

고정비용과 비선형 단위운송비용을 가지는 수송문제를 위한 이단유전알고리즘에 관한 연구

성 기 석[†]

강릉원주대학교 산업경영공학과

A Study on the Bi-level Genetic Algorithm for the Fixed Charge Transportation Problem with Non-linear Unit Cost

Kiseok Sung

Industrial and Management Engineering, Gangneung-Wonju National University

■ Abstract ■

This paper proposes a Bi-level Genetic Algorithm for the Fixed Charge Transportation Problem with Non-linear Unit Cost. The problem has the property of mixed integer program with non-linear objective function and linear constraints.

The bi-level procedure consists of the upper-GA and the lower-GA. While the upper-GA optimize the connectivity between each supply and demand pair, the lower-GA optimize the amount of transportation between the pairs set to be connected by the upper-GA.

In the upper-GA, the feasibility of the connectivity are verified, and if a connectivity is not feasible, it is modified so as to be feasible. In the lower-GA, a simple method is used to obtain a pivot feasible solution under the restriction of the connectivity determined by the upper-GA. The obtained pivot feasible solution is utilized to generate the initial generation of chromosomes.

The computational experiment is performed on the selected problems with several non-linear objective functions. The performance of the proposed procedure is analyzed with the result of experiment

Keywords : Fixed Charge, Non-Linear Unit Cost, Transportation Problem, Bi-level Genetic Algorithm

1. 고정비용과 비선형 단위당 운송 비용을 가지는 수송문제

네트워크분석 방법은 통신망, 상하수도망, 송전망, 교통망 등에서 발생하는 각종 문제들을 다루는데 있어서 매우 유용하다. 그러한 네트워크와 관련된 문제 중의 하나로서 우리가 잘 알고 있는 수송 문제는 1941년 Hitchcock[4]에 의해서 처음 제시된 이후에 많은 학자들에 의해서 연구되어왔고, 선형계획법을 사용하여 쉽게 그 해를 구할 수 있다. 그러한 수송문제를 좀더 복잡한 현실 상황에 맞게 고도화 한 것이 고정비용과 비선형인 단위당 운송 비용을 고려하는 수송문제이다.

각 공급지로부터 각 수요지로 재화를 수송하는 문제를 보자. 각 공급지에서 공급 가능한 양이 주어지고, 각 수요지에서 요구하는 수요량이 정해져 있으며, 이들 공급량에 대한 제약과 수요량에 대한 요구가 만족되어야 한다. 재화를 각 공급지로부터 각 수요지로 운송하되 최소의 비용으로 전체 공급지와 수요지 사이에 운송이 이루어져야 한다.

한편 수송에 있어서 발생하는 비용을 살펴보자. 어느 한 공급지로부터 어느 한 수요지로 운송이 이루어지려면, 수송 경로를 개설하고 유지하는 비용이 발생할 수 있다. 이러한 비용을 수송을 위한 고정비용으로 정한다. 즉, 고정비용은 각 공급지-수요지 쌍에 대하여 다르게 주어지는데, 공급지-수요지 쌍에 조금이라도 운송량이 있으면 그 공급지-수요지 쌍에 대하여 수송경로를 개설하여야 하므로 고정비용이 발생하게 된다. 반대로, 그 공급지-수요지 쌍에 운송량이 전혀 없으면 그 공급지-수요지 쌍에 대하여 수송경로를 개설하지 않아도 되므로 고정비용이 발생하지 않게 된다.

또한 실제 공급지와 수요지 사이에서 재화를 운송하는데 소요되는 운송비용은 운송량에 따라 비선형으로 증가하는 경우가 많다. 즉 단위당 운송비용이 비선형인 경우이다. 단위당 운송비용이 운송량의 제공에 비례하거나, 제공근에 비례하거나, 또는 공급지의 공급 가능량에 대한 비율에 비례하거나 등

등 여러 가지 형태의 비선형 단위당 운송비용이 있을 수 있다. 이러한 고정비용과 비선형 단위당 운송 비용을 가지는 수송문제의 수리모형은 다음의 식 (1)과 같이 나타낼 수 있다.

고정비용과 비선형 운송비용을 가지는 수송문제

$$\text{Min } G(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^m \sum_{j=1}^n [c_{ij}(\mathbf{x}) + f_{ij}y_{ij}] \quad (1)$$

$$s.t. \quad \sum_{j=1}^n x_{ij} \leq s_i, \quad i = 1, 2, \dots, m$$

$$\sum_{i=1}^m x_{ij} \geq d_j, \quad i = 1, 2, \dots, n$$

$$x_{ij} \geq 0, \quad \forall i, j$$

$$y_{ij} = \begin{cases} 1, & \text{if } x_{ij} > 0 \\ 0, & \text{otherwise} \end{cases}$$

m : 공급지수

n : 수요지수

$c_{ij}(\cdot)$: 공급지 i 에서 수요지 j 로의 수송비용 함수

x_{ij} : 공급지 i 에서 수요지 j 로의 운송량

y_{ij} : 공급지 i 에서 수요지 j 로의 수송경로 설치 여부

f_{ij} : 공급지 i 에서 수요지 j 로의 수송경로 설치 비용

s_i : 공급지 i 에서 공급 가능량

d_j : 수요지 j 에서 요구 수요량

$G(\mathbf{x}, \mathbf{y})$: 총 수송 비용 함수

$\mathbf{x} = (x_{ij}), \mathbf{y} = (y_{ij}), \quad i = 1, \dots, m, \quad j = 1, \dots, n$

2. 연구현황

1993년 Munakata[6] 등은 각호의 흐름 용량이 주어진 네트워크상에서 최대흐름 문제를 유전알고리즘을 사용하여 해를 구하려고 시도하였다. 이때 저자들은 유전알고리즘을 적용함에 있어서 두 가지 해결하기 어려운 점을 서술하였다. 첫째 각 호에서의 흐름이 그 호의 흐름용량을 만족하는 범위 내에서 실수(Real number)값을 가진다는 점과, 둘째 각 마디에서 흐름의 유입량과 유출량이 균형을 이루어야 한다는 점이다. 그들은 유전검색체를 실수 벡터 형태로 디자인하고, 각 호의 유입량과 유출량

이 균형을 맞춘 정도에 따라서 에너지레벨 이라는 개념을 도입하여 유전 연산자를 디자인하여 사용하였으나 그다지 좋은 결과를 보지 못하였다.

2001년 Nesa[7] 등은 고정비용은 없이, 비선형인 수송문제의 해를 구하는 진화적 방법을 제시한 바 있다. 이들은 비선형 목적함수를 가진 수송문제의 제약식을 균형식으로 만들고 임의의 가능해들로 초기 세대를 구성하고 진화시켜나가는 방법을 제안하였으나, 초기 세대를 구성하는 해들의 블록조합으로 구성되는 가능해의 범위 내에서만 해를 탐색할 수 있었고, 모든 가능해를 탐색하고 유지해나가는 방법을 제시하지 못했다.

2005년 Gen et al.[3] 등은 네트워크 디자인에 있어서 걸침나무를 이용하는 진화적 방법을 활용할 것을 제안하였으며, 2007년 Jo[5] 등은 고정비용과 비선형 단위당 운송비용이 주어진 수송 문제를 걸침나무를 이용하는 유전알고리즘을 사용하여 해를 구하였다. 그들의 방법에서 단위당 운송비용을 선형으로 완화하여 근사 최적해를 구하였다.

2012년 Xie and Jia[8] 등은 위의 Jo[5] 등이 테스트에 사용하였던 것과 동일한 수송문제를, 최소비용흐름을 이용하는 하이브리드 유전알고리즘을 사용하여 해를 구하였다. 이들의 방법 역시 비선형인 단위당 운송비용을 선형으로 완화하여 네트워크 흐름문제로 변환하여 근사 최적해를 구하였다.

2012년 성기석[2]은 선형제약식과 비선형 목적함수를 가지는 수리계획문제를 유전알고리즘을 이용하여 해를 구하는데 있어서 선형제약식을 다루는 방법을 제시하였다. 그는 부등제약식을 소거하고 등식인 제약식들만을 사용하였다. 선형계획법의 내부점 기법을 이용하여 초기의 가능해들을 구하고, 영공간사영법(Null space projection method)을 이용하여 해의 가능성을 유지하면서 유전연산을 수행할 수 있도록 하는 방법을 제시하였다. 또한 2013년 성기석[1]은 유전알고리즘에서 선형제약식을 다루는 방법을 활용하여, 비선형 단위비용을 가지는 최소비용흐름문제의 해를 구하는 유전알고리즘에 관한 연구를 발표하였다.

3. 문제의 특성과 해법 전략

이 연구에서 다루려고 하는 고정비용과 비선형 단위당 운송비용을 가지는 수송문제의 특성을 수리계획의 관점에서 보면, 목적함수가 비선형이며, 0-1 정수변수와 실수변수를 가진 혼합 정수계획이며, 선형제약식을 가진다는 점이다.

이러한 특성을 가진 수리계획 문제의 해를 구하는데 유전알고리즘을 사용하려면 몇 가지 사항을 고려해야 한다. 특히 고정비용과 비선형 단위당 운송비용을 가지는 수송문제의 해를 구하기 위해서 유전알고리즘을 사용할 때 고려해야 할 점들은 다음과 같다.

- 1) 각 수송경로 상에서 운송량이 연속적인 값을 가지는 실수 변수이다.
- 2) 각 수송경로 상에서 운송량의 존재 여부에 따라 고정비용이 추가되거나 추가되지 않도록 하기 위해서 0 또는 1의 이산적인 값을 가지는 변수를 사용하여야 한다. 그리고 0-1변수의 값에 따라 목적함수의 값이 이산적으로 증감한다. 또한 0-1변수의 값에 따라 고려대상이 되는 공급지-수요지 쌍의 집합이 달라지는데, 이는 곧 0-1변수의 값에 따라 실수변수인 운송량의 가능해의 영역이 달라진다는 것을 의미한다.
- 3) 각 공급지의 공급 가능량과 각 수요지의 요구 수요량을 만족해야 한다는 선형제약식을 가지고 있다.
- 4) 각 수송경로의 운송량에 따라 운송비용이 비선형으로 증가하므로 비선형 목적함수를 가지고 있다.

이러한 특성들을 각각 고려하여 유전알고리즘을 적용하기 위한 방법을 서술하면 다음과 같다. 먼저 위의 특성 중 1)번에 나타난 실수 변수의 특성은 0-1 비트 또는 이산 숫자 방식으로 구성된 유전염색체로는 표현하기 보다는 실수인 가능해 공간에서의 탐색에 유리하도록, 실수 공간상의 벡터로써 표현한다.

그리고 2)번 특성에 있어서 0-1변수의 경우 0-1 비트열 형태의 유전염색체로써 표현하기 쉽고 유전 알고리즘에도 적합하다. 그러나 0-1변수의 값이 어떻게 정해지는가에 따라서 고려대상이 되는 공급지와 수요지 집합이 달라지며, 또 그에 따라 운송량을 나타내는 실수변수의 해가 위치해야 할 가능해 공간이 달라진다. 그러므로 0-1변수들과 실수 변수들을 동시에 고려하는 유전염색체를 사용할 경우 해의 가능성을 유지하는데 어려움이 있다. 이러한 어려움들을 해결하기 위한 방법으로, 0-1변수를 다루는 부분과 실수변수를 다루는 부분으로 나누어서, 유전염색체를 완전히 분리하여 구성하고, 각각의 유전염색체에 대해서 별도의 유전알고리즘을 적용하는 이단유전알고리즘 방법을 사용한다.

한편 3)번 특성은 선형제약식에 대해서 해의 가능성을 유지하는 것에 관한 것이다. 이러한 선형제약식에 대하여 가능해를 유지하는 것과 관련하여, 성기석[2]이 제안한 가능해를 유지하기 위해서 선형제약식을 다루는 방법을 적용한다. 그의 방법에 의하면, 부등식으로 주어진 제약식을 모두 등식 형태로 바꾸어서 사용하여야 한다. 이를 위해서, 위의 식에 나타난 수송문제를 균형수송문제로 바꾸면 선형제약식을 모두 등식 형태의 제약식으로 바꿀 수 있다. 균형수송문제란, 각 공급지의 공급 가능량의 총합과 각 수요지의 요구하는 수요량의 총합이 일치하는 수송문제이다. 주어진 수송문제의 공급 가능량의 총합과 요구 수요량의 총합에 차이가 있을 경우, 그 차이만큼을 공급하거나 수요 요구하는 가상의 공급지나 수요지를 추가함으로써 균형수송문제로 바꿀 수 있다.

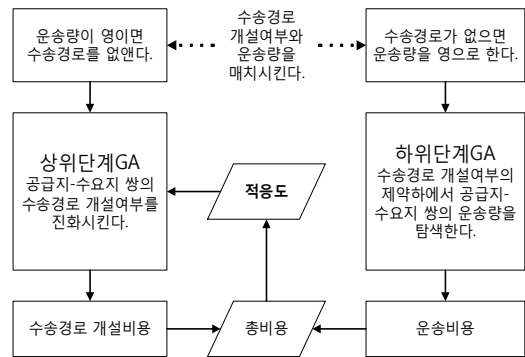
마지막으로 4)번 특성은 유전알고리즘을 적용할 경우, 도함수를 사용할 필요가 없이 주어진 변수 값에 대하여 함수의 값을 계산하면 되므로 어려움 없이 해결된다. 특히 위의 수리계획 식에서 나타난 단위당 운송비용 함수 $c_{ij}(\mathbf{x})$ 는 어떠한 형태의 비선형 함수라도 가능하다. 기존의 비선형 단위비용을 고려하는 수송문제에서는 공급지 i 와 수요지 j 사이의 운송비용함수 $c_{ij}(x_{ij})$ 는 공급지 i 와 수요지 j 사

이의 운송량 x_{ij} 에 관해서만 비선형인 함수로 정의되었으나, 본 논문에서는 모든 운송량 $x_{ij} > 0$ 에 관한 비선형 함수 $c_{ij}(\mathbf{x})$ 로 정의한다. 더 나아가 수송 경로 설치 여부를 나타내는 변수 y_{ij} 를 더 포함하여 비선형 함수 $c_{ij}(\mathbf{x}, \mathbf{y})$ 로 정의할 수도 있다. 그렇게 하여도 본 논문에서 제안하는 이단유전알고리즘을 적용할 수 있으며, 계산에 어려움이 없다. 이는 유전알고리즘이 가지는 특성이며 가장 강력한 장점이다. 본 논문에서는 이러한 유전알고리즘의 장점을 최대한 활용하면서 최적에 가까운 최선의 해를 구하고자 한다.

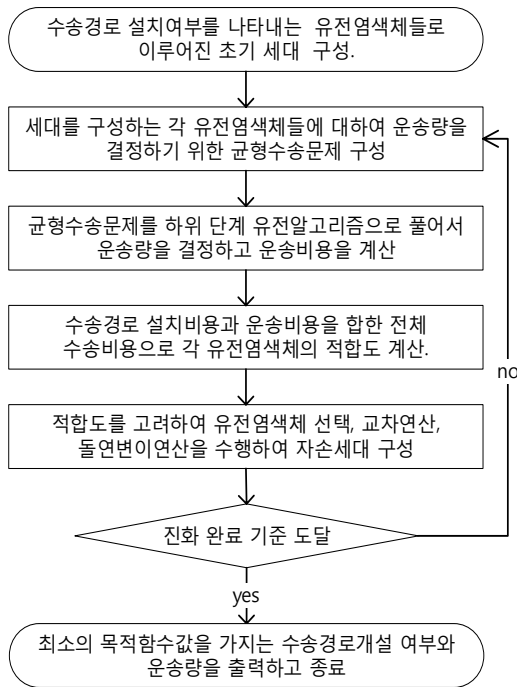
4. 이단유전알고리즘

본 연구에서 제안하는 이단유전알고리즘은 상위 단계와 하위 단계의 유전알고리즘으로 구성되어 있다. 상위 단계의 유전알고리즘에는 수송경로 설치 여부 최적화를 수행한다. 하위 단계에서는 수송경로가 설치된 공급지와 수요지들만으로 이루어진 비선형 단위비용 수송문제에서 각 공급지와 수요지 사이의 운송량을 최적화하는 유전알고리즘을 수행한다.

이러한 상위 단계에서의 최적화와 하위 단계에서의 최적화의 관계를 [그림 1]에 나타내었다. 그리고 이와 같이 상위 단계와 하위 단계의 유전알고리즘을 혼합하여 주어진 문제의 해를 구하는 계산 과정의 흐름을 [그림 2]에 나타내었다.



[그림 1] 상위 단계와 하위 단계의 유전알고리즘

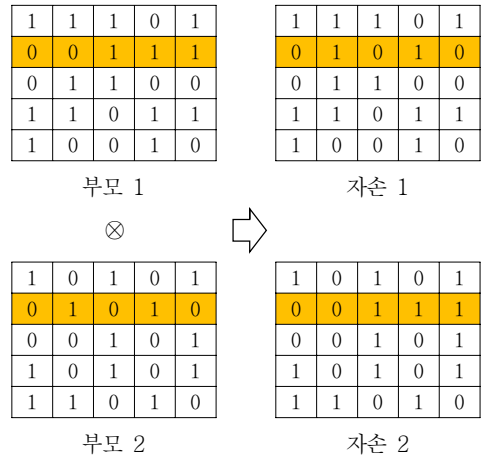


[그림 2] 이단유전알고리즘의 계산 흐름

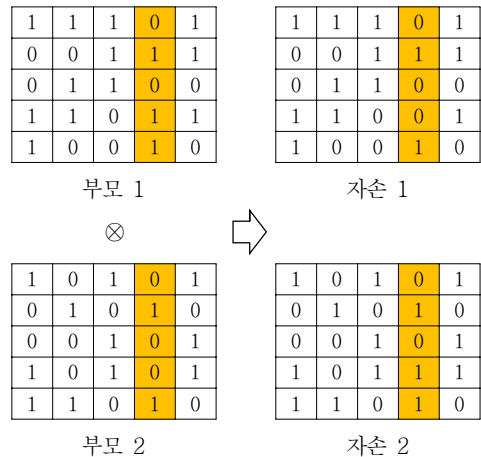
4.1 상위 단계 유전알고리즘

상위 단계의 유전알고리즘은 수송경로 설치여부 최적화를 수행한다. 이러한 수송경로 설치 최적화를 수행하기 위한 상위 단계 유전알고리즘의 유전염색체는 각 공급지와 수요지상의 모든 공급지-수요지 쌍에 대해서 수송경로 설치여부를 0과 1로 나타내는 사건행렬을 사용한다. 즉, 각 행렬 요소들에 대해서, 수송경로가 설치된 경우에는 1, 설치되지 않은 경우에는 0의 값을 가지는 행렬을 수송경로 설치 최적화를 위한 유전염색체로 사용한다.

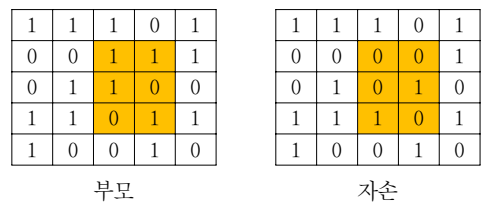
이렇게 구성된 유전염색체에 대하여 다음과 같이 유전연산자를 적용한다. 교차 연산자는 섹터교차, 행-교차, 열-교차, 행렬-교차 등의 교차연산자를 사용한다. 또한 돌연변이연산자는 섹터돌연변이, 단순돌연변이 등의 유전연산자를 사용한다. 아래 [그림 3]과 [그림 4]는 각각 행-교차와 열-교차 연산을, [그림 5]와 [그림 6]은 각각 섹터돌연변이와 단순돌연변이 연산을 나타낸 것이다.



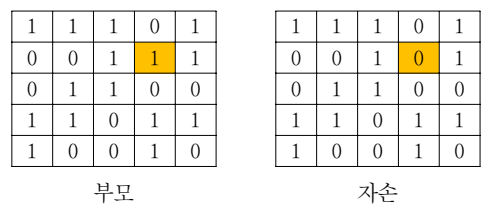
[그림 3] 행-교차



[그림 4] 열-교차



[그림 5] 섹터돌연변이



[그림 6] 단순돌연변이

교차연산과 돌연변이 연산의 결과 얻어지는 자손 염색체는 수송경로가 하나도 개설되지 않은 공급지나 수요지가 있을 수 있다. 이 경우 가능해를 구하지 못할 수 있으므로, 이를 검토하여 모든 공급지와 수요지에 대하여 적어도 하나의 수송 경로가 개설될 수 있도록 수정한다. 이러한 과정은 다음에 수송경로 검토 및 수정 과정에 나타내었다.

수송경로검토 및 수정과정

단계 1

- 1-1 : 연결된 경로가 하나도 없는 공급지 s_i 를 찾는다. 모두다 연결되어 있으면 단계 2로 간다.
- 1-2 : 연결된 경로의 수가 최소인 수요지 d_j 를 찾는다.
- 1-3 : 경로 $R(s_i, d_j)$ 를 개설하는 것으로 하고 단계 1-1로 간다.

단계 2

- 2-1 : 연결된 경로가 하나도 없는 수요지 d_j 를 찾는다. 모두다 연결되어있으면 계산을 끝낸다.
- 2-2 : 연결된 경로의 수가 최소인 공급지 s_i 를 찾는다.
- 2-3 : 경로 $R(s_i, d_j)$ 를 개설하는 것으로 하고 단계 2-1로 간다.

이와 같이 수송경로에 대한 가능성 검토와 수정이 끝난 자손 염색체들에 대해서 수송경로 개설비용을 계산한다. 수송경로가 개설된 공급지-수요지 쌍의 집합을 R 이라 할 때, 수송경로 개설비용은 $\sum_{i=1}^m \sum_{j=1}^n f_{ij}y_{ij} = \sum_{(i,j) \in R} f_{ij}$ 와 같이 계산한다.

그리고 각 자손염색체들에 대하여 하위 단계 유전 알고리즘을 수행하여 각 공급지와 수요지사이의 운송량을 최적화하여 운송비용 $C(x)$ 를 구하고, 그 운송비용과 수송경로 개설비용을 총합한 값인, 식 (1)에서의 목적함수 값 $G(x, y) = C(x) + \sum_{i=1}^m \sum_{j=1}^n f_{ij}y_{ij}$ 을 이용하여 각 자손염색체의 적응도를 $Fitness = 1/(1 +$

$G(x, y))$ 와 같이 구한다. 그리고 그 적응도 값을 기준으로 각 자손 염색체의 순위를 매기고, 이 순위를 사용하여 다음 세대의 재생산을 위한 부모 염색체를 확률적으로 선택한다.

선택한 부모 염색체들에 대해서 교차연산, 돌연변이연산 등을 수행하여 다음 세대를 생성한다. [그림 3]~[그림 6]에 교차연산과 돌연변이 연산 방법을 나타내었다. 본 연구의 실험에서 교차연산은 행-교차와 열-교차를 확률적으로 혼합하여 적용하는, 행열-교차를 사용하였다. 돌연변이 연산은 단순돌연변이를 사용하였다. 이러한 상위 단계 유전알고리즘은 미리 설정한 세대 수만큼 진화를 진행한 후 종료한다.

4.2 하위 단계 유전알고리즘

하위 단계에서는 수송경로가 설치된 공급지와 수요지의 집합으로 이루어진 비선형 단위비용 수송문제에서 각 공급지와 수요지 사이의 운송량을 최적화하는 유전알고리즘을 수행한다. 이러한 운송량을 최적화하는 하위 단계 유전알고리즘을 위한 유전염색체는 다음과 같이 구성한다.

하위 단계의 유전염색체는 수송경로가 개설된 공급지-수요지 쌍 사이의 운송량을 나타내는 실수벡터를 사용한다.

이때 운송량에 대하여 적용되는 선형제약식은 상위 단계에서 주어진 유전염색체가 나타내는 수송경로 개설 여부에 의해서 정해진다. 즉, 수송경로가 개설된 공급지-수요지 쌍의 집합에 대해서만 공급가능량과 요구수요량을 만족하도록 축소된 선형제약식을 적용한다.

또한 공급가능량의 총합과 요구수요량의 총합이 일치하는 균형수송문제로 변환하여 모든 선형제약식을 등식제약식으로 바꾼다. 즉, 상위 단계에서 주어진 각 염색체가 내포하는, 수송경로가 개설된 공급지-수요지 쌍의 집합에 따라 다음의 식 (2)와 같이 축소된 균형수송문제를 만든다. 그리고 유전알고리즘을 사용하여 식 (2)의 문제를 풀어서 해를 구한다.

구해진 해의 목적함수 값은 주어진 염색체가 내포하는 설치된 수송경로의 제약하에 최적화된 전체

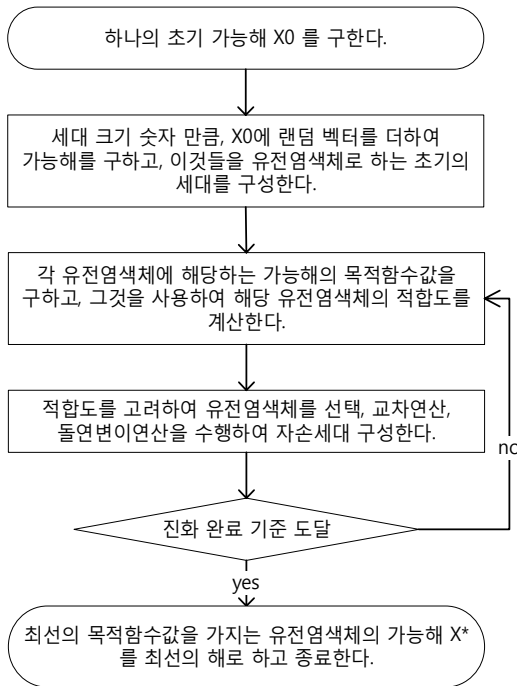
운송비용으로서, 상위 단계 유전알고리즘에서 해당 염색체의 적응도를 계산하는데 사용된다.

축소된 균형수송문제

$$\begin{aligned} \text{Min } C(\mathbf{x}) &= \sum_{(i,j) \in R} c_{ij}(\mathbf{x}) & (2) \\ \text{s.t. } \sum_{j|(i,j) \in R} x_{ij} &= s_i, \quad \forall i|(i,j) \in R \\ \sum_{i|(i,j) \in R} x_{ij} &= d_j, \quad \forall j|(i,j) \in R \\ x_{ij} &\geq 0, \quad \forall (i,j) \in R \\ \text{단, } \sum_{i|(i,j) \in R} s_i &= \sum_{j|(i,j) \in R} d_j \end{aligned}$$

R : 수송경로가 개설된 공급지-수요지 쌍의 집합

한편, 유전알고리즘을 사용하여 식 (2)의 균형수송문제의 해를 구하는 방법은 기본적으로 성기석 [2]의 방법을 사용한다. 그의 방법에 의하여 균형수송문제를 푸는 하위 단계 유전알고리즘의 계산 과정흐름은 다음의 [그림 7]에 나타내었다. 각 단계에 대한 자세한 설명은 다음과 같다.



[그림 7] 하위 단계 유전알고리즘의 계산 흐름

4.2.1 하위 단계 초기세대의 구성.

먼저 초기의 세대를 구성하기 위해서 식 (2)의 모든 제약식을 만족하는 초기의 중심이 되는 가능해 (초기중심가능해 : Initial Pivot Feasible Solution) 를 하나 구한다. 성기석[2]의 방법에 의하면 선형계획법의 내부점기법(Interior point method) 중의 하나인 이단계원아핀법(Two-phase primal affine-scaling method)의 제 1단계를 수행하여 하나의 초기중심가능해를 구한다. 그런데 본 연구에서는 균형수송문제의 특성을 이용하여 초기중심가능해를 구하고 이를 초기 세대의 유전염색체를 구성하는데 사용한다.

균형수송문제의 특성을 이용하여 초기중심가능해를 구하는 방법으로서 다음에 제시하는 수정복서모서리법과 균등배정법을 사용하였으며, 각 방법의 계산 단계는 다음과 같다.

수정복서모서리법

단계 0

- 0-1 : 모든 공급지와 수요지는 아직 배정이 완료되지 않은 것으로 한다.
- 0-2 : 모든 공급지 s_i 의 남은공급량 $M s_i$ 와 수요지 d_j 의 남은수요량 $M d_j$ 는 모두 원래의 공급량과 수요량으로 둔다.
- 0-3 : 공급지의 집합 S에서, 연결된 경로의 수가 최소인 공급지 s_i 를 선택한다. 단계 1로 간다.

단계 1

- 1-0 : 배정이 완료되지 않은 수요지나 공급지가 없으면 계산을 끝낸다.
- 1-1 : 배정이 완료되지 않은 수요지의 집합 D에서, 공급지 s_i 와 연결되어있는 수요지가 없으면 1-2로 가고, 있으면 1-3으로 간다.
- 1-2 : 배정이 완료되지 않은 수요지의 집합 D에서, 공급지 s_i 와 연결되지 않은 수요지 중에서 다른 공급지와 연결된 경로의 수가 최소인 수요지 d_j 를 선택하

고, 공급지 s_i 와 수요지 d_j 사이의 경로를 개설하는 것으로 수정하고 단계 3으로 간다.

- 1-3 : 배정이 완료되지 않은 수요지의 집합 D 에서, 공급지 s_i 와 연결되어 있는 수요지 중에서, 다른 공급지와 연결된 경로의 수가 최소인 수요지 d_j 를 선택하고 단계 3으로 간다.

단계 2

- 2-0 : 배정이 완료되지 않은 수요지나 공급지가 없으면 계산을 끝낸다.
- 2-1 : 배정이 완료되지 않은 공급지의 집합 S 에서, 수요지 d_j 와 연결되어 있는 공급지 d_j 가 없으면 2-2로 가고, 있으면 2-3으로 간다.
- 2-2 : 배정이 완료되지 않은 공급지의 집합 S 에서, 수요지 d_j 와 연결되지 않은 공급지 중에서 다른 수요지와 연결된 경로의 수가 최소인 공급지 s_i 를 선택하고, 공급지 s_i 와 수요지 d_j 사이의 경로 $R(s_i, d_j)$ 를 개설하는 것으로 수정하고, 단계 3으로 간다.
- 2-3 : 배정이 완료되지 않은 공급지의 집합 S 에서, 수요지 d_j 와 연결되어 있는 공급지 중에서, 다른 수요지와 연결된 경로의 수가 최소인 공급지 s_i 를 선택하고 단계 3으로 간다.

단계 3

- 3-1 : 공급지 s_i 와 수요지 d_j 의 남은공급량 M_{s_i} 과 남은수요량 M_{d_j} 을 비교하여 남은공급량 M_{s_i} 이 남은수요량 M_{d_j} 보다 많으면 3-2로 가고, 아니면 3-3으로 간다.
- 3-2 : 수요지 d_j 를 배정이 완료된 것으로 하고, 공급지 s_i 의 남은공급량은 $M_{s_i} = M_{s_i} - M_{d_j}$ 로 두고, 단계 1로 간다.
- 3-3 : 공급지 s_i 를 배정이 완료된 것으로 하고, 수요지 d_j 의 남은수요량을 $M_{d_j} = M_{d_j} - M_{s_i}$ 로 두고, 단계 2로 간다.

균등배정법

단계 0

- 0-1 : 모든 공급지 s_i 에 대해서, 공급량 M_{s_i} 를 공급지 s_i 에 연결된 수요지의 수 $n(R_{s_i})$ 로 나눈, 균등공급량 $E_{s_i} = M_{s_i} / n(R_{s_i})$ 를 계산한다.
- 0-2 : 모든 수요지 d_j 에 대해서 수요량 M_{d_j} 를 수요지 d_j 에 연결된 공급지의 수 $n(R_{d_j})$ 로 나눈, 균등수요량 $E_{d_j} = M_{d_j} / n(R_{d_j})$ 를 계산한다.
- 0-3 : 모든 연결된 경로 $R(s_i, d_j)$ 에 대해서 균등공급량 E_{s_i} 와 균등수요량 E_{d_j} 중에서 작은 것을 배정한다.
- 0-4 : 모든 공급지 s_i 에 대해서, 공급지 s_i 에 연결된 모든 경로 $R(s_i, :)$ 에 배정된 수요량을 뺀 것을 남은공급량 M_{s_i} 으로 한다. 남은공급량이 $M_{s_i} = 0$ 인 공급지 s_i 는 배정이 완료된 것으로 하고, 남은공급량이 $M_{s_i} > 0$ 인 공급지 s_i 는 배정이 완료되지 않은 것으로 한다.
- 0-5 : 모든 수요지 d_j 에 대해서, 수요지 d_j 에 연결된 모든 경로 $R(:, d_j)$ 에 배정된 공급량을 뺀 것을 남은수요량으로 한다. 남은수요량이 $M_{d_j} = 0$ 인 수요지 d_j 는 배정이 완료된 것으로 하고, 남은수요량이 $M_{d_j} > 0$ 인 수요지 d_j 는 배정이 완료되지 않은 것으로 한다.
- 0-6 : 배정이 완료되지 않은 공급지의 집합 S 에서, 연결된 경로의 수가 최소인 공급지 s_i 를 선택한다.
- 0-7 : 수정복서모서리법의 단계 1~단계 3을 수행한다.

수정복서모서리법은 수송단체법에서 초기기저가능해를 구하기 위해서 사용하는 복서모서리법과 유사한 방법이다. 복서모서리법을 수정하여 사용하는 이유는 모든 공급지-수요지 쌍에 대하여 수송경로

가 개설되지 않기 때문이다. 그 때문에 복서모서리법을 사용하여 운송량을 배정할 경우, 남은 공급 가능량 또는 요구 수요량을 배정할 수송경로가 존재하지 않을 수 있다. 이 경우에 새로운 경로를 추가로 개설함으로써, 남은 공급량 또는 수요량을 배정할 수 있도록 한다. 수정복서모서리법을 사용하여 얻어지는 해는 기저가능해에 가깝도록 구해지는 경향이 있는데, 이는 가능해 영역의 꼭지점에 해당하며, 수송단체법의 걸침나무 등의 방법으로 구해진 해와 유사한 특성을 갖는다.

균등배정법은 주어진 공급 가능량 또는 요구 수요량을 수송경로가 개설되어 수송이 가능한 모든 공급지-수요지에 대하여 될 수 있는 데로 균등하게 배정하여 가능해를 구하려는 것이다. 균등배정법은 먼저 각 공급지의 공급 가능량을 균등하게 수요지에 배분하여 배정하거나, 각 수요지의 수요량을 균등하게 배분하여 공급지에 배정하고, 그 균등배분량이 일치하지 않아서 남게 되는 공급 가능량이나 요구 수요량은 수정복서모서리법을 사용하여 배정한다. 이 방법에 의하여 구해진 해는 가능해집합의 내부점 중앙부에 자리하는 경향이 나타나며, 이는 내부점 기법에 의해 구해진 해와 유사한 특성을 갖는다.

본 연구에서는 가능한 한 내부점 기법에 의해서 구해진 해와 유사한 특성을 가지는 균등배정법을 사용하여 하나의 초기중심가능해 X^1 를 구한다. 그리고 다음과 같이 계산하여 영공간 투영법을 사용하여 초기세대를 구성하는 N개의 유전염색체를 모두 구한다.

먼저 식 (2)의 모든 제약식을 포함하는 행렬계수를 A라고 하면, 식 (2)의 제약식의 영공간에 투영하는 사영행렬(Projection matrix)은 $M = I - A^T(AA^T)^{-1}A$ 와 같이 구한다. 그리고 임의의 랜덤벡터 r 을 투영한 벡터 $T = Mr$ 을 구하고, $X^c = X^1 + \alpha^c T$, $c = 1, \dots, N$ 와 같이 N개의 유전 염색체를 구한다. 이때 α^c 는 X^c 가 비음조건을 만족하도록 하는 스케일 값이다. 이렇게 구해진 초기세대의 모든 유전염색체는 균형수송문제의 모든 제약식을 만족하게 된다.

4.2.2 적응도의 계산

구해진 각 유전염색체 X^c 에 대해서, 식 (2)에서의 목적함수 $C(X^c)$ 값을 구하고, 이것을 사용하여 하위 단계 유전염색체의 적응도를 구한다. 적응도 값은 $Fitness = 1/(1+C(X^c))$ 와 같이 구한다. 그리고 이 적응도 값을 기준으로 각 유전 염색체의 순위를 매기고, 다음 세대를 재생산하기 위한 부모 염색체를 확률적으로 선택한다.

4.2.3 교차연산

선택된 두 부모 염색체는 다음과 같이 단순교차연산을 수행하여 두 자손을 생성한다. 두 부모 염색체를 $X^p = (x_1, x_2, \dots, x_n)$, $Y^p = (y_1, y_2, \dots, y_n)$ 라고 하면, 임의의 k번째 요소에서 교차를 수행하여 $X^c = (x_1, x_2, \dots, x_k, y_{k+1}, \dots, y_n)$, $Y^c = (y_1, y_2, \dots, y_k, x_{k+1}, \dots, x_n)$ 를 구한다. 그리고 앞서 사용한 사영행렬 $M = I - A^T(AA^T)^{-1}A$ 을 사용하여, $X^0 = X^p + \alpha M(X^c - X^p)$, $Y^0 = Y^p + \alpha M(Y^c - Y^p)$ 와 같이 자손 염색체들을 구한다. 이때 α 와 β 는 각각 X^0 와 Y^0 가 비음조건을 만족하도록 하는 스케일 값이다. 이렇게 구해진 모든 자손 염색체는 균형수송문제의 모든 제약식을 만족한다.

4.2.3 돌연변이 연산

돌연변이 대상으로 선택된 염색체 X^p 에 대해서, 임의의 랜덤벡터 r 을 투영한 벡터 $T = Mr$ 을 구하고, $X^0 = X^p + \alpha T$ 와 같이 N개의 유전 염색체를 구한다. 이때 α 는 X^c 가 비음조건을 만족하도록 하는 스케일 값이다. 이렇게 돌연변이를 수행한 후의 유전 염색체는 균형수송문제의 모든 제약식을 만족한다.

4.2.4 하위 단계 유전알고리즘의 종료

적응도를 이용하여 하위 단계 유전알고리즘에서 다음 세대의 재생산을 위한 선택과 교차의 과정을 반복하며 진화하다가 미리 설정한 세대 수만큼 진화를 진행한 후 종료한다.

이렇게 하위 단계 유전알고리즘을 수행하여 그때까지 구한 운송비용 중 최소인 것을 상위 단계

유전알고리즘에서 해당 염색체의 고정비용 값과 합한 총비용을 해당 염색체의 적응도를 구하는데 사용한다.

5. 전산실험 및 비교

위와 같이 제시한 이단유전알고리즘이 고정비용과 비선형 단위당 운송비용을 가지는 수송문제를 성공적으로 해를 구할 수 있는지 전산실험을 수행하여 보았다. 전산실험을 위하여 문제의 크기를 공급지의 수와 수요지의 수가 각각 4×5 , 5×10 , 8×16 , 20×20 인 문제를 대상으로 하였다. 이 문제들의 단위당 운송비용과 고정비용 값은 2012년 Xie and Jia[8]이 그들의 논문에서 사용한 값을 그대로 사용하였다.

그리고 이들 문제의 목적함수인 총비용을 다음과 같이 다섯 가지의 함수로 설정하였다. 즉,

- 1) $G1(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n (u_{ij}x_{ij}^2 + f_{ij}y_{ij})$
- 2) $G2(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n (u_{ij}x_{ij}^2/100 + f_{ij}y_{ij})$
- 3) $G3(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n (u_{ij}\sqrt{x_{ij}} + f_{ij}y_{ij})$
- 4) $G4(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n (u_{ij}(1 + (x_{ij} - 10)^3/1000 + f_{ij}y_{ij}))$
- 5) $G5(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n (u_{ij}((x_{ij} - s_i/\sum_j y_{ij})^2 + (x_{ij} - d_j/\sum_i y_{ij})^2) + f_{ij}y_{ij})$ 와 같이 정하였다.

$G1(\mathbf{x})$ 의 경우에 단위당 운송비용은 운송량의 제곱에 비례한다. 이 경우 하위 단계의 균형수송문제의 목적함수는 볼록함수로 된다. $G2(\mathbf{x})$ 의 경우는 $G1(\mathbf{x})$ 의 경우와 동일하지만 고정비용에 비해서 단위당 운송비용의 비중을 줄인 것이다. $G3(\mathbf{x})$ 의 경우에 단위당 운송비용은 운송량의 제곱근에 비례한다. 이 경우 하위 단계의 균형수송문제의 목적함수는 오목함수로 된다. $G4(\mathbf{x})$ 의 경우에 단위당 운송비용은 운송량에 대한 3차 함수에 비례한다. 이 경우 하위 단계의 균형수송문제의 목적함수는 3차 함수이며, 오목이나 볼록함수가 아니다. 한편 $G5(\mathbf{x})$ 의 경우에 단위당 운송비용은 수요지와 공급지의

평균운송량에 대한 편차의 제곱 합에 비례하는 경우를 가정한 것이다. 이러한 목적함수들은 $G1(\mathbf{x})$ 에서부터 $G5(\mathbf{x})$ 까지 점차적으로 비선형성이 높아진다고 할 수 있다.

이와 같이 단위당 운송비용이 제곱에 비례하거나, 제곱근에 비례하거나, 제제곱에 비례하거나, 계단식으로 증가하거나, 공급지의 공급 가능량이나 수요지의 요구 수요량의 비율에 비례하거나 운송량의 편차에 비례하거나 등등 어떠한 형태의 비선형함수라 하더라도 제약 없이 제안된 이단유전알고리즘을 적용하여 해를 구할 수 있다.

제안된 알고리즘은 MATLAB 2016a으로 프로그래밍하였으며, Intel core i7-5500 CPU에서 계산을 수행하였다. 비교대상으로서 기존의 비선형혼합정수계획 모델을 기반으로 하는 상용 최적화 프로그램(가칭 TCMINLP)를 사용하였다. 한편 목적함수 $G1(\mathbf{x})$ 에 대해서는 2012년 Xie and Jia[8] 등이 그들의 논문에서 제시한 자료가 있어서 비교하는데 사용하였다.

<표 1>에 각 문제들에 대해서 구해진 해의 $G1(\mathbf{x})$ 목적함수 값을 사용하여 수행도를 비교하여 나타내었다. 표의 목적함수 값 중 BLGA-Total Cost 행은 본 논문에서 제시한 이단유전알고리즘에 의해 구해진 목적함수 값이고, HGA 행은 Xie and Jia[8] 등이 그들의 방법에 의해서 구한 목적함수 값이다. 마지막 행의 수행도는 BLGA-Total Cost와 HGA의 목적함수 값을 비교한 것으로서, 모두 다 최소 6.16%에서 최고 26.95% 정도까지 목적함수 값이 개선된 해를 찾았다.

이와 같이 목적함수 값을 많이 개선시킬 수 있었던 이유는, HGA의 경우 비선형인 단위당 운송비용을 선형으로 완화하여 해를 구하였고, BLGA의 경우에는 비선형인 단위당 운송비용을 그대로 적용하여 해를 구할 수 있었기 때문인 것으로 판단된다. 한편 HGA의 경우에 그들의 알고리즘으로는 목적함수 $G3(\mathbf{x}) \sim G5(\mathbf{x})$ 에 대해서 해를 구할 수가 없다.

〈표 1〉 각 문제의 목적함수 $G_1(\mathbf{x})$ 에 대한 성능비교

Problem ID		1	2	3	4	
Problem size		4×5	5×10	8×16	20×20	
Number of 0-1 integer variable		20	50	128	400	
Number of real variable		20	50	128	400	
U-GA number of generation		100	150	200	1,000	
U-GA population size		20	30	50	150	
L-GA number of generation		20	20	20	100	
L-GA population size		20	30	50	150	
Objective function value	BLGA	Total Cost	22,829	179,003	741,972	3,639,745
		Fixed Cost	1,532	12,508	87,856	184,557
		Trans. Cost	21,297	166,495	654,116	3,455,188
	HGA	27,127	245,038	805,941	3,878,824	
Performance : (HGA-BLGA)/HGA		15.84%	26.95%	7.94%	6.16%	

〈표 2〉 문제 20x20의 목적함수 $G_1(\mathbf{x})$ 에 대한 파라미터에 따른 성능비교

Run ID		1	2	3	4	5
U-GA number of generation		1,000	500	300	500	1,000
U-GA population size		150	300	250	150	60
L-GA number of generation		100	50	70	50	20
L-GA population size		150	300	200	150	500
Objective function value(BLGA)	Total Cost	3,639,745	3,671,512	3,688,176	3,843,314	3,908,435
	Fixed Cost	184,557	181,269	178,325	167,574	164,250
	Trans. Cost	3,455,188	3,490,243	3,509,851	3,675,740	3,744,185
Performance : (HGA-BLGA)/HGA		6.16%	5.34%	4.92%	0.92%	-0.76%

파라미터들을 어떻게 설정할 것인가는 유전알고리즘을 적용하는데 있어서 어려운 점 중의 하나이다. 특히 이단유전알고리즘에서는 상위 단계 유전알고리즘과 하위 단계 유전알고리즘의 파라미터를 어떻게 조화롭게 설정할 것인가를 고민하여야 한다. 전산실험에서는 상위 단계의 유전알고리즘에서 교차확률 0.9 돌연변이확률 0.3, 하위 단계 유전알고리즘에서 교차확률 0.7, 돌연변이확률 0.3을 사용하였다. 그리고 세대의 크기와 진화 세대 수는 문제의 크기에 따라 변경하였다. <표 2>에 20×20 문제의 해를 구하는데 있어서 이단유전알고리즘의 가장 기본적인 파라미터인 세대의 크기와 세대의

수에 대해서 비교하여 제시하였다. 한편 <표 2>에 나타난 목적함수 값의 구성을 살펴보면, 수송경로 개설비용인 고정비용이 적을수록 운송비용 부분이 많아져서 총비용이 더 커지는 것을 볼 수 있다.

<표 3>은 각 목적함수 별로 문제 크기 별로 BLGA와 TCMINLP로 계산한 결과를 보여준다. 표에서 Stop criteria는 TCMINLP의 경우 프로그램이 완료된 상황을 나타내고, BLGA의 경우 상위 단계와 하위 단계 유전알고리즘의 (세대 수)×(세대 크기)를 나타낸다. 표에 나타난 목적함수 값과 수행시간에 대한 분석을 다음의 <표 4>와 <표 5>에 정리하여 나타내었다.

〈표 3〉 각 목적함수 별 문제크기별 계산 시험 결과

Objective Function	Problem Size	Method	Objective Value	Run Time (sec)	Solution Status	Stop Criteria
G1	4×5	TCMINLP	22,825.8	0.1	Global Opt.	Finished
	4×5	BLGA	22,826.0	388.7		100×40_30×40
	5×10	TCMINLP	176,736.3	0.2	Global Opt.	Finished
	5×10	BLGA	179,003.3	173.2		150×30_20×30
	8×16	TCMINLP	709,504.2	0.2	Global Opt.	Finished
	8×16	BLGA	714,734.2	7,957.1		200×100_40×100
	20×20	TCMINLP	3,575,457.0	27.4	Global Opt.	Finished
	20×20	BLGA	4,210,336.3	10,567.5		200×100_40×100
G2	4×5	TCMINLP	979.1	3.7	Global Opt.	Finished
	4×5	BLGA	1,071.0	413.7		100×40_30×40
	5×10	TCMINLP	6,657.2	19.7	Global Opt.	Finished
	5×10	BLGA	7,502.1	6,405.4		200×100_40×100
	8×16	TCMINLP	42,459.1*	134,702.4	Feasible Sol'n	Interrupted
	8×16	BLGA	48,879.9	6,699.8		200×100_40×100
	20×20	TCMINLP	134,513.5*	109,334.1	Feasible Sol'n	Interrupted
	20×20	BLGA	141,755.2	9,138.0		200×100_40×100
G3	4×5	TCMINLP	698.0	0.1	Local Opt.	Finished
	4×5	BLGA	704.0	359.7		100×40_30×40
	5×10	TCMINLP	3,335.8	6.2	Local Opt.	Finished
	5×10	BLGA	3,574.3	149.0		150×30_20×30
	8×16	TCMINLP	17,809.0*	134,970.5	Feasible Sol'n	Interrupted
	8×16	BLGA	26,131.3	6,500.7		200×100_40×100
	20×20	TCMINLP	29,016.0*	111,254.8	Feasible Sol'n	Interrupted
	20×20	BLGA	51,522.8	7,839.5		200×100_40×100
G4	4×5	TCMINLP	1,086.0	21.3	Local Opt.	Finished
	4×5	BLGA	1,146.5	420.2		100×40_30×40
	5×10	BLGA	14,122.2	7,346.8		200×100_40×100
	5×10	TCMINLP	14,131.2	109,473.4	Feasible Sol'n	Interrupted
	8×16	BLGA	64,819.0	8,938.4		200×100_40×100
	8×16	TCMINLP	67,728.0	135,612.7	Feasible Sol'n	Interrupted
	20×20	BLGA	157,554.0	13,671.6		200×100_40×100
	20×20	TCMINLP	163,611.2	108,195.4	Feasible Sol'n	Interrupted
G5	4×5	BLGA	580.0	576.1		100×40_30×40
	4×5	TCMINLP	6,549.8	0.1*	Local Opt.	Finished
	5×10	BLGA	4,530.5	10,024.5		200×100_40×100
	5×10	TCMINLP	190,830.1	0.6*	Local Opt.	Finished
	8×16	BLGA	27,905.0	10,519.0		200×100_40×100
	8×16	TCMINLP	1,741,020.0	0.6*	Local Opt.	Finished
	20×20	BLGA	57,791.3	14,093.1		200×100_40×100
	20×20	TCMINLP	22,961,420.0	4.8*	Local Opt.	Finished

〈표 4〉 BLGA와 TCMINLP의 목적함수 값 비교

		목적함수 값이 좋은 것				
↑ 문 제 크 기	20×20	T	T*	T*	B	B
	8×16	T	T*	T*	B	B
	5×10	T	T	T	B	B
	4×5	T	T	T	T	B
		G1	G2	G3	G4	G5
목적함수 비선형성 →						

〈표 5〉 BLGA와 TCMINLP의 수행시간 비교

		수행시간이 좋은 것				
↑ 문 제 크 기	20×20	T	B	B	B	B*
	8×16	T	B	B	B	B*
	5×10	T	T	T	B	B*
	4×5	T	T	T	T	B*
		G1	G2	G3	G4	G5
목적함수 비선형성 →						

〈표 4〉는 상대적으로 더 좋은 목적함수 값 구한 방법을 비교하여 표시한 것이다. 문제의 크기가 클수록, 목적함수의 비선형성이 높을수록 BLGA의 성능이 더 나은 것을 보여준다. 표에서 T*로 표시한 경우 TCMINLP에서 더 나은 해를 찾았으나 그것이 최적해임을 확인하지 못해서 계속 탐색 수행을 하다가 수행시간이 30시간을 넘겨서 수행을 강제로 중단한 경우이다. 이 경우에 실질적으로는 TCMINLP가 더 좋은 해를 찾은 경우로 보았다.

〈표 5〉는 수행시간이 상대적으로 더 좋은 방법을 표시한 것이다. 역시 문제의 크기가 클수록, 목적함수의 비선형성이 높을수록 BLGA의 수행시간이 상대적으로 더 좋은 것을 보여준다. 표에서 B*로 표시한 경우는 TCMINLP의 수행시간은 적었으나 목적함수 값이 BLGA에 비해서 매우 큰 값을 가지는 지역 최적해로서 수행이 완료되었고, 더 이상 좋은 해를 찾을 수 없는 경우이다. 이 경우에 실질적으로는 BLGA의 수행시간이 더 좋은 경우로 보았다.

〈표 6〉, 〈표 7〉, 〈표 8〉은 각각 20×20 문제의 공급지-수요지 쌍에 대한 수송경로 개설비용이며 고정비용인 f_{ij} , 운송량의 제곱에 비례하는 단위당 운송비용인 u_{ij} , 공급 가능량 s_i 및 요구 수요량 d_j 에 관한 자료이다. 〈표 9〉는 20×20 문제의 목적함수 $G1(\mathbf{x})$ 에 대해서 이단유전알고리즘을 적용하여 구한 공급지-수요지 쌍에 대한 최선의 운송량 x_{ij} 을 나타낸다. 이 운송량에 해당하는 고정비용은 184,557이고 운송비용은 3,455,188이며 이들의 총합인 목적함수 값은 3,639,745이다.

한편 [그림 8]과 [그림 9]는 〈표 2〉에 나타난 20×20 문제의 목적함수 $G1(\mathbf{x})$ 에 대해서 수행한 계산 중 첫 번째 수행한 계산 과정에서, 세대가 진화하면서 목적함수 값이 수렴하는 모양을 보여주고 있다. [그림 6]의 그래프는 전체적인 진화의 과정에서 안정적으로 목적함수 값이 수렴하는 것을 보여주고 있으며, [그림 7]의 그래프는 진화의 초기 단계에서 빠른 속도로 목적함수 값이 수렴하고 있는 것을 보여주고 있다.

6. 결 론

고정비용과 비선형 단위당 운송비용을 가지는 수송문제는 0-1 혼합정수계획으로 모형화된다. 유전알고리즘을 사용하여 이 문제의 해를 구할 때, 0-1 정수변수로 표현되는 고정비용 부분과 실수변수로 표현되는 비선형 단위당 운송비용 부분이 혼합되어, 가능성을 유지하는 해의 집합을 탐색하는데 어려움이 있으며, 이런 이유로 좋은 해를 구하는 효과적인 방법에 대한 연구가 있어왔다.

본 연구에서는 이단유전알고리즘을 사용하여 가능성을 유지하면서 최적해를 탐색해 나가는 방법을 제안하였다. 상위 단계의 유전알고리즘에서는 각 공급지-수요지 쌍의 수송경로 설치에 대한 최적화를 수행하고, 하위 단계의 유전알고리즘에서는 수송경로가 설치된 공급지-수요지 쌍에 대한 최적 운송량을 결정한다.

〈표 6〉 문제 20x20의 고정비용

f_{ij}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	441	473	515	469	506	485	515	410	510	441	593	587	599	568	482	561	598	451	435	539
2	403	468	539	422	422	458	480	512	462	459	456	462	475	480	569	482	545	532	481	591
3	529	503	444	517	550	491	581	456	534	578	456	473	402	459	597	553	516	521	453	502
4	539	572	418	508	483	414	490	540	500	481	589	597	518	506	523	417	437	456	403	430
5	484	510	512	556	408	481	411	462	521	504	519	466	582	554	515	486	547	432	411	448
6	552	591	469	537	559	407	559	497	441	521	582	500	531	402	533	593	401	591	414	458
7	517	562	409	513	463	410	482	502	547	490	464	414	482	515	533	516	531	562	483	560
8	544	539	546	438	555	414	449	591	564	444	437	447	586	451	451	543	533	555	552	462
9	565	544	580	444	471	511	471	441	516	526	428	412	527	568	485	485	437	499	595	554
10	476	497	509	535	407	530	580	468	423	485	555	471	482	550	487	544	501	434	455	417
11	596	457	524	461	571	464	559	435	460	414	499	491	520	406	463	427	465	552	526	593
12	566	511	521	438	581	464	520	520	554	543	538	532	451	510	502	513	456	483	470	595
13	504	517	493	406	475	427	531	430	439	475	419	523	541	488	484	467	540	442	408	578
14	408	492	548	551	565	442	494	574	566	482	492	470	450	562	509	508	525	417	542	451
15	411	455	593	465	530	473	474	465	443	442	485	524	448	537	518	521	415	408	430	449
16	480	517	457	568	580	505	477	400	450	432	529	597	462	448	576	426	514	521	522	539
17	570	419	479	565	453	536	479	567	486	568	443	546	580	443	597	594	575	505	530	491
18	431	529	523	571	473	410	501	410	475	490	459	594	402	530	523	507	405	544	463	597
19	458	473	464	471	447	519	541	453	515	515	599	587	450	562	554	481	526	499	500	426
20	594	435	518	536	422	530	548	556	486	416	500	515	411	482	423	429	463	599	521	455

〈표 7〉 문제 20x20의 단위비용

u_{ij}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	22	40	34	20	32	22	22	36	25	20	26	28	26	39	40	28	25	23	23	33
2	28	28	34	26	33	24	23	32	21	29	39	25	36	27	26	39	33	33	28	22
3	31	34	39	37	20	31	39	29	34	30	30	29	27	28	25	21	25	40	21	28
4	27	30	23	29	25	33	31	23	39	33	30	28	22	36	29	35	32	37	20	24
5	21	22	26	22	20	31	33	31	37	21	24	34	29	27	34	34	39	31	21	35
6	28	29	30	24	26	22	32	23	39	22	29	25	38	35	25	34	25	21	20	26
7	36	26	24	30	25	27	20	30	24	38	32	35	39	26	21	21	33	28	40	22
8	39	33	27	23	30	31	40	22	20	27	31	39	31	28	37	37	34	35	40	27
9	30	28	34	23	28	31	37	31	28	30	24	33	30	34	38	27	26	26	23	31
10	24	32	27	38	30	24	34	35	32	36	23	36	24	40	21	21	36	27	29	22
11	22	23	21	35	31	31	24	29	35	35	28	38	35	21	33	34	20	29	28	25
12	40	36	34	28	35	25	27	29	39	22	33	27	22	23	21	29	40	31	30	40
13	24	27	28	25	30	22	30	40	31	39	38	29	34	21	35	34	30	23	24	26
14	36	21	30	35	36	26	40	36	34	39	38	28	22	40	36	34	28	20	23	23
15	30	28	22	25	33	37	30	23	38	27	26	36	24	29	38	34	28	27	28	38
16	32	39	30	26	38	25	25	23	27	20	35	37	25	34	24	37	35	29	21	28
17	27	34	26	36	23	32	40	25	39	22	40	33	32	38	32	25	38	21	29	35
18	25	27	28	31	25	32	24	21	38	22	33	38	24	39	27	35	30	36	37	28
19	24	26	28	23	33	22	24	34	30	23	40	38	20	28	26	29	38	37	23	22
20	20	23	27	23	40	38	32	35	23	25	20	37	36	36	26	29	31	36	31	24

〈표 8〉 문제 20x20의 공급가능량과 요구수량

s_i, d_j	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Supply	500	349	506	196	353	653	442	226	243	294	450	345	318	100	250	157	150	139	300	300
Demand	305	354	210	319	294	259	325	330	253	256	366	365	318	398	383	245	339	396	249	307

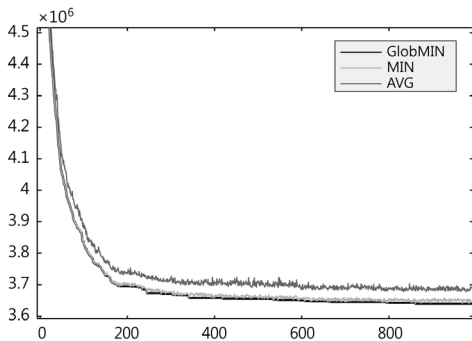
〈표 9〉 문제 20x20의 목적함수 $C_i(x)$ 에 대해서 구해진 해

x_{ij}	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	33.36	19.21	17.73	33.78	20.78	26.53	32	18.47	27.96	24.47	30.38	32.51	25.36	19.21	19.18	19.96	27.66	31.24	23.22	17.01
2	20.59	20.39	10.53	22.15	11.91	14.73	23.95	15.38	19.28	10.43	14.37	24.26	11.94	24.15	23.34	9.18	14.77	22.46	12.38	22.8
3	20.31	20.35	12.73	20.91	35.4	17.49	19.39	26.39	22.47	21.54	31.3	31.03	30.04	28.5	30.04	28.28	33.48	23.47	26.66	26.2
4	14.06	9.43	5.79	9.92	12.72	0	10.69	11.61	7.07	0	13.33	12.19	13.57	13.83	11.63	5.41	13.95	0	13.37	17.44
5	21.68	22.89	15.43	22.77	21.02	12.3	19.1	13.92	9.41	17.38	23.85	16.75	17.59	22.39	25.24	12.53	14.29	18.11	13.4	12.96
6	31.2	35.9	23.77	34.88	28.49	36.2	26.87	39.72	21.67	31.06	31.55	36.42	20.88	29.67	38.42	21.28	36.85	54.57	39.64	33.97
7	17.89	25.24	21.36	20.56	24.32	21.13	29.81	19.67	27.54	12.68	19.63	22.46	17.68	28.33	24.76	28.2	20.19	26.64	11.68	22.27
8	9	12.37	9.56	15.48	12.21	14.78	0	16.36	18.15	10.88	17.24	11.94	8.76	13.63	11.53	8.4	11.41	14.21	0.95	9.14
9	10.11	12.15	0	10.14	13.18	10.62	10.64	13.55	7.9	8.54	15.79	12.41	15.72	15.75	12.09	13.79	17.07	18.9	15.47	9.19
10	16.49	14.91	10.29	9.64	13.13	16.71	15.2	13.84	10.14	10.22	19.04	14.94	20.43	11.33	22.88	13.7	13.88	17.31	10.54	19.39
11	24.44	28.44	26.27	17.29	15.69	16.84	25.59	20.61	17.46	15.39	25.92	20.81	15.3	41.54	19.53	15.8	31.86	27.36	19.11	24.75
12	11.02	12.81	12.43	18.22	14.4	15.11	17.64	16.93	12.14	20.5	17.38	21.68	21.44	27.88	30.61	15.78	15.94	19.82	10.98	12.28
13	16.4	19.06	11.51	16.44	14.91	12.92	13.11	14.6	13.92	10.7	12.45	19.43	14.81	25.28	14.14	10.44	18.22	23.83	17.14	18.7
14	0	12.08	0	0	0	4.62	0	7.07	0	0	0	14.37	10.56	7.38	9.74	0	9.46	10.62	8.08	6.02
15	9.15	18.83	9.28	13.97	11.81	8.29	15.16	13.14	8.5	13.13	19.88	10.72	17.9	20.36	18.39	6.39	13.08	16.77	5.25	0
16	0	9.77	0	8.34	5.48	0	12.33	16.76	7.69	10.13	11.63	8.91	10.85	11.64	11.5	0	9.34	13.65	0	8.99
17	6.2	7.83	0	8.09	8.78	5.56	9.04	11.29	0	8.73	10.46	12.74	6.75	10.79	10.37	10.82	0	18.03	0	4.51
18	10.29	11.99	0	0	12.64	0	10.29	14.05	0	0	11.44	8.99	7.54	12.84	8.3	2.7	11.24	8.68	0	7.98
19	14.3	19.73	8.66	20.94	9.72	17.08	17.62	13.49	9.04	17.33	12.45	17.98	17.59	18.11	21.56	11.37	11.24	13.42	12.05	16.32
20	18.5	20.62	14.67	15.48	7.43	8.1	16.58	13.14	12.67	12.88	27.91	14.46	13.3	15.38	19.75	10.97	15.07	16.92	9.08	17.09

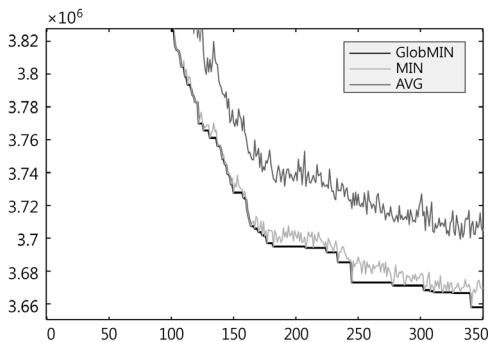
상위 단계와 하위 단계에서 유전염색체의 해의 가능성을 유지하기 위한 검토 과정을 수행하며, 필요한 경우 유전염색체를 수정하는 과정을 수행한다. 또한 균형수송문제의 특성을 이용하여 하나의 가능해를 구하고, 이를 이용하여 하위 단계의 유전 알고리즘에서 초기 세대를 구성하는 유전염색체를 구하는 방법을 사용하였다.

제안된 이단유전알고리즘을 프로그램으로 구현하고 기존에 사용하였던 테스트 문제를 대상으로 전산실험을 수행하였다. 목적함수는 비선형성이 점차 증가하는 다섯 종류를 정하였다.

실험결과에 의하면, 제안된 이단유전알고리즘이 목적함수의 비선형성의 정도에 관계없이 좋은 해를 찾을 수 있었다. 그리고 문제의 크기가 클수록, 목적함수의 비선형성이 높을수록 상대적으로 목적함수의 성능과 수행시간이 더 나은 것을 실험을 통하여 알 수 있었다.



[그림 8] 문제 20x20의 $G1(x)$ 에 대한 전체수렴과정



[그림 9] 문제 20x20의 $G1(x)$ 에 대한 초기수렴과정

추후 연구과제로서는 이단유전알고리즘을 수행하는 과정에서 중복하여 계산되는 염색체의 수를 줄임으로써 수행 시간을 단축하는 방법에 대한 연구가 필요하다.

참고 문헌

- [1] 성기석, “고정비용과 비선형 단위비용을 가지는 최소비용흐름문제를 위한 유전알고리즘”, 『한국경영과학회 추계학술대회』, 이화여자대학교, 2013.
- [2] 성기석, “유전알고리즘에서 선형제약식을 다루는 방법”, 『한국경영과학회지』, 제7권, 제4호(2012), pp.67-72.
- [3] Gen, M., A. Kumar, and J.R. Kim, “Recent network design techniques using evolutionary algorithms,” *Int. J. Production Economics*, Vol.98, No.2(2005), pp.251-261.
- [4] Hitchcock, F.L., “The distribution of a product from several sources to numerous localities,” *Journal of Mathematical Physics*, Vol.20, No.1-4(1941), pp.224-230.
- [5] Jo, J.-B., Y. Li, and M. Gen, “Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm,” *Computers and Industrial Engineering*, Vol.53, No.2 (2007), pp.290-298.
- [6] Munakata, T. and D.J. Hashier, “A Genetic Algorithm Applied to the Maximum Flow Problem,” *The Fifth International Conference on Genetic Algorithms*, Urbana-Champaign, IL, July(1993), pp.488-493.
- [7] Nesa, I. and P. Slobodan, “An Evolution Program for Non-Linear Transportation Problems,” *Journal of Heuristics*, Vol.7(2001), pp.145-168.
- [8] Xie, F. and R. Jia, “Nonlinear fixed charge transportation problem by minimum cost flow-based genetic algorithm,” *Computers and Industrial Engineering*, Vol.63, No.4(2012), pp.763-778.