Journal of the Society of Korea Industrial and Systems Engineering
Vol. 32, No. 2, pp.01－12, June 2009.

물류시스템에서 물류센터의 크기 확장계획모형에 대한 유전알고리듬

장석화† · 김재곤

인천대학교 산업경영공학과

Genetic Algorithm for Capacity Expansion Planning Model of the Distribution Centers in a Distribution System

Suk Hwa Chang† · Jae Gon Kim

Dept. of Industrial and Management Engineering, University of Incheon

Distribution centers in a distribution system that consists of the distribution centers and retailers supplies products to retailers. At the present, although total capacity of the distribution centers are enough to supply total demand of retailers, capacity of the distribution centers need to be expanded to satisfy the demand of retailers in case that future demand of the retailers will be increased. Capacity expansion model in a distribution system is to determine the location and size of expansion distribution centers that minimize costs among given distribution centers. Transportation amount from distribution center to retailers also is determined. The costs factors are the capacity expansion costs of the distribution centers and the transportation costs from the distribution centers to the retailers. A model is formulated, and a genetic algorithm based solution procedure is developed. A numerical example is shown and the algorithm is analyzed through examples.

Keywords : Capacity Expansion Model, Location and Size of Distribution Center, Distribution System

1. 서 언

물류센터와 대리점으로 이루어진 물류시스템에서 물류센터는 공장에서 제품을 받아 대리점에 공급하는 역할을 한다. 대리점의 수요량을 고려하여 물류센터는 공급부족이 발생하지 않도록 충분한 공급능력을 갖추고 있어야 한다. 물류센터의 크기에 비례하여 물류센터는 대리점에의 공급능력을 갖는다. 물류센터의 크기를 공급능력으로 정의하면 각 물류센터의 크기는 물류센터가 공급하는 대리점의 수요량을 충족시킬 수 있도록 충분하여야 한다.

미래계획기간에 물류수요가 증가하면 대리점의 수요도 증가한다. 현재는 물류센터의 공급능력이 대리점의 수요량을 충족시킬 정도로 충분하더라도 미래계획기간에 대리점의 수요가 증가하면 현재의 물류센터의 공급능력은 부족할 수 있다. 따라서 대리점의 미래 수요증가에 대비하여 현재시점에서 물류센터의 공급능력을 증가시키기 위해 물류센터를 확장할 필요가 있다.

물류센터의 확장은 기존 물류센터의 위치 또는 새로운 위치에서 이루어질 수 있다. 확장될 물류센터의 후보위치는 최소 한 곳 이상이 되어야 한다. 미래계획 기간에 예상되는 대리점의 수요량 증가를 대비하여 추가적

논문접수일 : 2008년 07월 22일　　논문수정일 : 2009년 02월 16일　　게재확정일 : 2009년 04월 21일
† 교신저자 shchang@incheon.ac.kr
※ 이 논문은 인천대학교 2008년도 자체연구비 지원에 의하여 연구되었음.

으로 적절한 위치에 물류센터를 확장할 때 물류센터의 확장위치와 확장크기는 관련 물류비용에 중요한 영향을 미치므로 물류비용을 고려하여 경제적으로 정해야 한다. 대리점의 현재수요량에 대해 물류센터로부터 수송량도 물류센터 확장 전인 현재시점과 확장 후인 미래시점에서 변할 수 있고, 이는 수송비용을 변화시킬 수 있다.

물류비용은 물류센터 확장시 발생하는 확장비용과 물류센터에서 대리점으로 수송비용이 있다. 물류비용을 최소화 할 수 있도록 적절한 위치에 적절한 크기로 물류센터를 확장해야 한다. 물류센터의 확장위치, 확장크기와 물류센터에서 대리점으로 수송량을 정하는데 있어 물류센터의 현재공급량, 대리점의 현재수요량과 미래 수요증가량의 합인 미래수요량, 물류센터 확장비용과 물류센터에서 대리점으로 수송비용 등을 고려한다.

다수의 물류센터와 대리점으로 구성된 물류시스템에서 물류센터 확장시 물류센터의 확장위치와 물류센터에서 대리점으로 수송량 등을 구하는 문제를 분석하여 수리적 모형으로 나타내고, 해를 구하는 알고리듬을 개발한다. 확장할 물류센터 후보 위치에 대해 확장 여부를 나타내는 유전자를 정의하여 유전알고리듬에 바탕을 둔 발견적 기법을 개발한다.

무제약 설비위치문제로 알려진 공장위치를 정하는 문제는 공장의 공급능력에 제한이 없는 것으로 Balinski[2]에 의해 처음으로 도입되었다. 이 문제는 고정비용과 수송비용을 최소화하는 물류센터의 위치와 물류센터가 공급하는 대리점을 정하는 문제와 동일한 것으로 위치이론에서 가장 중요한 모형의 하나로 분류될 수 있을 것이다. 설비가 서비스할 수 있는 능력에 제한이 있는 문제는 제약 설비위치문제가 된다. 무제약 설비위치문제와 제약 설비위치문제 모두 NP-hard 문제이다. 이러한 문제들의 해를 구하기 위해 많은 알고리듬, 해석적 방법 그리고 발견적 방법이 지난 수십년동안 개발되었다[1, 3, 5, 6, 10, 11, 14]. Wu et al.[19]은 하나의 위치에 일반적인 착수비용과 여러 설비가 가능한 제약 설비위치문제에 대해 해법으로 라그랑지안 휴리스틱 알고리듬을 개발하였다.

고정비용 설비위치 모형이 설명되고, 해를 구하는 절차로 add, drop, and exchange, Lagrangian relaxation, and branch and bound, heuristic method 등이 설명되었다[5, 16, 18]. 무제약 설비위치문제에 대해 Efroymson and Ray[6]와 Khumawala[11]은 LP relaxation을 적용하기 위해 일부 제약식을 통합하여 나타내었다. 그리고 Goldengorin[7]와 Korkel[13]등은 무제약 설비위치문제에 대해 분지한계기법으로 해를 구하는 방법을 연구하였다. Daskin and Owen[4]은 설비 위치 모형화의 전반적인 내용을 설명하였다. Klose and Drexl[12]은 설비 위치와 고객 할당문제에 대해 기존에 연구된 내용을 정리하였고, 기본가정, 수리적 복잡성과 계산적인 성과 등의 관점에 초점을 두고 다루었다.

고정 수송비용문제는 기존의 수송문제에서 고려하지 않은 고정비용을 고려한 것으로 공급지에서 수요지로 수송할 때 비용으로 수송량에 관계없는 고정비용에 수송량에 비례하는 가변비용을 합한 형태의 문제에 대해 해를 구할 수 있는 휴리스틱 기법을 다루고 있다[20, 21]. 여기서의 고정비용은 물류센터의 확장 등과 같은 고정비용의 가정이 아니라 공급지에서 수요지로 매 수송시 수송량에 비례하는 가변비용에 고정비용을 반영한 것이다.

본 논문은 다수의 물류센터와 다수의 대리점으로 이루어진 물류시스템에서 증가하는 대리점의 미래수요량을 공급하기 위해 물류센터를 확장할 때 관련 물류비용을 최소화하는 물류센터의 확장위치, 확장크기와 물류센터에서 대리점으로 수송량을 구하는 문제이다. 이는 이제까지 다루어지지 않은 문제로서 연구는 중요한 의미를 갖는다. 제 2장에서는 문제를 설명하고 이를 수리적 모형으로 나타낸다. 제 3장에서는 유전알고리듬을 이용하여 해를 구하는 절차를 설명하고, 그리고 제 4장에서는 수치적 예제를 사용하여 문제를 설명하고, 다양한 문제에 대해 알고리듬의 성과를 분석한다.

2. 물류센터의 확장위치 및 확장크기, 수송량 모형화

현재 물류센터의 공급능력이 대리점의 수요량을 충족시킬 수 있을 정도로 충분한 크기일지라도 미래에 대리점의 수요량이 증가하게 되면 물류센터의 공급부족이 발생한다. 미래 단일계획기간에 대리점의 수요량이 증가하게 되면 물류센터의 공급부족이 발생하지 않도록 현재 시점에서 물류센터를 확장하는 문제를 고려한다. 미래계획기간 동안에 대리점의 수요 증가량과 확장 가능한 물류센터 후보위치의 분포 등으로 비용 발생에 차이가 있으므로 물류센터의 확장위치와 확장크기를 정하는데 있어 물류비용을 최소화하도록 고려한다.

물류센터의 현재공급량은 미래에도 공급할 수 있는 물류센터의 크기가 될 것이다. 물류센터의 미래공급량은 물류센터의 현재공급량과 물류센터의 확장공급량의 합이 될 것이다. 그리고 대리점의 미래수요량은 현재수요량에 증가된 수요량을 합한 것이 될 것이다.

물류센터를 확장시 비용으로 확장비용과 수송비용을 고려한다. 확장비용은 현재 시점에서 미래 계획기간동안 수요 증가량에 해당하는 크기를 확장할 때 발생하는 것으로 확장 고정비용과 가변비용을 고려한다. 수송비

용은 미래 계획기간 동안에 발생하는 수송으로 인한 비용이다. 수송비용은 물류센터의 미래공급량으로 대리점의 미래수요량을 충족시키키기 위해 수송할 때 발생하는 것이다.

물류센터 확장에 따른 확장비용과 물류센터에서 대리점으로 수송비용의 합을 최소화하는 물류센터의 확장위치, 확장크기와 물류센터에서 대리점으로 수송량을 구한다. 이러한 문제를 해결할 수 있는 수리적 모형을 세우고 알고리듬을 개발한다. 문제를 분석하고 수리적 모형을 세우기 위해 가정과 부호를 정의한다.

2.1 가정

① 현재의 물류센터와 대리점의 위치가 주어져 있고, 또한 현재의 물류센터 위치를 포함하여 확장 가능한 물류센터의 후보위치도 주어져 있다.
② 물류센터의 현재공급량, 대리점의 현재수요량과 대리점의 미래 수요증가량 등은 알려져 있다.
③ 대리점은 다수의 물류센터에서 제품을 나누어 공급받을 수 있고, 물류센터에서 대리점으로 수송은 수송비용을 최소화하도록 이루어진다.
④ 현재 물류센터의 공급크기는 미래시점에도 유지되고, 추가로 물류센터의 확장크기의 합은 대리점의 수요증가량의 합에 해당한다.
⑤ 물류센터 확장 고정비용과 가변비용은 물류센터 위치에 따라 다를 수 있다.

2.2 부호

i = 물류센터를 나타내는 첨자
N = 물류센터의 수
j = 대리점을 나타내는 첨자
J = 대리점의 수
S_i = 현시점에서 물류센터 i의 크기
r_j = 대리점 j의 현재수요량
b_j = 대리점 j의 미래계획기간 수요 증가량
R_i = 대리점 j의 미래수요량, $R_j = r_j + b_j$
K_i = 물류센터 i에서 시설확장 착수비용
c_{1i} = 물류센터 i에서 시설확장 가변비용
c_i = 물류센터 i에서 대리점으로 단위당 단위거리 당 수송비용
$c_{i,j}$ = 물류센터 i에서 대리점 j로 단위당 비용
$d_{i,j}$ = 물류센터 i에서 대리점 j까지 거리

$$z_i = \begin{cases} 1, \text{물류센터 } i\text{의 시설이 확장되면,} \\ 0, \text{그렇지 않으면} \end{cases}$$

$$y_{ij} = \begin{cases} 1, \text{물류센터 } i\text{에서 대리점 } j\text{에 제품을 공급하면,} \\ 0, \text{그렇지 않으면} \end{cases}$$

$x_{i,j}$ = 물류센터 확장 후 물류센터 i에서 대리점 j로 수송량
w_i = 물류센터 i의 확장크기

2.3 모형 정식화

현재시점에서 물류센터의 위치와 공급량을 생각한다. 현재 다수의 위치에 물류센터가 설치되어 있고, 물류센터의 총 공급량은 대리점의 총 수요량보다 적지 않다. 각 물류센터의 위치와 공급량을 임의로 가정할 수 있다. 그러나 여기서는 현재의 물류센터의 위치와 공급량이 경제적인 상태임을 가정할 필요가 있다. 설치된 물류센터의 수가 주어지면, 물류센터의 위치와 공급량은 수송비용을 최소화하는 조건을 만족시키는 상태임을 가정한다. 즉, 물류센터의 수를 가정하면, 현재의 물류센터의 위치와 공급량은 식 (1)과 식 (2)를 만족시키면서 수송비용을 최소화하는 상태임을 가정한다. 물류센터 i의 공급량, $S_i > 0$이면, 현재시점에 물류센터가 설치되어 있는 곳이고, $S_i = 0$이면, 현재시점에 물류센터가 설치되어 있지 않는 곳이다. 그러나 현재 공급량과 관계없이 모든 후보위치는 현재시점에서 미래를 위해 확장 가능한 물류센터 후보위치가 될 수 있다.

$$\sum_{i=1}^{N} \sum_{j=1}^{J} r_j y_{ij} \le \sum_{i=1}^{N} S_i \tag{1}$$

$$\sum_{i=1}^{N} y_{ij} = 1 \tag{2}$$

대리점 j의 현재수요량 r_j에 미래 수요증가량은 b_j이다. 모든 대리점의 수요증가량에 대비하여 확장하여야 할 물류센터의 크기는 식 (3)과 같다. 이 물류센터 확장크기 B을 적절한 위치에 나누어 물류센터를 확장한다.

$$B = \sum_{j=1}^{J} b_j \tag{3}$$

수리적 모형에서 목적식은 비용함수이고 물류센터 확장 고정비용, 가변비용과 미래 시점에 물류센터에서 대리점으로 수송비용의 합으로 다음 식 (4)과 같다.

$$\sum_{i=1}^{N} K_{iz_i} + \sum_{i=1}^{N} c_{1i} w_i + \sum_{i=1}^{N} \sum_{j=1}^{J} c_i d_{ij} x_{ij} \tag{4}$$

물류센터의 공급량과 대리점의 수요량의 제약을 중심

으로 발생하는 제약식을 생각한다.

물류센터의 미래공급량은 기존 물류센터의 현재공급량에 확장된 물류센터의 공급량을 더한 값으로 식 (5)와 같고, 시스템 전체에서 확장크기는 B에 해당하는 것으로 식 (6)과 같다. 물류센터가 확장되지 않으면 확장크기는 0이고, 확장되면 최대 B까지 가능함을 나타내는 관계는 식 (7)과 같다.

$$\sum_{j=1}^{J} x_{ij} = S_i + w_i, \ \forall i \tag{5}$$

$$\sum_{i=1}^{N} w_i = B \tag{6}$$

$$w_i - Bz_i \le 0, \ \forall i \tag{7}$$

$$w_i \ge 0 \text{ and integer}, \ \forall i \tag{8}$$

식 (5)에서 물류센터 i의 확장크기 w_i는 물류센터 i에서 모든 대리점으로 수송량의 합에서 현재공급량을 뺀 것으로 $w_i \ge 0$을 만족시키기 위해 식 (9)을 만족해야 한다.

$$\sum_{j=1}^{J} x_{ij} - S_i \ge 0 \tag{9}$$

미래기간에 모든 물류센터에서 대리점 j로 수송량과 대리점의 수요량과 관련한 제약식은 다음 식 (10)과 같다.

$$\sum_{i=1}^{N} x_{ij} = R_j \tag{10}$$

변수가 갖는 범위에 대한 제약식이 있고, 다음 식 (11), 식 (12)과 같다.

$$x_{ij} \ge 0 \text{ and integer}, \ \forall i, \ \forall j \tag{11}$$

$$z_i = 0 \text{ or } 1, \ \forall i \tag{12}$$

목적함수 식 (4)에 식 (5)의 w_i을 정리하여 대입하면 식 (13)와 같다.

$$\sum_{i=1}^{N} K_{iz_i} + \sum_{i=1}^{N} \sum_{j=1}^{J} (c_{1i} + c_i d_{ij}) x_{ij} - \sum_{i=1}^{N} c_{1i} S_i \tag{13}$$

식 (13)에서 세 번째 항은 상수항으로 변수의 값에 영향을 주지 않으므로 변수의 식만으로 나타내면 식 (14)이 된다. 그리고 $c_{ij} = c_{1i} + c_i d_{ij}$로 하여 확장 가변비용이 포함된 새로운 비용으로 정의하여 목적식의 비용계수로 한다. 물류센터 확장 가변비용은 모형의 표현에는 영향을 주지 않는다. 가변비용이 없으면 $c_{1i} = 0$으로 하여 $c_{ij} = c_i d_{ij}$로 정의한다.

$$\sum_{i=1}^{N} K_{iz_i} + \sum_{i=1}^{N} \sum_{j=1}^{J} (c_{1i} + c_i d_{ij}) x_{ij} \tag{14}$$

대리점의 미래 수요 증가에 대비하여 물류센터를 확장할 때 물류센터 확장 고정비용과 물류센터에서 대리점으로 비용의 합을 최소화하도록 확장 물류센터 위치(z_i), 확장크기(w_i)와 물류센터에서 대리점으로의 수송량(x_{ij})을 구하는 수리적 모형은 다음 P1과 같다.

P1: Min $\quad Z1 = \sum_{i=1}^{N} K_{iz_i} + \sum_{i=1}^{N} \sum_{j=1}^{J} c_{ij} x_{ij}$

subject to

$$\sum_{j=1}^{J} x_{ij} - w_i = S_i, \ \forall i$$

$$\sum_{i=1}^{N} w_i = B$$

$$w_i - Bz_i \le 0, \ \forall i$$

$$\sum_{j=1}^{J} x_{ij} \ge S_i$$

$$\sum_{i=1}^{N} x_{ij} = R_j, \ \forall j$$

$$w_i \ge 0 \text{ and integer}, \ \forall i$$

$$z_i = 0 \text{ or } 1, \ \forall i$$

$$x_{ij} \ge 0, \ \forall i, \ \forall j$$

모형 P1은 제약식이 0, 1의 값을 갖는 변수를 포함한 선형 정수계획 모형이다. 선형정수계획 프로그램을 사용하여 해를 구할 수 있을 것이다. 문제의 크기가 아주 크지 않을 경우에 기존에 알려진 선형정수계획 프로그램을 이용하면 짧은 시간에 해를 구할 수 있을 것이다. 최적해 프로그램을 이용하는 방법 외에 추가적으로 유전 알고리듬을 이용하여 해를 구하는 방법을 개발한다.

3. 유전 알고리듬

확장 가능한 물류센터의 위치들 중에서 임의의 위치들을 확장하는 것을 하나의 대안으로 하여 비용을 구하고, 다른 대안들과 비용을 비교하여 비용을 최소화하는 물류센터의 확장위치, 확장크기 수송량 등을 구할 수 있을 것이다. 확장 가능한 물류센터의 후보위치의 수가

N개이고, 후보위치들 중에서 적어도 하나 이상의 위치에 물류센터를 확장하려면 가능한 집합은 2^N-1개가 될 것이다. 확장 가능한 물류센터 후보위치 중에서 임의의 후보위치들을 선택하여 확장하는 것으로 하고, 이 확장으로 발생하는 비용을 구하는 방식으로 알고리듬을 개발한다. 확장 가능한 물류센터 후보위치에서 각 후보위치에 물류센터가 확장되면 1로 하고, 확장되지 않으면 0으로 하여 유전알고리듬을 개발한다. 즉, 모든 확장 물류센터 후보위치는 확장 여부에 따라 유전자 변수의 값이 0, 1이 될 수 있다.

유전알고리듬을 적용하는 과정에서 유전자 개체 집합에 있는 개체에 대해 비용과 수송량을 구한다. 한 개체의 물류센터 확장 유전자 배열이 $Z=(z_1, z_2, \cdots, z_N)$이라 할 때 이 개체로 인해 발생하는 비용을 구한다. 대리점을 기준으로 대리점이 물류센터 중에서 가장 적은 수송비용과 두 번째 적은 수송비용과의 차이를 구하고, 이 차이가 가장 크게 발생하는 대리점에 먼저 수송량을 배정하기 위해 물류센터를 정하는 방식으로 해를 구하는 절차를 개발한다. 유전알고리듬의 적용과정에서 임의의 유전자 개체에 대한 해를 구하는 절차는 다음과 같다.

3.1 개체에 대한 해 절차

- **단계 1** : 유전자 개체 $Z=(z_1, z_2, \cdots, z_N)$이 주어진다. $z_i=1$인 i을 집합 S의 요소로 하는 확장물류센터 집합 S을 구한다. 누적 물류센터 확장크기 $A1=0$으로 둔다. 물류센터 i의 현재공급량을 S_i, 대리점 j의 미래수요량을 R_j로 한다. 그리고 총 확장크기 $B=\sum_{j=1}^{J} R_j-\sum_{i=1}^{N} S_i$을 계산한다. 물류센터 i에서 대리점 j로 수송량 표를 $T(i,j)$로 정의하고, 초기 수송량을 $T(i,j)=0$으로 둔다.
- **단계 2** : 물류센터 i에서, 대리점 j로 단위당 비용이 c_{ij}인 수송비용 표 $T1$을 정의한다. 수송비용 표 $T1$에서 물류센터 i의 공급량을 Q_i로 정의하고, $Q_i=S_i+z_iB$로 하여 $J+1$열에 넣는다. 표 $T1$에서 대리점 j의 수요량을 D_j로 정의하고, $D_j=R_j$로 하여 $N+1$행에 넣는다.
- **단계 3** : 수송비용 표 $T1$에서, $Q_i=0$인 모든 i행와 $D_j=0$인 모든 j열을 삭제한다. 수송비용 표 $T1$에서 삭제되지 않은 대리점 j에 대해 물류센터로부터 이 대리점으로 수송비용 중에서 두 번째로 가장 적은 수송비용에서 첫 번째로 가장 적은 수송비용을 뺀 값과 첫 번째 가장 적은 수송비용 행을 $N+2$, $N+3$행의 j열에 각각 기록한다.
- **단계 4** : 수송비용 표 $T1$에서, $N+2$행에서 가장 큰 값을 갖는 열 n과 $N+3$행의 n열에 기록된 행 m이 만나는 란 (m,n)에 할당할 수송량을 구한다. 즉, 물류센터 m에서 대리점 n로 수송량 $T(m,n)$을 다음의 순서로 구한다.
  (1) m행이 집합 S에 속하고, $A1+D_n<B$이면, (2)로 간다. m행이 집합 S에 속하고, $A1\le B$이고, $A1+D_n\ge B$이면, (3)으로 간다. m행이 집합 S에 속하지 않으면, (4)로 간다.
  (2) $SR=\min[Q_m, D_n]$을 구한다. m행에서 n열로 수송량, $T(m,n)=T(m,n)+SR$이다. 그리고 $B-Q_m>0$이고 $B-Q_m>D_n$이면, $A1=A1+D_n$로 한다. $B-Q_m>0$이고, $B-Q_m\le D_n$이면, $A1=A1+B-Q_m$로 한다. 그리고 $Q_m=Q_m-SR$, $D_n=D_n-SR$로 수정한다. 단계 5로 간다.
  (3) $B-Q_m\le 0$이면, m행에서 n열로 수송량, $T(m,n)=T(m,n)+Q_m-A1$이고, $Q_m=A1$, $D_n=D_n+A1-Q_m$로 수정한다.
  $B-Q_m>0$이면, m행에서 n열로 수송량, $T(m,n)=T(m,n)+(B-A1)$이고, $Q_m=Q_m-B+A1$, $D_n=D_n-B+A1$로 수정한다. $B-Q_m>0$이고 $A1-Q_m>0$이면, $A1=B$로 한다. $B-Q_m>0$이고 $A1-Q_m\le 0$이면 $A1=A1+B-Q_m$로 한다. 집합 S에 있는 행을 삭제한다. 단계 5로 간다.
  (4) $SR=\min[D_m, D_n]$로 정의한다. m행에서 n열로 수송량, $T(m,n)=T(m,n)+SR$이고, $Q_m=Q_m-SR$, $D_n=D_n-SR$로 수정한다. 단계 5로 간다.
- **단계 5** : 수송비용 표 $T1$에서, $\sum_{n=1}^{J} D_n>0$이면, 단계 3으로 가고, $\sum_{n=1}^{J} D_n=0$이면, 다음 단계로 간다.
- **단계 6** : 수송량 $T(i,j)$을 이용하여, 총 비용 $TC(Z)$을 다음 식 (15)과 같이 구한다.

$$TC(Z)=\sum_{i\in S} K_i+\sum_{i=1}^{N}\sum_{j=1}^{J} c_{ij}T(i,j) \tag{15}$$

3.2 유전 알고리듬 절차

식 (15)은 다음 유전알고리듬을 적용하여 해를 구하는 과정에서 한 개체의 비용을 구할 때 사용된다. 유전알고리듬을 적용하여 해를 구하는 과정에서 물류센터 후보위치들 중에서 임의의 위치들에 물류센터가 확장되었을 때 발생하는 확장비용과 수송비용의 합을 해를 구

하는 절차를 적용하여 최종적으로 식 (15)을 적용하여 구한다.

유전알고리듬(genetic algorithm)은 1970년대 말에 개발되어 여러 분야에서 대형문제를 해결하는데 매우 효과적으로 이용되고 있다. 자연에서 생물학적인 구조는 환경에 잘 적응하면서 더 낳은 성과를 주는 방향으로 점진적으로 진화되어 왔다는 이론을 응용한 방법이다. 유전알고리듬은 자연선택과 진화과정을 모방한 발견적 탐색으로 최적화기법의 하나이다. 유전알고리듬의 기본개념은 Holland[9]에 의해 제시되었고, Goldberg[8]은 유전알고리듬이 유연성, 견고함, 효율성 등에서 기존 탐색절차와 다르다고 제시하였다. 유전알고리듬을 설명을 위한 부호를 정의한다.

i = 물류센터 확장 후보지를 나타내는 첨자
z_i = 유전자를 의미하는 변수
g = 세대를 나타내는 첨자
G = 세대 수
$P(g)$ = 세대 g에서 해의 집단의 집합
M = 해의 집단에 있는 개체의 수
N = 개체의 크기로 한 개체에 있는 원소의 수
p_c = 교차율
p_m = 돌연변이율

문제에 대한 개체를 염색체 또는 스트링으로 부르고, 염색체의 길이 N은 $(z_1, z_2, \cdots, z_N)$의 형태로 나타내며, 여기서 z_i는 유전자를 의미한다. 유전자의 값은 대립유전자이다. z_i가 선택되는 값의 범위는 문제의 알파벳으로 부른다. 흔히 알파벳은 이진수 0, 1로 이루어져 있다. 해의 풀은 일정한 크기로 가정한 해의 집단으로 불리어진다. 재생산과정은 초기 해의 집단을 임의적으로 선택하여 시작된다. 각 염색체에 적응도로 불리는 양의 값을 할당하고, 적응도 값을 바탕으로 현재 해의 집단의 염색체는 재생산을 위해 선택된다. 재생산은 유전자에 교차와 돌연변이라는 유전자 조작을 적용하여 이루어진다.

교차는 두 개의 후손을 생산하기 위해 두 개의 부모 염색체의 부분스트링을 바꾸는 것이다. 교차는 교차비율로 불리는 확률 p_c만큼 발생시킨다. 돌연변이는 하나의 알파벳에서 다른 알파벳으로 대립유전자의 임의의 변화이다. 돌연변이는 확률 p_m만큼 발생한다.

유전알고리듬은 무제한 탐색 기법이다. 그러나 실제 문제들은 하나 이상의 제약을 갖고 있을 수 있다. 벌칙방법, 수리 알고리즘 등과 같은 제약을 다루는 다른 접근법이 Michalewicz[15]에서 제안되었다. 여기서는 물류센터가 한 곳도 설치되지 않으면 실현불가능이 된다. 즉, 유전자 개체 배열 $(z_1, z_2, \cdots, z_N)$에서 $\sum_{i=1}^{N} z_i = 0$이면 실현불가능이 된다. 이러한 경우를 해결하기 위해 $\sum_{i=1}^{N} z_i = 0$이면, 이 개체의 비용을 매우 크게 하는 방식과 이러한 결과의 개체가 발생하면 제거하여 개체 집합에 포함하지 않도록 교차 또는 돌연변이를 수행하는 방식을 적용할 수 있다. 여기서는 후자의 방식을 적용한다.

위치 i에 물류센터를 확장하느냐 따라 z_i을 0, 1로 나타낸다. 초기 해의 집단, 적응도 평가, 선별 및 교차, 돌연변이, 그리고 종료조건 등으로 이루어진 유전알고리듬의 절차는 다음과 같다.

- **단계 1:** (초기 해의 집단) $g \leftarrow 1$을 두고, 초기 해의 집단 $P(g)$을 임의로 생성한다.
  a, b $(a > b)$, M, G, p_c, p_m의 값을 준다.
- **단계 2 :** (적응도 평가) 해의 집단 $P(g)$에 있는 모든 개체의 적응도를 평가한다.
  (1) $P(g)$에 있는 모든 개체에 대해 "개체에 대한 해 절차"에 따라 비용을 구하고, 가장 적은 비용을 갖는 개체의 최소비용을 기록한다. 이 최소비용이 이전 세대까지의 최소비용보다 적으면 이 최소비용과 개체를 기록하고, 그렇지 않으면 이전 기록을 보존한다.
  (2) 적응도 평가는 해의 집단에 있는 모든 개체에 대하여 비용이 적은 개체부터 큰 개체의 순서로 정렬한 후 순위선별 방식으로 적응도 값을 부여한다. 즉 비용이 가장 적은 개체에는 a, 비용이 $j(j = 1, 2, \cdots, M)$번째로 적은 개체에는 $a-(a-b)(j-1)/(M-1)$을 부여하는 방식으로 차례로 해의 집단에 있는 모든 개체에 새로운 적응도 값을 부여한다. 가장 큰 비용을 갖는 개체에는 b을 부여한다. 해의 집단에 있는 모든 개체의 적응도의 합을 구하고, 각 개체의 적응도 값을 이 해의집단 개체의 적응도의 합으로 나누어 각 개체의 선별확률을 정한다.
- **단계 3 :** (선별 및 교차)
  (1) $P(g)$의 개체에 대해 차례로 배열하고 선별확률을 차례로 더한 누적확률을 구한다. [0, 1]사이의 난수를 발생시켜 이 난수가 누적확률에서 포함된 위치를 찾아 해당 개체를 찾아 $P(g+1)$에 넣는다. 이 과정을 $P(g)$에 있는 개체 수 M만큼 반복한다. 동일한 개체가 중복하여 선택될 수 있다.
  (2) [0, 1]사이의 난수를 발생시켜 $P(g+1)$에 있는 개체의 교차 후보자를 선택한다. 동일한 개체 번호

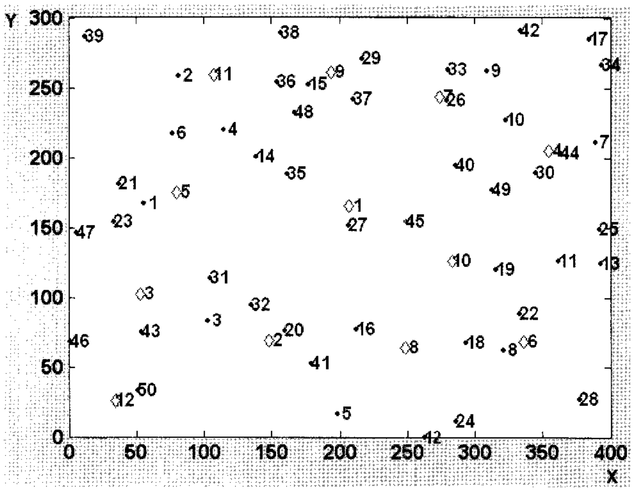
가 중복되게 발생하지 않도록 $p_c M$보다 적지 않으며 가장 적은 짝수만큼 무작위로 개체를 선택하여 교차 후보자로 둔다.

(3) 모든 교차 후보자에 대해 두 개체씩 차례로 짝을 정해 1점 교차방식으로 교차하여 두 자손을 생산한다. 생성된 두 자손을 $P(g+1)$에서 교차된 부모 개체와 동일한 위치의 개체를 $P(g+1)$에서 제거하고, 대신 생산된 자손을 $P(g+1)$에 넣는다.

- 단계 4 : (돌연변이) $P(g+1)$의 개체를 돌연변이 시킨다. 개체의 각 원소에 대해 [0, 1]사이의 난수를 발생시켜 난수에 해당하는 개체의 원소를 돌연변이 시킨다. 돌연변이가 발생한 원소가 포함된 개체를 $P(g+1)$에 두고, 돌연변이가 행해진 비율을 구하고, 이 비율이 p_m보다 적으면 계속하여 새로운 돌연변이를 수행하고, 이 비율이 p_m보다 크면 돌연변이를 멈춘다.
- 단계 5 : (종료조건) 종료조건을 만족하면 끝낸다. $g \geq G$이면, 멈춘다. $g < G$이면, $g \leftarrow g+1$ 두고, 단계 2로 간다.

4. 수치적 예제

<그림 1>과 같이 12개의 물류센터 후보 위치와 50개의 대리점으로 구성된 가상의 물류시스템을 생각한다.



<그림 1> 물류센터와 대리점의 배치

<표 1> 대리점의 위치좌표, 현재수요량 및 수요증가량

대리점번호	좌표(x_i, y_i)	현재 수요량	수요 증가량	대리점번호	좌표(x_i, y_i)	현재 수요량	수요 증가량
1	(56, 167)	184	150	26	(280, 241)	106	68
2	(82, 258)	93	92	27	(207, 151)	174	172
3	(103, 83)	88	72	28	(377, 27)	57	29
4	(115, 220)	190	118	29	(217, 271)	140	133
5	(199, 16)	70	39	30	(345, 189)	192	159
6	(77, 217)	191	157	31	(105, 114)	93	85
7	(389, 211)	155	117	32	(135, 94)	183	149
8	(321, 62)	177	117	33	(281, 263)	65	59
9	(309, 262)	81	49	34	(394, 266)	60	46
10	(323, 227)	118	71	35	(162, 188)	85	61
11	(362, 126)	62	41	36	(154, 254)	190	124
12	(263, 0)	178	113	37	(211, 242)	59	44
13	(393, 124)	134	72	38	(157, 289)	90	60
14	(139, 200)	98	61	39	(12, 286)	200	177
15	(177, 252)	106	73	40	(286, 194)	82	70
16	(212, 77)	180	122	41	(179, 52)	125	114
17	(385, 285)	106	93	42	(334, 291)	94	81
18	(294, 67)	61	48	43	(54, 75)	151	99
19	(316, 120)	80	74	44	(364, 203)	194	173
20	(160, 76)	57	39	45	(249, 154)	165	164
21	(37, 181)	135	70	46	(1, 68)	150	80
22	(333, 88)	68	48	47	(6, 146)	70	35
23	(34, 154)	128	75	48	(167, 232)	64	42
24	(286, 11)	68	35	49	(313, 177)	52	35
25	(392, 149)	165	127	50	(51, 33)	93	89

<그림 1>에서 ◇는 물류센터 후보위치, ●는 대리점의 위치이다. 각 위치는 가로 400km, 세로 300km에서 생성된 자료로 물류센터 사이의 최소거리는 70km, 대리점 사이의 최소거리는 20km로 하여 무작위로 생성한 것이다. 대리점의 위치, 대리점의 현재 수요량과 미래 수요량 증가치는 무작위로 생성된 자료로 <표 1>에 나타나 있다. 현재 수요량은 [50, 200]사이에서 무작위로 생성되었고, 미래 수요량 증가치는 현재 수요량에 [0.5, 1.0]사이의 난수를 곱하여 무작위로 생성하였다. 물류센터의 후보 위치 좌표는 <표 2>와 같고, 12개의 물류센터 후보 위치 중에서 현재 물류센터가 위치하고 있는 지역은 위치 3, 4, 5, 6, 7, 8이고, 이는 가까운 대리점의 수가 많은 물류센터부터 6개를 선택하여 지정한 것이다. 현재 물류센터 위치를 포함하여 모든 물류센터 후보 위치는 미래에 확장할 수 있는 위치가 된다. 현재 물류센터의 위치와 공급량은 대리점의 현재 수요량을 고려하였을 때 수송비용을 적게 하는 위치에 있으며, 각 물류센터의 공급량은 대리점의 총 수요량을 현재 물류센터의 수로 나눈 값에 [0.8, 1.2]사이의 값을 곱하여 정하였고, 공급량과 수요량의 차이를 현재 물류센터 중에서 가장 적은 번호의 물류센터에 더하여 균형을 맞추었다.

물류센터 확장 고정비용은 <표 2>에 나타난 것과 같고, 확장 가변비용은 0으로 한다. 확장고정비용은 [1x10^6, 2x10^6]사이에서 10,000단위로 무작위로 생성된 수이다. 물류센터 i의 수송비용, c_i =20/km/단위 하여, $c_{ij} = c_{1i} + c_i d_{ij}$로 하여 구한다. 해를 구하는 방식은 최적해 프로그램과 유전알고리듬 두 가지 방법으로 해를 구하였다. 최적해 프로그램은 Lingo8.0 프로그램을 사용하였다. 문제크기가 크지 않아 프로그램으로 쉽게 해를 구할 수 있었고, 최적 물류센터 확장위치와 확장크기는 <표 3>과 같고, 물류센터에서 대리점으로 수송량은 <표 4>와 같다. 이때 최소 총 비용은 1.4594$x$$10^7$이었다<표 7>.

유전자 알고리듬을 이용한 해는 알고리듬에서 제시한 절차에 따른 프로그램으로 개발하여 $G=100$, $M=20$, $a=20$, $b=1$, $p_c=0.3$, $p_m=0.02$로 하여 구했다. 물류센터 확장위치와 확장크기는 <표 5>와 같고, 물류센터에서 대리점으로 수송량은 <표 6>와 같다. 이때 발생한 최소 총비용은 1.4688$x$$10^7$이었다<표 7>. 유전알고리듬의 난수로 인한 해의 가변성이 발생할 수 있기 때문에 동일문제에 대해 반복실험을 하였고, 반복실험에서 일부의 해를 제외하고 거의 대부분이 이 비용의 해가 발생하였다. <그림 2>는 세대별 개체집합의 평균비용과 누적세대 최소비용을 보여주고 있다. 유전알고리듬 방식을 적용하여 해를 구할 때 세대가 진행됨에 따라 세대별 집합에 속한 개체들의 평균비용과 누적세대 최소비용이 감소하는 형태임을 보여주고 있다.

두 방법 사이에서 총비용의 차이는 매우 적게 나타났다. 그러나 물류센터 2와 4의 확장크기가 다르고, 수송량이 일부 다르게 나타났다. 해를 구하는데 걸리는 시간은 문제의 규모가 크지 않아 짧게 나타났고, 최적해

<표 2>물류센터의 위치 좌표, 현재공급량 및 고정비용

물류센터 번호	좌표(x_i, y_i)	현재 공급량	확장 고정비용 (x1000)	물류센터번호	좌표(x_i, y_i)	현재 공급량	확장 고정비용 (x1000)
1	(207, 165)	0	1660	7	(274, 243)	821	1690
2	(148, 69)	0	1180	8	(248, 64)	1165	1680
3	(53, 102)	0	1640	10	(194, 261)	821	1880
4	(354, 205)	850	1170	11	(283, 126)	0	1010
5	(80, 175)	1044	1540	12	(107, 259)	0	1310
6	(336, 68)	1206	1620	13	(35, 26)	0	1780

<표 3> 최적해 방식 : 물류센터의 확장위치와 확장크기

물류센터 번호	현재 공급량	확장크기	미래 공급량	물류센터 번호	현재 공급량	확장크기	미래 공급량
1	0	0	0	7	821	0	821
2	0	1581	1581	8	1165	0	1165
3	0	0	0	9	821	0	821
4	850	1013	1863	10	0	0	0
5	1044	0	1044	11	0	1827	1827
6	1206	0	1206	12	0	0	0

<표 4> 최적해 방식 : 수송량

물류센터	대리점(수송량)
2	3(160), 20(96), 27(111), 31(178), 32(332), 41(239), 43(250), 46(33), 50(182)
4	7(272), 10(189), 17(199), 25(292), 30(351), 34(106), 44(367), 49(87)
5	1(334), 21(205), 23(203), 46(197), 47(105)
6	8(294), 11(103), 13(206), 18(109), 19(154), 22(116), 24(103), 28(86), 45(35)
7	9(130), 26(174), 33(124), 40(152), 42(175), 45(66)
8	5(109), 12(291), 16(302), 27(235), 45(228)
9	15(179), 29(273), 35(10), 37(103), 38(150), 48(106)
11	2(185), 4(308), 6(348), 14(159), 35(136), 36(314), 39(377)

<표 5> 유전알고리듬 방식 : 물류센터의 확장위치와 확장크기

물류센터 번호	현재 공급량	확장크기	미래 공급량	물류센터 번호	현재 공급량	확장크기	미래 공급량
1	0	0	0	7	821	0	821
2	0	1470	1470	8	1165	0	1165
3	0	0	0	9	821	0	821
4	850	1124	1974	10	0	0	0
5	1044	0	1044	11	0	1827	1827
6	1206	0	1206	12	0	0	0

<표 6> 유전알고리듬 방식 : 수송량

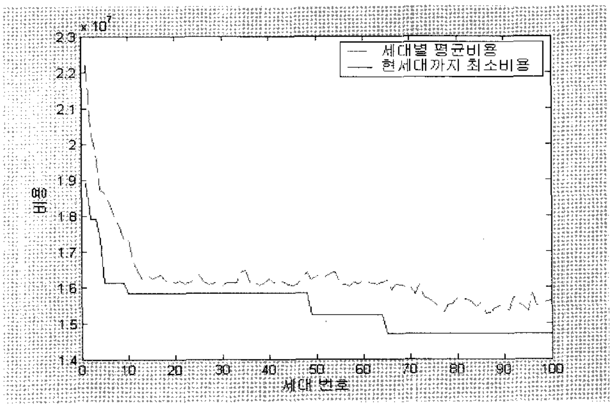
물류센터	대리점(수송량)
2	3(160), 20(96), 31(178), 32(332), 41(239), 43(250), 46(33), 50(182)
4	7(272), 10(189), 17(199), 25(292), 30(351), 34(106), 44(367), 45(111), 49(87)
5	1(334), 21(205), 23(203), 46(197), 47(105)
6	8(294), 11(103), 13(206), 18(109), 19(154), 22(116), 28(86), 45(138)
7	9(130), 26(174), 33(124), 40(152), 42(175), 45(66)
8	5(109), 12(291), 16(302), 24(103), 27(346), 45(14)
9	15(179), 29(273), 35(10), 37(103), 38(150), 48(106)
11	2(185), 4(308), 6(348), 14(159), 35(136), 36(314), 39(377)

프로그램이 유전알고리듬 방식보다 더 빨리 해를 구할 수 있었다. 문제의 크기가 크지 않을 경우에 유전알고리듬 방식이 해를 구하는데 걸리는 시간에서 좀 길게 나타났지만 최적해 프로그램과의 시간차이는 의미가 없을 정도로 적었다.

<표 7> 최소 비용

방 식	총 비용
최적해 방식	1.4594×10^7
유전알고리듬 방식	1.4688×10^7

문제의 크기가 큰 다수의 예제를 통하여 본 알고리듬



<그림 2> 세대별 평균비용과 누적세대 최소비용

으로 얻은 결과와 분지한계기법의 최적해 프로그램으로 얻은 결과를 비교한다. <표 8>에 나타난 것과 같이 물류센터의 수 N과 대리점의 수 J을 기준으로 3종류의 문제유형 (N, J)에 대해 각 유형별로 7개씩의 문제를 무작위로 생성한다. 문제유형 (N, J) = (30, 150), (50, 200)은 물류센터와 대리점의 위치가 가로 500km, 세로 400km에서 생성한 것이고, 나머지 모든 자료는 예제에서 설명한 동일한 상황에서 문제유형별로 동일 조건에서 자료를 생성하였다. 문제유형이 다르면 공간크기 등에 차이가 있지만 동일 문제유형에서는 동일 조건을 다르게 자료를 생성하였다.

〈표 8〉 비용 비율 및 실행시간 비교

문제유형 (N, J)	일련번호	본 알고리듬 비용 / 최적해 비용	본 알고리듬 실행시간	최적해 프로그램 실행시간
(20, 100)	1	1.021	55~60초	15분 이상
	2	1.033	55~60초	1분 33초
	3	1.002	55~60초	4분 23초
	4	1.027	55~60초	15분 이상
	5	1.023	55~60초	1분 59초
	6	1.003	55~60초	15분 이상
	7	1.018	55~60초	15분 이상
	평균	1.018	55~60초	11분 이상
(30, 150)	1	1.024	5~6분	40분 이상
	2	1.023	5~6분	40분 이상
	3	1.033	5~6분	40분 이상
	4	1.008	5~6분	40분 이상
	5	1.003	5~6분	40분 이상
	6	1.012	5~6분	40분 이상
	7	1.019	5~6분	40분 이상
	평균	1.017	5~6분	40분 이상
(50, 200)	1	0.994	27, 34, 43분	90분 이상
	2	1.036	27, 34, 43분	90분 이상
	3	0.999	27, 34, 43분	90분 이상
	4	1.029	27, 34, 43분	90분 이상
	5	0.988	27, 34, 43분	90분 이상
	6	1.017	27, 34, 43분	90분 이상
	7	0.961	27, 34, 43분	90분 이상
	평균	1.003	27, 34, 43분	90분 이상

본 알고리듬에서 유전자 집단의 개체의 수 M과 세대수 G는 문제유형이 (20, 100)일 경우 (M, G) = (20, 150), 문제유형이 (N, J) = (30, 150)일 경우 (M, G) = (30, 150)로 하였다. 양 문제유형의 각 일련번호에 대해 조건을 동일하게 하여 3회씩 실험하였다, 문제유형 (N, J) = (50, 200)는 모든 일련번호에 대해 3회 실험 중에서 첫 번째는 (M, G) = (40, 150), 두 번째는 (M, G) = (50, 150), 세 번째는 (M, G) = (50, 150)로 하여 1회씩 조건을 다르게 하여 실험하였다. <표 8>의 3번째 열은 최적해 프로그램 비용을 기준으로 본 알고리듬의 비용 비율정도를 나타내기 위한 것으로 본 알고리듬을 통하여 얻은 비용을 최적해 알고리듬을 통하여 얻은 비용으로 나누어 나타낸 값이다. 이 값이 1보다 크면 최적해 프로그램이 더 적은 비용을 나타낸 것이다. 본 알고리듬은 초기해의 난수로 인한 결과의 가변성을 고려하여 각 문제유형의 일련번호에 대해 3회씩 실험을 수행하여 얻은 비용을 평균하여 사용하였다. <표 8>의 4번째 열은 본 알고리듬에 대한 실행시간으로 주어진 조건에 대해 결과가 얻어질 때까지 행해진 시간이다. <표 8>의 5번째 열은 3열의 최적해 비용을 얻은 최적해 프로그램의 실행시간으로 무한정 실행을 계속할 수 없어 문제유형 (N, J) = (20, 100)는 최소 15분, 문제유형 (N, J) = (30, 150)는 최소 40분, 문제유형 (N, J) = (50, 200)는 최소 90분으로 하여 설정시간이 경과되는 시간까지 행한 실행시간이다. 예를 들어, 문제유형 (N, J) = (20, 100) 일련번호 1에서 최적해 프로그램 실행시간 15분 이상으로 표현된 것은 실행시간 15분이 초과되는 시간까지 가장 좋은 최적해 비용으로 이 비용은 문제에 대한 최적해가 아닐 수 있다. 모든 문제의 경우에 최적해 프로그램의 실행시간이 본 알고리듬의 실행시간보다 크게 하였다.

Intel Pentium(R) D CPU 3.00GHz PC를 사용하여 실험하였다. 결과를 보면 문제유형이 (N, J) = (20, 100)과 (30, 150)일 경우 본 알고리듬 비용과 최적 프로그램의 비용과의 사이에 오차가 평균 2% 이내임을 알 수 있다. 실행시간은 문제유형 (N, J) = (20, 100)인 경우에 본 알고리듬은 1분 이내의 해를 구할 수 있었지만 최적해 프로그램은 7개의 문제에서 3개의 문제는 5분 이내에 해를 구할 수 있었고, 나머지 4개의 문제는 실행시간이 15분 이상 계속되어 15분까지만 실행한 것으로, 실제 프로그램이 종료되기까지 실행시간은 상당히 오래 지속될 것으로 예상된다.

문제유형 (N, J) = (30, 150)인 경우에 본 알고리듬의 실행시간은 5~6분 사이이지만 최적 프로그램의 실행시간은 40분 이상이다. 40분으로 종료되지 않아 40분까지만 실행하였지만 실제 최적해를 얻기까지 계속하면 실행시간은 상당히 오래 걸릴 것이다. 문제유형 (N, J) = (50, 200)은 본 알고리듬의 비용과 최적 프로그램의 비용과의 사이에 큰 차이가 없는 것으로 나타났다. 7개의 문제에서 4개의 문제에서는 본 알고리듬의 비용이 더 적

게 나타난 것을 알 수 있다. 이는 문제의 크기가 큰 규모일 경우에 최적해 프로그램이 실행시간 90분으로는 최적해를 나타내지 못할 수 있음을 알 수 있다. 본 알고리듬의 실행시간은 (M, G) = (40, 150)일 때 27분, (M, G) = (50, 150)일 때 34분, (M, G) = (50, 180)일 때 43분 등이다. 반면 최적 프로그램의 경우 모두 90분 이상 계속되어 90분까지만 실행하였지만 최적해를 구하기 위해서는 90분 이상 매우 긴 시간의 실행을 해야 할 필요가 있다.

최적해 프로그램의 경우 문제의 크기가 적을 경우에 짧은 시간 내에 최적해를 얻을 수 있지만 문제의 크기가 커질 경우에 최적해를 구하는데 있어 예상할 수 없는 매우 긴 시간이 소요되거나 최적해를 주지 못하고 유사 최적해로 프로그램이 종료될 수도 있다. 반면 유전알고리듬을 활용한 본 알고리듬은 문제의 크기가 클 경우에 상대적으로 적은 시간에 오차는 있지만 최적해에 가까운 해를 줄 수 있음을 알 수 있다.

5. 결 론

본 논문은 물류센터와 대리점으로 이루어진 물류시스템에서 미래의 단일계획 기간에 대리점의 수요가 증가할 때에 대리점에 제품을 공급하는 물류센터의 크기를 증가시키기 위해 물류센터를 확장하는 문제에 대해 연구하였다. 물류센터의 확장위치, 확장크기 및 수송량 등을 정할 때 물류센터 확장비용과 물류센터에서 대리점으로 수송비용을 고려하여 비용이 가장 적게 발생하도록 하였다.

물류센터 후보 위치에 물류센터의 설치 여부를 유전자로 하여 개체를 표현하고, 유전알고리듬을 이용하여 해를 구하는 절차를 개발하였다. 유전알고리듬을 적용하는 단계에서 각 개체에 대한 해를 구하는 절차를 개발하였다. 대리점을 기준으로 대리점이 물류센터 중에서 가장 적은 수송비용과 두 번째 적은 수송비용과의 차이를 구하고, 이 차이가 가장 크게 발생하는 대리점에 먼저 수송량을 배정하는 방식으로 해를 구하는 절차를 개발하였다.

문제에 대한 수리적 모형은 적은 규모의 문제의 경우에 최적해 프로그램으로 빠른 시간에 해를 시간에 구할 수 있다. 그러나 큰 규모의 문제는 최적해 프로그램이라도 해를 구하는데 있어 시간이 매우 오래 걸리거나 유사 최적해만을 얻음을 알 수 있었다. 그리고 여기서 개발한 알고리듬도 최적해 프로그램과 같이 최적해에 근접한 해를 제공할 수 있을 것이다. 제시한 유전알고리듬은 최적해 프로그램 방법과 함께 해를 구할 수 있는 추가적인 새로운 방법이 될 것이다. 해를 구하는 새로운 다른 방식의 개발을 통하여 해법의 다양성을 제공할 것이다.

계획 기간이 단일 기간 이상으로 이산형 유한 다기간이고, 기간이 지남에 따라 대리점의 수요가 계속하여 증가할 경우에 주어진 계획기간동안에 매 확장시점에서 어느 위치에 물류센터를 확장하며, 어느 정도의 크기로 확장할 것인가를 정하는 물류센터의 확장계획에 대한 연구가 필요할 것이다. 그러나 이 문제에 대한 모형화와 최적해를 구하는 것은 매우 어려운 문제가 될 것으로 생각된다.

참고문헌

[1] Akinc, U. and Khumawala, B.; "An efficient branch and bound algorithm for the capacitated warehouse location problem," *Management Science*, 23(6) : 585-594, 1997.

[2] Balinski, M.; "Integer programming : methods, uses, computation," *Management Science,* 12 : 254-313, 1965.

[3] Beasley, J. E.; "An algorithm for solving large capacitated warehouse location problems," *European J. of Operational Research,* 33(3) : 314-325, 1988.

[4] Daskin, M. S. and Owen, S. H.; Location Models in Transportation. R. Hall, *Handbook of Transportation Science*, Kluwer Academic Publishers, Norwell, MA : 311-360, 1999.

[5] Drezner, Z.; *Facility Location: a Survey of Applications and Methods*, Springer, New York, 1995.

[6] Effroymson, M. A. and Ray, T. L.; "A branch and bound algorithm for plant location," *Operations Research*, 14 : 361-368, 1966.

[7] Goldengorin, B., Ghosh, D. and Sierksma, G.; "Branch and peg algorithms for the simple plant location problem," *Computers and Operations Research*, 30 : 967-981, 2003.

[8] Goldberg, D. E.; Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.

[9] Holland, J. H., Adaptation in Neural and Artificial Systems, 2nd ed., Ann Arbor, MI : University of Michigan Press, 1992.

[10] Jacobsen, S. K.; "Heuristic for the capacitated plant location model," *European J. of Operational Research*, 12 : 253-261, 1983.

[11] Khumawala, B. M.; "An efficient branch and bound algorithm for the capacitated warehouse location prob-

lem," *Management Science*, 18(12) : B718-731, 1972.

[12] Klose, A. and Drexl, A.; "Facility location models for distribution system design," *European J. of Operational Research*, 162 : 4-29, 2005.

[13] Korkel, M.; "On the exact solution of large scale simple plant location problem," *European J. of Operational Research*, 39 : 157-173, 1989.

[14] Krarup, J. and Pruzen, P. M., "The simple plant location problem : survey and synthesis," *European J. of Operational Research*, 12(1) : 36-81, 1983.

[15] Michalewicz, Z.; Genetic Algorithms + Data Structures = Evolution Programs, 3rd ed., AI series, New York : Springer-Verlag, 1996.

[16] Sule, D. R.; *Logistics of Facility Location and Allocation*, Marcel Dekker, Inc., New York, 2001.

[17] Taha, H. A.; *Operations Research*, 6th Edition, Prentice Hall International, Inc, 1997.

[18] Tomkins, J. A, White, J. A., Bozer, Y. A., Frazelle, E. H., Tanchoco, J. M. A. and Treviño, J.; *Facilities Planning*, Wiley, New York, 1996.

[19] Wu, L. Y., Zhang, X. S., Zhang, J. U.; "Capacitated facility location problem with general setup cost," *Computers and Operations Research,* 33 : 1226-1241, 2006.

[20] Kowalski, K. and Lev, B.; "On step fixed-charge transportation problem," *Omega*, 36 : 913-917, 2008.

[21] Adlakha, V. and Kowalski, K.; "A simple heuristic for solving small fixed-charge transportation problems," *Omega*, 31 : 205-211, 2003.