기 전에 수행하는 선처리로 반입전 선처리(First-Preprocessing)라고 하고, 두 번째 부분은 컨테이너를 반입한 후 수행하는 선처리로 반입후 선처리(Later-Preprocessing)라고 한다. 하나의 컨테이너가 반입될 때마다 2부분의 선처리 모두 수행할 수 있으며, 경우에 따라, 하나의 선처리만 수행할 수도 있다. 선처리의 실행 여부는 조건에 따라 달라진다. 다음은 본 논문에서 사용하는 기호를 나타낸 후, 알고리즘에 대하여 자세히 설명한다.

- $C_s$  : 하나의 베이를 구성하는 스택의 개수
- $C_e$  : 하나의 베이에서 컨테이너가 적재되지 않은 스택의 개수
- $C_f^{m1}$  : 반입전 선처리 수행시 f 열의 컨테이너들 중 1회 이동을 하는 컨테이너들의 수
- $C_f^{m2}$  : 반입전 선처리 수행시 f 열의 컨테이너들 중 2회 이동을 하는 컨테이너들의 수
- $C_A$  : 반입후 선처리 수행시 이동한 컨테이너들의 수
- $X_c$  : 현재 반입 중인 컨테이너의 인덱스
- $X_f^u$  : 스택의 f열에서 위에서 u번째에 적재되어 있는 컨테이너의 인덱스, 3열의 최상위 컨테이너의 인덱스가 4일 경우,  $X_3^1$ 가 4로 나타낸다.

- $S_t$  : 선처리의 과정으로, 하나의 열에서 다른 열  $t$ 로 컨테이너를 이동해야 할 때, 반입 컨테이너의 이동도 포함하여,  $t$ 열이 될 수 있는 스택들의 집합
- $D_{f,t}$  :  $f$ 열에서  $t$ 열로 하나의 컨테이너를 이동할 때, 발생하게 되는 '상단 컨테이너들의 인덱스 합'의 차이, 그 값은  $X_f^2 - X_t^1$ 가 된다.
- $L_B$  : 반입전 선처리에 대해 허용되는 최대 이동횟수 (LMF)
- $L_A$  : 반입후 선처리에 대해 허용되는 최대 이동횟수 (LML)

2.3.1. 반입전 선처리 (First-Preprocessing)

컨테이너가 반입될 때마다 일정한 조건들을 먼저 점검한 후 그 조건들이 만족할 때 반입전 선처리를 수행 하게 된다. 그리고 반입전 선처리를 수행할 경우, 반입전 선처리로 인한 컨테이너 이동을 최소화하는 스택을 선택하여 수행하게 된다. 아래는 이런 조건들과 선택 기준들을 나타내었다.

- 선처리 발생 조건 :  $X_c < \min_{f=1}^{C_s} \{X_f^1\}$
- 선처리 수행 조건 :  $C_f^{m1} + 2C_f^{m2} \leq L_B$
- 스택 선택 기준 A :  $\min_{f=1}^{C_s} \{C_f^{m1} + 2C_f^{m2}\}$
- 스택 선택 기준 B :  $\min_{f=1}^{C_s} \left\{ \sum_{t \in S} D_{f,t} \right\}$ , 단  $f \neq t$

하나의 컨테이너가 장치장에 반입될 때마다, 먼저 반입전 선처리의 발생 조건이 만족하는지를 체크한다. '선처리 발생 조건'은 반입하는 컨테이너를 어느 스택에 적재하더라도 반출시 재취급을 할 수 밖에 없는지를 점검하기 위한 것으로, 반입하는 컨테이너의 인덱스  $X_c$ 가 적재되어 있는 모든 스택들의 최상단의 인덱스보다 우선순위가 낮은 지 점검하게 된다. '선처리 발생 조건'이 만족하면, '선처리 수행 조건'을 체크하여 선처리를 제한된 이동횟수  $L_B$  이내에 완료할 수 있는지를 점검하게 된다.

두 조건이 만족되면 반입전 선처리를 수행하게 된다. 이때 수행하는 작업은, 먼저 반출시에 재취급이 발생하지 않도록 기존의 컨테이너를 이동한 후에, 반입하는 컨테이너를 적재하는 작업이다. 경우에 따라, 반입 컨테이너의 적재 후에도 컨테이너를 이동하는 작업이 추가될 수 있는데 이 이동도 반입전 선처리에 포함된다. 이를 위해 반입하는 컨테이너를 적재할 최적의 스택을 선택해야 하는데 최적의 스택을 선택할 기준은 두 가지로 구성된다. 첫째 조건인 '스택 선택 기준 A'는 반입전 선처리가 가능한 스택들 중 반입전 선처리로 발생하는 컨테이너 이동 횟수가 가장 적은 스택을 선택하기 위한 것이다. 반입전 선처리 중 발생하는 컨테이너 이동은 이동횟수를 기준으로 1회 이동과 2회 이동으로 나눌 수 있다. 1회 이동은 다른 스택의 최상단 컨테이너보다 우선순위가 높거나 같은 컨테이너들의 이동이다. 이런 컨테이너들은 다른 스택으로의 1회의 이동만으로 반출시에 재취급이 발생하지 않게 된다. 2회 이동은 다른 스택의 최상단 컨테이너보다 우선순위가 낮은 컨테이너들의 이동으로, 다른 스택으로의 이동에 원래 스택으로의 복귀 이동을 추가해서 2회 이동을 해야 하는 경우이다. 반입전 선

처리를 수행하면 대상 스택에 따라서 1회 이동만 발생하거나, 2회 이동만 발생하거나, 또는 두 가지가 동시에 발생하게 되는데 이런 이동 횟수들의 합이 최소인 스택을 선택하여야 하며, 이것이 '스택 선택 기준 A'가 된다. 두 번째 '스택 선택 기준 B'는 '스택 선택 기준 A'를 만족하는 스택이 여러 개일 경우, 반입전 선처리에 따른 이동 후의 최상단 인덱스의 합이 최소가 되는 이동을 선택하기 위한 기준이다. 이는 다음 컨테이너의 반입시 재취급 발생가능성을 줄이기에 가장 적합한 스택을 선택하기 위한 기준으로 반입전 선처리를 수행한 후의 모든 스택들의 인덱스 합이 최소가 되는 스택을 선택하는 것이다. 반입전 선처리 과정에 반입컨테이너의 적재가 포함되기 때문에 반입전 선처리가 끝나면 하나의 컨테이너 반입이 완료된다.

2.3.2. 반입후 선처리 (Later-Preprocessing)

반입후 선처리는 컨테이너의 반입이 완료된 직후에 수행하는 선처리로서 반입전 선처리처럼 일정한 조건들이 만족할 때 수행을 하게 된다. 반입후 선처리를 수행할 경우, 반입후 선처리로 인한 컨테이너 이동을 최소화하는 스택을 선택하여 수행하게 된다. 아래는 이런 조건들과 선택 기준들을 나타내었다.

- 선처리 발생 조건 :  $C_e = 0$
- 선처리 수행 조건 :  $C_A \leq L_A$
- 스택 선택 기준 :  $D_{f,t} < 0$  &  $\min_{f=1, t=1}^{C_s, C_s} \{D_{f,t}\}$ , 단  $f \neq t$

하나의 컨테이너를 반입한 직후, 재취급이 발생할 가능성을 줄이기 위하여 수행하는 것이 반입후 선처리이다. 컨테이너 반입이 진행되다보면 빈 스택이 점점 줄어들게 되는데, 빈 스택  $C_e$ 이 하나도 남지 않게 될 때부터 재취급이 발생하게 되므로 빈 스택의 수가 0이 되는 것이 '선처리 발생 조건'이 된다. '선처리 발생 조건'이 만족되는 시점에서, '선처리 수행 조건'을 체크하여 반입후 선처리를 제한된 이동횟수  $L_A$  이내에 완료할 수 있는지를 점검하게 된다. 두 조건이 만족되면 반입후 선처리를 수행하게 되는데, 반입 후 선처리로 스택 상단의 인덱스 합이 줄어드는 이동이 가능한 스택을 대상으로 ( $D_{f,t} < 0$ ) 반입후 선처리를 수행하게 된다. 이런 반입후 선처리가 가능한 스택이 여러 개 있을 경우, 줄어드는 폭이 가장 큰 스택을 선정하여 먼저 반입후 선처리를 수행하게 된다. 앞에서 언급한 것과 같이 반입전 선처리와 반입후 선처리의 수행은 같이 수행할 수도 있고, 하나씩만 수행할 수도 있다.

3. 연구방법론

3.1. 고려할 요소

여기서는 선처리 알고리즘의 수행과 관련하여 고려해야할 요소들을 알아본다. 먼저 두 선처리를 어떻게 적용하느냐에 따라 알고리즘의 성능이 달라질 수 있다. 그리고 각 선처리에 허용되는 이동 횟수에 제한을 가할 수 있는데 이들 또한 성능에 영향을 줄 수 있다. 그리고 컨테이너들에게 인덱스를 어떻게 부여할 것인가도 성능에 영향을 미칠 수 있다. 이런 요소들의

특성에 대하여 알아본다. 그리고 컨테이너 터미널 운영과 관련하여 결정할 사항들도 알아본다.

### 3.1.1. 적용하는 선처리의 종류(Kind of Preprocessing : KOP)

선처리는 기능으로 볼 때 반입전 선처리 부분과 반입후 선처리 부분으로 나누어진다. 반입전 선처리와 반입후 선처리는 특성이 다르므로 당연히 그 효과도 다르다. 선처리를 수행할 때 상황에 따라 아래와 같이 세 가지 방식으로 선처리 부분들을 실행하게 되는데, 그기에 따른 효과도 달라질 것이다.

- 1) 반입 전 선처리만 실행
- 2) 반입 후 선처리만 실행
- 3) 두 가지 선처리 모두 실행

### 3.1.2. 성능에 영향을 미치는 요소

선처리를 실행할 때 주어진 상황에 맞게 여러 가지 요소들에 변화를 주어서 실행하게 된다. 이런 요소들에 대한 자세한 사항들을 알아본다.

#### 3.1.2.1. 반입전 선처리 최대 이동횟수(Limit of Maximum-number for First-Preprocessing : LMF)

반입전 선처리는 앞에서 언급한 것처럼 이동 횟수에 제한을 둘 수 있다. ‘반입전 선처리 최대 이동횟수’는 반입전 선처리 수행에 허용되는 컨테이너 이동의 최대 횟수이다. 반입전에 수행하는 선처리의 경우, 반입되는 컨테이너가 바로 재취급 여부를 알 수 있는 상황이라, 반출시의 재취급이 발생하지 않는 경우에는 선처리가 필요 없게 된다. 반출시의 재취급이 발생할 상황일 경우에는 반입전 선처리로 재취급을 막을 수 있으므로, 반출시의 작업에 성능을 높이는 중요한 요소가 될 수 있다.

#### 3.1.2.2. 반입후 선처리 최대 이동횟수(Limit of Maximum-number for Later-Preprocessing : LML)

반입후 선처리도 앞에서 언급한 것처럼 이동 횟수에 제한을 둘 수 있다. ‘반입후 선처리 최대 이동횟수’는 반입후 선처리 수행에 허용되는 컨테이너 이동의 최대 횟수이다. 반입한 후의 선처리는 다음에 어떤 컨테이너가 반입될지 알 수 없는 상황에서 수행하는 선처리이므로, 반입후 선처리로 인한 반출시 재취급을 방지하는 효과는 반입전 선처리보다 낮을 수 있지만, 반입후 선처리의 이동횟수에 가하는 제한도 작업 성능에 변화를 주는 중요한 요소 중 하나이다.

#### 3.1.2.3. 인덱스수 (Number of Container Index : NCI)

컨테이너의 무게를 기준으로 장치장에 적재를 하게 되는데, 무게를 몇 개의 그룹으로 나누어, 같은 그룹의 컨테이너에 유사한 무게의 컨테이너에 반입과 반출을 위한 우선순위를 부여하게 된다. 이 우선순위를 인덱스라고 하며 모든 컨테이너마다 서로 다른 인덱스를 부과하여 모든 컨테이너가 다른 우선순위를 갖도록 할 수도 있다. 인덱스로 컨테이너를 그룹화하면, 그룹 당 컨테이너 수에 따라, (다르게 말하여 NCI에 따라) 적재하는 순서를 정하기 쉬워지기도 되고 어려워지기도 되니까, 반입작업이나 반출작업의 효율이 바뀌게 된다.

#### 3.1.2.4. 가득 찬 스택 제외(Exception of Full Stack : EFS)

컨테이너가 모두 찬 스택에는 추가로 적재할 수 있는 공간이 없어서 가득 찬 스택으로 이동을 할 수가 없다. 그래서 임시 저장 공간으로 활용할 수는 없다. 그러나 가득 찬 스택에 있는 컨테이너를 가득 차지 않은 스택으로 이동은 가능하기 때문에 반입전 선처리에서는 반입 컨테이너를 최종적으로 적재하는 스택으로 사용할 수가 있고, 반입후 선처리에서는 최종적으로 선처리 대상 스택으로 사용할 수가 있다. 가득 찬 스택 제외 옵션은 가득 찬 스택에 있는 컨테이너를 가득 차지 않은 스택으로 이동하는 것조차 제한을 가하는 옵션이다. 이로 인하여 성능에 어떤 영향이 미치는지를 알아보려고 한다.

### 3.1.3. 컨테이너 반입과 관련하여 결정할 사항

컨테이너 장치장 운영과 관련하여 결정하여야 할 사항이며, 이 값들이 어떻게 정해지는가에 따라 선처리 작업의 효율이 크게 영향을 받는다. 그래서 3.2.2.에서 언급하는 요소들을 분석함으로써 이들 값을 결정하는데 활용할 수 있다.

#### 3.1.3.1. 선처리와 재취급에 대한 중요도.

선처리와 재취급은 상관관계가 있다. 선처리 횟수가 늘어나면 재취급 횟수는 줄어들게 되고, 선처리 횟수가 줄어들면 재취급 횟수는 늘어난다. 그리고 ‘선처리 횟수와 재취급 횟수의 합’과 선처리 횟수도 상관관계가 있다. 선처리 횟수가 변하면 두 횟수의 합은 따라서 변하지만 선형적으로 변하지는 않는다. ‘선처리 횟수와 재취급 횟수의 합’이 커질수록 이동에 드는 비용이 증가하므로, 두 횟수의 합은 중요한 요소이다. 그리고 컨테이너 터미널의 성능에 가장 큰 영향을 미치는 것은 재취급 횟수이므로 이 또한 중요한 요소이다. 결국, 재취급의 횟수와 ‘선처리 횟수와 재취급 횟수의 합’ 중 어느 것이 중요한가를 결정해야 한다.

#### 3.1.3.2. 컨테이너 그룹 수준

인덱스를 통한 컨테이너의 그룹은 어느 수준으로 할 것인가도 결정해야 할 사항이다. 인덱스의 개수는 컨테이너 그룹의 개수를 나타내며 인덱스의 개수가 많아질수록 재취급 발생 가능성이 높아지는 것은 당연하다. 인덱스의 개수를 낮출수록 재취급 발생 가능성이 줄어들지만 선박의 안정성을 고려하면 무조건 낮출 수는 없다. 성능에 중대한 영향을 미치는 인덱스의 개수도 결정해야 할 사항이다.

## 3.2. 실험

### 3.2.1. 실험 방법

위에 언급한 요소들은 알고리즘의 성능에 직, 간접적으로 성능에 영향을 미친다. 이들이 어느 정도 영향을 줄 수 있는지 분석하기 위하여 여러 요소들을 변화시키면서 시뮬레이션을 실시한다. 이를 위하여 이 논문에서 제안하는 알고리즘을 Microsoft Visual C++을 사용하여 구현하였고, 복잡도가 높은 실험을 위하여 최대 컨테이너를 보유한 베이들을 대상으로 실험을 실시하였다. 현장에서 수용하는 최대 컨테이너 수가 12열 8단의 경우 89개, 8열 6단의 경우 43개이므로 이러한 베이들로 1,000개씩 무작위로 생성하여 실험을 실시하였다. 여러

가지의 요소가 서로 영향을 미칠 수 있기 때문에 하나의 요소에 대하여 실험을 할 경우 다른 요소들의 값은 단순화 시켜서 실험을 실시하였다. 각 시뮬레이션의 결과로서 평균 선처리수, 평균 재취급수 또는 이들의 합을 필요에 따라서 일부만 또는 모두를 나타내었으며 그 값들을 비교함으로써 각 요소들이 알고리즘의 성능에 미치는 영향을 알아보았다. 다음은 실험을 위하여 설정한 요소들을 나타내었다. 표현을 간단하게 하기 위하여 위의 여러 요소들을 위의 설명의 제목에 나타낸 약어로 표기한다.

3.2.2. 실험 조건

시뮬레이션에서 사용한 시험 조건들을 아래 <Table 1>에 나타내었으며 시험 결과들은 4.2 실험 결과 분석에서 설명하였다. 실험조건을 <Table 1>로 나타낸 것은 실험조건이 상당히 많아서 각 실험에 대한 설명을 간략하게 나타내기 위함이다. 표에서 사용한 √기호는 실험에 적용한 요소들을 나타내기 위한 것으로, Test 1 실험을 예로 들어 설명한다. Test 1 실험은 ‘반입전 선처리 최대 이동횟수’의 영향을 분석하기 위한 실험으로, 사용한 베이는 12W8H(W는 스택을 H는 단을 나타내었다)이며, 선처리는 반입전 선처리만 적용하였다. LMF는 변화시키면서, LML은 고려하지 않았다. 이는 반입후 선처리가 적용되지 않았기 때문이다. NCI는 89로 고정된 값을 사용하였으며, EFS는 적용하지 않았다는 뜻이다.

Table 1: Test conditions

Important Factors		Test							
		1	2	3	4	5	6	7	8
Bay	12W8H	√	√	√	√				
	8W6H					√	√	√	√
Preprocess	only First	√				√			
	only Later		√				√		
	simultaneously			√	√			√	√
LMF	variable	√		√		√		√	
	fixed				1				1
LML	variable		√				√		
	fixed			1	1			1	1
NCI	variable			√				√	
	fixed	89	89		89	43	43		43
EFS	apply								
	no apply	√	√	√		√	√	√	

4. 연구 결과

4.1. 단일 결과 분석

4.1.1. 반입전 선처리 최대 이동횟수 (LMF)

먼저 반입전 선처리만 적용할 경우 반입전 선처리 최대 이동횟수인 LMF를 변화시키면서 선처리 횟수가 어떻게 변하는지를 알아보았다. LMF는 반입전 선처리를 수행할 때 허용하는

최대 이동횟수를 의미하며, 그 수가 증가할수록 선처리를 더 많이 수행할 수 있게 되며 그로 인하여 부적합한 위치로 적재되는 컨테이너 수가 줄어드는 효과가 있으리라 기대된다. 그래서 재취급 횟수의 변화도 함께 나타내었다. 아래 그래프에서 선처리 횟수와 재취급 횟수 그리고 그들의 합을 나타내었는데 <Figure 3>은 12열, 8단의 베이들을 대상으로 한 실험으로 <Table 1>의 Test 1의 실험결과이며, <Figure 4>는 8열, 6단의 베이들을 대상으로 실시한 실험으로 <Table 1>의 Test 5의 실험결과이다. 그래프에서 가로축은 LMF를 세로축은 컨테이너 평균 이동횟수를 나타내었다.

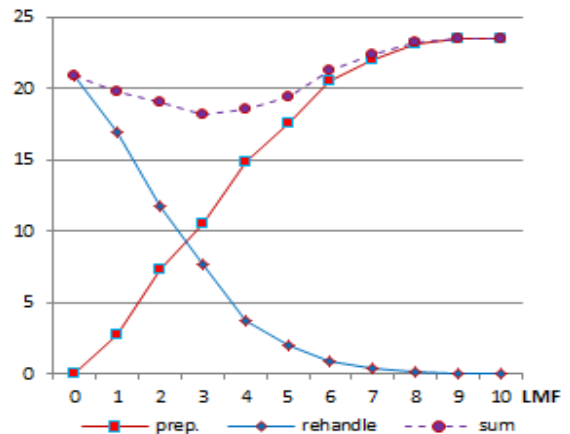


Figure 3: Performance Changes with LMF Changes

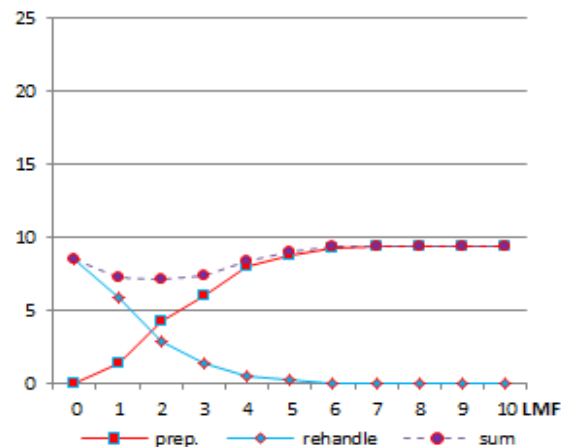


Figure 4: Performance Changes with LMF Changes

두 그래프 모두 LMF가 증가할수록 선처리 횟수는 늘어나며, 재취급 횟수는 줄어드는 것을 알 수 있다. 두 횟수의 합을 나타낸 이유는 두 횟수의 합이 전체 컨테이너의 이동, 즉 필요한 전체 작업을 의미하기 때문이다. 선처리의 횟수가 늘어나는 것은 컨테이너가 반입되는 시점에 컨테이너 이동이 늘어난다는 것이며, 이로 인해 재취급이 줄어든다는 것은 선박에 적재

하는 시점의 컨테이너 이동이 줄어든다는 것을 나타낸다. 컨테이너 터미널의 성능에서 볼 때, 컨테이너 전체이동인 두 이동의 합도 중요하지만 재취급이 줄어드는 것이 상당히 중요하다. 그리고 재취급을 완전히 없애려면 LMF를 적당한 수 이상 높이면 가능해진다. <Figure 3>의 경우 10, <Figure 4>의 경우 7이상이면 재취급 횟수가 0이 된다. 즉 재취급을 완전히 없애는 것도 가능해진다는 것이다. 그리고 그래프에서 나타난 것처럼 두 값의 합이 중요할 경우 12열 8단의 경우는 LMF가 3일 때, 8열 6단의 경우 LMF가 2일 때 가장 좋은 것을 알 수 있으며, 이 값은 베이에 따라 달라진다. 두 그래프가 비슷한 모양을 나타내고 있으나 <Figure 3>의 값이 <Figure 4>보다 큰 값을 나타내는 것은 취급하는 컨테이너의 수가 89대 43으로 차이가 있기 때문이며 이는 아래의 실험에서도 같은 현상이 반복되고 있다.

4.1.2. 반입후 선처리 최대 이동횟수 (LML)

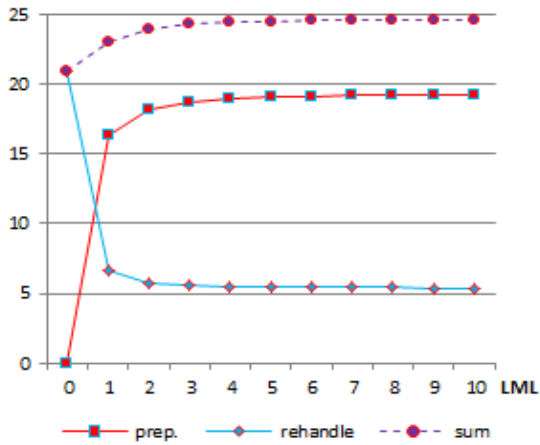


Figure 5: Performance Changes with LML Changes

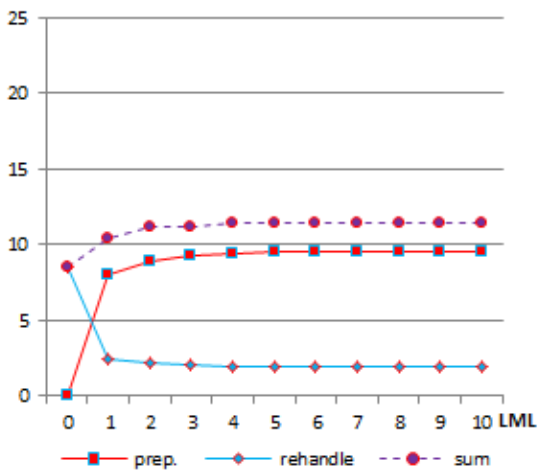


Figure 6: Performance Changes with LML Changes

다음은 반입후 선처리만 적용할 경우의 실험 결과인데, 반입후 선처리 최대 이동횟수인 LML을 변화시키면서 선처리 횟

수, 재취급 횟수, 그리고 그 합이 어떻게 변하는지를 알아보았다. LML은 반입후 선처리를 수행할 때 허용하는 최대 이동횟수를 의미하며, 그 수가 증가함에 따른 효과도 LMF의 경우와 유사하리라 짐작된다. 그 결과는 위의 두 그래프에 나타나 있는데 <Figure 5>는 12열, 8단의 베이들을 대상으로 한 실험으로 <Table 1>의 Test 2의 실험 결과이며, <Figure 6>은 8열, 6단의 베이들을 대상으로 실시한 실험으로 <Table 1>의 Test 6의 실험 결과이다. 그래프의 가로축은 LML을, 세로축은 컨테이너 평균 이동횟수를 나타내었다. 그래프에 보면 알 수 있듯이 LML이 0에서 1로 변할 때 재취급 감소 효과가 크다는 것을 알 수 있다. 그리고 LML이 2이상 증가하더라도 선처리 횟수의 증가에 따른 재취급 감소의 효과는 크지 않은 것을 알 수 있다. 그리고 선처리 횟수와 재취급 횟수의 합은 계속 증가하지만 어느 값이상이면 거의 변화가 없다는 것을 알 수 있다. 이는 LML의 영향이 제한적이라는 것을 나타낸다.

4.1.3. 인덱스 수 (NCI)

인덱스는 위에서 설명한 것처럼 반입되는 컨테이너들을 무게를 기준으로 그룹화하는 것이다. 즉 유사한 무게의 컨테이너에 같은 인덱스를 부과함으로써 컨테이너 반입작업이나 반출작업의 성능을 높일 수 있다. 인덱스의 수(NCI)를 변화시켜서 나타나는 성능의 변화를 아래의 <Figure>들에서 확인할 수 있다. <Figure 7>은 12열, 8단의 베이들을 대상으로 한 실험으로 <Table 1>의 Test 3의 결과이며, <Figure 8>은 8열, 6단의 베이들을 대상으로 실시한 실험으로 <Table 1>의 Test 7의 결과이다. 그래프의 가로축은 인덱스 값, 세로축은 컨테이너 이동횟수(선처리 횟수와 재취급 횟수의 합)를 나타내며, 4개의 선들은 LML을 1로 정해둔 상태에서, LMF=1, LMF=4, LMF=7, LMF=10인 경우를 나타내었다. 그래프에서 알 수 있듯이 NCI가 증가할수록 컨테이너의 이동수는 높아지는 것을 알 수 있는데, 이는 거꾸로 말하면 인덱스가 감소할수록, 즉 그룹 당 컨테이너 수가 많아질수록 성능이 높아진다는 것이다. <Figure 7>에서 제일 위의 선인 LMF=10인 경우, NCI가 5일 때 이동횟수는 3.117인데 NCI가 최대인 89일 때 이동횟수는 31.722로써, 그 비율은 1/10보다 낮은 것을 볼 수 있다. 인덱스를 적당하게 활용함으로써 성능을 높일 수 있다는 것이다.

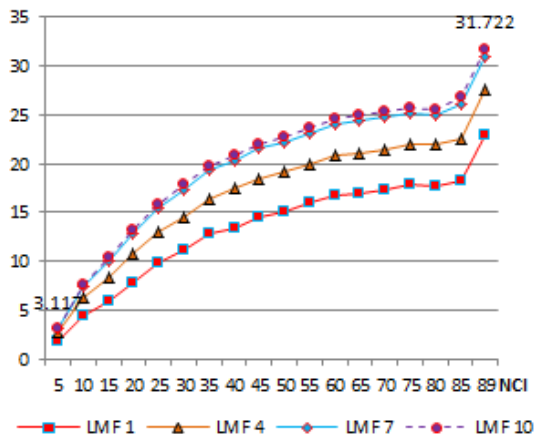


Figure 7: Performance Changes with NCI Changes

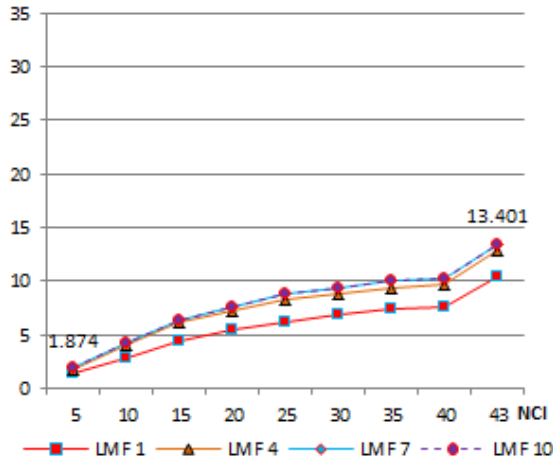


Figure 8: Performance Changes with NCI Changes

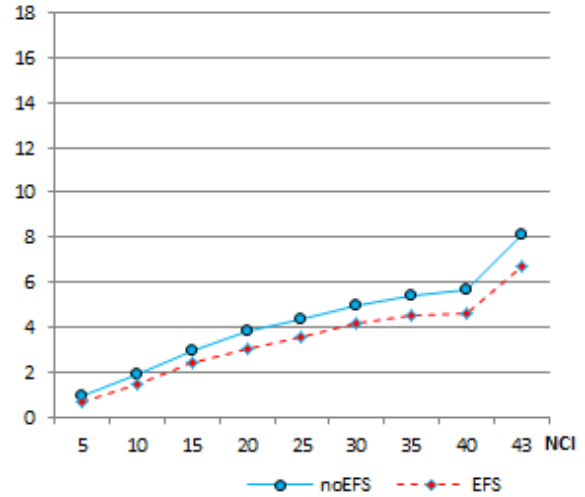


Figure 10: Performance Changes with EFS or noEFS

4.1.4. 가득 찬 스택 제외 (EFS)

다음은 가득 찬 스택을 선처리 작업의 대상에서 제외함으로써 생기는 성능의 변화를 실험한 것이다. 가득 찬 스택은 안전한 상태로 간주하여 작업의 대상에서 제외시키는 것인데 이렇게 하는 것이 성능에 얼마만한 영향을 미치는 가를 보기 위한 실험으로 아래의 <Figure>들에서 확인할 수 있다. <Figure 9>는 12열, 8단의 베이들을 대상으로 한 실험으로 위의 <table 1>의 Test 4의 결과이며, <Figure 10>은 8열, 6단의 베이들을 대상으로 실시한 실험으로 위의 <table 1>의 Test 8의 결과이다. <Table 1>의 Test 4와 Test 8의 EFS항목이 비어 있는 것은 이 실험이 EFS를 시험하는 것이기 때문이다. 그래프의 가로축은 NCI, 세로축은 선처리로 인한 컨테이너의 이동횟수를 나타내며, NCI가 변할 때 가득 찬 스택을 제외할 경우(EFS)와 제외하지 않을 경우(noEFS)의 그래프를 나타내고 있다. 그림에서 보듯이 가득 찬 스택을 제외하는 경우에 효과가 크다는 것을 볼 수 있다. 즉, 가득 찬 스택을 이동의 대상에서 제외하는 것은 이동 가능성을 줄이는 효과가 있다고 볼 수 있겠다.

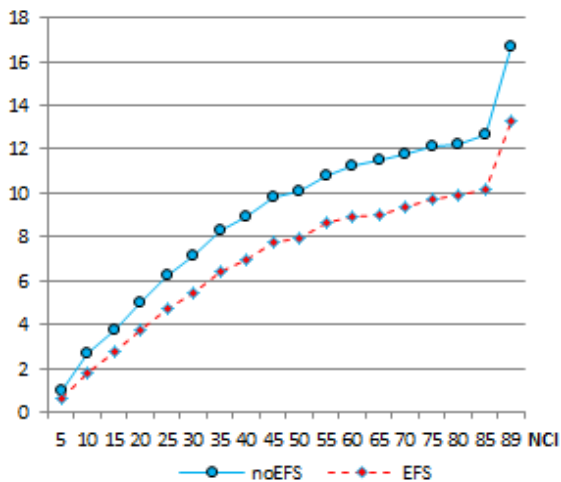


Figure 9: Performance Changes with EFS or noEFS

4.2. 연관 결과 분석

4.2.1. 반입전 선처리 최대 이동횟수 (LMF)와 반입후 선처리 최대 이동횟수 (LML)

4.1.1.을 보면 LMF를 증가하면 재취급의 횟수가 꾸준히 감소하는 것을 알 수 있다. LML를 증가할 경우는 0에서 1이 될 때는 상당한 감소효과가 있지만 그 이상으로 증가할 경우 감소효과가 LMF에 비하여 낮다는 것을 4.1.2.에서 알 수 있다. 이 두 결과를 함께 고려한다면 선처리 전체 횟수에 제한이 있을 경우, LMF와 LML의 값에 대하여 최적의 수를 선택할 수 있을 것이다. 즉 만약 선처리 작업을 1회만 하여야 한다면, LML을 1로 설정하는 것이, 다시 말해 반입후 선처리를 1회 허용하는 것이, 효과적이라는 것을 알 수 있다. 그러나 그래프에서 나타났듯이 그 이상의 횟수로 선처리 작업을 해야 한다면 LML은 1로 그 나머지는 LMF에 적용하는 것이 효과적이 될 것이다. 그리고 4.1.1에서 언급한 것처럼 LMF를 충분히 높이면 재취급을 완전히 없앨 수 있다는 것은 의미 있는 결과이다.

5. 연구결과 토론 및 시사점

5.1. 연구요약 및 시사점

컨테이너 터미널에서 컨테이너를 반입하고, 반출하는 작업에서 재취급은 성능의 주요한 저해요소라고 할 수 있으며, 이를 줄이는 방법에 관하여 많은 연구가 있어 왔다. 재취급을 줄이는 시점을 기준으로 본다면 모든 컨테이너의 반입이 완료된 후 재정돈을 실시함으로써 재취급을 줄이는 방법과 이 논문에서 제시하는 방법처럼 반입이 진행되는 중에 재취급을 줄이는 방법으로 나눌 수 있을 것이다. 본 논문에서는 반입이 진행되는 중간에 선처리를 실행함으로써 재취급을 줄이는 방법으로 반입전 선처리와 반입후 선처리라는 기능을 가진 선처리 알고리즘을 제시하고, 이 방안에서 영향을 미칠 수 있는 다양한 요소들에 대하여 성능을 분석하여 보았다. 4.1.1.과 4.1.2.



에서 알 수 있듯이 재취급을 줄이려면, 선처리 최대 이동횟수를 높이는 것이 상당히 효과적이다. 그리고 컨테이너 이동횟수 전체를 고려하면, <Figure 3>에서는 LMF가 3일 때, <Figure 4>에서는 LMF가 2일 때가 가장 효과적이었다. 4.1.3.을 통해서 NCI는 줄수록 효과적이라는 것을 알 수 있었다. 그러나 NCI를 무작정 줄이는 것은 현실적으로 어려움이 발생한다. NCI를 줄이는 것은 선박에 컨테이너를 적재했을 때 선박의 안정성에는 악영향을 미치기 때문이다. 4.1.4.에서 알아본 것처럼 가득 찬 스택을 제외하는 방안은 성능에 긍정적인 요소를 미친다. 이는 다른 부정적인 요소 없이 성능에 긍정적인 요소만 있는 것으로 보인다. 그리고 주어진 상황에 맞게 요소를 결정해서 반입 작업시 선처리를 수행하는 것이 필요할 것으로 보인다. 예를 들면 재취급이 중요하고, 선박의 안정성을 고려할 경우 최소로 가능한 NCI가 5라고 가정하면, LMF와 LML을 최대한 높이고, NCI 최소인 5로 값으로 설정한 후 EFS로 반입 작업을 하면 가장 성능이 높은 작업이 될 수 있겠다. 그리고 NCI뿐 아니라 LMF와 LML의 합이 예를 들어 4로 제한되어 있다면, LML은 4.2.1.에서 언급한대로 1로 선택한 후 그 나머지를 LMF에 할당하는 방법으로 설정한 후 작업하여야 할 것이다. 물론 전체 컨테이너 이동수가 중요한 상황일 경우에는 허용되는 범위 내에서 LMF는 베이의 열과 단에 따라 정해지는 최적의 값으로 설정하고, LML은 허용된다면 1로 설정한 후, 주어진 NCI에서 작업을 하여야 할 것이다. NCI는 미리 결정되어야 하지만 나머지 값들은 컨테이너가 반입되는 시점에 주어지는 상황에 맞는 값으로 설정하는 것이 필요할 것이고 그때는 이 실험의 결과들을 활용하는 것이 효과적일 것으로 생각된다.

## 5.2. 연구의 한계 및 향후 연구 방향

본 연구는 선처리 알고리즘을 제안하면서, 알고리즘의 여러 요소들이 컨테이너 터미널의 성능에 미치는 영향에 대하여 알아보았다는 점에서 의의를 가지지만 몇 가지 한계점이 있다. 영향을 미치는 요소들을 두 가지 이상 동시에 변화시키는 실험을 하지 못했다는 것이다. 하나의 요소를 변화시키면서 성능에 미치는 영향은 알아보았으나, 두 가지 이상을 동시에 변화시킬 경우, 너무 많은 경우의 수가 발생하고 그 영향의 분석이 너무 복잡하기 때문이었다. 그리고 시험대상으로 선택한 베이의 종류도 2가지로 너무 단순한 면이 있었다. 각 베이의 종류 당 1,000가지의 베이를 대상으로 실험을 하였으나, 베이의 종류를 더 다양하게 하지 못한 점이 있다. 그래서 향후 연구로 시험 대상 베이와 영향을 미치는 요소들을 다양하게 변화시키면서 그들 상호간의 영향도 알아보는 연구가 필요하리라 생각한다. 향후 본 연구와 관련지어서 추가로 연구해 볼 수 있는 분야로, 선처리와 재정돈을 연계하여 전체적인 성능을 높이는 방안에 대한 연구가 있을 것이다. 그동안은 선처리 작업은 반입 중에 실시를 하는 작업이고 재정돈 작업은 반입 작업이 끝난 후 실시를 하는 작업이므로 두 작업을 별도로 연구해 왔었다. 그 두 작업을 연계하여 수행하는 것이 각각의 작업을 나누어서 할 때와 효과가 다를 것으로 생각된다. 즉, 선처리 작업을 수행할 때 재취급이 발생될 것이 예상될 경우, 재정돈을 할 예정인지 아닌지에 따라, 지금과 다른 선처리를 수행함으로써 선처리와 재취급의 전체 성능을 높일 수 있을 것으로 생각되기 때문이다.

## References

- Bae, J. W., Park, Y. M., & Kim, K. H. (2006). Assignment and Operation Sequencing for Remarshalling of a Vertical Yard Block in Automated Container Terminals. *Journal of Korean Navigation and Port Research*, 30(6), 457-464.
- Caserta, M., Voß, S., & Sniedovich, M. (2011). Applying the Corridor Method to a Block Relocation Problem. *OR Spectrum*, 33, 915-929.
- Ha, B. H., & Kim, S. S. (2012). A\* Algorithm for Optimal Intra-bay Container Pre-marshalling Plan, *Journal of the Korean Institute of Industrial Engineers*, 38(2), 157-172.
- Han, X., Lu, Z., & Xi, L. (2010). A Proactive Approach for Simultaneous Berth and Quay Crane Scheduling Problem with Stochastic Handling Time. *European Journal of Operational Research*, 207(3), 1327-1340.
- Hirashima, Y., Ishikawa, N., & Takeda, K. (2006). A New Reinforcement Learning for Group-Based Marshalling Plan considering Desired Layout of Containers in Port Terminals. *Proc. IEEE Conf. Networking, Sensing and control*, 670-675.
- Imai, A., Sasaki, K., Nishimura, E., & Papadimitriou, S. (2006). Multi objective Simultaneous Stowage and Load Planning for a Container Ship with Container Rehandle in Yard Stacks. *European Journal of Operational Research*, 171(2), 373-389.
- Jeong, Y. H., Kim, K. H., Woo, Y. J., & Seo, B. H. (2012). A Simulation Study on a Workload based Operation Planning Method in Container Terminal. *Industrial Engineering and Management Systems*, 11(1), 103-113.
- Kang, J. H., Oh, M. S., Ru, K. R., & Kim, K. H. (2004). Method of Inbound Container Positioning for Minimal Rehandling Considering Weight. *Proceedings of The 2004 KIIS Fall Conference*, 271-278.
- Kang, J. H., Oh, M. S., Ru, K. R., & Kim, K. H. (2005). Sequencing Container Moves for Intra-Block Remarshalling in a Container Terminal Yard. *Journal of the Korean Institute of Industrial Engineers*, 29(1), 83-90.
- Kang, J., Ryu, K. R., & Kim, K. H. (2006). Deriving Stacking Strategies for Export Containers with Uncertain Weight Information. *Journal of Intelligent Manufacturing*, 17, 399-410.
- Kim, J. E., Park, K. Y., Park, T. J., & Ryu, K. R. (2009). Container Selecting Methods for Remarshalling Considering Restricted Idle Time of Crane in an Automated Container Terminal. *Journal of Korean Navigation and Port Research*, 33(10), 715-722.
- Kim, K. H. (2007). Decision making Problems for the Operation of Container Terminal. *Journal of the Korean Institute of Industrial Engineers*, 33(3), 290-302.
- Kim, K. H., Kim, K. Y., & Ko, C. S. (1997). Load Scheduling Using a Genetic Algorithm in Port Container Terminals. *Journal of the Korean Institute of Industrial*

- Engineers*, 23(4), 645-660.
- Kim, K. H., Park, Y. M., & Ryu, K.-R. (2000). Deriving Decision Rules to Locate Export Containers in Container Yards. *European Journal of Operational Research*, 124, 89-101.
- Kim, K. H., & Hong, G. P. (2006). A Heuristic Rule for Relocating Blocks. *Computers and Operations Research*, 33, 940-954.
- Kim, K. H., & Park, K. T. (2003). A Note on a Dynamic Space-allocation Method for Outbound Containers. *European Journal of Operational Research*, 148, 92-101.
- Kim T. K., Yang, Y. J., Bae, A. K., & Ryu, K. R. (2014). Optimization of Dispatching Strategies for Stacking Cranes Including Remarshalling Jobs. *Journal of Korean Navigation and Port Research*, 38(2), 155-162.
- Lee, S. W. (2011). A Genetic Algorithm for the Container Pick-Up Problem. *IE Interface*, 24(4), 362-372.
- Lee, Y., & Lee Y.-J. (2010). A Heuristic for Retrieving Containers from a Yard. *Computers and Operations Research*, 37, 1139-1147.
- Mohammad B. A., Nima Safaei, B., Nikbakhsh, J. (2009). A genetic algorithm to solve the storage space allocation problem in a container terminal. *Computers & Industrial Engineering*, 56(1), 44-52.
- Nilsson, N. J. (1998). *Artificial Intelligence: A New Synthesis* (pp. 142-160). San Francisco, CA: Morgan Kaufmann Publishers, Inc.
- Oh, M. S., Kwang, J. H., Yu, K. R., & Kim, K. H. (2005). A Heuristic Approach to Scheduling Multiple Cranes for Intra-Block Remarshalling. *Journal of Korean Navigation and Port Research*, 29(5), 447-455.
- Park, K. Y., Park, T. J., & Ryu, K. R. (2010). Iterative Container Reselection Methods for Remarshaling in a Container Terminal. *Journal of Korean Navigation and Port Research*, 34(6), 503-509.
- Park, Y. K., & Kwak, K. S. (2011). Export container preprocessing method to decrease the number of rehandling in container terminal. *Journal of Korean Navigation and Port Research*, 35(1), 77-82.
- Park, Y. K. (2016). Comparison fo Algorithm based on the Container Remarshalling Efficiency Factor in Port Distribution. *Journal of Distribution Science*, 14(5), 107-114.
- Wan, Y. W., Liu, J., & Tsai, P. C. (2009). The Assignment of Storage Locations to Containers for a Container Stack. *Naval Research Logistics*, 56, 699-713.
- Zhang, C., Liu, J., Wan, Y. W., Murty, K. G., & Linn, R. J. (2003). Storage Space-Allocation in Container Terminals. *Transportation Research Part B*, 37, 883-903.