The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 18, No. 3, pp.209-214, Jun. 30, 2018. pISSN 2289-0238, eISSN 2289-0246

https://doi.org/10.7236/JIIBC.2018.18.3.209
JIIBC 2018-3-27

부패상품 임시물류센터에 대한 위치 문제 개선

Improvement for Facility Location Problem of Perishable Commodities

이상운*

Sang-Un, Lee*

요 약 본 논문은 부패상품의 수송시간 제약조건 L^* 이내로 총 수송비용을 최소화시키는 물류센터 위치를 결정하는 문제의 단순한 알고리즘을 제안하였다. 이 문제에 대해 Lee[4]는 후보 물류센터 위치를 선정하고, 각 물류센터의 총 수송비용을 구한 후, 상위 2개 물류센터에 대해 최적의 위치로 이동시키는 복잡한 방법을 제안하였다. 반면에 본 논문에서는 서브트리 개념을 도입하여 L^* 이내가 되는 후보 물류센터의 위치를 간단히 결정하고, 근방 개념을 도입하여 요구량이 집중화된 지역 상위 2개 지역에 대해 총 수송비용이 최소화되는 지점을 단순히 최적의 위치로 결정하였다.

Abstract This paper suggests simple algorithm of facility location problem in perishable commodities that satisfy with minimum total transportation cost and within the transportation time constraint L^*. For this problem, Lee[4] suggests very complex algorithm that decides candidate facility locations, computes total transportation cost for each candidate facility location, then moving the location to optimal location for top two facilities. On the other hand, this paper simply determines the candidate facility locations within L^* using subtree concept, and decides optimal minimal total transportation cost for top two locations in centralized area of required quantity using neighborhood concept.

Key Words : Perishable Commodity, Transportation Time Constraints, Facility Location Problem

Ⅰ. 서 론

일반적으로, 어떤 제품이 운송중의 고온, 고습, 등의 원인으로 인해 부패 또는 변질의 위험이 있는 경우 부패하기 쉬운(perishable) 제품이라고 한다. 부패하기 쉬운 제품을 어떠한 지역에 공급하기 위해서는 양질의 상태를 유지시킬 수 있는 수송시간 제약이 가해지며, 이 시간 내에 모든 지역에 수송할 수 있는 임시 물류센터를 결정하는 것이 정책적으로 매우 중요한 사안이다.[1,2]

부패하기 쉬운 제품을 모든 요구 지역에 수송하기 위해서는 수송 시간 제약조건을 만족시켜야 하며, 또한, 수송비용을 최소화시킬 수 있는 임시 물류센터 위치를 결정해야 한다. 총 수송비용은 부패로 인한 손실비용과 수송비용의 합으로 계산된다.[1]

부패하기 쉬운 제품의 임시물류센터 위치를 결정하는 방법은 1-center problem, 1-median problem, 1-absolute center problem 방법 등이 적용되고 있다.[3] 그러나 이들 방법은 수송시간 제약사항을 고려하지 않고 단지 수송비용을 최소화 시키는 제약조건만을 만족시킨다.

Zhang과 Yang[1]은 각 지역에서 최대 거리 2 지점의

*정회원, 강릉원주대학교 과학기술대학 멀티미디어공학과
접수일자 : 2018년 4월 1일, 수정완료 : 2018년 5월 12일
게재확정일자 : 2018년 6월 8일

Received: 1 April, 2018 / Revised: 12 May, 2018 / Accepted: 8 June, 2018
*Corresponding Author: sulee@gwnu.ac.kr
Dept. of Multimedia Eng., Gangneung-Wonju National University, Korea

평균값이 수송제약시간보다 작은 지점에 대해 위치를 이동시키는 p-center 방법으로 총 수송비용을 최소화시키는 지점을 임시 물류센터 위치로 결정하였다. 그러나 Zhang과 Yang[1]이 제안한 방법은 임시 물류센터의 후보 위치들에서 각 요구 지점들 간의 최단거리 계산 오류로 인해 임시 물류센터 위치를 잘못 선정하는 오류를 범하였다. 이러한 Zhang과 Yang[1]의 p-center 방법의 오류를 바로 잡고, 정확한 위치를 결정할 수 있는 방법으로 Lee[4]은 L^* 제약조건을 만족하는 후보 물류센터 위치를 결정하고, 물류센터 위치들 중에서 총 배송비용 F_j가 최소인 지점을 최종적으로 결정하는 방법을 제안하였다. 그러나 이 방법은 후보 물류센터 위치를 결정하는 과정이 복잡하며, 최종적으로 결정된 2곳의 정확한 위치를 찾는 과정이 1Km 간격으로 많은 계산횟수를 필요로 하는 단점을 갖고 있었다.

본 논문은 서브트리(subtree) 개념을 도입하여 수송시간 제약조건 L^*을 충족시키는 정확한 위치를 빠르게 결정하고, 근방(neighborhood) 개념을 도입하여 요구량이 집중된 지점 2곳에 대해서만 총 수송비용 F_j가 최소치를 갖는 2곳 중 1곳을 단순히 결정하는 알고리즘을 제안한다.

2장에서는 부패하기 쉬운 제품의 총 수송비용을 계산하는 문제를 정의하고, Zhang과 Yang[1]과 Lee[4]의 방법을 고찰한다. 3장에서는 최적의 물류센터 위치를 결정하는 알고리즘을 제안하고, Lee[4]의 방법과 알고리즘의 단순성을 비교해 본다.

II. 관련연구와 문제점

부패하기 쉬운 상품을 v_i에서 v_j로 수송하는 상품의 초기수량 $W_{ij}(0)$에 대해 부패 속도가 일정한 경우 부패율은 식 (1) 미분방정식을 따른다.[4,5]

$$\frac{dW_{ij}(t)}{dt} = -\theta W_{ij}(t) \tag{1}$$

여기서 W_{ij}는 물류센터 v_i에서 요구 지역 v_j로 t 시점에서 부패되지 않은 상품의 수송물량이며, θ는 부패속도 계수이다. 차량 속도를 v_{ij}, $\alpha = 1/v_{ij}$, 최단 거리를 l_{ij}라 하면 v_i에서 v_j까지 소요시간 t_{ij}는 $t_{ij} = \alpha l_{ij}$가 된다. 따라서 v_j 지역에서의 요구량은 식 (2)로 표현된다.

$$W_{ij}(t_{ij}) = d_j \tag{2}$$

따라서 $W_{ij}(0) = e^{\theta t_{ij}} d_j$가 되며, h를 단위 수량당 수송비용이라 하면 v_i에서 v_j까지 수송비용(transportation cost) Ft_{ij}는 식 (3)으로 계산된다.

$$Ft_{ij} = \sum_{j=1}^{n} e^{\theta t_{ij}} h d_j l_{ij} \tag{3}$$

상품의 단가를 c라 하면 부패로 인한 손실 비용(perishability lost) Fd_{ij}는 식 (4)로 계산된다.

$$Fd_{ij} = \sum_{j=1}^{n} \alpha d_j (e^{\theta t_{ij}} - 1) \tag{4}$$

결국, i에 위치한 임시 물류센터에서 모든 지역 j까지의 총 수송비용 F_j는 $F_j = Ft_{ij} + Fd_{ij}$로 식 (5)로 표현된다.[4,6]

$$F_j = \sum_{j=1}^{n} [e^{\theta t_{ij}} h d_j l_{ij} + \alpha d_j (e^{\theta t_{ij}} - 1)] \tag{5}$$

그림 1은 Zhang과 Yang[1]이 제시한 부패하기 쉬운 상품의 지역별 요구량과 거리를 표현한 망이다[4]. 이 사례는 8개 지역으로 구성된 분배망에서 각 지역 간 이동가능 도로 거리 (Km)와 지역별 요구량 (demands) d_j가 주어졌다. 여기서 상품의 단가 (c)는 5, 수송 속도는 40 Km/h, 부패 상수 (θ)는 0.01, 수송단가 (h)는 3.0112, 최대 허용 수송시간 (R^*)은 4시간이다.

그림 1의 예제에서 $\alpha = 1/40 = 0.025$, 물류센터에서 모든 지역으로 최단거리로 배송한다고 가정하면 최대 허용거리 $L^* = R^*/\alpha = 160$Km이다.

Zhang과 Yang[1]은 이 예제에 대해 Dijkstra 알고리즘[7]을 적용하여 표 1과 같이 최단거리를 구하였다.[4]

Zhang과 Yang[1]은 1-Absolute Center Problem 방법[3]을 변형시킨 p-Center 방법을 적용하였다. 여기서 $p \geq 1$이다. 모든 지역에서의 최단거리가 최대치인 2개 지점 v_{i1}과 v_{i2} 까지 최단거리 값을 l_{i1}와 l_{i2}로 하여 $r(o_i) = (l_{i1} + l_{i2})/2$로 한다. $r(o_i) = (l_{i1} + l_{i2})/2 < L^*$인 지역 v_i를 선택하여 $(v_i, \cdots, v_{i1})$경로와 $(v_i, \cdots, v_{i2})$ 경로에 대해 v_i에서 $L^* - r(o_i)$ 만큼 떨어진 지점을 v_i'와 v_i''로 물류센터의 후보 $v_4'(v_4'')$, v_7, v_7', v_7''를 결정한다.

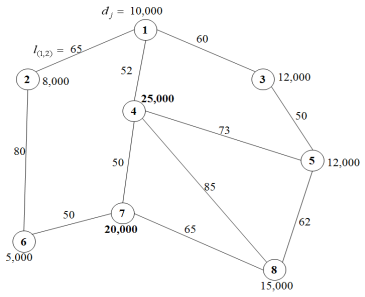


그림 1. 부패하기 쉬운 상품 분배 망
Fig. 1. Distribution Network for Perishable Commodities

표 1. 최단거리와 경로
Table 1. Shortest Distance and Path

$L(P)=$

	①	②	③	④	⑤	⑥	⑦	⑧
①	0	65 (1,2)	60 (1,3)	52 (1,4)	110 (1,3,5)	145 (1,2,6)	102 (1,4,7)	137 (1,4,8)
②	65 (2,1)	0	125 (2,1,3)	117 (2,1,4)	175 (2,1,3,5)	80 (2,6)	130 (2,6,7)	195 (2,6,7,8)
③	60 (3,1)	125 (3,1,2)	0	112 (3,1,4)	50 (3,5)	205 (3,1,2,6)	162 (3,1,4,7)	112 (3,5,8)
④	52 (4,1)	117 (4,1,2)	112 (4,1,3)	0	73 (4,5)	100 (4,7,6)	50 (4,7)	85 (4,8)
⑤	110 (5,3,1)	175 (5,3,1,2)	50 (5,3)	73 (5,4)	0	173 (5,4,7,6)	123 (5,4,7)	62 (5,8)
⑥	145 (6,2,1)	80 (6,2)	205 (6,2,1,3)	100 (6,7,4)	173 (6,7,4,5)	0	50 (6,7)	115 (6,7,8)
⑦	102 (7,4,1)	130 (7,6,2)	162 (7,4,1,3)	50 (7,4)	123 (7,4,5)	50 (7,6)	0	65 (7,8)
⑧	137 (8,4,1)	195 (8,7,6,2)	112 (8,5,3)	85 (8,4)	62 (8,5)	115 (8,7,6)	65 (8,7)	0

Zhang과 Yang[1]는 $1'', 4'(4''), 7', 7''$의 최단거리 계산 오류로 인해 물류센터의 위치를 v_4임에도 불구하고 v_7'로 잘못 결정하였다.

Lee[4]는 Zhang과 Yang[1]의 거리계산 오류를 바로잡고 물류센터 위치를 v_4로 정확히 계산하는 알고리즘을 제안하였다. Lee[4]의 물류센터 위치 알고리즘은 2단계를 수행한다. 첫 번째 단계는 $L^* = 160\,Km$제약조건을 만족하는 물류센터 위치를 결정하며, 두 번째 단계는 물류센터 위치들 중에서 총 배송비용 F_j가 최소인 지점을 결정한다.

L^* 제약조건을 만족하는 물류센터 위치는 다음과 같이 수행한다.

Step 1. 각 지역 $(v_i, v_j), 1 \le i \le 8, 1 \le j \le 8$간 최단거리를 Dijkstra 알고리즘[8]으로 계산한다.

Step 2. 최단거리 행렬 L의 각 행에 대해 최대값 $\max l_{ij} \le L^*$인 지점을 기본적인 후보 물류센터로 선택한다.

Step 3. $l_{ij} > L^*$를 갖는 지점에 대해 $l_{ij} > L^*$ 양 끝단 (v_i, v_j)을 기점으로 하는 경로를 선택한다. $l_{ij} > L^*$인 값의 경로 $(p_{ij} = v_i, v_k, \cdots, v_l, v_j)$에 대해 v_i에서 v_k방향으로 $m_{ij} = l_{ij} - L^*$ 만큼, v_j에서 v_l방향으로 $m_{ij} = l_{ij} - L^*$ 만큼 이동시킨 위치를 후보 물류센터에 추가로 포함시킨다.

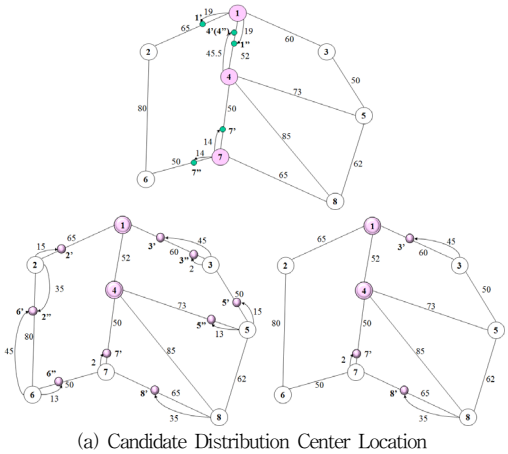
Step 4. 결정된 물류센터 각 지점으로부터 모든 요구지역까지의 최단거리 l_{ij}를 계산하여 $l_{ij} > L^*$이면 후보 집합에서 삭제한다.

총 배송비용 F_j가 최소인 지점을 최적의 위치로 결정하는 과정은 다음과 같이 수행한다.

Step 5. 최종 물류센터들 각각의 총 배송비용 F_j가 최소치인 지역 2곳을 선택한다.

Step 6. $v_4(25,000), v_6(5,000)$인 경우의 최적 해 후보지 (v_7', v_4)와 $v_4(5,000), v_6(25,000)$인 경우의 (v_4, v_7') 경로 상에서 1 Km 간격으로 F_j를 계산하여 최소치 지점을 최종 물류센터 위치로 결정한다.

Lee[4]의 방법을 요약하면 그림 2와 같다.



(a) Candidate Distribution Center Location

$L'=$

	1	2	3	4	5	6	7	8
1	0	65	60	52	110	145	102	137
3'	15	80	45	67	95	160	117	152
4	52	117	112	0	73	100	50	85
7'	100	132	160	48	121	52	2	67
8'	132	160	147	80	97	80	30	35

(b) Shortest Distance from Distribution Centers

후보 물류센터	$v_4(25,000), v_6(5,000)$		$v_4(5,000), v_6(25,000)$	
	F_j	최적해	F_j	최적해
v_1'	3.1669×10^6	v_4	3.5150×10^6	v_7'
v_1	2.6823×10^6		3.2708×10^6	
v_1''	2.4663×10^6		3.0943×10^6	
$v_4'(v_4'')$	2.6530×10^6		3.2848×10^6	
v_4	$\underline{1.9859 \times 10^6}$		2.6036×10^6	
v_7'	2.4110×10^6		$\underline{2.5840 \times 10^6}$	

(c) Total Transportation Cost of Distribution Center

그림 2. Lee[4] 방법
Fig. 2. Lee[4] method

Lee[4]는 물류센터 후보를 기본적으로 $\{1,4\}$를 얻고, 추가적으로 $\{2', 2'', 3', 3'', 5', 5'', 6', 6'', 7', 8'\}$를 추가하여 $\{1,2', 2'', 3', 3'', 4, 5', 5'', 6', 6'', 7', 8'\}$로 결정하였다. 다음으로 $\{2', 2'', 2'', 5', 5'', 6', 6''\}$를 삭제하여 최종적으로 $\{1, 3', 4, 7', 8'\}$로 결정하였다. 결정된 후보 물류센터 위치들 각각에서 요구 지역까지의 최단 거리 L'를 구하고, 총 배송비용 F_j의 최소치를 가진 2개 후보지를 결정한 결과 $v_4(25{,}000), v_6(5{,}000)$인 경우 (v_7', v_4) $v_4(5{,}000), v_6(25{,}000)$인 경우 (v_4, v_7')을 얻었다. 마지막으로 $v_4(25{,}000), v_6(5{,}000)$인 경우의 최적해 후보지 (v_7', v_4)와 $v_4(5{,}000), v_6(25{,}000)$인 경우의 (v_4, v_7') 경로 상에서 1 Km 간격으로 F_j를 계산하여 최소치 지점을 최종 물류센터 위치로 각각 v_4와 v_7'(v_7에서 2Km 떨어진 지점)를 결정하였다.

이와 같이 Lee[4]는 정확한 물류센터 위치를 결정하는 장점은 갖고 있지만 후보 물류센터를 결정하는 과정이 복잡하며, 최종적으로 결정된 후보지의 2곳에 대해 $l(v_4, v_7) = 50$ 경로상의 정확한 위치를 결정함에 있어 1Km 간격으로 F_j를 50회 수행하여 결정하는 매우 복잡한 과정을 거치는 단점을 갖고 있다.

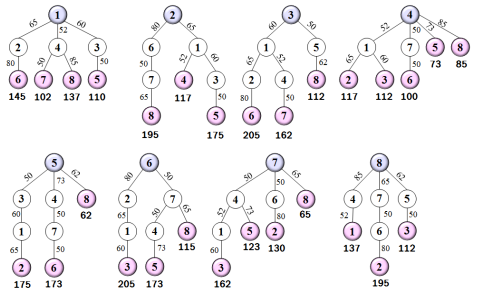
3장에서는 Lee[4]의 방법보다 간단히 해를 얻을 수 있는 알고리즘을 제안한다.

Ⅲ. 최단거리 트리를 이용한 물류센터 최적위치 결정 알고리즘

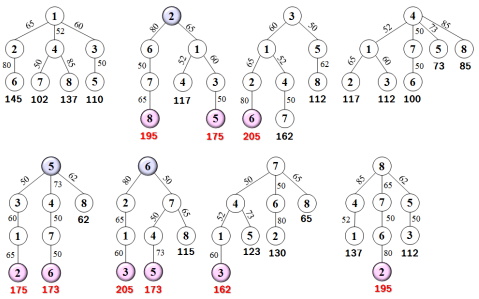
본 장에서 제안하는 최적의 물류센터 위치 결정 알고리즘은 $L^* = 160$를 충족시키는 정확한 후보 위치를 결정하고, 다음으로 후보 위치들 중에서 닫힌 이웃(closed neighborhood)의 요구량이 최대가 되는 상위 2개 지점(top 2)만을 대상으로 총 운송비용 F_j를 계산하여 최소치를 갖는 지점을 최적의 물류센터로 결정하는 단순한 방법을 적용한다. 제안된 알고리즘은 최단거리를 결정하는 Dijikstra's 알고리즘과 트리의 근 노드(root node)와 단 노드(terminal node) 개념을 결합시켜 적용한다.

Phase Ⅰ. 후보 위치 결정

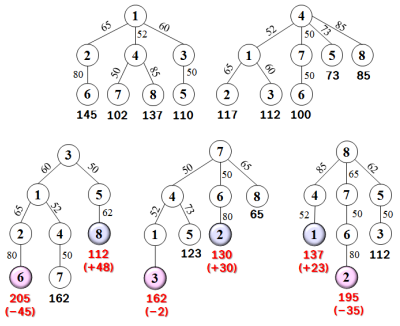
Step 1. 각 지역을 근 노드로 하는 나머지 다른 지역까지의 최단거리 트리(tree)를 형성한다.



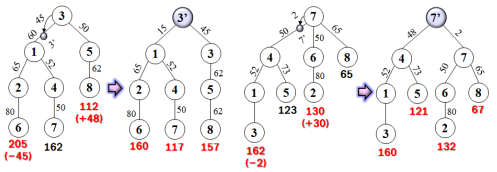
Step 2. Step 1 트리에 대해 근 노드를 제거한 부분트리(subtree)의 단 노드(terminal node)들 중 $L^* = 160$이상이 되는 단 노드 개수가 2개 이상인 지역(#2, #5, #6)은 후보에서 삭제한다.



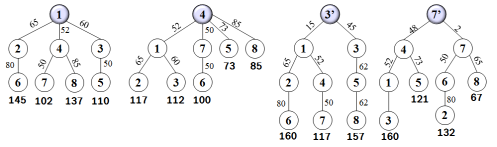
Step 3. Step 2에서 결정한 후보 지역들 중 $L^* = 160$ 이상 최대길이 단 노드의 감소 허용 길이 $(\alpha = L_i - L^*)$와 $L^* = 160$ 이하의 가장 근접한 단 노드의 증가 허용 길이$(\beta = L^* - L_i)$를 계산하여, $\alpha > \beta$인 지역(#8)을 후보에서 삭제한다.



Step 4. Step 3에서 $L^* = 160$이상 최대길이 단 노드의 감소 허용 길이 $\alpha = L_i - L^* = 0$로 할 수 있는 위치로 근 노드를 이동시킨다. (3→3', 7→7')



이 결과 $L^*=160$을 충족시키는 최종적으로 얻은 후보 위치들은 1,4,3'와 7'이다.



Phase II. 최적 위치 결정

Step 5. $N_G[v_i]$ 요구량 합을 구하여 Top 2를 후보로 결정한다. Top 2위치에 대해 식 (5) $F_j=\sum_{j=1}^{n}[e^{\theta t_{ij}}hd_jl_{ij}+\alpha d_j(e^{\theta t_{ij}}-1)]$를 계산하여 $\min F_j$인 위치 j를 최종 위치로 결정한다.

요구량	①: 10,000, ②: 8,000, ③: 12,000, ④: 25,000, ⑤: 12,000, ⑥: 5,000, ⑦: 20,000, ⑧: 15,000		
지역 v_i	$N_G[v_i]$ 요구량 합	순위	선택
1	1+2+3+4=55,000	3	-
4	1+4+5+7+8=**93,000**	**1**	v_4
3'=3	1+3+5=34,000	4	-
7'=7	4+6+7+8=**65,000**	**2**	$v_{7'}$

요구량	①: 10,000, ②: 8,000, ③: 12,000, ④: 5,000, ⑤: 12,000, ⑥: 25,000, ⑦: 20,000, ⑧: 15,000		
지역 v_i	$N_G[v_i]$ 요구량 합	순위	선택
1	1+2+3+4=35,000	3	-
4	1+4+5+7+8=**62,000**	**2**	v_4
3'=3	1+3+5=34,000	4	-
7'=7	4+6+7+8=**65,000**	**1**	$v_{7'}$

물류 센터	$v_4(25,000), v_6(5,000)$		$v_4(5,000), v_6(25,000)$	
	F_j	최적해	F_j	최적해
v_4	1.9859×10^6	v_4	2.6036×10^6	v_7'
v_7'	2.4556×10^6		2.4803×10^6	

본 논문은 Lee[4]와 동일한 결과를 얻는 물류센터 위치를 결정하였지만 먼저, $L^*=160$를 충족시키는 물류센터 위치를 정확히 결정함에 있어 서브트리 개념을 도입하여 L^*를 초과하는 서브트리의 경로길이를 L^*로 설정하는 위치 이동 길이를 단순히 결정하였다. 이외 같이 최적 위치를 단순히 결정한 이유는 ①과 ④는 $L^*=160$을 초과하는 서브트리가 전혀 없어 이들 위치에서 이동시키면 ① 또는 ④로의 총 수송비용이 증가하여 보다 많은 수송비용이 발생하여 위치이동을 하지 못하며, 3→3'의 경우 (3,6) 경로의 205를 $L^*=160$으로 감소시켜야만 하는 제약조건으로 인해 부득이 총 수송비용 F_j가 증가하더라도 ③에서 ①의 (3,1) 경로 상의 ③에서 +45, +48로 이동시킨 지점이 3'가 되며 이는 ③에서 [45,48] 범위로 3'가 ③에서 멀어질수록 ③으로의 수송비용이 증가하므로 최적의 위치는 가능한 위치 범위 [45,48]의 하한치인 ③에서 45 Km 떨어진 지점이 된다. 마찬가지로 7→7'의 경우 (7,4) 경로 상의 ⑦에서 +2, +30으로 [2,30]이며, 7'가 ⑦에서 멀어질수록 ⑦로의 수송비용이 증가하므로 최적의 위치는 가능한 위치의 범위 [2,30]의 하한치인 ⑦에서 2 Km 떨어진 지점이 된다. 따라서 하한치 지점 이외의 지점들에 대해서는 총 수송비용 F_j를 구하여 최적 해가 되는지를 검증할 필요성이 없다.

다음으로, 근방 개념을 도입하여 요구량이 집중적으로 몰려 있는 지점을 물류센터로 단순히 결정하는 방법으로 Lee[4]의 복잡한 방법을 최대한으로 단순화시킬 수 있었다.

Ⅳ. 결 론

본 논문은 부패하기 쉬운 상품을 총 수송비용을 최소화시키면서 수송시간 제약조건을 만족시키는 최적의 물류센터 위치를 결정하는 문제에 대한 알고리즘을 제안하였다.

이 문제에 대해 Lee[4]는 정확한 물류센터 위치를 결정하는 장점은 갖고 있지만 후보 물류센터를 결정하는 과정이 복잡하며, 최종적으로 결정된 후보지의 2곳에 대해 $l(v_4,v_7)=50$ 경로상의 정확한 위치를 결정함에 있어 1Km 간격으로 F_j를 50회 수행하여 결정하는 매우 복잡한 과정을 거치는 단점을 갖고 있었다. 반면에 제안된 알고리즘은 서브트리 개념을 도입하여 L^*를 초과하는 서브트리의 경로길이를 L^*로 설정하는 위치 이동 길이를 단순히 결정하였으며, 근방 개념을 도입하여 요구량이 집중적으로 몰려 있는 지점을 물류센터로 단순히 결정하는 방법으로 Lee[4]의 복잡한 방법을 단순화시켰다.

제안된 알고리즘은 지역별 요구량이 변경되는 다양한 상황에서도 빠르게 최적의 위치를 찾을 수 있는 장점을 갖고 있다.

References

[1] M. Zhang and J. Yang, "Optimization Modeling and Algorithm of Facility Location Problem in Perishable Commodities Emergency System," The International Conference on Natural Computation (ICNC), IEEE Computer Society, pp. 246-250, Aug. 2007. doi:10.1109/ICNC.2007.525

[2] K. H. Hahn, H. Hwang, and S. W. Shinn, "A Returns Policy for Distribution Channel Coordination of Perishable Items," European Journal of Operational Research, Vol. 152, No. 3, pp. 770-780, Feb. 2004. doi:10.1016/S0377-2217(02)00753-1

[3] H. A. Eiselt and C-L, Sandblom, "Decision Analysis, Location Models, and Scheduling Problems, Part II: Location and Layout Decisions, Chapter 2. Location Models on Networks," pp. 169-210, Springer-Verlag Berlin Heidelberg, 2004. ISBN:978-3-540-24722-7

[4] S. U. Lee, "Facility Location Problem for Blood Logistics Center," Journal of The Institute of Internet, Broadcasting and Communication (IIBC), Vol. 12, No. 2, pp. 135-143, Apr. 2012. doi:10.7236/JIWIT.2012.12.2.135

[5] H. S. Hwang, "A Stochastic Set-covering Location Model for Both Ameliorating and Deteriorating Items," Computers & Industrial Engineering, Vol. 46, No. 2, pp. 313-319, Apr. 2004. doi:10.1016/j. cie.2003.12.010

[6] O. Berman, Z. Drezner, and G. O. Wesolowsky, "Locating Service Facilities Whose Reliability is Distance Dependent," Computers and Operations Research, Vol. 30, No. 11, pp. 1683-1695, Sep. 2003. doi:10.1016/S0305-0548(02)00099-0

[7] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms (2nd ed.), Section 24.3: Dijkstra's Algorithm", MIT Press and McGraw-Hill. pp. 595 - 601, 2001. ISBN-13:978-0262032933

저자 소개

이 상 운(정회원)

- 1987년 : 한국항공대학교 항공전자공학과 (학사)
- 1997년 : 경상대학교 컴퓨터과학과 (석사)
- 2001년 : 경상대학교 컴퓨터과학과 (박사)
- 2003년 : 강원도립대학 컴퓨터응용과 전임강사
- 2004년 ~ 2007년 2월 : 국립 원주대학 여성교양과 조교수
- 2007년 3월 ~ 현재 : 강릉원주대학교 멀티미디어공학과 정교수
- 관심분야 : 소프트웨어 프로젝트 관리, 개발 방법론, 분석과 설계 방법론, 시험 및 품질보증, 소프트웨어 신뢰성, 최적화 알고리즘
- E-Mail : sulee@gwnu.ac.kr