

# 복수 트랜스퍼 크레인을 활용하는 블록 내 재정돈 계획 탐색

(Searching for an Intra-block Remarshalling Plan for Multiple Transfer Cranes)

오 명 섭 <sup>†</sup>      강 재 호 <sup>\*\*</sup>      류 광 렬 <sup>\*\*\*</sup>      김 갑 환 <sup>\*\*\*\*</sup>  
(Myung-Seob Oh)    (Jaeho Kang)    (Kwang Ryel Ryu)    (Kap Hwan Kim)

**요 약** 본 논문에서는 담금질 알고리즘을 이용하여 블록 내 재정돈 계획을 수립하는 방안을 제안한다. 블록 내 재정돈이란 컨테이너를 선박에 싣는 적하 작업을 효율적으로 수행하기 위하여 하나의 장치장 블록 여기저기에 흩어져 있는 대상 컨테이너들을 적하 순서에 맞춰 몇 개의 베이로 모으는 작업이다. 유효하면서 좋은 재정돈 계획을 수립하기 위해서는 적하 순서를 충분히 고려하는 동시에 재정돈에 소요되는 시간을 가능한 줄여야 한다. 본 제안 방안은 대상 컨테이너들의 목표 위치를 탐색할 대상으로 두어 해로 표현하고, 대상 컨테이너들의 초기 위치, 목표 위치 및 적하 순서를 함께 고려하여 컨테이너 이동을 위한 부분 순서를 생성한다. 생성한 부분 순서의 제약하에 트랜스퍼 크레인간의 간섭을 고려하여 구체적인 재정돈 계획을 수립하고 그 예상 작업 시간으로 해를 평가한다. 시뮬레이션 실험 결과 본 제안 방안이 다양한 환경에서 유효하면서 효율적인 재정돈 계획을 수립할 수 있음을 확인하였다.

**키워드** : 블록 내 재정돈, 트랜스퍼 크레인 작업 계획, 컨테이너 터미널, 담금질 알고리즘, 최적화, 탐색

**Abstract** This paper applies simulated annealing algorithm to the problem of generating a plan for intra-block remarshalling with multiple transfer cranes. Intra-block remarshalling refers to the task of rearranging containers scattered around within a block into certain designated target areas of the block so that they can be efficiently loaded onto a ship. In generating a remarshalling plan, the predetermined container loading sequence should be considered carefully to avoid re-handlings that may delay the loading operations. In addition, the required time for the remarshalling operation itself should be minimized. A candidate solution in our search space specifies target locations of the containers to be rearranged. A candidate solution is evaluated by deriving a container moving plan and estimating the time needed to execute the plan using two cranes with minimum interference. Simulation experiments have shown that our method can generate efficient remarshalling plans in various situations.

**Key words** : Intra-block remarshalling, Scheduling transfer cranes, Container terminal, Simulated annealing, Optimization, Search

## 1. 서 론

컨테이너 터미널의 생산성은 선박에서 컨테이너를 내리는 양하(unloading) 작업과 선박에 컨테이너를 싣는 적하(loading) 작업의 효율에 큰 영향을 받는다. 양하와 적하 작업을 효율적으로 수행하기 위해서는 작업 중에 발생하는 지연이 최소화되어야 한다. 지연이 발생하는 빈도가 상대적으로 낮은 양하 작업과는 달리, 적하 작업은 장치장(yard)에 쌓여 있는 대상 컨테이너들의 위치에 따라 지연 발생 정도가 크게 달라진다. 예를 들어 연이어 적하할 컨테이너들이 장치장 여기저기에 흩어져 있다면, 이들을 장치장에서 꺼내어 내부 트럭(yard

· 이 논문은 교육인적자원부 지방연구중심대학육성사업(차세대물류IT기술연구사업단)의 지원에 의하여 연구되었음

<sup>†</sup> 비 회 원 : 부산대학교 컴퓨터공학과  
oms1226@pusan.ac.kr

<sup>\*\*</sup> 정 회 원 : 부산대학교 컴퓨터공학과 박사후연수연구원  
jhkang@pusan.ac.kr

<sup>\*\*\*</sup> 종신회원 : 부산대학교 컴퓨터공학과 교수  
kr Ryu@pusan.ac.kr

<sup>\*\*\*\*</sup> 비 회 원 : 부산대학교 산업공학과 교수  
kapkim@pusan.ac.kr

논문접수 : 2006년 2월 6일

심사완료 : 2006년 5월 1일

truck)에 전달하는 트랜스퍼 크레인(transfer crane)의 이동이 빈번해진다. 또한 지금 적하할 컨테이너의 상단에 이후에 적하할 컨테이너들이 쌓여 있다면, 해당 컨테이너를 꺼내기 위하여 상단에 있는 컨테이너들을 임시로 옮겨야 하는 재취급(re-handling, reshuffling)이 발생한다. 트랜스퍼 크레인의 빈번한 이동과 재취급 발생은 적하 작업을 지연시키는 주된 요인들이다.

이와 같은 원인들로 인해 발생하는 적하 작업의 지연을 최소화하기 위해서는 적하 작업이 시작되기 전에 블록 내에 흩어져 있는 대상 컨테이너들을 적하 순서에 맞춰 재취급이 발생하지 않도록 모으는 작업이 필요하다. 이 작업을 블록 내 컨테이너 재정돈(intra-block remarshalling; 이하 재정돈으로 칭함)이라 한다. 근래에 들어 컨테이너 터미널 생산성 향상을 위하여 자동화 트랜스퍼 크레인(automated transfer crane)을 도입하는 추세이므로, 추가의 인력 투입 없이 여유 시간에 재정돈을 자동으로 수행할 수 있는 여건이 마련되어 가고 있다. 이에 따라 효율적인 재정돈 계획을 수립하는 방안에 대한 관심도 점차 높아지고 있다.

재정돈 계획을 수립하기 위해서는 먼저 좋은 재정돈 계획의 요건들을 정리할 필요가 있다. 좋은 재정돈 계획은 크게 두 가지 특징으로 요약될 수 있다. 첫 번째는 적하 작업 시 재취급에 의한 지연이 발생하지 않도록 컨테이너들을 모을 수 있어야 한다. 대상 컨테이너들을 모으게 되면 적하 시 트랜스퍼 크레인의 과도한 이동에 의한 지연은 해소되므로, 적하 작업 지연의 상당 부분은 재취급 때문에 발생하게 된다. 따라서, 사전에 계획된 적하 순서를 충분히 고려하여 재정돈 계획을 수립하여야 한다. 두 번째는 재정돈에 소요되는 시간이 최소화되어야 한다. 재정돈 소요 시간을 줄이기 위해서는 각각의 대상 컨테이너를 목표 위치로 옮길 때 중간에 일시 임시 위치에 놓는 일 없이 한 번에 옮기는 것이 바람직하다. 재정돈 과정에서 임시로 컨테이너를 옮겨 두었다가 이후에 다시 옮기는 작업, 즉 재취급이 발생하면 재정돈 작업의 효율이 저하될 수 있기 때문이다. 또한 대상 블록 내에 설치된 트랜스퍼 크레인이 여럿인 경우 이들을 함께 활용하는 계획을 수립하는 것이 재정돈 시간 단축에 보다 유리하다. 이 경우 트랜스퍼 크레인간에는 상호 접근하여 작업하는 것이 불가능한 최소 간섭 거리가 있으므로 이를 고려하여 재정돈 계획을 수립하여야 한다.

본 논문에서는 담금질 알고리즘(simulated annealing [1])을 이용하여 복수의 트랜스퍼 크레인을 활용하는 재정돈 계획을 수립하는 방안을 제안한다. 본 제안 방안은 흩어져 있는 대상 컨테이너들을 옮겨놓을 목표 위치를 탐색의 대상으로 삼는다. 따라서 대상 컨테이너들의 목표 위치 집합을 하나의 해로 표현한다. 해로 표현된 대

상 컨테이너들의 목표 위치를 사전에 미리 주어진 대상 컨테이너들의 초기 위치 및 적하 순서와 통합적으로 고려하면 재정돈 시 재취급이 발생하지 않는 컨테이너 이동 부분 순서를 생성할 수 있다. 생성된 부분 순서는 적하 시 재취급이 발생하지 않으며, 재정돈 시에도 재취급이 발생하지 않는 컨테이너 이동 순서들의 집합이다. 부분 순서에 의한 이동 순서상의 제약 하에서 트랜스퍼 크레인간의 간섭을 고려하여 재정돈 소요 시간을 가능한 줄일 수 있도록 각 트랜스퍼 크레인이 옮길 컨테이너들과 그 순서를 결정하여 실행 가능한 구체적인 재정돈 계획을 수립한다. 수립된 재정돈 계획의 예상 작업 소요 시간으로 해를 평가한다. 현재해에서 일부 컨테이너들의 목표 위치를 변경하면, 현재해와 조금 다른 이웃해를 생성할 수 있다. 본 연구에서는 두 컨테이너의 목표 위치를 맞추어 이웃해를 생성한다. 생성한 이웃해를 이용하여 현재해로부터 도출한 부분 순서와 조금 다른 부분 순서를 얻을 수 있다. 담금질 알고리즘은 이웃해로 구체적인 재정돈 계획을 수립하여 평가해보고, 조건이 만족되면 현재해를 이웃해로 치환한다. 이러한 과정을 반복하여 탐색을 진행한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서 본 연구의 배경이 되는 컨테이너 터미널과 블록 내 재정돈 문제에 대하여 자세히 소개하고, 관련된 연구들을 정리한다. 3장에서는 초기해 생성 방안, 부분 순서를 도출하는 방안, 그리고 현재해로부터 이웃해를 생성하는 방안에 대하여 기술한다. 또한 도출된 부분 순서의 제약하에서 트랜스퍼 크레인간 간섭을 고려하여 실행 가능한 재정돈 계획을 수립하는 방안도 함께 소개한다. 4장에서는 제안한 방안을 적용한 실험 결과를 정리하여 분석하고, 마지막 5장에서 결론과 향후 연구로 매듭을 짓는다.

## 2. 컨테이너 터미널의 블록 내 재정돈 문제

본 장에서는 본 논문의 연구 배경이 되는 컨테이너 터미널에 대하여 소개하고, 대상 문제인 블록 내 재정돈 문제를 정의한 다음 관련된 연구들을 정리한다.

### 2.1 컨테이너 터미널

그림 1은 컨테이너 터미널의 구조를 간략하게 보이고 있다. 컨테이너 터미널은 크게 해측의 선석(berth)과 육측의 장치장으로 구성된다. 선석에는 선박으로부터 수입 컨테이너를 내리는 양하 작업과 선박에 수출 컨테이너를 싣는 적하 작업을 수행하는 안벽 크레인(quay crane)들이 있다. 양하된 컨테이너들과 적하할 컨테이너들은 터미널 내부의 장치장이라는 공간에 임시로 적재되어 이후 작업에 대기하는데, 그림에서의 장치장은 선석과 수평으로 배치되어 있는 여러 블록들로 이루어져 있다. 각 블록에는 컨테이너를 받아 장치장에 장치시키는 반입

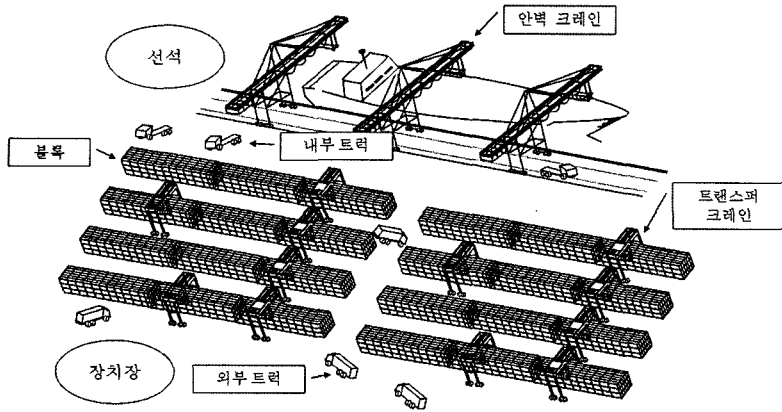


그림 1 컨테이너 터미널

작업과 컨테이너를 장치장에서 내보내는 반출 작업을 수행하는 트랜스퍼 크레인들이 있다. 선석과 장치장 사이의 컨테이너 운송은 내부 트럭이 담당하며, 터미널 외부와 장치장 사이의 컨테이너 운송은 외부 트럭이 수행한다.

컨테이너 터미널의 생산성을 높이기 위해서는 관련된 장비들이 지연 없이 작업을 처리하여야 하는데, 특히 선박 체류 시간과 직결되는 안벽 크레인들의 작업 지연을 최소화하는 것이 무엇보다도 중요하다. 적하 작업은 미리 계획된 순서대로 컨테이너들을 장치장에서 반출하여 선박에 싣는 작업이므로, 적하 대상 컨테이너들의 장치장 내 위치가 작업의 효율에 큰 영향을 준다. 적하 순서상 연속되는 컨테이너들이 장치장 여기저기에 흩어져 있거나, 지금 적하할 컨테이너를 반출하기 위하여 상단의 컨테이너들을 임시로 옮겨야 하는 재취급이 발생하면 장치장에서 작업이 지연되어 적하 작업 전체의 효율이 떨어지게 된다. 따라서 적하 대상이 되는 컨테이너가 장치장으로 반입되는 시점에 적하 작업을 고려하여 장치할 위치를 적절히 결정하는 것이 이상적이다. 수출 컨테이너가 터미널에 반입되는 시점에는 해당 컨테이너를 실을 선박, 목적항, 무게와 같은 정보가 함께 통지되므로, 터미널에서는 사전에 장치장 공간 계획을 수립하여 동일한 종류(규격, 선박, 목적항, 무게 등을 고려하여 구분)의 컨테이너들을 지정한 공간에 모아 장치함으로써 트랜스퍼 크레인의 불필요한 이동을 최소화하고자 한다.<sup>1)</sup> 하지만 환적(trans-shipment)<sup>2)</sup> 화물의 처리, 장치

할 공간의 부족, 컨테이너 정보의 미비<sup>3)</sup>, 그리고 트랜스퍼 크레인이 현재 수행하고 있는 다른 작업들에 대한 지연 최소화 등의 이유로 항상 적하 작업에 유리하도록 컨테이너들을 장치하지 못할 수 있다. 더구나 적하 계획은 대부분의 대상 컨테이너들이 장치장에 반입된 이후에 수립되므로 컨테이너 반입 시점에 장치 위치를 잘 결정하여 적하 작업의 효율을 증대시키는 데는 그 한계가 있다.

### 2.2 블록 내 재정돈 문제

그림 2는 본 연구에서 대상으로 하는 블록 내 재정돈 문제의 간단한 예를 보이고 있다. 하나의 장치장 블록은 여러 개의 베이로 구성되는데, 그림에서는 총 14개의 베이를 가진 블록을 예로 보이고 있다. 하나의 베이는 여러 단과 열로 구성된다. 그림에서의 베이는 3단 3열 구조를 가지고 있다. 재정돈하기 전에 대상 컨테이너들이 위치한 베이를 원천 베이(source bay)라 부르며, 컨테이너들을 재정돈하여 모을 베이를 목표 베이(target bay)라 부른다. 본 연구에서는 재정돈 수행 이전에 목표 베이들이 비워져 있다고 가정하였다. 그림에서는 두 개의 원천 베이  $S_1$ 과  $S_2$ 에 대상 컨테이너들이 쌓여져 있으며, 각 대상 컨테이너를 어느 목표 베이로 옮길 지와 해당 목표 베이 내에서의 적하 순서를 각각 문자와 숫자로 표시하고 있다. 예를 들어,  $S_1$ 에 장치된 컨테이너 7W의 경우  $T_{70}$ 로 옮겨져야 하는 컨테이너들 중에서 7번째로 적하 될 컨테이너이다.

그림 3은 재정돈 계획이 만족시켜야 하는 두 가지 제약 조건을 나열하고 있다[4]. 제약 조건 1은 재정돈 후 각 목표 베이에서 재취급 없이 컨테이너들을 반출할 수

1) 목적항에서 효율적으로 양하하기 위하여 동일 종류의 컨테이너들을 모아 연달아 적하할 수 있도록 적하 계획이 수립될 가능성이 높다.  
 2) 하나의 선박에서 양하된 후 해당 컨테이너 터미널 또는 인접 컨테이너 터미널의 다른 선박에 적하되는 컨테이너 화물을 의미한다. 이들 컨테이너들은 대부분 양하 작업을 가장 효율적으로 수행할 수 있는 위치에 장치된다.

3) 선박의 균형을 위하여 수출 컨테이너는 무게 정보를 고려하여 장치장 내 위치가 결정된다. 하지만 무게 정보가 부정확하게 전달되는 경우가 빈번하다[2,3].

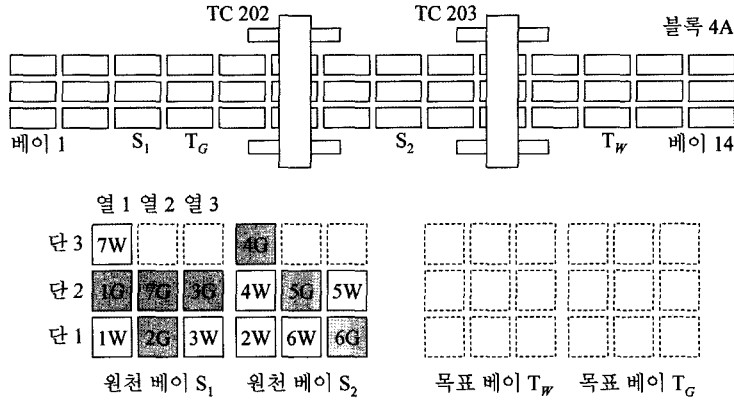


그림 2 블록 내 재정돈 문제의 예

**계약 조건 1:** 적하 작업을 위한 제약 조건  
 재정돈 이후, 각 대상 컨테이너는 재취급 없이 적하 될 수 있어야 한다.

**계약 조건 2:** 재정돈을 위한 제약 조건  
 재정돈 동안 각 대상 컨테이너는 재취급 없이 한 번에 목표 베이로 옮겨져야 한다.

그림 3 블록 내 재정돈을 위한 제약 조건들

있어야 한다는 조건으로, 적하 순서가 빠른 컨테이너가 적하 순서가 늦은 컨테이너 아래에 장치되지 않으면 만족시킬 수 있다. 이 제약 조건을 지키지 않고 재정돈을 수행하면 적하 작업 시 재취급이 발생하여 적하 작업의 효율이 떨어지게 된다. 제약 조건 2는 재정돈 중에 각 대상 컨테이너를 재취급 없이 한 번의 이동으로 원천 베이에서 목표 베이로 이동시켜야 한다는 조건으로, 이는 재정돈에 소요되는 시간을 줄이기 위하여 부가한 제약 조건이다. 재정돈 작업 시에 재취급이 발생하면 트랜스퍼 크레인(이하 크레인)이 수행하여야 하는 부가적인 작업이 발생하므로 재정돈 소요 시간이 늘어날 수 있다. 본 연구에서는 이 두 제약 조건을 지키면서 작업 소요 시간이 가능한 짧은 계획을 수립하고자 한다. 둘 이상의 크레인을 재정돈에 활용하는 경우 이들간에는 물리적 간섭에 의하여 동시에 작업이 불가능한 거리가 있으므로 이를 고려하여 재정돈 계획을 수립하여야 한다.

**2.3 관련 연구**

블록 내 재정돈과 관련한 연구로는 선석에 수평으로 배치된 블록을 대상으로 한 연구[4-6]와 수직으로 배치된 블록을 대상으로 한 연구[7]가 수행된 바 있다. 선석에 수평으로 배치된 블록을 대상으로 한 연구로, Kim과 Bae는 하나의 블록에 흩어져 있는 컨테이너들의 초기 위치와 해당 컨테이너들을 실을 선박의 공간 정보(ship profile)를 함께 고려하여 크레인의 움직임을 최소화하면서 컨테이너들을 모으는 방안을 제안하였다[5]. 이 연구

는 먼저 이동시킬 컨테이너의 수를 최소화하기 위하여 베이 정합(bay matching) 과정을 통하여 어떤 종류의 컨테이너들을 어느 베이에 모을 것인지 결정한다. 그리고, 컨테이너를 옮기는 크레인 작업 계획은 순회 판매자 문제(traveling salesman problem)로 변환하여 수립하는 방안을 제안하였다. 이 연구는 적하 계획이 수립되기 이전 단계에서 재정돈하는 것을 가정하였으므로, 제안된 방안은 각 컨테이너의 베이 내 구체적인 위치와 적하 순서상의 선후 관계를 고려하지 않았다. 이에 비해 본 연구에서 제시하는 방안은 적하 순서가 결정된 후 개별 컨테이너의 베이 내 위치까지 고려하여 재정돈을 계획하는 문제를 다루고 있다. 또한 본 연구는 하나의 크레인을 사용한다고 가정한 기존 연구와는 달리 복수 크레인을 활용하는 계획을 수립한다.

이주호와 최용석은 블록들이 선석에 수직으로 배치된 장치장에서 블록 내 재정돈<sup>4)</sup> 문제를 다루었다[7]. 수직으로 배치된 블록은 양 끝 단에 외부 트럭과 내부 트럭이 정차하도록 설계되어 있어 양적하 및 외부 반입출 작업의 효율을 높이기 위해서는 대상 컨테이너들을 블록의 양 끝 단에 미리 옮겨 놓는 것이 유리하다. 이 연구에서는 크기가 달라 상호 교차 가능한 두 대의 크레인을 운영한다고 가정하고, 각 컨테이너의 장치장 내 예상 체류 시간을 이용하여 옮길 대상 컨테이너들을 선정하는 방안을 제안하였다. 이 연구는 Kim과 Bae의 연구

4) 해당 논문에서는 이적이라는 용어를 사용하였다.

[5]와 같이 명시적인 적하 계획이 수립되기 전에 이후에 발생할 작업들에 대비하여 컨테이너들을 미리 옮기는 상황을 가정하였다.

강재호, 오명섭, 류광렬, 그리고 김갑환은 수평 배치의 블록 구조를 가진 장치장에서 적하 순서가 결정된 상황에서 블록 내 재정돈 계획을 수립하는 휴리스틱 방안을 소개하였다[4]. 이 연구에서는 대상 컨테이너들의 초기 위치와 적하 순서를 동시에 고려하여 그림 3의 두 가지 제약 조건을 만족하는 각 컨테이너의 목표 위치를 결정하는 휴리스틱과 결정된 목표 위치를 초기 위치 및 적하 순서와 함께 고려하여 컨테이너 이동을 위한 부분 순서를 도출하는 방안을 소개하였다. 또한 이렇게 생성한 부분 순서로부터 간단한 그리디(greedy) 휴리스틱을 이용하여 크레인 작업 계획을 생성하는 방안도 함께 제안하였다. 이 후 이들은 복수 크레인을 활용하는 재정돈 계획을 수립하기 위하여 상태 전이 그래프(state transition graph)를 이용하여 크레인간의 간섭을 파악하고 회피하는 방안을 제안하였다[6].

본 연구는 재정돈 계획을 위한 휴리스틱을 소개한 기존 연구[6]를 확장하여, 컨테이너들의 목표 위치를 담금질 알고리즘으로 탐색함으로써 보다 효율적인 재정돈 계획을 수립하는 방안을 제안한다.

**3. 탐색 기법을 이용한 재정돈 계획 수립**

본 장에서는 지역적 탐색(local search) 기법의 하나인 담금질 알고리즘을 본 문제에 적용하기 위하여 컨테이너들의 목표 위치를 설정하여 초기해를 생성하는 방

안과 목표 위치를 초기 위치 및 적하 순서와 함께 활용하여 부분 순서를 도출하는 방안을 소개한다. 현재해를 변형하여 이웃해를 생성하고, 생성된 이웃해가 유효한지 판단하는 방안도 함께 제시한다. 또한 부분 순서의 제약 하에 각 컨테이너 이동 작업을 크레인에 순차적으로 할당함으로써 실행 가능한 재정돈 계획을 수립하는 방안을 소개한다.

**3.1 초기해 생성 방안**

본 절에서는 그림 3에서 제시한 제약 조건들을 만족하도록 대상 컨테이너들의 목표 위치를 결정하는 휴리스틱 RFSDH(Re-handling-Free Slot Determination Heuristic)[4]를 소개한다. RFSDH에 의하여 결정된 컨테이너들의 목표 위치는 본 연구에서 탐색을 위한 초기해로 사용된다. 휴리스틱을 이용하여 초기해를 생성하는 이유는 대상 컨테이너들의 목표 위치를 무작위로 생성해서는 제약 조건들을 만족하는 해를 얻기가 대단히 어렵기 때문이다.

그림 4는 그림 2의 재정돈 문제를 대상으로 RFSDH를 적용하는 과정을 순차적으로 보여주고 있다. (a)의 초기 상태에서 제약 조건 2를 만족하면서 바로 옮길 수 있는 컨테이너들은 원천 베이들의 상단에 위치한 6개의 컨테이너 7W, 7G, 3G, 4G, 5G, 5W이다. 이들 컨테이너 중에서 가장 많은 수의 컨테이너가 옮겨져야 하는 목표 베이를 찾아낸다. 이 경우에는  $T_G$ 로 향하는 컨테이너가 4개(7G, 3G, 4G, 5G)이므로,  $T_G$ 가 선택된다.  $T_G$ 로 향하는 컨테이너들 중에서 스택 하나를 채울 수 있을 만큼의 컨테이너들을 선택한다. 본 예에서는 하나

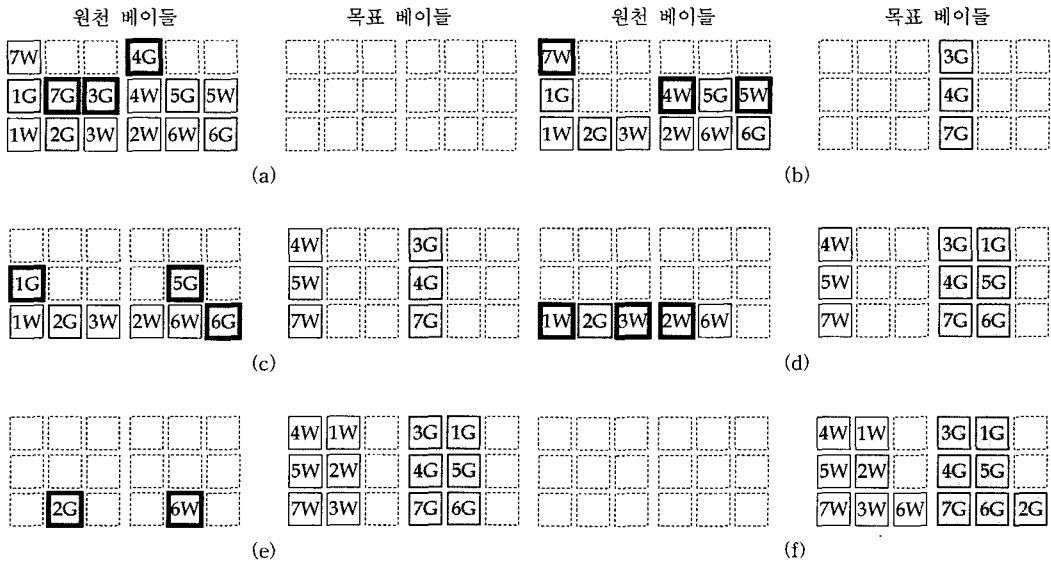


그림 4 RFSDH를 이용한 컨테이너 목표 위치 결정의 예(초기해 생성)

의 스택에 최대 3개의 컨테이너를 장치할 수 있으므로, {7G, 3G, 4G, 5G} 중에서 3개의 컨테이너를 선택한다.

예를 들어 7G, 4G, 5G 3개의 컨테이너가 선택되었다면, 이들 컨테이너를  $T_G$ 의 스택 하나에 7G → 5G → 4G의 순으로 쌓아야 제약 조건 1을 지키면서 컨테이너들을 이동시킬 수 있다. 다른 컨테이너들을 선택하더라도 선택된 컨테이너간에는 이미 부여된 적하 순서에 의하여 하나의 이동 선형 순서(linear order, total order)가 존재한다. 선택 대상 컨테이너들은 모두 원천 베이 스택의 상단에 위치하는 컨테이너들이므로 복잡한 이동 순서 계획 없이도 두 제약 조건을 모두 만족시키면서 옮길 수 있다. 한 스택에 장치할 컨테이너들을 선택하는 구체적인 절차는 그림 5와 같다. 이 절차는 가능한 하단에 옮겨야 할 컨테이너들이 많이 쌓여 있는 컨테이너를 우선적으로 선택한다. 초기 상태 (a)에서 그림 5의 절차를 적용하여 7G, 3G, 4G가 선택되면, 7G → 4G → 3G 순으로 스택에 장치하게 된다. 장치 위치가 결정된 컨테이너들을 원천 베이에서 목표 베이로 옮긴 것으로 표기하면 (b)와 같은 상태를 얻을 수 있다. 상태 (b)에서도 마찬가지로 방법으로 목표 베이를 선정하고 해당 목표 베이의 스택 하나를 채울 수 있을 만큼의 컨테이너들을 선택한다. 그림의 각 상태에서 선택된 컨테이너들은 진한 사각형으로 표시하였다. 동일한 방법을 반복하여 적용하면 최종적으로 (f)와 같이 모든 컨테이너들의 목표 위치가 결정된 상태를 얻을 수 있다.

그림 5에서 제시한 절차는 동를 발생 시 임의로(randomly) 결정하는 경우가 있으므로, 그림 4의 컨테이너들의 목표 위치는 RFSDH로 생성할 수 있는 여러 가능한 형태 중의 하나이다. 대상 컨테이너의 수가 많아지

고, 베이의 단과 열 수가 커질수록 RFSDH로 생성 가능한 초기해의 수도 많아진다.

### 3.2 부분 순서 도출

3.1절에서 소개한 RFSDH는 제약 조건 1과 2를 고려하여 컨테이너들의 목표 위치를 결정한다. 컨테이너들의 초기 위치와 결정된 목표 위치 및 적하 순서로부터 제약 조건 1과 2를 모두 만족하는 컨테이너 이동 순서들을 하나의 부분 순서로 표현하는 예를 그림 6에 보이고 있다. 그림 6의 (a)는 컨테이너들의 초기 위치를 보이고 있다. 원천 베이에서는 각 스택의 상단에 있는 컨테이너를 먼저 이동시켜야 제약 조건 2를 만족하므로, 하나의 스택 내에 있는 컨테이너간에는 이동 순서의 제약이 있다. 예를 들어 맨 왼쪽 스택의 경우 7W → 1G → 1W의 순서로 컨테이너를 옮겨야 한다. 실선 화살표로 이러한 이동 순서상의 제약을 표현하였다. 그림 6의 (b)는 컨테이너들의 목표 위치를 보이고 있다. 각 목표 베이의 스택 별로 컨테이너들이 적하 순서에 따라 정렬되어 있으므로, 제약 조건 1은 이미 만족한다. 제약 조건 2를 만족하기 위해서는 아래쪽에 위치한 컨테이너를 상단의 컨테이너보다 먼저 옮겨야 한다. 예를 들어 맨 왼쪽 스택의 경우 7W → 5W → 4W 순서로 컨테이너를 옮겨야 한다. 이러한 이동 순서상의 제약은 점선 화살표로 그림에 표시하였다. 그림 6의 (c)는 이 두 가지 이동 순서상의 제약을 하나의 부분 순서 그래프로 표현한 것이다. 컨테이너 이동 순서 계획 시 이 부분 순서 그래프에 나타난 제약을 지키면, 제약 조건 1과 2를 모두 만족시킬 수 있다.

### 3.3 이웃해 생성

본 절에서는 탐색을 위하여 현재해를 변형하여 이웃

- 단계 1. 원천 베이 스택의 상단에 위치하여 재취급 없이 바로 옮길 수 있는 컨테이너들의 집합을  $M$ 으로 둔다.
- 단계 2.  $M$ 에서 가장 많은 컨테이너를 보유한 목표 베이  $T$ 를 선정한다.
  - 단계 2.1. 동물이 발생하면, 그 중에서 무작위로 목표 베이  $T$ 를 선정한다.
- 단계 3.  $M$ 에서  $T$ 로 향하는 컨테이너들의 부분 집합을  $M_T$ 로 둔다.
- 단계 4.  $M_T$ 에 소속된 컨테이너들 중에서  $T$ 의 스택 하나를 채울 수 있는 수만큼 아래의 과정(단계 4.1 ~ 단계 4.3)을 반복하여 컨테이너를 하나씩 선정한다.
  - 단계 4.1. 자신의 하단에 가장 많은 대상 컨테이너들이 장치되어 있는 컨테이너를 선택한다.
  - 단계 4.2. 동물이 발생하면, 그 중에서 자신의 바로 하단에  $T$ 로 향하는 컨테이너가 있는 컨테이너를 선택한다.
  - 단계 4.3. 여전히 동물이 발생하면, 그 중에서 무작위로 하나를 선정한다.

그림 5 스택에 장치할 컨테이너들을 선택하는 절차

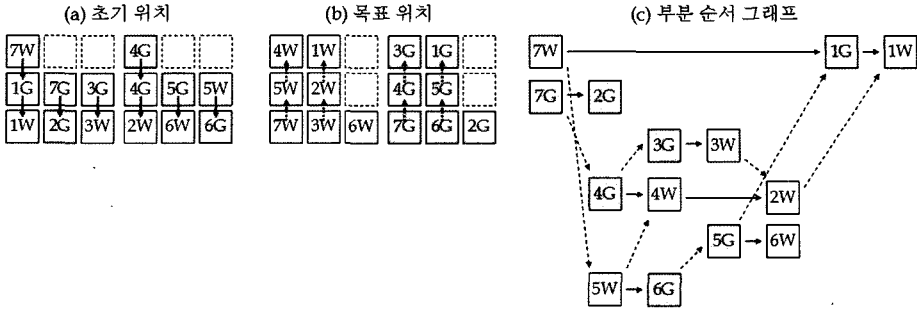


그림 6 초기 위치와 목표 위치로부터 부분 순서를 생성하는 예

해를 생성하는 방안을 소개한다. 그림 7의 (a)는 현재해를 보이고 있다. 이웃해는 현재해에서 동일한 목표 배이의 다른 스택에 장치할 두 컨테이너를 무작위로 선정하고 그 위치를 서로 교환하여 생성한다. 이 때 동일한 스택에 적하 순서가 빠른 컨테이너가 적하 순서가 늦은 컨테이너의 하단에 위치하면 적하 시 재취급이 발생하게 되므로, 제약 조건 1을 만족하기 위하여 컨테이너 교환 후 적하 순서에 맞춰 스택 내의 컨테이너들을 정렬한다. 예를 들어 그림 7의 (b)는 현재해에서 2G와 4G를 교환한 후 정렬한 경우이다. 그림에서 교환된 컨테이너들은 굵은 사각형으로 표시하였다. 그림 7의 (c)는 7W와 6W를 교환하여 생성하였는데, 이 경우에는 스택들을 정렬할 필요가 없다. 컨테이너 하나를 다른 스택으로 옮겨서 이웃해를 생성할 수도 있다. 그림 7의 (d)는 4W를 첫 번째 스택의 빈 공간 즉 가상의 컨테이너와 위치를 바꾸어 생성한 이웃해이다. 빈 공간과의 교환을 허용함으로써 각 스택에 쌓을 컨테이너 수가 현재해와 다른 이웃해도 생성할 수 있다.

3.4 이웃해 타당성 검증

현재해에서 두 컨테이너의 교환으로 생성할 수 있는 모든 이웃해가 그림 3의 두 제약 조건을 항상 만족하는

것은 아니다. 따라서, 생성된 이웃해가 제약 조건들을 만족하는지 즉 유효한지(feasible) 검증하는 과정이 필요하다. 이웃해가 제약 조건들을 만족한다면 3.2절에서 소개한 방식으로 도출한 그래프에 순환 경로(cycle)이 존재하지 않아야 한다. 도출된 그래프에 순환 경로가 존재한다면, 두 제약 조건을 모두 만족하면서 순환 경로에 소속된 컨테이너들을 옮길 수 있는 선형 순서를 만들 수 없기 때문이다.

그림 8은 그림 7의 이웃해 1, 2, 3의 유효성을 확인하기 위하여 각각에 대하여 부분 순서를 도출한 결과를 보이고 있다. 그림 7(b)의 이웃해 1은 그림 8(a)와 같은 부분 순서를 도출하게 된다. 이 경우 생성된 그래프에는 순환 경로가 존재하지 않으므로 이웃해 1은 유효하다고 판단할 수 있다. 마찬가지로 그림 7(d)의 이웃해 3은 그림 8(c)와 같은 부분 순서를 도출하는데 이 경우에도 유효한 이웃해임을 확인할 수 있다. 이에 비해 그림 7(c)의 이웃해 2로부터 생성한 그림 8(b)의 그래프에는 하나의 순환 경로 5W → 6G → 5G → 6W → 5W가 존재한다. 따라서 이 경우에는 순환 경로를 이루는 이 4개의 컨테이너들(5W, 6G, 5G, 6W) 중 하나 이상의 컨테이너를 재정돈 작업 시 재취급하여야 모든 컨테이너들

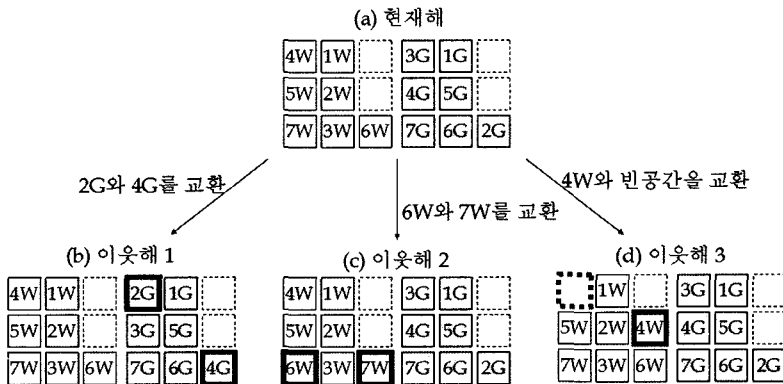


그림 7 현재해(a)에서 이웃해(b, c, d)를 생성하는 예

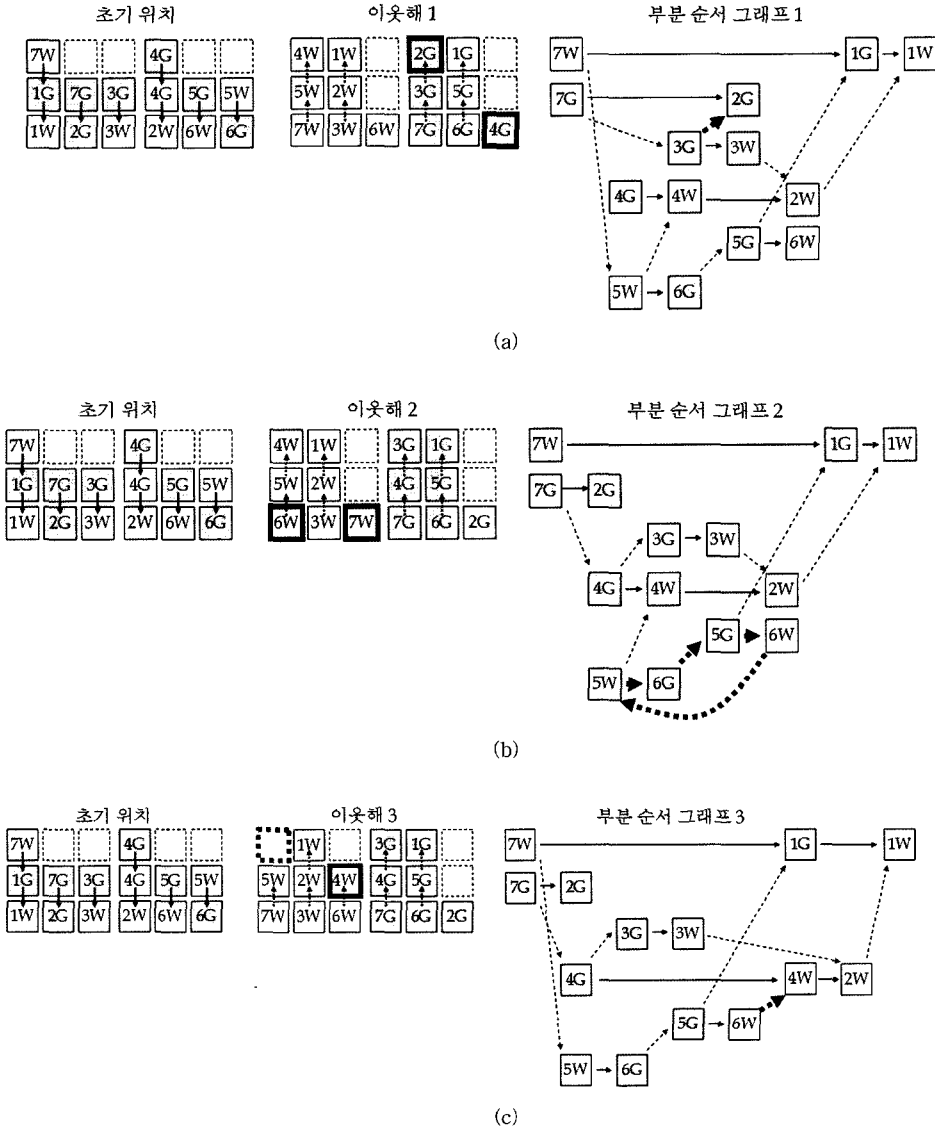


그림 8 생성한 이웃해의 유효성 검증의 예

을 정해진 목표 위치로 옮길 수 있다. 생성한 이웃해로부터 도출한 그래프에 순환 경로가 존재하면 새로운 이웃해 생성을 시도한다.

주어진 그래프 상에 순환 경로가 존재하는지 여부를 확인하는 작업은 변(edge)의 수에 비례하는 선형 시간에 수행할 수 있다[8]. 본 연구에서는 유효성 검증을 보다 효율적으로 수행하기 위하여 그림 9와 같은 부분 순서 그래프의 특성을 이용하였다. 이러한 특성에 의해서 현재해로부터 도출한 부분 순서 그래프에 새로 추가된 변(그림 8에는 굵은 선으로 표시)들을 지나는 순환 경로가 존재하는지 검사하면 된다. 예를 들어 그림 8의 (c)

의 경우 추가된 변은  $\langle 6W, 4W \rangle$ 이므로, 순환 경로가 존재하는지는 여부는 세 개의 변들( $\langle 6W, 4W \rangle$ ,  $\langle 4W, 2W \rangle$ ,  $\langle 2W, 1W \rangle$ )만 따라가며 조사하면 확인할 수 있다.

### 3.5 재정돈 계획의 생성과 평가

앞에서 소개한 방안으로 생성한 부분 순서로부터 수행 가능한 재정돈 계획을 수립하고 평가하기 위해서는 각 대상 컨테이너를 이동시킬 크레인을 선정하고, 대상 컨테이너 간의 이동 순서를 결정하여야 한다. 본 절에서는 부분 순서로부터 구체적인 크레인 작업 계획을 수립하고 평가하는 방안을 소개한다.

그림 10은 주어진 부분 순서의 제약하에서 크레인 작



1. 부분 순서 그래프는 순환 경로가 없는 방향성 있는 그래프(acyclic directed graph)이다.
2. 방향성 있는 그래프  $G = \langle V, E \rangle$ 가 부분 순서 그래프라면 ( $V =$  노드들의 집합,  $E =$  변들의 집합),
  - 2.1  $G$ 에서 일부 변들이 제거된 그래프  $G' = \langle V, E' \rangle$ ,  $E' \subset E$  도 부분 순서 그래프이다.
  - 2.2  $G$ 에 일부 변들이 추가된 그래프  $G'' = \langle V, E'' \rangle$ ,  $E \subset E''$  가 부분 순서 그래프가 아니라면, 하나 이상의 순환 경로가  $G''$ 에 존재하며, 존재하는 각 순환 경로는 추가된 변들  $E'' = E' - E$  중에 적어도 하나의 변을 지나가야 한다.

그림 9 부분 순서 그래프의 특성

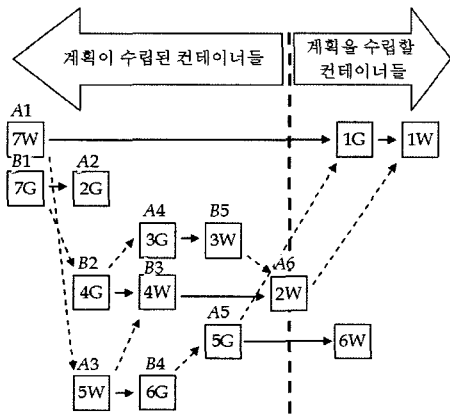


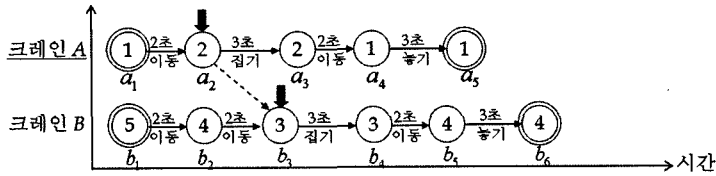
그림 10 크레인 작업 계획 수립 과정의 한 시점

업 계획을 수립해나가는 과정 중 한 시점을 보이고 있다. 이동 작업을 수행할 크레인인 순서가 결정된 컨테이너들은 관련 정보를 해당 컨테이너 위에 표기하였다. 예를 들어 2G의 경우 크레인 A가 2번째로 이동시킬 컨테이너이다. 이 예에서는 2대의 크레인 A와 B를 재정에 활용한다고 가정하였다. 크레인 A의 경우 7W → 2G → 5W → 3G → 5G → 2W 순서로 작업을 수행하고, 크레인 B는 7G → 4G → 4W → 6G → 3W 순서로 작업을 수행하기로 계획되었다. 시뮬레이션을 통해 현재 상황에서 먼저 작업이 끝나는 크레인에 새로운 작업을 추가로 할당하는 과정을 반복함으로써 전체 컨테이너에 대한 이동 작업 계획을 완성한다.

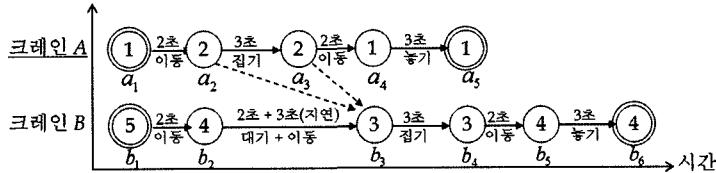
그림 10과 같이 일부만 결정된 작업 계획에서 크레인 B가 3W를 이동시키는 작업을 먼저 완료할 것으로 예상된다면, 3W 이동 이후에 수행할 새로운 작업을 크레인 B에 할당한다. 현재 상황에서 부분 순서 상의 제약에 의하여 크레인 B에 할당 가능한 작업은 1G와 6W이다. 이 중에서 어느 작업을 크레인 B에 할당하고, 작업

중에 간섭이 발생한다면 어느 크레인에 작업 우선권을 주는 것이 바람직한지 추정하기 위하여 크레인간의 간섭을 효율적으로 고려할 수 있는 상태 전이 그래프로 시뮬레이션을 수행한다[6].

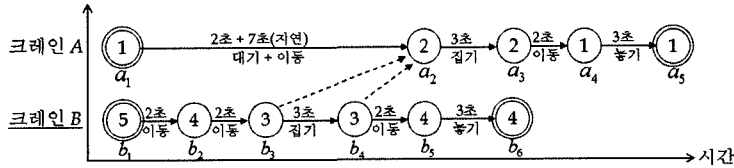
그림 11은 상태 전이 그래프를 이용하여 크레인간의 간섭을 파악하고 회피하는 예를 보이고 있다. 그림 11의 (a)는 크레인 A, B 각각이 주어진 작업을 완료하는데 필요한 동작들(operations)을 상태 전이 그래프로 표현한 것이다. 각 노드에는 해당 크레인이 위치할 베이 번호를 숫자로 나타내었으며, 화살표에는 상태 전이에 필요한 동작과 소요 시간을 기입하였다. 이해를 돕기 위하여 동작 소요 시간은 간단한 수치를 사용하였다. A, B 두 크레인이 최소 2배이 간격 이상 떨어져 있어야 작업이 가능하다고 가정하였다. 상태 전이 그래프로 시뮬레이션을 수행해보면 그림 11의 (a)와 같이 크레인 A가 상태  $a_2$ , 그리고 크레인 B가 상태  $b_3$ 일 때 간섭이 발생할 수 있다. 그림에서 간섭이 발생한 상태 위에는 굵은 화살표를 표기하였다. 간섭 발생 시 크레인 A에 작업 우선권이 있다면, 크레인 B는 크레인 A의 상태  $a_2$ 와 간섭이 발생하지 않는 이전 상태 노드  $b_2$ 에서 대기하여야 한다. 그림에서 이러한 제약을 점선 화살표로 나타내었다. 간섭 회피를 위한 제약을 만족시키기 위하여 크레인 B의 상태 전이 그래프를 수정한 결과를 그림 11의 (b)에 보이고 있다. 이 경우 작업 지연이 3초 발생함을 알 수 있다. 반대로 크레인 B에 작업 우선권이 있다면, 그림 11의 (c)와 같은 결과를 얻게 된다. 이 경우에는 7초의 지연이 발생하였다. 이러한 방식으로 모든 할당 가능한 작업에 대하여 각각의 크레인이 작업 우선권을 가진 경우를 시뮬레이션하고 그 중에서 가장 작업 지연이 적은 경우를 선택하여 부분적으로 확정된 계획에 추가하는 과정을 반복한다. 재정돈 계획이 완성되면 전체 작업 예상 시간으로 해를 평가한다.



(a) 시뮬레이션을 통한 간섭 파악



(b) 간섭을 회피하기 위하여 수정한 상태 전이 그래프 (크레인 A에 우선권이 있는 경우)



(c) 간섭을 회피하기 위하여 수정한 상태 전이 그래프 (크레인 B에 우선권이 있는 경우)

그림 11 상태 전이 그래프를 이용한 간섭 파악과 회피의 예

이상으로 초기해 생성 방안, 이웃해 생성 방안, 해의 평가 방안을 정의하였으므로, 지역적 탐색 기법들을 적용하기 위하여 필요한 요소들이 모두 갖추어졌다. 간단하게는 생성한 이웃해의 품질이 현재해보다 나으면 해당 이웃해로 이동하는(move, accept) 언덕 오르기(hill-climbing) 탐색 기법을 적용할 수 있다. 하지만, 언덕 오르기 탐색의 경우 국소 최저점(local optimum)에 도달하면 해의 품질을 더 이상 개선시키지 못하는 단점이 있다. 이러한 단점을 극복하기 위하여 본 연구에서는 담금질 알고리즘[1,9,10]을 사용하였다. 담금질 알고리즘은 이웃해의 품질이 현재해의 품질보다 나쁘더라도 확률적으로 이동을 허용한다. 이동할 확률은  $e^{-\delta/T}$ 와 같이 정의된다. 수식에서  $\delta$ 는 현재해와 이웃해의 평가 값 차이이며,  $T$ 는 탐색이 진행됨에 따라 점차 낮아져 최종적으로는 0에 접근하는 온도이다. 즉, 탐색 초기에는 현재해와 이웃해의 평가 값 차이가 크더라도 이동할 수 있게 함으로써 탐험(exploration)을 허용하고, 탐색 후반부에는 이웃해의 품질이 현재해보다 좋은 경우에 이동하는 빈도를 보다 높임으로써 집중(concentration)을 강화한다. 담금질 알고리즘은 언덕 오르기 탐색 기법이 국소 최저점에서 더 이상 해가 개선되지 않는 문제점을 어느 정도 극복할 수 있으며, 순회 판매자 문제[11], 흐름 공장(flow shop) 계획 문제[12], 주파수 할당 문제[13],

VLSI 설계에서 평면 배치 계획(floor-planning) 문제 [14] 등에 효과적으로 적용된 바 있다.

#### 4. 실험 결과

본 장에서는 제안 방안을 적용한 실험 결과를 정리하여 분석한다. 하나의 블록에는 총 33개 베이가 있으며, 각 베이는 자동화 컨테이너 터미널에서 많이 활용되는 6단 9열로 구성되었다고 가정하였다. 대상 컨테이너들의 적하 순서는 임의로 생성하였으며, 총 196개(4개 베이 분량)의 컨테이너를 재정돈하는 계획을 수립한다고 가정하였다. 재정돈에는 교차 불가능한 두 대의 크레인을 사용하며, 크레인들이 상호 간섭 없이 작업할 수 있는 최소 거리는 5베이로 두었다. 목표 베이 위치에 따른 본 제안 방안 및 비교 실험 방안들의 성능을 평가하기 위하여 그림 12와 같이 목표 베이 위치가 다른 세 가지 시나리오를 설정하였다. 원천 베이의 수는 4, 5, 6, 8, 16, 29개로 두되, 목표 베이들을 제외한 나머지 베이들 중에서 무작위로 골라 원천 베이로 설정하였다. 원천 베이 수를 29개로 설정한 실험은 목표 베이들을 제외한 블록 내 나머지 모든 베이들에 대상 컨테이너들이 골고루 흩어진 경우이며, 원천 베이 수를 4개로 설정한 실험은 196개의 대상 컨테이너들이 블록 내에 흩어져 있는 몇몇 베이에 최대로 집중된 경우이다. 원천 베이 수가

표 1 RFSDH-1 대비 탐색 방안 별 재정돈 작업 시간 단축 비율 (단위: %)

시나리오	탐색 방안	원천 베이 수					
		4	5	6	8	16	29
시나리오1	RFSDH-10000	3.8	6.0	9.2	12.6	11.3	11.7
	HC	4.0	5.7	9.2	12.0	11.3	11.4
	SA	<b>4.6</b>	<b>7.1</b>	<b>10.8</b>	<b>14.4</b>	<b>15.1</b>	<b>14.9</b>
시나리오2	RFSDH-10000	7.5	11.8	13.3	14.5	15.5	14.3
	HC	7.7	11.5	11.5	12.7	14.1	13.2
	SA	<b>10.3</b>	<b>13.5</b>	<b>15.3</b>	<b>15.4</b>	<b>16.3</b>	<b>14.8</b>
시나리오3	RFSDH-10000	8.8	10.0	11.0	10.6	7.6	5.4
	HC	9.1	9.1	10.7	9.6	6.5	5.0
	SA	<b>11.0</b>	<b>11.0</b>	<b>12.1</b>	<b>11.8</b>	<b>7.7</b>	5.0

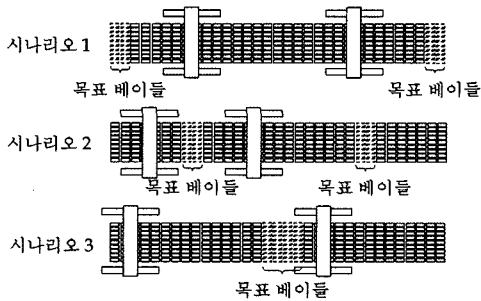


그림 12 블록 내 목표 베이 위치 설정의 세 가지 시나리오

많을수록 대상 컨테이너들이 블록 내에서 흩어진 정도는 증가한다.

실험을 위하여 기존 연구[6]에서 제안한 RFSDH를 1만회 반복 수행하여 가장 좋은 결과를 선택하는 RFSDH-10000, 지역적 탐색 기법의 하나인 언덕 오르기(hill-climbing) 탐색 기법(HC)을 적용하는 방안, 그리고 본 연구에서 제안하는 담금질 알고리즘(SA)을 이용하는 방안 이 세 가지 방안을 구현하였다. SA는 원천 베이 수에 따라 여러 온도 계획(temperature schedule)을 실험해보고 가장 좋은 성능을 보인 온도 계획을 채택하였다. 각 시나리오 및 원천 베이 수에 따라 10개의 서로 다른 문제를 생성하여 각 방안을 적용한 후 그 성능을 평균하였다. 공평한 비교를 위하여 1회의 실험에서 각 방안은 1만회씩 재정돈 계획을 수립하고 평가하였다. 부분 순서로부터 간섭을 고려하여 구체적인 작업 계획을 수립하는 과정이 전체 연산 시간의 대부분을 차지하므로 동일한 횟수의 재정돈 계획을 수립하는 방안들끼리의 비교는 공평하다 할 수 있다. 1만회의 재정돈 계획을 수립하는데 평균 13분 정도 소요되었다.<sup>5)</sup>

표에 비교 실험 결과를 정리하였다. 표에서의 수치는 RFSDH를 1회 수행하여 수립한 재정돈 계획의 작업 소

요 시간<sup>6)</sup>에 대비하여 해당 방안 적용 시 단축할 수 있는 시간을 비율로 나타낸 것이다. 따라서 수치가 클수록 소요 시간이 짧은 좋은 재정돈 계획을 수립한 것이다. 각 시나리오 및 원천 베이 수에 따른 실험에서 가장 좋은 성능을 보인 경우는 표에서 회색 바탕에 나타내었다. 전반적으로 본 연구에서 제안한 SA를 이용한 방안이 가장 좋은 결과를 보였으며, HC의 경우 국소 최저점(local optimum)에 빠지는 경향이 있어 몇몇 경우를 제외하고는 RFSDH-10000보다 성능이 떨어졌다. 이러한 결과는 본 논문에 제안하는 방안이 적절한 시간 내에 효율적인 재정돈 계획을 수립할 수 있음을 보여 준다.

### 5. 결론 및 향후 연구

블록 내 재정돈 작업은 컨테이너 터미널의 적하 작업 효율을 높이기 위하여 장치장에서 수행하는 중요한 작업의 하나이다. 본 논문에서는 담금질 알고리즘을 활용하여 재정돈 계획을 수립하는 방안을 제안하였다. 본 제안 방안은 대상 컨테이너들의 목표 위치를 탐색의 대상으로 삼아 해로 표현하고, 컨테이너들의 목표 위치를 바꾸어 이웃해를 생성하였다. 해에 표현된 컨테이너들의 목표 위치를 주어진 초기 위치 및 적하 순서와 함께 고려하여 재취급이 발생하지 않는 부분 순서를 도출하고, 도출된 부분 순서 상의 제약하에서 크레인간의 간섭을 고려하여 재정돈 계획을 수립하고 해를 평가하였다. 실험 결과 본 제안 방안이 다양한 환경하에서 유효하면서 효율적인 재정돈 계획을 적절한 시간 내에 수립할 수 있음을 확인하였다.

본 연구에서는 선석에 수평으로 배치된 장치장 블록에서 교차가 허용되지 않는 크레인들을 대상으로 재정돈 계획을 수립하였다. 향후 선석에 수직으로 배치된 블록과 교차 가능한 크레인들이 설치된 경우에도 효율적인 재정돈 계획을 수립하기 위하여 본 제안 방안을 확장하는 연구가 요청된다.

5) 3.2GHz 펜티엄 CPU를 탑재한 컴퓨터에 윈도우 XP 운영체제를 설치하고 Visual C++를 이용하여 소프트웨어를 개발하여 실험하였다.

6) RFSDH를 1만회 수행한 후 결과들의 평균을 취하였다.

참고문헌

[1] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P., "Optimization by Simulated Annealing," *Science*, Vol. 220, pp. 671-680, 1983.

[2] 강재호, 류광렬, 김갑환, "부정확한 무게 정보를 가진 수출 컨테이너를 위한 장치 위치 결정 규칙 생성 방법", *한국항해항만학회 논문지*, 제29권, 제6호, pp. 573-581, 2005.

[3] 강재호, 강병호, 류광렬, 김갑환, "기계학습을 이용한 수출 컨테이너의 무게그룹 분류", *한국지능정보시스템학회 논문지*, 제11권, 제2호, pp. 59-79, 2005.

[4] 강재호, 오명섭, 류광렬, 김갑환, "컨테이너 터미널 장치장에서 블록 내 이적을 위한 컨테이너 이동 순서 계획", *한국항해항만학회지 논문지*, 제29권, 제1호, pp. 83-90, 2005.

[5] Kim, K. H. and Bae, J.-W, "Re-Marshaling Export Containers," *Computers and Industrial Engineering*, Vol. 35, No. 3-4, pp. 655-658, 1998.

[6] 오명섭, 강재호, 류광렬, 김갑환, "복수 크레인을 활용한 블록 내 컨테이너 이적 계획", *한국항해항만학회 논문지*, 제29권, 제5호, pp. 447-455, 2005.

[7] 이주호, 최용석, "자동화 컨테이너터미널의 이적작업에 관한 시뮬레이션 연구", *한국항해항만학회 추계학술대회 논문집*, pp. 203-208, 2004.

[8] Nivasch, G., "Cycle Detection Using a Stack," *Information Processing Letters*," Vol. 90, No. 3, pp. 135-140, 2004.

[9] Aarts, E. and Korst, J. *Simulated Annealing and Boltzman Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing*, John Wiley & Sons, 1989.

[10] Aarts, E. and Korst, J., "Simulated annealing," *Local Search in Combinatorial Optimization*, John Wiley & Sons, pp. 91-120, 1997.

[11] Triki, E., Collette, Y., and Siarry, P., "A Theoretical Study on the Behavior of Simulated Annealing Leading to a New Cooling Schedule," *European Journal Operational Research*, Vol. 166, No. 1, pp. 77-92, 2005.

[12] Low, C., "Simulated annealing heuristic for flow shop scheduling problems with unrelated parallel machines," *Computers and Operational Research*, Vol. 32, No. 8 pp. 2013-2025, 2005.

[13] Björklund, P., Värbrand, P., and Yuan, D., "Optimized planning of frequency hopping in cellular networks," *Computers and Operational Research*, Vol. 32, No. 1, pp. 169-186, 2005.

[14] Ho, S.-Y., Ho, S.-J., Lin, Y.-K., and Chu, W. C.-C., "An orthogonal simulated annealing algorithm for large floorplanning problems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 12, No. 8, pp. 874-876, 2004.



오명섭

부산대학교 컴퓨터공학과. 2004년 부산대학교 컴퓨터공학 학사. 2006년 부산대학교 컴퓨터공학 석사. 2006년 1월~현재 팬택계열 SKY 내수총괄 연구원. 관심분야는 인공지능, 최적화, 지능형물류시스템



강재호

부산대학교 컴퓨터공학과 박사후연수연구원. 1995년 부산대학교 컴퓨터공학 학사. 1997년 부산대학교 컴퓨터공학 석사. 2004년 부산대학교 컴퓨터공학과 박사. 1999년 3월~2000년 1월 (주)해동EMC 연구원. 2000년 2월~2004년 1월 동아대학교 지능형통합항만관리연구센터 연구원. 2005년 8월~2006년 4월 부산대학교 차세대물류IT사업단 Post Doc. 2006년 5월~현재 야후코리아(주) 검색 R&D 파트 근무. 관심분야는 기계학습, 최적화, 지능형물류시스템, 정보검색



류광렬

부산대학교 컴퓨터공학과 교수. 1979년 서울대학교 전자공학과 학사. 1981년 서울대학교 전자공학과 석사. 1983년 3월~1984년 8월 충북대학교 컴퓨터공학과 전임강사. 1992년 University of Michigan 전기 및 컴퓨터공학과 박사. 1992년 3월~1993년 2월 Scientific Research Lab., Ford Motor Company 선임연구원. 1993년~현재 부산대학교 정보컴퓨터공학부 교수. 관심분야는 인공지능, 기계학습, 지능형물류시스템, 정보검색



김갑환

부산대학교 산업공학과 교수. 1977년 서울대학교 산업공학과 학사. 1979년 한국과학기술원 산업공학과 석사. 1979년 3월~1982년 9월 한국과학기술연구소 및 한국생산기술연구원 연구원. 1987년 한국과학기술원 산업공학과 박사. 1984년 3월~현재 부산대학교 산업공학과 교수. 관심분야는 생산시스템의 설계, 계획 및 운영의 최적화