

## 2단계 수송네트워크 상에서의 장기적 비용 최적화

정재헌\*

### A Cost Optimization Problem on a 2 Level Transportation Network

Jaeheon Jung\*

#### ■ Abstract ■

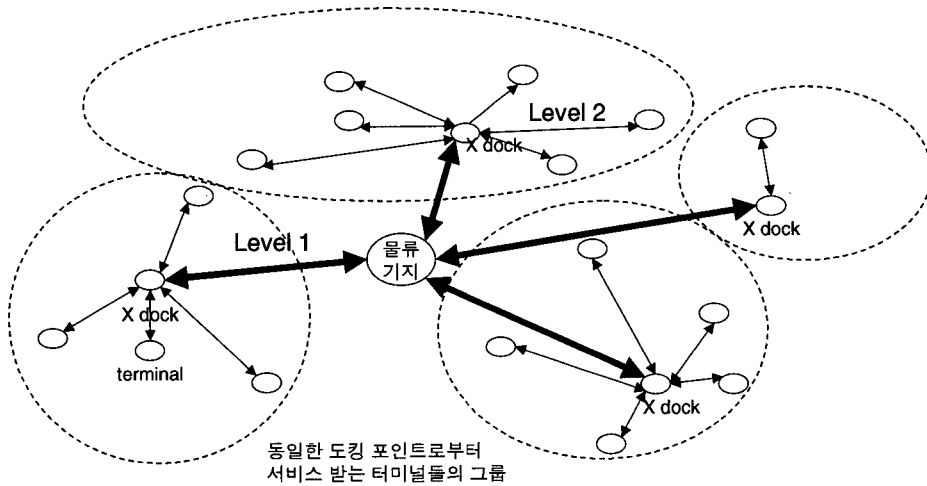
In our model, a fleet of vehicles start from docking point to collect loads at the terminals assigned to the point. Then the docking points are connected to the hub by primary vehicle routes starting at the hub. This vehicle visit all the docking points to collect the loads which have been collected by the secondary vehicles. Our goal is to minimize the long-run cost of setting up the docking points and vehicle operation cost by deciding the location of the docking points and the routes optimally. We propose an heuristic algorithm to solve this and tested it though various experiments.

Keyword : Mathematical Programming, Dynamic Programming, Supply Chain Management, Transportation, Vehicle Routing, Cross-docking Policy

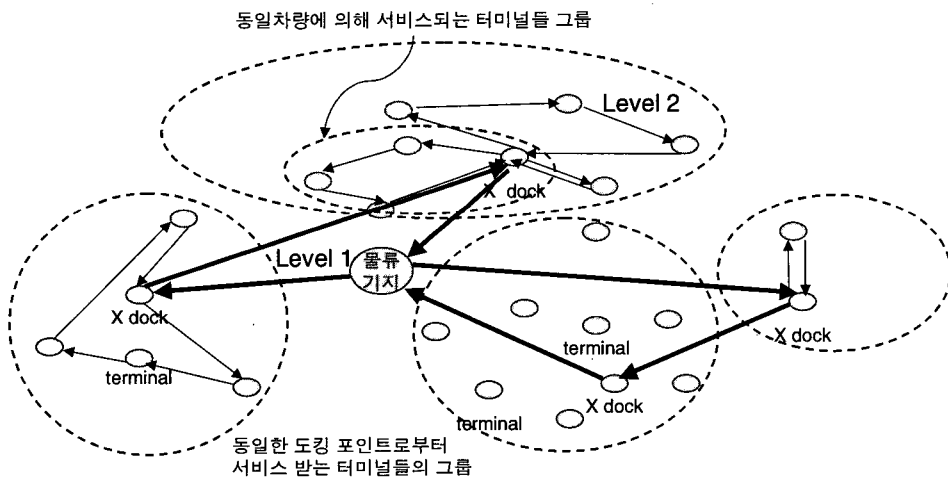
### 1. 서론

작은 화물 수송을 요하는 많은 터미널이 특정 물류기지(동시에 차량기지), 곧 네트워크의 허브에 의존하고 있을 때, 수송 차량을 그들 터미널에 배정하는 한 가지 방법은 [그림 1]과 같은 2단계 수

송 네트워크를 이용하는 것이 될 수 있다. [그림 1]에 표시된 바와 같이 허브에서 도킹 포인트로 가는 것이 1차 수송네트워크이고, 도킹 포인트에서 터미널까지 타 차량을 이용하여 수송되는 것이 2차 수송네트워크를 이루어 2단계 수송네트워크를 구성한다. 큰 규모의 화물은 도킹포인트로 옮겨져서, 작



[그림 1] 전형적인 2단계 수송 네트워크 모델 : 도킹 포인트(x dock)의 결정 및 도킹 포인트로의 주문 배정



[그림 2] 새로운 2단계 수송 네트워크 모델 : 도킹 포인트(x dock)의 결정 및 도킹 포인트와 차량으로의 주문 배정, 차량 순회 경로의 결정

은 물량으로 쪼개져서 환적되어 재 수송된다. 이와 같은 전략은 크로스 도킹 전략으로 널리 알려져 있다. 주로 재고 정책 측면에서 알려진 크로스 도킹 전략에서 도킹 포인트는 수송 과정의 조정자 역할을 하며 그 자체가 재고를 유지하지는 않는다(크로스도킹 정책의 경영적인 측면은 Stalk et al.[18]를 참조할 수 있다).

한편, 크로스 도킹 전략은 수송 전략 측면에서는 크게 주목받지 않았다. 이와 같은 맥락에서 크로스

도킹 전략의 비용 최적화 분야는 보다 많은 연구가 필요하다. 본 논문은 크로스 도킹 전략의 2단계 수송 네트워크의 비용적인 측면에 주목한다.

물류 기지에서 모든 터미널에 화물을 직접 배달하는 방법을 택할 때보다, 그것이 중간의 크로스도킹 포인트를 거쳐서 갈 때, 수배송 관리상 여러 이점을 가져올 수 있다. 예컨대, 만약 철도로 화물을 모든 목적지로 운반한다고 하자. 이 경우 값싼 철도라는 수송 수단이 물류기지에서 모든 터미널까

지 직접 화물을 운송하게 함으로서 최저 비용 운송을 달성할 수는 있겠으나, 그 경우 운송 상의 여러 불리함을 감수할 수 없는 경우가 생긴다. 고객이 특정 물품의 배송을 예컨대, 8시간 내에 배송하는 것을 원할 때, 철도를 이용한다면 그 말단 터미널까지는 8시간 이상이 걸릴 수도 있고, 철도로 갈 수 없는 터미널도 존재한다. 그러나 예를 들어, 중간지인 크로스도킹이 가능한 지점까지 최대 6시간 만에 간다고 하면 도킹 포인트에서는 자동차를 이용하여 환적 시간까지 감안하더라도 8시간 안에 배송할 수 있다. 거꾸로 자동차를 이용하여 물류기지에서 모든 최종 터미널까지 화물을 수송한다면 시간내에 운송할 수는 있을지 모르지만 수송 비용이 훨씬 많이 들 수도 있다. 이와 같이 크로스 도킹 전략은 특정 서비스 수준과 수송비용 상의 트레이드 오프상의 선택에 훨씬 많은 다양한 가능성을 제시해주는 수송 전략으로 채택될 수 있다. 여기서 주로 재고 정책의 관점에서 주목 받아온 크로스 도킹 전략이 수송전략으로도 충분히 의미가 있음을 주목할 필요가 있다. 이러한 관점에서 수송비용, 환적 비용, 도킹 포인트 유지 비용 등을 모두 포괄하여 최적의 도킹 포인트 위치 선정 및 차량 운영 결정을 모델화시킨 2단계 수송 네트워크 최적화에 관한 연구는 중요하다.

크로스 도킹 포인트에 해당하는 Break Bulk, Hub 또는 Relay point(또는 gateway)를 고려한 2단계 수송네트워크에서 최적 차량 또는 경로 조합에 의한 차량 운송 비용 최적화에 관한 연구(D. Kim *et al.*[13], W.B. Powell and Y. Sheffi[17]) 또는 도킹 포인트의 최적 위치 등도 함께 고려하는 네트워크 최적화문제에 관한 연구(Powell[16], T. Aykin[4], CS. Sung *et al.*[19], CS. Sung *et al.*[20])는 다수 있다. 또 이들을 개괄할 수 있는 문헌은 K.A. Ware *et al.*[3]과 Minoux[15]가 있다. 이 밖에도 실용적인 관점에서 크로스 도킹 전략을 바라본 연구는 U.M. Apte *et al.*[2]를 참조할 수 있다.

그러나 이들 문헌이 다루는 수송 네트워크들은

그 안에 하나의 차량이 다수의 화물을 싣고(혼적), 여러 상차 또는 하차지를 순회하는 부분 네트워크(Vehicle Routing)는 포함하지 않고 있다. 또한 [그림 2]와 같이 묘사되는 본 논문의 수송 네트워크 모델과 정확히 일치되는 모델을 타 문헌에서 다루는 적이 없으며, 다만 혼적 및 Vehicle Routing 개념이 포함되지 않은 본 모델의 원형이 되는, 크로스 도킹 포인트를 수송네트워크에 포함시켜 비용 최적화를 고려한 연구를 Minoux[15]에서 찾을 수 있다(그 모델의 2단계 수송네트워크는 [그림 1]과 같다). 본 논문의 모델과 기존 문헌의 모델의 차이점은 다음과 같이 강조될 수 있다.

- 1) 도킹 포인트의 가능한 지점들의 집합  $I=(1, 2, \dots, m)$ 은 Minoux[15]의 전형적인 2단계 네트워크 상에서와 같이 미리 위치가 지정되어 있지 않다. 전형적인 2단계 네트워크 모델에서는 도킹 포인트 후보로 지정된 지점은 최적화 결과 터미널로 기능할 수 없고, 터미널로 기능하기로 결정된 지점은 최적화 결과에서 도킹포인트로 기능할 수 없다. 본 논문의 모델에서는 모든 터미널이 도킹 포인트 후보로 기능할 수 있다. 즉 본 논문에서 제시하는 최적화 결과로 모든 터미널 중에서 최소 비용을 제시하는 터미널이 도킹포인트로 지정되고, 나머지 터미널은 순수 터미널로만 기능한다. 물론 특정 지점만을 도킹 포인트로 이용되게 할 수도 있다. 이는 나머지 터미널의 도킹 포인트 설치비용을 아주 크게 함으로서 본 논문의 모델에서 가능하다.
- 2) 도킹 포인트와 물류기지사이 그리고 도킹 포인트와 터미널 사이는 [그림 1]에서 보는 바와 같이 나무 구조(tree)로 연결되어지지 않고 모두 환형(cycle)의 구조 [그림 2]를 이룬다.

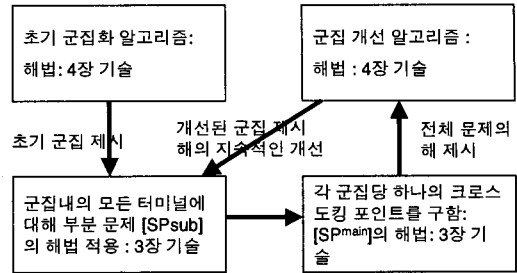
여기서 한 그룹의 터미널은 동일 수송수단에 의해 동시에 방문되는 보다 작은 그룹으로 나뉘지고, 그 수송수단은 특정 도킹 포인트에서 출발하여 2차

수송 네트워크를 이룬다. 이 도킹 포인트는 다시 다른 도킹 포인트 및 물류기지와 또 다른 유형의 수송로인 1차 수송 네트워크로 연결된다. 이 경우 우리의 모델은 최적의 1, 2차 차량 경로 및 터미널의 차량에의 배정을 동시에 최적화한다.

이 모델은 기존의 출판된 어떤 연구도 다루지 않았으며, 기존의 2단계 계층 수송 네트워크 모델에 비해서 다음과 같은 점에서 보다 쓸모 있다.

- 1) 모든 터미널이 도킹 포인트가 될 수 있도록 지정되어 중심 물류기지에 직접 연결될 수 있어 현실적인 적용에 있어 보다 유연성이 있다. 도킹 포인트 후보군이 관리자에 의해 미리 전제되는 기존 모델에 비해 우리의 모델은 보다 일반적이다.
- 2) 차량 하나가 다수의 터미널을 연결함으로써 단순한 왕복형의 루트(line haul)보다 경제적이다.
- 3) 현실적으로 이 모델은 우리가 도킹 포인트를 특정 지점에 설치하는데 드는 비용과, 수송수단을 운용하는데 드는 비용(유류비, 인건비, 환적 비용, 도킹 포인트 유지 비용)의 총합을 장기적으로 최소화하고자 할 때 이용될 수 있다. 즉 이 모델은, 수송수단이 모두 자가 소유이고 그 수송수단을 구입하는데 드는 비용은 기투자된 비용으로 취급되면서, 주된 수송 원가 절감의 초점이 운용비용과 도킹 포인트와 관련된 고정비용의 감소가 맞춰질 때, 유용하다.

본 논문에서 우리는 위에서 기술한 대규모의 2단계 계층 수송네트워크의 최적화를 위한 모델을 개발시켰다. 이 모델은 NP hard문제이나, 우리는 근사 해법을 고안하였다. 이 해법은 부분적인 근사해를 구하는 방법과 부분 최적해를 구하는 방법을 조합함으로써 이루어진다. 이 해의 유용성을 보여주기 위해 Minoux[15]의 전형적인 2단계 네트워크의 최적화 모델을 이용한 해법과 계산 실험을 통한 비교 결과가 본 논문의 마지막 절에 제시되어 있다.

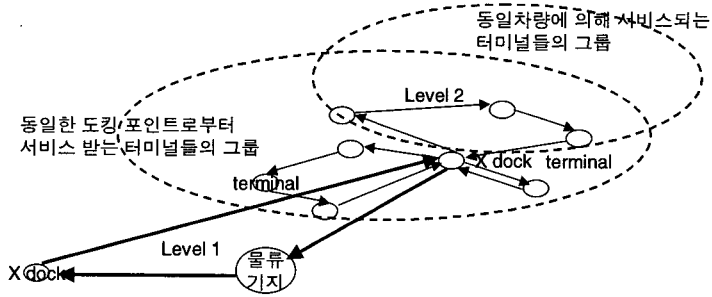


[그림 3] 해법의 흐름도

본 논문 2장에서 어떻게 2단계 수송 네트워크가 수리적 모델로 정형화될 수 있는지가 보여지고, 3절에서 동일한 도킹 포인트에 연결될 터미널들을 하나의 그룹으로 묶는 군집화(grouping)의 결정이 이루어진 상태에서, 각 군집별로 어떤 터미널에 도킹 포인트를 설치할 것인가를 결정하는 문제에 대한 해법을 제시한다. 4장에서는 동일한 도킹 포인트에 연결될 터미널들을 묶는 군집화(grouping) 결정의 휴리스틱 해법(초기 군집화 알고리즘)을 제시한다. 동시에 4장에서, 3장의 알고리즘을 이용해 군집을 지속적으로 개선해나가는 과정의 종합적인 알고리즘(군집개선 알고리즘)이 제시된다. 전체적인 해법의 구조와 동시에 해법의 기술 방법은 [그림 3]과 같다. 그리고 5장에서 타 해법과의 비교를 위한 계산 실험 결과가 제시된다.

## 2. 모 델

우리의 모델에서, 수송수단은 도킹 포인트에서 그 도킹 포인트에 배정된 터미널의 화물들을 수집하기 위해 출발한다. [그림 4]에 보여진 바와 같이 터미널은 동일한 도킹 포인트를 위한 집단으로 군집화되어 있는 상태 아래에서 다시 동일한 차량에 의해 배송되는 그룹으로 나뉜다(2차 수송네트워크). 이 도킹 포인트는 [그림 2]에서 보이는 대로, 다시 중심이 되는 물류기지에서 출발하는 1차 네트워크의 1차 수송수단에 의해 다른 도킹포인트와 함께 화물을 물류기지로부터 배송받는다. 1차 및 2차 수송네트워크는 모두 전형적인 Traveling Salesman 문제의 해에 의해 주어지는 경로를 따라 가는 것이 최적이며, 2차

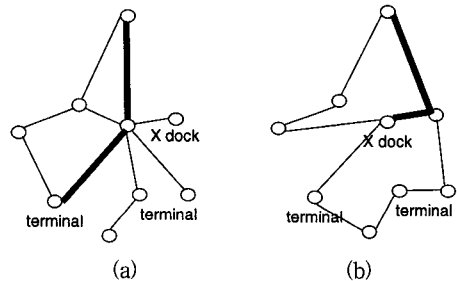


[그림 4] 모델 묘사

(1차) 수송 네트워크에서 동일 차량에 의해 배송되는 터미널(도킹 포인트)의 결정은 배송 계획 문제 (Vehicle Routing Problem)의 해를 따른다. 이하에서 자주 쓰이는 문자 기호의 설명은 다음과 같다.

- $f_i$  : 도킹 포인트를 터미널  $i$ 에 설치할 때 드는 비용, 단  $f_0$ 는 물류기지 설치비용.
- $c_{ij}$  : 2차적인 수송 라인이 터미널  $i$ 에서  $j$ 로 연결될 때, 그 수송 비용. 만약 터미널  $i$ 가 도킹 포인트라면  $c_{ij}$ 는  $i$ 에서의 환적 비용을 포함.
- $c_{ij}^1$  : 1차적인 수송 라인이 터미널  $i$ 에서  $j$ 로 연결될 때의 수송 비용.
- $Q_1$  : 1차 수송수단의 용량
- $Q_2$  : 2차 수송수단의 용량
- $C$  : 한 개의 도킹포인트가 감당할 수 있는 최대의 화물용량
- $N$  : 모든 터미널들의 집합, 그리고 그 크기  $|N| = n$ .
- $Y$  :  $= \{Y_1, Y_2, \dots, Y_p\}$ 는 주어진 동일한 도킹포인트에서 출발하는 차량들에 의해 방문되는 터미널들 곧 군집들의 집합, 그 집합의 크기 곧 군집의 숫자는  $p$ .
- $R$  :  $= (r_1, r_2, \dots, r_p, \dots, r_p/r_p \in Y_p)$ 는 각 터미널 그룹  $Y_p, p=1, \dots, p$ 에 대응하는 도킹포인트들의 집합
- $a_i$  : 터미널  $i$ 를 위해 배송해야 하는 화물수량

우리는 수학적으로 이 모델을 정형화하기 위해 K 트리 개념을 도입한다. K 트리는 트리 구조의 그래프에 K개의 추가적인 에지가 더해진 그래프로 이해할 수 있다. 즉  $n$ 개의 노드를 가진 주어진 그래프에서, 우리는 [그림 5]에서 보는 바와 같이 K 트리를 그 그래프를 span 하는  $n+K$  에지의 집합으로 정의한다. K 트리의 상세한 내역은 ML. Fisher[10]을 참조할 수 있지만, 그림에서 보듯이 K개의 에지가 트리에 추가되었을 때, 특별한 경우는 K개의 환형의 차량 순회 (Vehicle Routing) 경로를 의미하게 된다. 즉 그림에서 보듯이 2개의 에지가 트리에 포함되었을 때, 왼쪽은 단순한 2-tree이지만 오른쪽의 특별한 경우는, 역시 마찬가지로 2-tree이지만 2개의 차량 순회 경로를 의미하는 2개의 환이다. 우리는 모델의 정형화에 K 트리를 이용하여 feasible한 VRP의 해가 K 트리에 포함된다고 명시하여 표현을 간략히 한다.



[그림 5] 2-trees : (a), (b) 모두 두 개의 추가적인 에지(굵은 선)를 가진 tree들. 그러나 (a)는 단순한 2-tree이고, (b)는 2개의 차량 루트를 가진 VRP의 가능해이다. 2개의 차량 루트를 가진 VRP 가능해는 2-tree의 특별한 경우이다.

이때,  $KT_b$ 를  $Y_b$ 의 모든 터미널 노드를 span하는 K 트리라고 하고,  $KT_0$ 를 집합 R과 중심 물류기지 0를 span하는 K 트리라고 할 때,  $y=(y/y \in KT_b \cup KT_0)$ 의 벡터가 정의된다.  $y_{ij}^p(y_{ij}^0)$ 는 만약 예지  $(i, j)$ 가  $KT_b(KT_0)$ 의 한 예지로 선택되면 1이고 그렇지 않으면 0이다. 그 예지는 무방향으로  $y_{ij}^p$ 는  $y_{ji}^p$ 와 동일한 값을 가진다(따라서 아래 문제 [P]의 두 번째 목적항에서 중복을 피하기 위해 부등호가 추가됨). 그리고 아래 수식에서,  $\lceil x \rceil$ 를  $x$ 보다 작지 않는 가장 작은 정수라고 할 때, 임의의  $S \subset N$ 에 대해  $a(S) = \sum_{i \in S} a_i$ 일 때,  $r_b(S)$ (b는 임의의 b)는  $\lceil a(S)/b \rceil$ 를 의미한다. 그러면 우리의 모델은 다음과 같이 정형화된다.

[P]

$$\begin{aligned} \text{Min } P, Y, R \left[ \sum_{p=1}^P \text{Min } y_p \left[ \sum_{i, j \in Y_p} c_{ij} y_{ij}^p \right] \right. \\ \left. + \text{Min } y_0 \left[ \sum_{i \in R \cup \{0\}, j \in R, j > i} (f_i + c_{ij}) y_{ij}^0 \right] \right] \end{aligned}$$

s.t.

$$\sum_{i \in R \cap Y_p} \sum_{j \in Y_p} y_{ij}^p = 2r_{Q_2}(Y_p) \quad \text{for } p=1, 2, \dots, P \quad (1)$$

$$\sum_{j \in Y_p} y_{ij}^p = 2 \quad \text{for all } i \in Y_p, i \notin R, p=1, 2, \dots, P \quad (2)$$

$$\sum_{i \in S} \sum_{j \in S} y_{ij}^p \geq 2r_{Q_2}(S) \quad \text{for all } S \subset Y_p \text{ and } i \notin R \\ \text{with } |S| \geq 2, p=1, 2, \dots, P \quad (3)$$

$$\sum_{i \in R} y_{0i}^0 = 2r_{Q_1}(N) \quad (4)$$

$$\sum_{j \in R \cup \{0\}} y_{ij}^0 = 2 \quad \text{for } i \in R \cup \{0\} \quad (5)$$

$$\sum_{i \in S} \sum_{j \in S} y_{ij}^0 \geq 2r_{Q_1}(S') \quad \text{for all } S' \subset R, \text{ with } |S'| \leq 2 \quad (6)$$

$$\sum_{j \in Y_p} y_{ij}^0 = 0 \quad \text{for all } i \in R \cap Y_p, \text{ for } p=1, 2, \dots, P \quad (7)$$

$$\sum_{i \in Y_p} a_i \leq C \quad \text{for } p=1, 2, \dots, P \quad (8)$$

$$y_{ij}^0, y_{ij}^p \in \{0, 1\} \quad \text{for all } i, j, p=1, 2, \dots, P \quad (9)$$

제약식 (1)~식 (3)은 각각의  $Y_p$ 안에 있는 노드로 구성된 Y 벡터로 표현되는 K 트리가 Vehicle Routing Problem(VRP)의 가능해가 될 수 있도록 보장한다.  $Y_p$ 의 도킹 포인트는 거기에서 출발하는 각각 하나의 차량 당 출발 및 배송 후 회송을 의미하는 2개의 예지로 연결되어야 하고, 또 도킹포인트는 차량 용량을 감안하여,  $r_{Q_2}(Y_p)$ 의 숫자 즉 그만큼 필요한 차량이 각각 출발 및 배송 후 회송하여야 한다. 따라서, 제약식 (1)은  $Y_p$ 로 이뤄지는 K 트리의 해에서 도킹 포인트가  $r_{Q_2}(Y_p)$ 의 2 배수의 예지를 가져야함을 보장한다. 제약식 (2)는 도킹포인트가 아닌 모든 터미널이 들어오고 나오는 차량 로를 의미하는 2개의 예지를 가지고 있어야 함을 의미한다. 제약식 (3)은 널리 알려진 sub-tour 방지 제약식으로 모든 가능한  $Y_p$ 의 부분집합에 연결된 차량루트는 가용한 적재 용량 제한  $Q_2$ 를 가지고 있는 차량이 그 부분집합의 모든 터미널의 로드를 감당할 수 있을 만큼의 숫자만큼 들어오고 나와야 함(필요한 차량수의 2배수)을 의미한다. 이것은 단순한 Traveling Salesman Problem의 sub-tour 방지 제약식에서 우변이 2(차량 수는 1)로서 단독으로 단절된(disconnected) 부분 그래프가 존재하지 않도록 하는 식이 보다 일반화된 제약식이다. 제약식 (4)~(6)은 위 제약식 (1)~(3)과 마찬가지로 도킹 포인트 집합인 R 과 물류기지로 이루어진 K 트리가 feasible한 VRP 해가 되어야 함을 보장한다. 제약식 (7)은 각  $Y_p$  집합안에서 오직 하나만이 도킹 포인트가 될 수 있음을 의미한다. 제약식 (8)은 하나의 도킹 포인트에 배정될 터미널들의 화물 크기의 합계는 하나의 도킹 포인트가 다룰 수 있는 최대 화물 용량 C에 의해 제약받고 있음을 의미한다.

### 3. 부분 모델의 해법

여기서 P와 Y는 4장 군집화 알고리즘에서의 해로 주어졌다고 가정된다. 우리는 이 군집 Y를 가지고, 문제 [P]의 해를 찾으려고 시도한다. 그리고 도킹 포인트들의 집합(집합 R)은 곧 기술될 두 번째 문제 [ $SP^{main}$ ]의 해로서 주어지는데, [ $SP^{main}$ ] 해법 중에 설명되었지만, 특정 군집내의 모든 터미널들이 한번은 도킹 포인트로서 시도되어, 그 첫 번째 문제 [ $SP^{sub}$ ]의 해를 생산하는 것이 [ $SP^{main}$ ]를 푸는 전제가 되므로, 첫 번째 문제를 먼저 기술한다(해법의 흐름도 [그림 3]를 참조). 그리고, 문제 [P]에서 제약식 (8)은 4장 군집화 알고리즘에서 고려될 것이므로 제거된다. 그러면 문제 [P]는 2개의 부분으로 쪼개진다. 그 첫 번째 문제는 다음과 같이 정식화된다.

[ $SP^{sub}$ ]

$$\sum_{\beta=1}^P \text{Min}_{R, y^p} \left[ \sum_{i, j \in Y_p} c_{ij} y_{ij}^p \right]$$

s.t.

$$\sum_{i \in R \cap Y_p} \sum_{j \in Y_p} y_{ij}^p = 2r_{Q_2}(Y_p) \quad \text{for } p=1, 2, \dots, P \quad (10)$$

$$\sum_{j \in Y_p} y_{ij}^p = 2 \quad \text{for all } i \in Y_p \notin R, p=1, 2, \dots, P \quad (11)$$

$$\sum_{i \in S} \sum_{j \in S} y_{ij}^p \geq 2r_{Q_2}(S) \quad \text{for all } S \subset Y_p \text{ and } i \notin R \quad \text{with } |S| \geq 2, p=1, 2, \dots, P \quad (12)$$

$$y_{ij}^p \in \{0, 1\} \quad \text{for all } i, j, p=1, 2, \dots, P \quad (13)$$

이 문제는 다시 작은 부차적인 문제로 분해되어 다음과 같이 각  $Y_p$ 에 대해서 풀릴 수 있다.

[ $SP_p^{sub}$ ]

$$\text{Min}_{y^p} \left[ \sum_{i, j \in Y_p} c_{ij} y_{ij}^p \right] \quad \sum_{i \in R \cap Y_p} \sum_{j \in Y_p} y_{ij}^p = 2r_{Q_2}(Y_p) \quad (14)$$

$$\sum_{j \in Y_p} y_{ij}^p = 2 \quad \text{for all } i \in Y_p \notin R, \quad (15)$$

$$\sum_{i \in S} \sum_{j \in S} y_{ij}^p \geq 2r_{Q_2}(S) \quad \text{for all } S \subset Y_p \text{ and } i \notin R \quad \text{with } |S| \geq 2 \quad (16)$$

$$y_{ij}^p, y_{ij}^p \in \{0, 1\} \quad \text{for all } i, j, p=1, 2, \dots, P \quad (17)$$

미리 주어진  $Y_p$ 에 대해서, 이것은 각 도킹 포인트  $i \in R \cap Y_p$ 를 차량 출발기지로 하는 VRP의 해를 구하는 문제로 귀착된다. 이 VRP 문제에 대해서는 많은 해법이 있으나(대표적인 것으로 Agarwal *et al.*[1], Bramel *et al.*[5], Wright *et al.*[6], Desrosiers *et al.*[7], Fisher *et al.*[11], Liu and Shen[14] 등이 있다), 본 논문 마지막 절의 계산실험에서는 Liu and Shen[14]에 있는 휴리스틱 해법을 사용한다.

$k_p \in R$  and  $k_p \in Y_p$ 일 때, [ $SP_p^{sub}$ ]의 해를  $S_p(k_p)$ 라고 하면, 우리는 본 문제 [P]의 두 번째 부분 문제인 [ $SP^{main}$ ]를 다음과 같이 정식화할 수 있다. 여기서 첫 번째 부분 문제의 전제가 되는 도킹 포인트들의 집합 R을 구한다. 위의  $S_p(k_p)$  값은 모든 터미널  $i \in Y_p$ 를 잠재적 도킹 포인트로 가정하여, 각각의 터미널에 대해 미리 구하여 아래 문제에 대입한다.

[ $SP^{main}$ ]

$$\min_{R=\{k_1, k_2, \dots, k_p\}} \left\{ \left[ \sum_{\beta=1}^P S_{\beta}(k_{\beta}) + f_{k_{\beta}} \right] + \min_{y^p} \sum_{i < j \in R \cup \{0\}} c_{ij} y_{ij}^p \right\}$$

s.t.

$$\sum_{i \in R} y_{0i}^p = 2r_{Q_1}(N) \quad (18)$$

$$\sum_{j \in R \cup \{0\}} y_{ij}^p = 2 \quad \text{for all } i \in R \quad (19)$$

$$\sum_{i \in S} \sum_{j \in S} y_{ij}^p \geq 2r_{Q_1}(S) \quad \text{for all } S \subset R, \text{ with } |S| \leq 2 \quad (20)$$

$$\sum_{j \in Y_p} y_{ij}^0 = 0 \quad \text{for all } i \in R \cap Y_p, \text{ for } p=1, 2, \dots, P \quad (21)$$

$$y_{ij}^0, y_{ij}^p \in \{0, 1\} \quad \text{for all } i, j, p=1, 2, \dots, P \quad (22)$$

제약식 (18)~(22)는 R에 속하는 각 노드와 물류기지로 이루어지는 K 트리가 feasible한 VRP 해가 되도록 보장한다. 이 문제는 군집 Y가 주어졌을 때, 각 군집  $Y_p$ 에서 선택된 유일한  $k_p$ 를 구하여, 그들과 물류기지가 이루는 K 트리의 수송비용과 각  $k_p$ 의 설치비용에, 이들과 함께 각  $k_p$ ,  $p=1, 2, \dots, |p|$ 에서 출발하는 2차 차량의 수송 비용 기여합  $\sum_{p=1}^P S_p(k(p))$ 을 최소화하는 것이다. 다시 말해서 여기서 해가 구해지면 그 해는 곧 원 문제 [P]의 최종해가 된다.

- 1) 이 문제의 해법을 설명하기 전에 먼저 제약식 (18)~(20)을 제외한 아래 [SP<sup>main</sup>] 문제의 해법을 고려해보자. 이 문제의 해는 아래의 동적 계획법(Dynamic Programming : DP)을 적용함으로써 얻어진다.  $S_p(\emptyset) = 0$ 으로 정한다.

[SP<sup>main</sup>]

$$\min_{R, y_{ij}^0} \text{ s.t. } i, j \in R \cup \{0\} [f_j + c_{ij} + S_p(j)] y_{ij}^0 \quad (23)$$

s. t.

$$\sum_{j \in Y_p, j \neq i} y_{ij}^0 = 0 \quad \text{for all } i \in R \cap Y_p, \text{ for } p=1, 2, \dots, P \quad (24)$$

$$y_{ij}^0 \in \{0, 1\} \quad \text{for all } i, j, p=1, 2, \dots, P \quad (25)$$

다음  $KT_{m,L}(E)$ 을 m단계에서 L개의 에지들(=  $EC(CE(E))$ )를 가진 minimum K 트리의 최소의 식 (23) 값을 주는 해라고 정의한다. 여기서  $CE(E)$ 는 다음의 조건을 만족하는 집합으로 정의된다.

(1)  $e \in CE(E)$ 이면, 에지들의 집합  $\{e\} \cup E$ 는 환을 이루지 않는다.

(2) 이 집합 에지들의 한 끝점이 집합  $Y_p$ 에 속할 때, 다른 끝점은 같은  $Y_p$ 에 속하지 않는다.

그러면 다음의 동적 계획법이 적용된다.

$$KT_{m,L}(E \cup \{e\}) = \min_{e \in CE(E)} C(e) + KT_{m-1, L-1}(E) \text{ when } m \geq 2$$

$$KT_{0,1}(\{e\}) = \min_{e \in N} C(e) \text{ when } m < 2$$

$$C(e) = c_{ij} + f_j + S_q(j) \text{ where } e = (i, j),$$

$$i \in Y_p \text{ and } j \in Y_q \text{ when } m \geq 2$$

$$C(e) = c_{0j} + f_j + S_p(j) \text{ where } e = (0, j),$$

$$j \in Y_p \text{ when } m = 1$$

이 DP 해는 규모가 매우 큰 (huge) 해 공간을 갖는다. 해 공간을 크게 줄여 DP를 풀기 위해 우리는 다음의 알고리즘(Greedy Algorithm)을 개발했다.

단계 1 : 중심물류기지에서부터 모든 노드  $i \in N$ 에 대해 [SP<sup>sub</sup>]의 해  $S_{p(i)}$ 를 찾아서  $KT_{0,1}$ 의 해를 구한다. 모든 해와 해당하는 에지들을 집합 T와  $E_L$ 에 각각 넣는다. 그리고  $L=1$ 로 놓는다.

단계 2 : T에서 가장 작은  $L^s$ 개의 해를 찾아 T에서 제거하고, 동일한 값을 가진 최소 해(들)을  $KT_{m,L^1}(E_{L^1}), KT_{m,L^2}(E_{L^2}), \dots, KT_{m,L^s}(E_{L^s})$ 로 표시한다.

단계 3 :  $L = n-1$ 이 되면 중지한다. 해  $E_{L^1}, E_{L^2}, \dots, E_{L^s}$ 를 저장한다.

단계 4 :  $KT_{m+1, L+1}(E_{L^s} \cup \{e\})$ ,  $s=1, \dots, S$ 의 해를 구하여, T에 저장하고, 그 해(들)에 해당하는 에지(들)를 집합  $E_{L+1}$ 로 저장한다.  $L = L+1$ 로 놓는다. 그리고 단계 2로 간다.



2) 위의 알고리즘을 통해서 얻은  $[SP^{main}]$ 의 DP 해는 우리에게  $R$ 과 이 도킹포인트 집합과 물류 기지를 spanning 하는 최소 트리를 준다. 우리가 전에 제거하였던 제약식 (18)~(20)을 다시 복원하면, 우리는 VRP의 feasible한 해 속에서  $R$ 을 구할 필요가 있다. 제약식이 복원된  $[SP^{main}]$ 에 대해서 기존에 알려진 좋은 해법은 없다. 우리는 제약식을 복원하여 직접 문제를 풀지 않는다. 궁극적으로 제약식이 복원된  $[SP^{main}]$ 도 우리에게 주는 해는  $R$ 이며, 또 그  $[SP^{main}]$ 의 해인  $R$ 이 만드는 트리가  $R$ 의 VRP 해의 근사치임을 착안한다.  $[SP^{main}]$ 에서 여러 개의 이러한 좋은 근사치를 주는  $R$ 을 구한다면, 그  $R$ 에 VRP의 해법을 적용하여 구한 여러 해 중에서 최소값으로  $[SP^{main}]$ 의 휴리스틱해를 구할 수 있음에 착안하고자 한다. 우리는 이러한 방법을 위하여, 약간 변경된 DP 알고리즘을 통해서  $[SP^{main}]$ 의 최적 해만이 아니라 여러 개의 다수 해를 특정 기준에 의해 구하고(단계 3), 다시 이를 이용해  $[SP^{main}]$ 의 휴리스틱해를 구한다. 상세한 알고리즘은 다음과 같다.

단계 1 : DP 알고리즘 1단계와 같다.

단계 2 : DP 알고리즘 2단계와 같다.

단계 3 :  $L = n - 1$ 이고, 해의 값이 직전 해와 비교해서  $S\%$  이내에 있으면 이를 직전 해로 그 값과 예지 집합  $E_L^*$ 를 가능한 해 중의 하나로 저장하고 단계 2로 간다.  $L = n - 1$ 이고 그 해가 이 해의 값이 직전 해와 비교해서  $S\%$  바깥에 있으면 중지한다. 본 논문의 알고리즘을 테스트하는 계산실험에서  $S$ 는 10으로 고정되었다.

단계 4 :  $KT_{m+1, L+1}(E_L \cup \{e\}), s = 1, \dots, S$ 의 해를 구한다. 그 최소해 및 해당하는 예지를 각각  $T$  및 집합  $E_L$ 에 더한다. 그리고 2 단계로 간다.

단계 5 : 모든 저장된 가능한 해들에 대해서, VRP 휴리스틱을 적용하고 그 해중에서 최소의 것을 골라  $[SP^{main}]$ 의 휴리스틱 해로 결정한다.

## 4. 군집화 알고리즘

여기서 초기 군집화의 알고리즘과 이들 군집과 3장의 알고리즘을 동시에 이용하여 지속적으로  $[P]$ 의 해를 개선해나가는 전체적인 휴리스틱 알고리즘을 제시한다. 아래에서  $LOAD$  값은 300, 600, 1000, 1500, 2000로 주어지고 이들 주어진 각각의 값에 대해 아래 알고리즘을 적용, 최적해를 구한다.

### 4.1 초기 군집화 알고리즘

단계 1 : 모든 터미널을 물류기지와 터미널 사이의 래디안 각이 증가하는 순서대로 정렬한다.  $l = 1$ 로 초기 세팅한다.

단계 2 : 터미널을 위에서 정해진 순서대로 세어  $\sum_{j=1}^L a_j \leq LOAD$ 인 최대의  $L$ 을 정한다. 여기서  $LOAD$ 는 외부에서 정해진 상수이다. 여기서 세어진 모든 터미널을 한 그룹으로 묶는다.  $l = L$ 로 세팅하고 단계 2를 모든 터미널에 대해 반복 적용하여, 모든 터미널을 그룹핑한다.

단계 3 : 단계 2까지에서 얻어진 군집들에 대해서 3장의 알고리즘을 적용하고 그 해를  $Sol_{LOAD}$ 로 놓는다.

### 4.2 군집 개선 알고리즘

단계 4 : 특정 군집  $c$ 종의 아직 고려되지 않은 터미널  $r$ 을 택해 다른 군집  $c' \neq c$ 의 모든 터미널사이의 위치를 고려한다. 즉 군집  $c'$ 가 이루고 있는 차량 순회 경로(Vehicle Routing Problem)의 해 : 3장  $[SP_c^{sub}]$ 의

해  $S_c(r_c)$  중의 모든 터미널 사이의 위치 중에서  $r$ 이 그 위치에 삽입되었을 때, 기존의  $c$ 와  $c'$ 가 이루는 차량 순회 비용합  $S_c(r_c) + S_{c'}(r_{c'})$ 이 가장 큰 절약을 가져오는 위치를 선택하여 그 위치에 삽입한다. 그리고 만약 그런 절약을 가져오는 위치가 없으면 다음의 고려되지 않은  $c'$ 을 고려한다. 이와 같이 모든 다른 군집  $c'$ 를 고려한다.

단계 5: 군집  $c$  중의 모든 터미널  $r$ 을 택하여 단계 5를 반복한다.

단계 6: 모든 군집을  $c$ 로 택하여 단계 4와 단계 5를 반복한다.

단계 7: 단일한 도킹 포인트만 가지고 그에 할당된 터미널이 없는 군집  $c$ 의 도킹 포인트  $r_c$ 를 다른 군집  $c'$ 로 이동시켜 가장 적은 차량 순회 비용을 가지는 위치로 삽입하여 차량 순회 비용과 도킹 포인트 설치 비용 합이 줄어드는가를 체크하여, 총 비용의 감소가 있으면,  $r_c$ 를 이동시킨다. 즉  $f_{r_c} + 2 * c_{0r_c} > S_{c \cup (r_c)}(r_c)$ 이면  $r_c$ 를 이동시킨다.

단계 8: 단계 4에서 단계 7까지 REP횟수만큼 반복하거나, 더 이상 총 비용의 감소가 없으면 현재 해를  $Sol_{LOAD}$ 로 놓고 다음 단계로 간다. 우리는 REP=3의 값을 계산실험에 이용하였다.

단계 9: 다음 3장의 알고리즘을 단계 8까지 얻어진 군집에 적용한다. 이것의 해가 현재의  $Sol_{LOAD}$ 보다 작고, 단계 8까지의 과정이 특정상수 REP1만큼 반복되지 않았으면,  $Sol_{LOAD}$ 를 갱신하고 여기서 얻어진 각 군집의 도킹 포인트를 가지고 단계 4로 간다. 이 해가  $Sol_{LOAD}$ 보다 크거나 단계 11까지의 과정이 REP1만큼 반복되었으면  $Sol_{LOAD}$ 를 최종해로 하여 멈춘다.

우리는 계산실험에서 REP1=3의 값을 사용하였다.

## 5. 계산 실험

우리의 알고리즘이 휴리스틱이므로 그 품질을 파악하기 위한 방법은 최적해를 비교하는 것이 가장 이상적이다. 그런데 본 모델의 고도의 combinatorial한 특징 때문에 합리적인 시간내에 최적해를 구한다는 것은 불가능하다. 그리고 본 논문의 최적 수송망 설계와 일치하는 모델은 기존의 어떤 연구도 제안한 바가 없어, 본 모델과 동일한 기존의 모델을 푸는 휴리스틱과 비교하여 휴리스틱의 품질을 평가하는 것도 불가능하다.

우리는 다른 대안으로서, 순회경로를 쓰지 않는 [그림 1]의 2단계 네트워크 최적화를 이용하였다. 이 경우, 크로스 도킹 포인트 후보가 정해지면, 도킹 포인트 설치여부와 설치된 도킹 포인트에 터미널을 배정하는 문제의 최적화는, Bramel *et al.*[5]에 묘사된 바 있는 아래의 용량 제약 저장고 문제 (Capacitated Concentrator Location Problem) 모델 [HP]을 이용하여 구할 수 있고, 이 모델은 터미널의 수가 크지 않는 경우, 정수 계획법을 사용하여 최적의 해를 구할 수도 있다. 우리는 CPLEX v9.0를 이용 최적해(터미널수(n)=150))를 구하였다.

[HP]에서 후보 도킹 포인트( $j$  인덱스로 표시됨, 그 집합은 Seed로 표기)를 사용하며, 그 후보 도킹 포인트  $j$ 와 터미널  $i$ 와의 거리를  $2c_{ij}$ 의 왕복거리를 연결비용(Connection Cost)으로 설정한다. 또 물류 기지에서 후보도킹 포인트까지의 왕복거리  $2c_{0j}$ 와 도킹 포인트 설치비용( $f_j$ )의 합이 후보 도킹 포인트  $j$ 의 고정 비용으로 이용된다. 후보 도킹 포인트가 설치되면  $y_j=1$ 이며, 그렇지 않으면,  $y_j=0$ 으로 결정된다. 터미널  $i$ 가 후보 도킹포인트  $j$ 에 배정되면  $z_{ij}=1$ 이며 그렇지 않으면 0이다. 모델에서 제약식 (26)은 하나의 후보 도킹 포인트가 다룰 수 있는 용량(C)제약을 의미하며, 제약식 (27)은 최소

한 하나의 후보 도킹포인트에 터미널이 배정되어야 함을 의미한다.

[HP]

$$\text{Min}_{z_{ij}, y_j} \sum_{j \in \text{seed}} (f_j + 2c'_{0j})y_j + \sum_{j \in \text{seed}} \sum_i 2c_{ij} z_{ij}$$

st.

$$\sum_{i \notin \text{seed}} \sum_{j \in \text{seed}} a_i z_{ij} \leq Cy_j \quad \text{for } j \in \text{seed} \quad (26)$$

$$\sum_{j \in \text{seed}} z_{ij} - 1 = 0 \quad \text{for } i \notin \text{seed} \quad (27)$$

$$z_{ij}, y_j \in \{0, 1\} \quad (28)$$

이 모델은 정확히 [그림 1]에 묘사된 네트워크의 최적화에 해당된다. 이 모델에서 주어진 해의 값과 우리의 휴리스틱이 주는 해를 단순 비교한다면, 순회경로는 언제나 단순 선형왕복 경로(line-haul)에 비해 항시 운행 비용의 절감을 가져오므로 공정한 비교가 아니다. 그러므로 우리는 [HP]에서 나온 크로스 도킹 포인트들과 각 도킹 포인트에 배정된 터미널들의 군집에 대해, Vehicle Routing Problem을 적용하여 선형 경로를 순회경로로 바꾸어 비교한다. 우리는 이 휴리스틱의 해를 [H2]의 해로 표기한다.

이제 나머지 문제는 후보 도킹 포인트 집합(Seed)을 어떻게 정하는 것이 [H2]의 해를 최소화하는가이다. 이를 위해 다양한 방법을 시도하여(모든 터미널을 후보 도킹포인트로 하는 방법을 포함, 매우 다양한 방법이 시도됨) 최소해를 가져오는 아래의 알고리즘 [H2]를 고안하였다. 이 알고리즘에서 최적의 군집화를 위한 후보 크로스 도킹 포인트 집합(Seed) 다수가 주어지며, 각각의 Seed에 대응한 [HP] 최적화의 결과로 나온 크로스 도킹 포인트들과 각 도킹 포인트에 배정된 터미널들의 군집에 대해 VRP 휴리스틱을 적용시켜 다수의 해들을 구하였다. 이때 다수의 해중 최소값인 H2로 표기된 해와 우리의 해인 H1을 비교하는 방향으로 계산 실험을 실시하여, 우리 해법의 우수성을 증명하려 했다.

H2

단계 1: 최초로  $sn=1$ 로 설정하고 모든 터미널을 물류기지과 터미널 사이의 래디안각이 증가하는 순서대로 정렬한다.

단계 2: 다음에  $l=sn$ 으로 놓고, 터미널을 위에서

정해진 순서대로 세어  $\sum_{j \in L} a_j \leq LOAD$ 인 최대의  $L$ 을 정한다. 여기서  $LOAD$ 는 외부에서 정해진 상수이다.  $l, l+1, \dots, L$ 중에서 최소의 도킹 포인트 설치를 위한 고정비  $f_j$ 를 주는 터미널  $j$ 를 후보 도킹 포인트 집합의(seed) 한 원소로 정한다. 그리고 다시  $l=L+1$ 로 놓고 새로운 새로운  $L$ 을 정하고, 동일한 방법으로 후보 도킹 포인트를 고른다. 위 과정을 모든 터미널에 대해서 실시한다.

단계 3: 주어진 Seed를 적용한 [HP]의 최적해를 CPLEX를 통해 구한다. 이들 해로서  $z_{ij}=1$ 인 모든 터미널  $i$ 는  $y_j=1$ 인 도킹 포인트  $j$ 에 배정되어 군집을 이룬다.  $y_j=0$ 인 도킹 포인트  $j$ 는 이용되지 않는다.

단계 4: 위의 해에서 구해진, 이용되는 도킹 포인트와 그에 배정된 터미널에 대해 도킹 포인트를 중심으로한 2차 수송 네트워크의 Vehicle Routing Problem를 풀고, 또 이들을 이용하여, 이용되는 모든 도킹 포인트와 물류 기지 사이의 1차 수송 네트워크의 Vehicle Routing Problem를 풀어 식 (23)의 공식을 적용, 문제 [P]의 해를 구한다.

단계 5:  $sn$ 을 1 증가시키고 단계 2에서 단계 4를 되풀이 한다. 모든 터미널이  $sn$ 으로 한번은 시도될 때까지 단계 1에서 단계 4까지 실행한 후, 단계 4의 최소해를 H2로 정한다.

요약하면, [H2]는 [그림 1]의 네트워크 최적화 모델인 [HP]로서 [그림 2]의 모델을 근사하여 다수 반복하여 해를 구한 다음, 그 중 [그림 2]의 네트워크의

최소해를 주는 것을 H2로 택한다. 우리의 계산실험에서는  $LOAD = 400$ 으로 최대한 작게 설정하여 거의 모든 터미널이 한 번은 도킹 포인트 후보가 되게 하여, 가능한 [H2]의 최적해가 얻어지게 하였다.

## 5.1 상수(parameter) 설정과 문제 생성

먼저 우리는 컴퓨터에 의한 난수 발생에 의해 문제를 자동 생성한다. 문제를 기술한다면,

- (1)  $n$ 개의 터미널의 위치는  $2L \times 2L$  정사각형 안에 random하게 위치하고 있다.  $L=30$ 으로 하였다. 중심물류기지는 그 중심에 위치한다. 모든 터미널의 화물 크기는 [50 200]의 범위에서 일양(uniform) 확률 분포의 값을 가지고 있게 생성되었다.
- (2) 일반성을 상실하지 않는 범위 내에서, 두 종류의 후보 도킹 포인트  $j$ 의 설치 비용  $f_j$ 을 상정하였다. 한 가지는  $f_j=0$ 이며 다른 한 가지는  $f_j$ 가 [25 100] 사이에 일양 확률 분포의 값을 가지고 있는 경우이다.
- (3) 후보 도킹 포인트로서의 각 터미널과 중심 물류기지사이의 거리는 각 터미널과 터미널 사이의 거리의  $Adj$ 배가 되도록 하였다. 즉  $c_{ij} = Adj * c_{ij}$ 으로 상정하였다. 이것은 후보 도킹 포인트와 물류기지 사이의 일차 수송 네트워크를 통한 연결이 2차 수송 네트워크를 통한 방법보다 값싼 수송수단에 의하여 이루어지므로 이를 감안하기 위해서이다. 우리는  $Adj$ 를 0.5로 고정하여 모든 실험을 진행하였다.

모든 알고리즘은 C로 코딩되었고 실험은 Compaq Presario 2800, 256MB Ram에서 진행되었다. 다양한 상수의 변동에 따른 효과를 보고자 다음과 같은 계산 실험이 고안되었다.

- (1) 도킹 포인트 설치 비용 증감의 효과 : 모든 case에 포함됨.

- (2) 터미널 수 증감의 효과 : case 1 vs. case 2 vs. case 3
- (3) 한 도킹 포인트가 다룰 수 있는 용량 변동 : case 2 vs. case 4

각 경우에 대해 우리는 앞에서 언급된 바와 같이, 두 가지의  $f_j$  값에 대한 실험을 진행하였다. case 4의 경우 개별 운송 수단의 용량( $Q_2$ )과 한 도킹 포인트가 다룰 수 있는 용량( $C$ )이 두 배로 증가하는 효과를 보는 경우이다. 각각의 case에 적용된 상수들은 다음과 같다. 각 케이스별로 10개의 문제를 풀어, H1과 H2/H1 값을 각각 표(<표 1>~<표 4>)로 정리하였다.

$$\text{Case 1 : } n=50, C=2000, Q_1=10000, \\ Q_2=700, a_j=UNIF[50.0, 200.0]$$

$$\text{Case 2 : } n=100, C=2000, Q_1=10000, \\ Q_2=700, a_j=UNIF[50.0, 200.0]$$

$$\text{Case 3 : } n=150, C=2000, Q_1=10000, \\ Q_2=700, a_j=UNIF[50.0, 200.0]$$

$$\text{Case 4 : } n=100, C=4000, Q_1=10000, \\ Q_2=1400, a_j=UNIF[50.0, 200.0]$$

## 5.2 계산 실험의 결과

대부분의 경우, 우리는 H1이 그에 대응하는 알고리즘 H2보다 우월함을 발견할 수 있다. 부분적으로 H1이 H2보다 큰 값을 보여 열등함을 보이는 경우도 있으나, 최악의 경우에도 최대 4%이내로서 휴리스틱에 의한 결과로서 감안할 수 있는 수준이다. 아래에서 상수 변화의 효과를 살펴보자.

(1)의 도킹 포인트 설치 비용 증감의 효과를 본다면, case 3의 경우, 도킹 포인트 설치비용이 증가할수록( $f_j=0$ 에서  $f_j \sim [25 100]$ 로 될 때) 추가 비용 절감 효과가 뚜렷하다. 도킹 포인트 설치비용이 증가하면, 개별도킹 포인트에 배정되는 터미널 수가 설치비용 절감을 고려해 증가하여, 그 도킹 포인트로부터 출발하는 개별 운송 수단이 개개의 터미널을

<표 1> 계산실험 case 1 :  $n=50, C=2000, Q_1=10000, Q_2=700, a_j=UNIF [50.0, 200.0]$

문제	$f_j=0$		$f_j \sim UNIF \sim (25.0, 100.0)$	
	H1	H2/H1	H1	H2/H1
1	507.83	1.35	1100.54	1.08
2	563.72	1.09	1009.72	1.15
3	577.34	1.08	1031.78	1.09
4	548.25	1.15	1082.58	1.09
5	580.82	1.05	1089.86	1.05
6	596.12	1.03	1095.26	1.05
7	685.12	0.97	1060.84	1.04
8	601.12	0.98	924.81	1.23
9	575.76	1.10	1063.78	1.08
10	627.31	0.99	1093.70	1.08
평균	586.34	1.08	1055.29	1.09

<표 4> 계산실험 case 4 :  $n=100, C=4000, Q_1=10000, Q_2=1400, a_j=UNIF [50.0, 200.0]$

문제	$f_j=0$		$f_j \sim UNIF \sim (25.0, 100.0)$	
	H1	H2/H1	H1	H2/H1
1	551.70	1.19	904.81	1.28
2	557.53	1.15	927.97	1.25
3	586.80	1.12	779.05	1.43
4	552.65	1.20	963.53	1.21
5	582.80	1.06	888.67	1.25
6	596.12	1.03	1054.46	1.11
7	597.54	1.12	772.76	1.44
8	606.60	1.07	1062.81	1.09
9	597.13	1.06	601.70	1.82
10	635.15	1.05	727.70	1.55
평균	586.40	1.11	868.34	1.34

<표 2> 계산실험 case 2 :  $n=100, C=2000, Q_1=10000, Q_2=700, a_j=UNIF [50.0, 200.0]$

문제	$f_j=0$		$f_j \sim UNIF \sim (25.0, 100.0)$	
	H1	H2/H1	H1	H2/H1
1	507.83	1.35	1100.54	1.08
2	563.72	1.09	1009.72	1.15
3	577.34	1.08	1031.78	1.09
4	548.25	1.15	1082.58	1.09
5	580.82	1.05	1089.86	1.05
6	596.12	1.03	1095.26	1.05
7	685.12	0.97	1060.84	1.04
8	601.12	0.98	924.81	1.23
9	575.76	1.10	1063.78	1.08
10	627.31	0.99	1093.70	1.08
평균	586.34	1.08	1055.29	1.09

<표 3> 계산실험 case 3 :  $n=150, C=2000, Q_1=10000, Q_2=700, a_j=UNIF [50.0, 200.0]$

문제	$f_j=0$		$f_j \sim UNIF \sim (25.0, 100.0)$	
	H1	H2/H1	H1	H2/H1
1	785.94	1.10	1370.45	1.08
2	709.28	1.14	1399.47	0.99
3	696.10	1.17	1478.64	1.05
4	831.34	0.99	1395.72	1.06
5	807.19	1.03	1430.07	1.02
6	835.38	0.98	1507.33	1.01
7	834.82	0.99	1453.08	0.99
8	866.91	0.98	1414.47	1.04
9	788.26	1.03	1379.34	1.10
10	842.48	1.00	1466.68	0.97
평균	799.77	1.04	1429.53	1.03

위해 운행해야 할 평균 거리는 더욱 크게 될 것이다. 이 경우, 흔적을 하지 않고 하나의 터미널만 왕복하는 전통적인 수송네트워크 모델에 기반한 해인 H2에 비해, 다수의 터미널을 순회하는 우리의 모델에 기반한 H1의 우월성이 더욱 드러나는 경우라고 해야 하겠다. 즉 개별 운송수단이 하나의 터미널이 아닌 다수의 터미널을 운행함으로써 발생하는 거리 운행 비용 절감이 더욱 커지므로, 그에 기반한 H1의 우월성이 더욱 드러난다고 할 것이다. 그 밖의 경우 H1의 우월성은 도킹 포인트 설치비용 증가에 따라 조금 감소하거나 거의 같다. 이들 경우는, 전체 터미널의 수가 작기 때문에, 동일 운송 수단이 배송하는 터미널 그룹들의 숫자가 작아서 운행 비용 절감 효과가 크지 않거나(case 1), 아니면 전체적으로 운행 비용 절감 효과가 그리 크게 나타나지 않는 경우라고 하겠다. 우리의 해에서는 3장의 해법에서 S값을 증가시킴으로서 H1의 휴리스틱 해의 품질을 더욱 좋게 할 수 있다.

(2)의 터미널 수 증감 효과의 경우, 터미널 수인  $n$ 이 증가할수록 H1의 H2에 대한 우월성은 약해지는데, 이는 터미널 수가 증가하면 할수록, 터미널의 위치는, 고정된  $2L \times 2L$  정사각형안에 위치하게 되므로, 좁은 면적에 보다 많이, 즉 뻥뻥히 터미널이 들어서게 되는 경우가 되어, 개별 운송 수단이 운행하

는 운행 거리는 더욱 짧아지기 때문이다. 즉 (1)의 경우에서 언급된 운행 비용 절감에 기반한 H1의 우월성이 약해지는 경우라고 하겠다. 그러나  $n$ 의 값이 증가하더라도, H1이 H2에 비해 열등하게 되지는 않음을 알 수 있다.

(3)의 한 도킹 포인트가 다룰 수 있는 용량 변동 효과를 본다면, <표 4>에서 개별 도킹 포인트가 다루는 화물용량과 2차 수송수단의 용량이 커진 경우에는, 앞에서 언급한 바와 같이 개별 운송 수단이 개개의 터미널을 위해 운행해야 할 평균 거리가 커지고, 개별 운송 수단이 배송해야 할 터미널의 숫자도 많아지므로 운행 비용 절감효과가 더 커져 H1의 우월성이 확대되는 경우라고 하겠다.

이상에서 본 바와 같이 본 논문이 제시하는 알고리즘은 하나의 개별 운송 수단이 개개의 터미널을 위해 운송해야 할 평균 거리가 증가하면 할수록 그 우수성이 더욱 드러난다고 할 것이다. 터미널 수가 증가하면, S값을 증가시켜 해의 품질을 좋게 할 수 있다. 계산시간이 문제가 될 수도 있는데,  $n=150$ 의 경우(case 3) 최대 10분이며, H1이 휴리스틱임을 감안할 때, 장기적인 전략의 결정 수단으로 큰 무리가 없다.

## 6. 결 론

2단계 수송 네트워크에서 고정 비용과 수송 비용을 포함한 가변 비용 최소화 모델은 기존의 연구 성과가 있으나, 혼적 개념이 포함된, 한 차량이 여러 터미널을 방문하는 Vehicle Routing을 고려한 2단계 수송 네트워크 [그림 2]상에서의 고정 및 가변 비용 최소화의 모델은 본 논문이 처음으로 시도한다. 그것의 실용적인 의미는 2단계 수송 네트워크를 설계할 때, 장기적으로 크로스 도킹 포인트의 위치를 어디에 설정하여야 그 도킹 포인트의 설치 비용 및 수송 비용 합을 최소화할 것인가에 대한 의사 결정 지원에 있다고 할 것이다. 본 논문은 이 모델의 근사 해법을 제시하고, 그 품질을 컴퓨터 계산실험을 통해 검증하였다.

## 참 고 문 헌

- [1] Agarwal, Y., K. Mathur, and H.M. Salkin, "A Set-Partitioning-Based Exact Algorithm for the Vehicle Routing Problem," *Networks*, Vol.19(1989), pp.731-749.
- [2] Apte, U.M. and S. Viswanathan, "Effective Cross Docking for Improving Distribution Efficiencies," *International Journal of Logistics*, Vol.3(2000), pp.291-302.
- [3] Armacost, A.P., C. Barnhart, and K.A. Ware, "Composite Variable Formulations for Express Shipment Service Network Design," *Transportation Science*, Vol.36(2002), pp.1-20.
- [4] Aykin, T., "Networking Policies for Hub-and-Spoke Systems with Application to the Air Transportation System," *Transportation Science*, Vol.29(1995), pp.201-220.
- [5] Bramel, J. and D. Simchi-Levi, "Location Based Heuristic for General Routing Problems," *Operations Research*, Vol.43(1995), pp.649-660.
- [6] Clarke, G. and J. Wright, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points," *Operations Research*, Vol.12(1964), pp.568-581.
- [7] Desrosiers, J., F. Soumis, and M. Desrochers, "Routing with Time Windows by Column Generation," *Networks*, Vol.14(1984), pp.545-565.
- [8] Desrochers, M. and T.W. Verhoog, "A New Heuristic for the Fleet Size and Mix Vehicle Routing Problem," *Computers and Operations Research*, Vol.18(1991), pp.263-274.
- [9] Fisher, M.L., "A Polynomial Algorithm for the Degree-constrained Minimal K-tree Problem," *Operations Research*, Vol.42, No.4 (1994), pp.775-779.
- [10] Fisher, M.L., "Optimal Solution of Vehicle

- Routing Problems Using Minimal K-trees," *Operations Research*, Vol.42, No.4(1994), pp.626-640.
- [11] Fisher, M. and R. Jaikumar, "A Generalized Assignment Heuristic for Vehicle Routing", *Networks*, Vol.11, No.2(1981), pp. 109-124.
- [12] Golden, B.L., A. Asad, L. Levy, and F.G. Gheysins, "The Fleet Size and Mix Vehicle Routing," *Networks*, Vol.7(1984), pp.113-148.
- [13] Kim, D., C. Barnhart, K. ware, and, G. Reinhardt, "Multimodal Express Package Delivery : A Service Network Design Application," *Transportation Science*, Vol.33(1999), pp.391-407.
- [14] Liu, F.-H. and S.Y. Shen, "The Fleet Size and Mix Vehicle Routing Problem with Time Windows," *Journal of Operational research Society*, Vol.50(1999), pp.721-732.
- [15] Minoux, M., "Network Synthesis and Optimum Network Design Problems : Models, Solution Methods and Applications," *Network*, Vol.19(1989), pp.313-360.
- [16] Powell, W.B., "A Local Improvement Heuristic for the Design of Less-than-Truckload Motor Carrier Networks," *Transportation Science*, Vol.20(1986), pp.246-257.
- [17] Powell, W.B. and Y. Sheffi, "Design and Implementation of a Interactive Optimization System for Network Design In the Motor Carrier Industry," *Operations Research*, Vol. 37(1989), pp.12-29.
- [18] Stalk, G.P. and L.E. Shulman, "Competing on Capabilities : The New Role of Corporate Strategy," *Harvard Business Review*, Vol. 70(1992), pp.57-69.
- [19] Sung, C.S. and H.W. Jin, "Dual-based Approach for a Hub Network Design Problem under Non-restrictive Policy," *European Journal of the Operational Research*, Vol.132 (2001), pp.88-105.
- [20] Sung, C.S. and S.H. Song, "Integrated Service Network Design for a Cross-docking Supply Chain Network," *Journal of the Operational Research Society*, Vol. 54(2003), pp.1283-1295.