

Heuristics for Rich Vehicle Routing Problem : A Case of a Korean Mixed Feed Company

Dong Hoon Son · Hwa-Joong Kim[†]

Graduate School of Logistics, Inha University

다특성 차량경로문제에 대한 휴리스틱 알고리즘 : 국내 복합사료 업체 사례

손동훈 · 김화중[†]

인하대학교 물류전문대학원

The vehicle routing problem is one of the vibrant research problems for half a century. Many studies have extensively studied the vehicle routing problem in order to deal with practical decision-making issues in logistics. However, developments of new logistics strategies have inevitably required investigations on solution methods for solving the problem because of computational complexity and inherent constraints in the problem. For this reason, this paper suggests a simulated annealing (SA) algorithm for a variant of vehicle routing problem introduced by a previous study. The vehicle routing problem is a multi-depot and multi-trip vehicle routing problem with multiple heterogeneous vehicles restricted by the maximum permitted weight and the number of compartments. The SA algorithm generates an initial solution through a greedy-type algorithm and improves it using an enhanced SA procedure with three local search methods. A series of computational experiments are performed to evaluate the performance of the heuristic and several managerial findings are further discussed through scenario analyses. Experiment results show that the proposed SA algorithm can obtain good solutions within a reasonable computation time and scenario analyses show that a transportation system visiting non-dedicated factories shows better performance in truck management in terms of the numbers of vehicles used and trips for serving customer orders than another system visiting only dedicated factories.

Keywords : Rich Vehicle Routing Problem, Simulated Annealing Algorithm, Local Search Methods

1. 서론

차량경로문제는 Dantzig and Ramser[4]에 의하여 소개된 이후 지난 반세기 동안 최적화 연구분야에서 가장 많은 연구가 이루어지고 있는 문제 중 하나이다. 차량경로 문제가 학계로부터 광범위하게 연구될 수 있었던 이유는 해당 문제가 지니는 학술적 가치의 중요성뿐만 아니라,

차량경로문제에서 다뤄지는 의사결정 문제가 다양한 산업영역의 물류문제를 해결하는 데에 도움이 될 수 있기 때문이다. 특히 지난 수십 년 동안 진행된 물류분야의 혁신과 글로벌화는 산업계로 하여금 새로운 물류관리 방법의 필요성을 절감하는 계기가 되었다. 그 결과 새로운 물류전략에 대응하는 다수의 차량경로문제 파생연구들이 수행되는 계기가 마련되었다.

차량경로문제는 일반적으로 다항시간 내에 풀기 어려운 난제로 간주된다. 이러한 이유로 상당수의 차량경로 문제 관련 연구들은 해 탐색 기법을 고안하는 데에 주력하였다. 차량경로문제에 대해 최적 알고리즘으로 접근한

연구로써 Fukasawa et al.[5]와 Baldacci et al.[1]은 135개의 방문 노드에 대한 차량경로 해를 도출하는 방법을 제안하였다. 하지만 실제 산업현장에서 고려되는 차량경로 문제는 앞서 제시한 선행연구에서 상정한 문제보다 복잡한 구조를 지니고 있다. 이러한 이유로 효율적, 효과적으로 차량경로문제를 접근할 수 있는 휴리스틱 알고리즘의 개발과 성과측정 연구는 차량경로결정 연구분야에서 중요한 입지를 확보하고 있다.

본 연구는 Kim et al.[8]의 연구에서 제시된 차량경로 문제를 인용하여 이에 대한 휴리스틱 알고리즘을 제안한다. 상기 선행연구는 국내 배합사료 생산기업이 직면한 물류활동 의사결정 문제를 복수의 공장으로부터 최대가능탑재 무게 및 선적공간 제약이 존재하는 복수의 이종 트럭들을 활용하여 공장으로부터 제품을 적재한 트럭들이 고객들의 수요를 충족시키는 다회 방문 차량경로문제로 환원하여 이를 혼합정수계획모형(Mixed integer programming model)으로 제시하고 있다. 본 연구는 해당 선행연구의 연구 성과를 개선하여 동일한 차량경로문제에서 시뮬레이티드 어닐링(Simulated annealing, SA) 알고리즘을 활용한 해 탐색 기법을 제안한다.

일반적으로 SA 알고리즘의 활용가치는 다음과 같이 요약할 수 있다. 우선, SA 알고리즘은 해 탐색 과정 중에 알고리즘이 지역해에 교착되어 해 탐색 범위가 국한되는 문제를 확률적으로 완화시켜주는 기능을 내재하고 있다. 이러한 확률적 접근방법은 SA 알고리즘이 보다 광범위한 해의 영역을 탐색하는 데에 기여한다. 또한 SA 알고리즘은 다양한 이웃해 탐색 기법과 함께 구현될 수 있으며, 비교적 단순한 해 탐색 절차를 통해 최적해를 탐색하기 때문에 차량경로문제에 대한 해 탐색 기법으로서 범용성을 확보하는 데에 유용하다. 일반적으로 실제 산업현장에서 마주하는 차량경로문제는 다수의 제약조건이 고려되기 때문에, 문제의 복잡도가 증가할수록 휴리스틱 알고리즘의 해 탐색 기법이 특화되는 경우가 많다. 그 결과 새로운 유형의 차량경로문제에 대해 동일한 휴리스틱 알고리즘을 적용하여 해를 도출하는 데 제약이 따른다. 메타 휴리스틱 알고리즘의 하나로써 SA 알고리즘은 지역해 탐색 작업을 반복적으로 수행함으로써 최적해에 근사하도록 구현된 탐색 기법이다. 이러한 탐색 기법 구현의 단순함과 유연성 때문에 현재에도 차량경로문제에 대한 탐색 기법으로 널리 쓰이고 있다[12, 20, 25].

본 연구는 서론을 포함하여 총 6장으로 구성되어 있다. 제 2장은 본 연구와 관련이 있는 선행연구들의 연구결과와 연구의의를 설명하고 본 연구가 지니는 차별점을 제시한다. 제 3장과 제 4장은 본 연구에서 상정한 차량경로문제를 설명하고 이를 토대로 개발된 휴리스틱 알고

리즘을 제안한다. 제 5장에서는 본 연구에서 제안하는 휴리스틱의 성능을 평가한 후, 가상데이터를 이용한 실험결과를 서술하였다. 마지막으로 제 6장에서는 본 연구의 결론을 요약하고 연구의 의의를 적시한다.

2. 선행 연구

앞서 언급한 바와 같이, 주어진 차량경로문제에 대한 적절한 휴리스틱 알고리즘을 선택하기 위해서는 문제에서 고려되는 제약조건 유형과 특성을 고려할 필요가 있다. 본 연구에서 고려한 차량경로문제는 상당한 종류의 제약조건이 고려되고 차량경로 네트워크상에는 복수 공장, 복수 차량이 고려된다. Lahyani et al.[10]과 Vidal et al.[21]에 따르면 이와 같이 고도의 연산 복잡도를 지니는 문제를 다특성 차량경로문제로 통칭하고 있다. 다특성 차량경로문제를 정의함에 있어 아직까지는 구체적인 분류기준이 존재하지 않는다. Lahyani et al.[10]은 이에 대한 기준으로 4가지 이상의 운영(정책)적 측면의 제약조건과 6가지 이상의 물리적 측면의 제약조건이 고려되는 차량경로문제는 다특성 차량경로문제로 정의하고 있다. 하지만 Wen et al.[22]의 연구의 경우, 용량제약이 있는 전통적인 차량경로문제를 다루는 대표적인 선행연구의 연산복잡도를 기준으로 다특성 차량경로문제를 분류하고 있다. 본 연구는 앞서 언급한 선행연구들의 관점에서 볼 때 모두 다특성 차량경로문제 조건에 충족한다고 판단하였다. 아울러 차량경로문제의 경우 전통적으로 연구가 활발한 문제이기에 상당한 수의 선행연구가 존재한다. 하지만 선행연구를 분석해 볼 때 다특성 차량경로문제가 새로운 분류기준으로 언급되기 시작하는 시점은 2000년대 중반으로 파악된다. 따라서 본 연구에서는 비교적 최근에 발표된 선행연구들 가운데 다특성 차량경로문제로서 언급되는 연구들을 중심으로 문헌분석을 실시하였다.

Pellegrini[13]은 다특성 차량경로문제 연구의 시발점에서 거론되는 중요한 연구성과를 제시하였다. 본 연구는 이종의 복수트럭에 대하여 복수의 차량운행 시간제약(time window constraint)과 차량이동시간에 대한 최대허용시간을 고려한 차량경로문제를 연구하였다. 아울러 Pellegrini[13]은 Solomon[19]이 활용한 최근접 이웃탐색방법(nearest neighbor method)과 노드 교차 방법(swap method)을 혼합한 휴리스틱 알고리즘을 활용하여 제시한 차량경로문제에 대한 해법을 제시하였다. 여기서 최근접 이웃탐색방법은 차량경로를 규정하는 탐색방법으로써, 본 연구에서는 차량의 다음 방문 노드를 결정함에 있어 임의적으로 선택하여 차량경로를 구성하도록 구현하였다.

Goel and Gruhn[6]은 이종 복수 트럭에 대하여 차량의 무게제약, 차량운행시간의 제약 그리고 차량경로 설정의 시종점 운영제약을 고려하여 배달과 수거문제가 고려되는 차량경로문제를 다루었다. 이들은 대규모 이웃해 탐색 (large neighborhood search) 방법을 응용하여 주어진 차량 경로문제에 대한 해법을 제시한 후, 임의로 생성한 50~200개의 고객에 대한 차량경로문제를 생성하여 휴리스틱 알고리즘의 성능을 평가하였다.

Ropke and Pisinger[15, 16]은 백홀(backhaul) 차량경로 운영특성을 지니는 배달과 수거문제에 대한 차량경로문제를 제시하였다. 본 문제는 복수 차고지 차량경로 네트워크에서의 차량운행시간 제약과 차량의 무게제약을 고려하여 정의되었다. 이들은 본 문제를 풀기 위하여 Shaw [18]이 제안한 대규모 이웃해 탐색 방법을 응용한 휴리스틱 알고리즘을 새롭게 제안하였고, 1,000여 명의 고객에 대한 배송서비스 차량경로문제를 생성하여 486개의 벤치마크 예제들 가운데 183개의 예제에 대해 본 연구에서 제안하는 휴리스틱 알고리즘이 보다 좋은 결과를 제시할 수 있음을 제시하였다.

Wen et al.[24]은 공급자에서부터 고객으로까지의 제품배송에 대해 크로스도킹 운영시스템이 고려되는 차량 경로문제에 대해 타부(tabu) 휴리스틱 알고리즘 기반의 해 탐색 기법을 제안하였다. 아울러 Wen et al.[23]은 복수 이종 트럭을 활용하여 트럭에 대한 무게 제약 및 트럭 운전수에 대한 노동규칙 등을 고려한 차량경로문제를 다루었다. 이들은 먼저 이를 혼합정수계획모형을 제시한 후, 이에 대해 가변 이웃해 탐색(variable neighborhood search) 방법을 이용하여 이에 대한 해법을 제시하였다.

Rieck and Zimmermann[14]는 차량운행 시간제약과 차량이 화물을 선적과 관련된 도킹제약을 고려하여 배송과 수거가 동시에 일어나는 차량경로 문제를 고려하였다. 본 논문에서는 다회 방문 제약을 고려하여 차량경로문제를 고려하지는 않았지만, 특정한 차량이 다회 운영될 수 있다는 점을 고려함으로써 다회 방문 제약이 고려되는 차량경로문제와 유사한 측면이 있다. 이들은 주어진 차량경로문제를 혼합정수계획모형으로 제시한 후, Solomon [19]에서 제시된 벤치마킹 데이터를 이용하여 10~30명의 고객이 고려되는 문제에서 개발한 모델의 컴퓨터 계산성능을 평가하였다.

Santillan et al.[17]은 차량경로문제와 차량에 화물을 선적하는 문제를 동시에 고려하는 복합문제를 고려하였다. 이들은 개미군집 휴리스틱 알고리즘(ant colony algorithm)을 이용하여 차량의 경로를 결정하고 상자 채우기 해 탐색(bin packing technique) 기법을 이용하여 차량에 대한 화물선적 문제를 해결하였다. 이들은 개발한 휴리스틱을 이용하여 Solomon[19]에서 제시된 예시문제를 토대로 개

발된 휴리스틱 알고리즘의 성능을 논하였다.

Cho et al.[3]은 소요차량을 최소화하는 목적 하에 차량경로 결정문제에 대한 2단계 발견적 기법을 제시하였다. 본 연구는 차량의 용량제약과 가용가능 시간제약을 고려하여 삽입방법(insertion method)을 활용하여 완성된 초기해를 생산한 후, 이탈방법(removal method)를 적용하여 해당 초기해를 개선하는 방법으로 최선해를 도출하였다. Lee and Yu[12]는 복수물류센터에 대하여 유전알고리즘 기반의 차량경로결정 문제를 다루었다. 본 연구는 복수물류센터에 대한 차량경로문제를 결정하는 데에 필요한 문제 복잡도를 경감시키기 위하여 우선 각 물류센터가 담당하는 고객의 권역을 결정한 후(clustering), 각 물류센터별 차량경로문제를 순차적으로 결정하였다.

앞서 제시한 선행연구들의 연구결과를 토대로 본 연구의 의의를 요약하면 다음과 같다. 우선, 본 연구는 휴리스틱 알고리즘을 제안한 이후, 이를 활용하여 서로 다른 공장이 차량을 공동으로 이용하여 물류서비스를 행하는 경우와 그렇지 않은 경우에 대한 비교실험을 추진한다. Kim et al.[8]의 연구에서는 비교적 작은 규모의 실제 사료기업의 물류데이터를 활용하여 이에 대한 운영시점을 제시하고 있다. 하지만 지금까지 문헌 조사한 바로는, 이 점에 주목하여 실험을 추진한 연구는 없었다. 따라서 본 연구는 Kim et al.[8]에서 제시한 데이터를 응용하여 대규모 가상의 예제 문제를 생성한 후, 이에 대한 실험결과를 논함으로써 연구의의를 갖는다. 또한 본 연구에서 고려되는 차량경로문제는 실제 국내 사료기업의 물류문제를 해결하기 위한 방안으로써 서로 다른 공장들이 차량을 공동으로 운영하는 데에 필요한 현실적인 운영제약들이 고려된다. 이러한 제약은 다른 선행연구에서 찾아보기 힘든 것이다[8]. 따라서 이에 대한 휴리스틱 알고리즘을 제안한다는 것은 실제 산업계의 물류문제를 해결하는 데에 실질적으로 기여한다는 점에서 또 다른 연구 의의가 있다고 할 수 있다.

다특성 차량경로문제와 관련된 더 많은 선행연구 분석 결과는 Hartl et al.[7], Vidal et al.[21], Lahyani et al.[10]에 자세히 소개되어 있다. 본 연구의 문헌조사에서 다뤄지지 않은 그 외의 연구들에 대한 조사결과는 상기한 선행연구 분석연구를 통해 확인하길 권한다.

3. 문제 설명

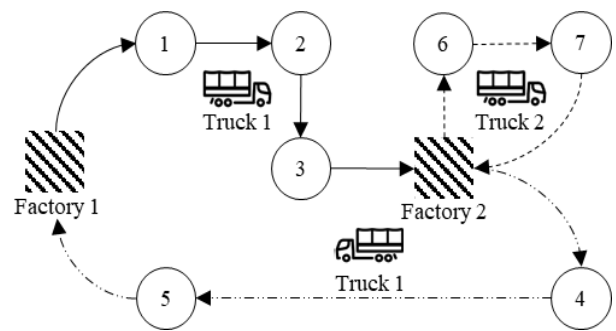
본 연구에서 제안하는 휴리스틱 알고리즘을 효율적으로 설명하기 위해서는 Kim et al.[8]의 연구에서 제시된 차량경로문제 정의와 문제 수립에 필요한 전제조건에 대하여 다시 언급할 필요가 있다. 앞서 언급한 바와 같이,

본 선행연구에서는 두 가지 경우의 차량경로문제를 고려하고 있다. 첫 번째는 각 공장이 담당하는 고객들이 오직 해당 공장에 속한 트럭에 의해서만 주문이 충족되는 경우이다. 두 번째 경우는 각 공장이 담당하는 고객들이 다른 공장에 속한 트럭을 이용하여 주문을 충족할 수 있도록 운영조건이 완화된 경우이다. 즉, 첫 번째 문제는 두 번째 문제에서의 복수 차고지 차량경로문제를 단일 차고지 차량경로문제로 축소시킨 경우이다. 따라서 첫 번째 문제는 서로 다른 공장이 독립적으로 고객들의 주문에 대응하는 문제라고 볼 수 있다. 따라서 본 연구에서는 서로 다른 공장이 차량을 공동으로 운영하는 경우를 중심으로 해당 문제의 운영계약 및 물리적 제약조건을 설명하고자 한다.

본 연구에서 고려하는 차량경로문제는 복수의 사료제품을 생산하는 공장, 사료 운송에 활용되는 트럭, 그리고 해당 제품을 소비하는 고객들이 실제적으로 존재한다. 해당 문제에서 생산공장들은 지리적으로 흩어져 있으며 각 공장은 고객들의 제품 주문을 충족시키기 위해 운송서비스를 수행하는 트럭을 확보하고 운영하고 있다. 해당 물류네트워크 상에 존재하는 고객들은 오직 하나의 공장에서만 제품을 인도 받을 수 있다. 각 고객의 주문은 배송처와 주문량으로 표시되며, 본 문제에서 고객의 주문배송처는 각 고객의 고정된 위치로 표현된다. 제품의 주문량은 개별적으로 구분이 가능한 제품단위가 아닌 중량으로 표현된다. 각 공장에 속한 트럭들은 공장이 담당해야 하는 고객들로부터 접수된 주문을 충족하기 위하여 공장에서 제품을 적재한 후 운송서비스를 수행한다. 이때 각 트럭은 각 고객의 수요를 충족시키기 위해 화물을 배송하기 위해 소요되는 이동거리를 최소화할 목적으로 경로와 스케줄링이 수립된다.

본 문제에서는 트럭에 대한 두 가지의 서로 다른 물리적 제약이 고려된다. 그 중 첫 번째 제약은 트럭의 선적공간 제약이다. 각 트럭은 몇 개의 격실로 구분된 화물적재공간으로 구성되어 있다. 각 격실에는 오직 하나의 고객 주문에 대한 화물만이 채워질 수 있다. 즉, 임의의 격실에 특정 고객의 주문에 대한 제품을 적재한 후, 격실 내부의 공간이 남아도 다른 고객의 주문의 제품을 동일한 격실에 추가로 적재할 수 없음을 의미한다. 두 번째 제약은 트럭의 탑재 중량 제한으로써 트럭이 공장에서 고객의 주문 제품을 적재할 때 트럭이 제품을 탑재할 수 있는 최대중량을 초과할 수 없다는 제약이다. 즉, 각 격실에 적재된 제품의 총중량은 트럭별로 주어진 최대가능 탑재중량을 초과할 수 없다는 의미이다. 이 때문에 화물을 주문한 고객의 수 뿐만 아니라 각 고객별로 주문한 화물의 규모에 의해서도 트럭의 운송스케줄에 영향을 미친다. 단, 본 문제에서는 트럭의 연료제약은 고려하지 않는다.

본 연구에서 고려하는 차량경로문제는 트럭이 자신이 속한 공장으로부터 제품을 적재하여 해당 공장이 담당하는 고객에게만 제품을 인도할 뿐만 아니라, 다른 지역에 있는 다른 공장에 방문하여 제품을 적재한 후, 해당 공장이 담당하는 고객들에게 제품을 전달할 수 있도록 한다. 하지만 만약에 임의의 트럭이 자신이 속하지 않은 다른 공장으로부터 제품을 적재하여 고객에게 운송서비스를 제공할 경우, 반드시 제품을 적재한 공장이 담당하는 고객에게만 제품을 전달할 수 있다. 또한 모든 운송서비스를 마친 트럭은 반드시 자신이 속한 공장으로 반드시 귀환해야 한다. 이와 같은 운영제약을 도식적으로 설명하면 다음과 같다. <Figure 1>에는 공장 1과 2가 있으며, 각각의 공장은 트럭 1과 2를 보유하고 있다고 가정한다. 이때 고객 1, 2, 3의 제품주문은 공장 1이 전담하며 고객 4, 5, 6, 7의 경우 공장 2가 전담한다. 트럭 1이 공장1로부터 제품을 적재한 후 고객 1, 2, 3을 방문하여 제품을 인도하였다. 이때 트럭 1은 자신이 속한 공장 1로 귀환할 수 있지만, 해당 그림에서는 해당 트럭이 공장 2를 방문하여 제품을 추가로 적재한 이후 공장 2가 담당하는 고객 4, 5의 주문에 대응하는 모습을 확인할 수 있다. 이때문에 공장 2에 속한 트럭 2는 고객 4, 5에 대한 방문을 하지 않고도 고객 6, 7만의 주문에 대응함으로써 모든 고객의 주문이 충족되는 모습을 확인할 수 있다. 이에 반해 서로 다른 공장이 독립적으로 물류서비스를 고객들에게 제공하는 경우, 트럭 1은 공장 2를 방문할 수 없다. 따라서 트럭 2는 고객 6, 7을 방문한 후, 고객 4, 5를 추가적으로 방문해야 한다.



<Figure 1> An Example of the Vehicle Routing Problem Obtained from Kim et al.[8]

4. 시뮬레이티드 어닐링 알고리즘

본 연구에서 제안하는 휴리스틱 알고리즘은 SA 알고리즘을 활용하여 개발되었다. SA 알고리즘은 Kirkpatrick et al.[9]에 의하여 처음으로 소개된 해 탐색방법으로써

금속공학의 담금질에서 착안점을 얻어 개발된 메타 휴리스틱 알고리즘이다. SA 알고리즘은 하나의 초기해를 토대로 한정된 수의 계산과정을 통해 최적의 해를 찾는다. 여기서 계산과정 반복횟수는 초기온도(T_0)와 냉각속도(r)에 따라 영향을 받으며, 초기온도는 $e^{-\Delta/T_0} = F_0$ 같이 계산된다. 여기서 F_0 는 예비실험을 통해 결정되는 고정된 파라미터 값이며 Δ 는 SA 알고리즘이 보유하고 있는 최고의 해가 초기온도 결정 과정에서 발견되는 이웃해에 의해서 갱신될 때의 두 해의 목적값 차이의 누적치로 정의된다. 일반적으로 SA 알고리즘은 초기온도가 높은 단계에서부터 계산과정을 시작한다. 이때 온도는 SA 알고리즘 해 탐색 과정에서 지역해에 교착되는 문제를 확률적으로 완화하는 역할을 수행하며, 이러한 이탈 확률은 온도가 높을수록 증가된다. 따라서 일반적인 SA 알고리즘은 초기온도가 높은 상태에서 해 탐색을 시작한다. 이 온도는 해 탐색 과정이 반복될수록 냉각스케줄 $T_k = r \cdot T_{k-1}$ 에 따라 점진적으로 냉각된다.

여기서 T_k 는 k 번째 반복(iteration)에서의 온도를 의미한다. 이러한 과정을 거쳐서 온도가 특정 임계지점보다 낮아진다면 해 탐색 과정이 종료된다. 각각의 계산과정에는 온도가 고정된 상태에서 이웃해를 탐색하는 과정이 내부반복 횟수 L 만큼 수행된다. 이 내부반복 수는 SA 알고리즘이 주어진 해를 기준으로 주위의 이웃해를 탐색하는 횟수를 제어한다. 본 연구에서는 $L = np$ 로 정의하며, 이때 n 은 주어진 차량경로문제에 존재하는 모든 고객의 수를 의미하고 p 는 생산공장의 수를 의미한다. 주어진 내부 반복 횟수 동안 SA 알고리즘을 우수한 이웃해를 탐색하는 절차를 거친다. 이후, 본 SA 알고리즘은 선정된 이웃해는 최고의 가능해(best feasible solution)을 갱신하며, 냉각스케줄에 따라 온도를 낮춰간다. SA 알고리즘의 특성 중 하나는 반복적인 이웃해를 탐색 과정에서 비교적 우수하지 못한 해를 탐색의 기준으로 삼아 알고리즘이 지역해에 교착되는 것을 피한다는 것이다. 이때 교착을 해소하는 절차는 확률적으로 이루어지는데 본 연구에서는 이때의 확률은 $e^{-\Delta/T_k}$ 로 정의하였다. 따라서 온도가 낮아질수록 SA 알고리즘의 지역해 교착 회피 가능성이 낮아진다고 할 수 있다.

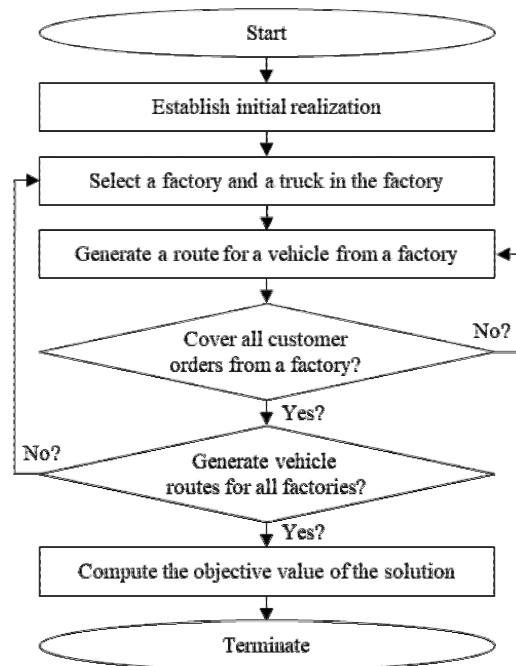
본 장에서는 SA 알고리즘을 구현하기 위하여 초기해 생성방법과 해당 알고리즘 내에서 이웃해를 탐색하는 방법에 대해서 설명하고자 한다. 아울러 해당 알고리즘의 동작을 위해 필요한 파라미터 값들은 다음 장에서 요약하도록 하겠다.

4.1 초기해 생성

제안된 SA 알고리즘의 초기해 탐색지역은 초기해의 생성방법에 따라 달라진다. 본 연구에서는 초기해를 생성하기

위하여 탐욕적 해 탐색 기법을 활용한다. 즉, 모든 고객의 주문이 트럭을 이용한 화물인도로 충족될 수 있도록 각 차량의 경로를 결정함에 있어 공장 또는 마지막 방문 고객의 위치로부터 가장 가까운 고객의 위치를 순차적으로 방문하도록 경로가 구성된다. 이때 각 차량의 최대가능 탑재중량 및 선적공간 제약들을 고려하여 두 조건이 모두 충족되었을 경우에만 차량의 새로운 경로가 결정되는 방법으로 초기해를 생성하였으며 결정된 트럭의 이동경로에 대한 운송비용은 고려하지 않는다. 본 SA 알고리즘에서는 어떠한 트럭이 운송서비스를 수행할지 결정하기 위해 최대가능 탑재중량이 가장 큰 트럭부터 경로를 결정하였다. 이때 탑재중량이 큰 트럭이 탑재중량이 작은 트럭보다 우선적으로 배송활동이 투입되는 데에는 서로 다른 종류의 트럭 간의 단위거리당 운송비용이 무차별하기 때문에 탑재중량이 큰 트럭을 이용하는 것이 더 많은 고객을 방문할 가능성이 높아져 비용절감에 유리하기 때문이다.

초기해 생성과정에서는 임의의 트럭이 자신이 속하지 않은 다른 공장의 제품을 적재하여 운송서비스를 연속하는 경우의 수는 고려하지 않았다. 즉, 본 연구에서 상정하는 차량경로문제를 단일 차고지 문제로 환원하여 각각의 공장이 담당해야 하는 고객 수요에 대해 서로 독립적으로 운송서비스를 완결하는 상황을 가정하여 초기해를 구성하였다. 이와 같은 배송활동은 실제 현업의 사례를 모방한 것으로써 이에 대해 Kim et al.[8]의 연구에 보다 구체적으로 설명되어 있다. <Figure 2>는 초기해 생성과정을 도식적으로 요약하고 있다.



<Figure 2> Procedure for Generating the Initial Solution

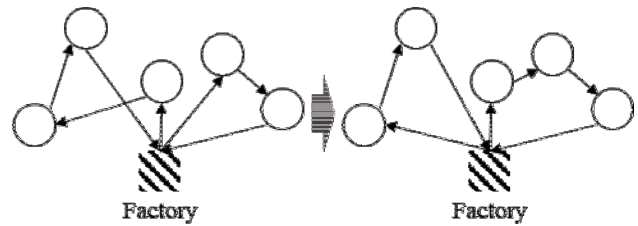
4.2 이웃해 생성

Laporte[11]와 Vidal et al.[21]에 의하면 일반적인 차량 경로문제에 대해 간단하고 명료한 이웃해 탐색 기법으로도 충분히 좋은 해를 도출할 수 있음을 선행연구 분석을 통해 주장하고 있다. 그리고 탐색 기법의 임의성은 주어진 탐색 기법이 좀 더 넓은 이웃해 영역을 탐색하는 데에 기여한다는 점도 밝히고 있다. 본 연구에서 고려하는 차량 경로문제는 특정한 공장에서부터 다수의 차량이 고객의 수요를 충족하기 위하여 입출고를 반복한다. 따라서 SA 알고리즘 개발 시 탐색 기법의 이웃해 탐색 능력과 더불어 주어진 알고리즘의 수렴성의 성능도 고려해야 했다. 따라서 본 연구는 두 가지 고민을 모두 해결하는 데에 유효할 것으로 기대되는 세 가지의 이웃해 탐색 기법을 구현하여 SA 알고리즘을 개발하는 데에 활용하였다.

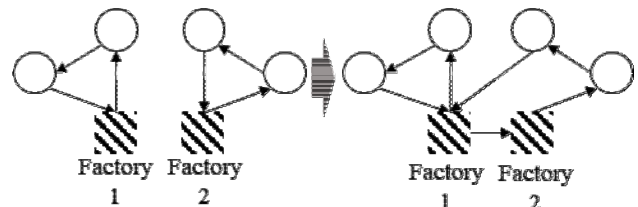
상기 제시한 세 가지의 이웃해 탐색 기법을 <Figure 3>에 요약하여 보여주고 있다. <Figure 3(A)>가 보여주고 있는 첫 번째 이웃해 탐색 기법은 일련의 2-opt 탐색으로써, 한 공장에서 시작하는 임의의 두 차량경로를 선택한 후, 거리가 가장 가까운 방식으로 차량의 경로를 재생성하는 것으로 정의할 수 있다. 이때 차량의 경로가 생성될 때는 차량의 무게제한 및 선적공간 제한을 고려하여 생성되었으며, 두 제약 중 한 가지라도 충족되지 않는 경우 트럭을 공장으로 되돌아오게 함으로써 차량의 경로를 완결한다. 나머지 두 가지 이웃해 탐색 기법은 <Figure 3(B)>를 통해 설명할 수 있다. 그 중 하나는 특정 차고지에서 출발하여 운송활동을 마친 차량이 다른 공장에 방문하여 제품을 선적한 후 해당 공장의 고객에게 운송서비스를 제공한 후 다시 본래의 공장으로 되돌아오도록 구현된 이웃해 탐색 기법이다. 해당 탐색 기법을 구현하기 위해 보다 구체적으로 설명하면 다음과 같다. 우선, 임의의 두 공장을 선택한 후, 각 공장의 완결된 차량경로를 하나씩 선택한다. 이때 두 차량경로를 담당하는 차량들 가운데 최대 적재가능 무게제한이 큰 차량이 나머지 차량의 차량경로를 책임질 수 있도록 차량경로를 재설정한다. 즉, 두 번째 이웃해 탐색 기법은 서로 다른 두 공장이 차량을 서로 공유하는 상황을 구현하기 위해 제안된 것이다. 이와는 반대로 세 번째 이웃해 탐색 기법은 두 번째 이웃해 탐색 기법을 통해 생성되어 특정 공장의 차량이 다른 공장에 의해 서비스를 받는 경우를 해소하기 위해 구현되었다. 즉, 세 번째 이웃해 탐색 기법은 서로 다른 공장이 차량을 서로 공유하는 것을 제한하기 위하여 고안되었다.

본 연구에서 제안하는 SA 알고리즘에서는 위에서 제안된 세 가지의 이웃해 탐색 기법을 모두 적용하여 세 가지의 서로 다른 후보해를 도출한 후, 이들 가운데 가장

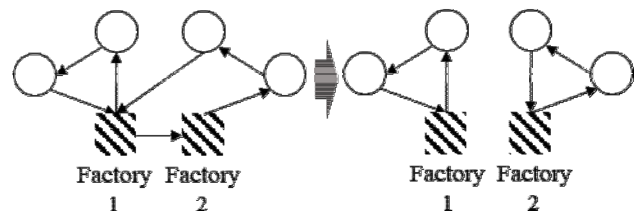
우수한 후보해를 선택하여 본 SA 알고리즘이 보유하고 있는 최우수해와 비교하여 해를 갱신하는 과정으로 구현되었다. 이때, 가장 좋은 후보해는 내부반복 수만큼 이웃해 탐색을 실시하여 도출된 후보해들 가운데 목적값이 가장 우수한 후보해를 의미한다.



(a) First local search method



(b) Second local search method



(c) Third local search method

<Figure 3> Examples of Local Search Methods

5. 실험 결과

5.1 SA 알고리즘 성능평가

본 장에서는 제 4장에서 개발된 SA 알고리즘의 성능을 평가하고 민감도 분석을 수행한다. SA 알고리즘을 동작하기 위해 파라미터 r 과 F_0 는 각각 0.99와 0.7로 부여하였다. 이와 같은 파라미터 값들은 일련의 선행실험으로부터 도출된 알고리즘 해의 수렴도와 목적값의 변화 관찰값을 토대로 결정된 것들이다. 본 연구에서 알고리즘의 성능평가와 민감도 분석 시 컴퓨터의 사양은 Intel(R) Core(TM) i7-2600 CPU@ 3.40GHz, RAM 4.00GB이며 본 연구의 SA 알고리즘은 Visual Studio 2013 C++로 구현되었다. 이때 알고리즘의 성능평가를 위해 상업용 소프트웨어 CPLEX 12.6.1.0을 이용하여 수행되었다.

SA 알고리즘의 성능평가를 위해 생성된 예제문제 생성방법을 요약하면 다음과 같다. 본 연구는 Kim et al.[8]의 실험연구에서 활용된 국내 배합사료 생산기업의 실제 물류활동 및 고객주문 데이터와 생산공장과 고객들 간의 트럭 이동거리를 바탕으로 데이터를 가상으로 생성하였다. 그리고 본 SA 알고리즘과 상업용 소프트웨어의 성능비교를 위하여 10개의 예제문제를 5개의 서로 다른 고객의 수(10, 20, 30, 40, 50)에 따라 독립적으로 생성하였다. 따라서 본 실험을 위하여 총 50개의 가상의 예제문제가 생성되었다. 이때의 고객의 수는 각 공장이 담당해야 하는 고객의 수를 의미하는 것으로서, 만약 공장이 p 개 존재한다면 고객의 수가 10으로 부여된 경우 주어진 차량 경로문제에 존재하는 총 고객의 수는 $10p$ 임을 의미한다. 각 예제문제에는 2개의 생산공장이 존재하며 각 공장에는 4대의 트럭이 운송서비스를 위해 대기하고 있다. 각 트럭의 최대 운송가능 횟수는 4회로 한정하였으며 각 트럭의 최대가능 탑재중량은 15,000kg과 10,000kg 중 임의로 부여되었다. 트럭의 선적가능 공간의 수에 대하여, 최대가능 탑재중량이 15,000kg 트럭의 경우 선적가능 공간의 수가 4개, 그리고 최대가능 탑재중량이 10,000kg인 트럭의 경우 3개가 있다고 가정하여 데이터를 생성하였다. 각 고객의 수요의 크기는 $U(2,000\text{kg}, 4,500\text{kg})$ 에서 임의로 생성하였으며, 이때 $U(a, b)$ 는 $[a, b]$ 구간에서의 연속균등분포를 의미한다. 그 외의 나머지 파라미터(고객 간의 이동거리, 고객과 생산공장 간의 이동거리, 트럭의 운영비용)들은 Kim et al.[8]의 데이터 생성법을 인용하였다. 휴리스틱 성능비교를 위하여 고려되는 대조해는 CPLEX의 연산과정을 통해 도출된 결과를 활용한다. 이때 각 예제문제에 대하여 최대 연산가능 시간을 두 시간으로 제한하였으며, 주어진 연산시간 내에 최적해를 도출하지 못한 경우 CPLEX가 도출한 휴리스틱 해 값(상한값)을 비교대상으로 선정하여 SA 알고리즘이 도출한 해와 비교하였다.

<Table 1>는 휴리스틱 알고리즘의 성능평가 결과를

요약하고 있다. 해당 표에는 CPLEX와 SA 알고리즘이 도출한 결과의 차이를 백분율로써 평균값으로 표현한 결과(Average percentage gap)와 예제 문제를 푸는 데에 소요된 연산시간의 평균값을 보여주는 결과(CPLEX CPU, SA CPU)를 보여주고 있다. 이때 CPLEX와 SA 알고리즘이 도출한 해품질의 차이는 $100 \cdot (O_{SA} - O_C) / O_C$ 로 계산되었으며, 여기서 O_{SA} 와 O_C 는 각각 휴리스틱 알고리즘과 CPLEX가 도출한 목적값을 의미한다. 이때의 목적값은 트럭의 이동거리에 대한 운송비용으로 측정되며 구체적인 계산 모형은 본 논문 부록에 수리모형의 목적식으로 요약되어 있다.

해당 표에서 볼 수 있듯이, 본 연구에서 제안하는 SA 알고리즘은 전반적으로 양호한 성능을 보여주는 것으로 사료된다. CPLEX의 경우 각 공장이 담당하는 고객의 수가 10인 경우에 대해 10문제 중에서 오직 1문제만 주어진 연산시간 동안 최적해를 도출하였고, 그 외의 경우 모두 최적해를 도출하지 못하였다. 이는 문제의 규모가 증가할수록 계산 복잡도가 증가하기에 나타나는 문제로 보인다. 이에 반면에 본 연구에서 제안하는 SA 알고리즘은 CPLEX보다 비교적 빠른 시간 내에 결과를 도출하는 것을 확인할 수 있었으며, 고객의 수가 50인 경우 평균적으로 주어진 문제를 4분 내에 풀어내는 것을 확인할 수 있다. SA 알고리즘이 도출한 해와 CPLEX가 도출한 해의 품질 측면에서 볼 때, 본 연구에서 제안하는 SA 알고리즘은 문제의 크기가 작은 경우에 있어 CPLEX보다 열등한 결과를 보여주는 것을 볼 수 있다. 고객의 수가 10, 20인 경우 본 SA 알고리즘의 해의 목적값은 CPLEX가 도출한 해의 최적해 또는 상한값보다 각각 3.56%, 6.01%의 괴리를 보였기 때문이다. 하지만 고객의 수 30, 40, 50의 경우 본 SA 알고리즘이 도출한 해는 CPLEX가 도출한 경우보다 6.55%, 9.17%, 17.24% 우수한 결과를 보여주는 것으로 드러나 대규모 물류네트워크에서의 본 SA 알고리즘의 적용타당성을 일부 입증하고 있다. CPLEX와 SA 알고리즘이 도출하는 해들의 목적값 차이를 표준편차 분석을 하면, 본 SA

<Table 1> Performance of the SA Algorithm

Number of customers served by a factory	Average percentage of solution gap ^a (%)	Std. dev. of the solution gap ^b (%)	Average CPU time ^c	
			CPLEX	SA
10	3.56	12.00	6743.05	2.05
20	6.01	7.41	7200.00	15.25
30	-6.55	9.40	7200.00	49.80
40	-9.17	9.06	7200.00	118.55
50	-17.24 ^d	7.29	7200.00	241.30

^aaverage percentage gap of SA solutions from the solutions of CPLEX.

^baverage standard deviation of the average percentage gap of SA solutions from the solutions of CPLEX.

^caverage computation time (in seconds) taken by CPLEX and SA to solve given test instances.

^dnote that only four test instances gave feasible solutions from CPLEX for given number of customers severed by a factory.

알고리즘은 주어진 문제들에 대해 평균적으로 7.29%~12%의 편차를 드러내는 것을 확인할 수 있다. 고객의 수가 10일 때의 표준편차가 12%로 드러난 점을 해석할 때, 본 SA 알고리즘이 최적해에 대해 비교적 강건한 결과를 내놓지 못하는 측면이 있다고 할 수 있다. 하지만 나머지 문제에서는 대체적으로 7~9% 정도의 표준편차 분석결과를 보여주고 있으며, 이는 문제의 크기에 대해 일관적인 상승 또는 하향의 패턴으로 드러나지는 않고 있다. 따라서 현업에서 본 SA 알고리즘을 이용하여 물류활동 계획을 수립할 경우, 이와 같은 SA 알고리즘의 특성을 고려할 때 운송거리 변동에 따른 배송비용의 편차가 두드러질 수 있음을 인지해야 할 것이다.

5.2 시나리오 분석결과

본 연구는 해당 SA 알고리즘이 본 연구에서 상정하는 차량경로문제의 제약조건을 토대로 대규모 물류 네트워크에서도 최적해에 근사하거나 납득할 수 있을 정도의 질 좋은 해를 생산할 수 있다는 가정 하에 시나리오 분석을 실시하였다. 본 연구에서 수행되는 시나리오 분석은 단일 차고지 차량경로문제와 복수 차고지 차량경로문제에서의 성과를 비교한다. 이때 복수 차고지 차량경로문제는 본 연구에서 제안하는 모형에서와 같이 트럭이 서로 다른 공장의 화물을 적재하고 해당 공장이 담당하는 고객들에게 운송서비스를 제공하여 차량을 통합 운영하는 경우를 의미하며, 단일 차고지 차량경로문제는 서로 다른 공장이 독립적으로 자신들의 고객들에게 제품을 전달하는 경우를 의미한다(본 연구에서는 표기의 편의를 위하여 전자의 경우를 Multi-depot case, 후자의 경우를 Single-depot case로 표기한다). 본 연구에서는 SA 알고리즘이 도출하는 해와 그에 따른 트럭운영의 성과로 구분하여 두 가지의 시나리오 분석결과를 제시한다. 실험조건은 앞서 휴리스틱 알고리즘 성능평가 때 고려된 파라미터 생성법 및 문제설정과 동일하다. 단, 서로 다른 공장이 독립적으로 물류활동을 하는 경우의 해를 도출하기 위하여 SA 알고리즘을 구현할 당시 고려되었던 두 번째와 세 번째 이웃해 탐색 기법은 제외하여 실험결과를 도출하였다. 즉, 단일 차고지 차량경로문제에서는 <Figure 3>에서 제시된 이웃해 탐색 기법 2, 3은 고려되지 않았다.

<Table 2>와 <Table 3>은 두 시나리오 분석 결과를 요약하여 제시하고 있다. <Table 2>는 본 연구의 SA 알고리즘이 Single-depot case와 Multi-depot case로부터 도출한 해의 목적값의 차이와 주어진 문제에서 도출된 해들의 변동성을 비교하고 있다. 이때 본 표에서 Average percentage of solution gap은 $100 \cdot (OS-OM)/OM$ 로 정의되었으며, OS와 OM는 각각 Single-depot case과 Multi-depot case 해

<Table 2> Comparison Results between Single-Depot and Multi-Depot Cases in Terms of Solution Quality

Number of customers served by a factory	Average percentage gap ^a (%)	Ratio of std. dev. ^b
10	-3.74	0.99
20	-4.66	0.66
30	-7.70	0.74
40	-11.27	0.94
50	-14.66	0.98

^aaverage percentage gap of SA solutions between the single-depot and multi-depot cases.

^bratio of standard deviations of solutions between the single-depot and multi-depot cases.

값을 의미한다. Ratio of standard deviation은 Single-depot case와 Multi-depot case에서 SA 알고리즘이 도출한 해의 변동성을 비교한 것으로써, Std.m/Std.s로 정의된다. 이때 Std.s와 Std.m은 각각 Single-depot case와 Multi-depot case의 예제문제에서 도출된 해의 표준편차를 의미한다. <Table 2>에서와 같이 본 연구에서 개발된 SA 알고리즘은 Multi-depot case에서 Single-depot case보다 질적으로 우수한 해를 생산하고 있음을 확인할 수 있다. Average percentage gap에 따르면 각 공장이 담당하는 고객의 수가 10일부터 50일 때까지 -3.74%에서 -14.66%로 그 값이 점진적으로 확장하는 것을 볼 수 있다. 즉, Multi-depot case는 Single-depot case보다 서로 다른 공장이 차량을 공유하는 시스템을 갖춤으로써 차량의 이동거리 및 이동시간을 줄일 수 있음을 실험적으로 드러내고 있는 것이다. 또한 본 SA 알고리즘은 Multi-depot case의 경우에 대해 Single-depot case의 문제조건보다 보다 변동성이 적인 해를 도출하고 있음을 확인할 수 있다. Ratio of standard deviation에서 확인할 수 있듯이 Multi-depot case에서 예제문제에 대해 도출되는 해의 편차가 Single-depot case에서보다 상대적으로 낮다는 것을 보여준다. 즉, 임의의 기업이 Single-depot case와 유사한 물류시스템에서 Multi-depot case와 유사한 물류시스템으로 전환할 경우, 물류비 절약은 물론, 물류비의 변동성을 낮출 수 있어 경영의 재무적 측면의 가시성을 보다 확보할 수 있음을 의미한다.

<Table 3>은 앞서 수행된 시나리오 분석의 결과를 트럭운영 측면에서 재조명하고 있는 결과를 요약하여 보여준다. 본 표에서 Number of trucks used는 각 예제문제에 대해 활용된 실제 트럭의 평균치를 보여주며, Number of trips of trucks는 트럭에 의해서 수행된 평균 출고횟수 혹은 트럭이 공장을 떠나서 다시 본래의 공장으로 돌아오는 경우의 평균 횟수를 의미한다. 마지막으로 Capacity utilization은 최대 적재가능 무게대비 출고 당시 선적한 화물의 평균 무게의 비율을 의미한다. 본 표에서 확인할

<Table 3> Comparison Results between Single-Depot and Multi-Depot Cases in Terms of Vehicle Operations

Number of customers served by a factory	Single-depot case						Multi-depot case					
	Number of trucks used		Number of trips of trucks		Capacity utilization(%)		Number of trucks used		Number of trips of trucks		Capacity utilization(%)	
	15-ton	10-ton	15-ton	10-ton	15-ton	10-ton	15-ton	10-ton	15-ton	10-ton	15-ton	10-ton
10	0.8	1.2	3.5	5.4	22.0	45.2	0.8	1.2	3.5	5.5	22.1	41.1
20	1.3	1.6	9.7	10.3	40.4	43.6	1.3	1.6	8.8	9.4	44.2	47.6
30	2.4	2.3	16.1	14.7	32.2	48.8	2.1	1.9	14.9	12.9	33.3	54.3
40	3.1	2.9	21.3	19.2	43.0	41.0	2.9	2.6	20.9	18.9	44.5	39.6
50	2.9	3.7	20.5	27.4	38.1	61.5	2.7	3.3	18.9	24.5	41.3	67.7

수 있듯이, 각 공장이 담당하는 고객의 수가 증가할수록 필요로 하는 실제 트럭의 수와 출고횟수가 증가함을 확인할 수 있다. 하지만 실제 사용된 트럭의 종류별 활용 수량과 출고횟수 측면에서 볼 때 고객의 수 증가에 따라 트럭 종류별로 의미 있는 차이를 드러내지는 않았다. 아울러, Capacity utilization 측면에서 볼 때 일반적으로 10톤 트럭의 경우 15톤 트럭의 경우보다 수치가 더 높게 나오므로써, 화물 적재측면에서 10톤 트럭이 운영효율성이 비교적 높게 나왔다. Single-depot case와 Multi-depot case로부터 도출된 해의 차이를 조명한 경우, Multi-depot case의 해가 대체적으로 더 적은 트럭의 수를 활용하는 것으로 드러났으며, 아울러 출고 횟수 또한 Single-depot case보다 낮게 드러나는 것을 확인할 수 있다. 또한 Capacity utilization 측면에서 볼 때, Multi-depot case가 모든 고객의 수에 대해 일관적으로 우수한 운영효율성을 보이는 것은 아니지만, Single-depot case의 경우보다 대체적으로 3% 정도의 적재효율이 증가하는 것을 볼 수 있다. 이는 서로 다른 공장이 서로의 트럭을 융통성 있게 운용함으로써 유휴 트럭의 감소, 출고횟수 감소 및 적재효율 증가를 유도할 수 있음을 보여주는 결과라고 할 수 있다.

6. 결론

본 연구는 Kim et al.[8]에서 제안된 다특성 차량경로 문제에 대한 휴리스틱 알고리즘을 제안하는 것을 목표로 추진되었다. 본 연구에서 고려된 차량경로문제는 복수의 공장으로부터 트럭의 화물탑재 중량 및 선적공간 제약이 존재하는 복수의 이종 트럭들을 활용하여 고객의 주문을 충족시키는 다회 방문 차량경로문제이다. 이에 대해 본 연구는 SA 알고리즘을 활용하여 해당 문제에 대한 해 탐색 기법을 제안하였다. SA 알고리즘의 초기해는 차량 경로를 구성함에 있어 트럭의 현재 위치로부터 가장 가까운 고객을 방문하도록 탐욕적 탐색 기법을 활용하여 구현되었다. 아울러, 본 연구에서 제안되는 SA 알고리즘은 세 가지의 이웃해 탐색 기법을 통해 최적해를 탐색한다.

첫 번째 이웃해 탐색 기법은 임의의 두 차량경로를 재설정하는 것이다. 즉, 차량의 무게제한 용량이 큰 차량부터 가장 최근의 방문지로부터 거리가 짧은 순으로 임의로 선택된 두 차량경로에 포함된 고객들에 대한 방문순서를 재설정하는 것이다. 두 번째 이웃해 탐색 기법은 서로 독립적으로 물류활동을 수행하는 공장들 간의 물류 네트워크를 구축한다. 따라서 해당 이웃해 탐색 기법은 SA 알고리즘이 단일 차고지 차량경로문제로부터 복수 차고지 차량경로문제에 대한 이웃해 영역을 탐색할 수 있도록 가능케 한다. 마지막으로 세 번째 이웃해 탐색 기법은 두 번째 이웃해 탐색 기법을 통해 구축된 서로 다른 공장들 간의 물류 네트워크를 해체하는 방식으로 구성되었다. 즉, 해당 이웃해 탐색 기법은 두 번째 이웃해 탐색 기법을 통해 생성된 새로운 이웃해 탐색영역을 배제하는 것이라고 할 수 있다.

본 연구에서 제안된 SA 알고리즘의 성능은 상업용 소프트웨어 CPLEX로부터 도출된 해와 비교평가를 통해 이루어졌다. 평가 결과를 요약하자면, 비교적 작은 규모의 예제문제(각 공장이 담당하는 고객의 수가 10, 20인 경우)에 대해서는 대체적으로 CPLEX가 도출한 해가 SA 알고리즘에 비해 3~6% 정도 우수한 해를 제시하는 것으로 드러났다. 하지만 주어진 예제문제의 규모가 커질수록(각 공장이 담당하는 고객의 수가 30, 40, 50인 경우) SA 알고리즘의 해가 CPLEX가 도출한 해보다 우수한 결과를 도출하는 것으로 확인되었다. 아울러 SA 알고리즘은 각 공장이 담당하는 수가 50인 경우에도 4분 내에 해를 도출하는 것으로 드러나 CPLEX보다 비교적 빠른 시간 내에 양질의 해를 도출하는 것으로 평가되었다.

본 연구는 이러한 실험결과를 바탕으로 시나리오 분석을 실시하여 Single-depot case와 Multi-depot case에서의 해의 품질과 트럭운용 측면에서의 차이를 비교하였다. 해의 품질 측면에서의 실험결과를 볼 때 Multi-depot case의 경우 Single-depot case보다 우수한 해를 도출할 수 있음을 보였다. 이는 Multi-depot case가 서로 다른 공장에서의 트럭을 서로 사용할 수 있게 함으로써 불필요한 트럭의 이동을 줄여 물류비 절감에 이르게 할 수 있음을 보여주는

것이라고 할 수 있다. 아울러 Multi-depot case로부터 도출된 해는 Single-depot case에서의 경우보다 해의 변동성 또한 낮은 것으로 실험적으로 확인되어, 기업으로 하여금 물류비 변동성에 대한 리스크를 완화할 수 있음을 보여주었다. 아울러 트럭운영의 성과측면에서 볼 때 Multi-depot case는 Single-depot case보다 비교적 적은 수의 트럭으로도 고객의 주문에 대응할 수 있음을 확인할 수 있었으며, 서로 다른 공장이 서로의 트럭을 고객주문 대응에 활용함으로써 출고횟수 감소 및 적재효율을 증진시키는 것을 확인하였다. 하지만 본 SA 알고리즘으로부터 도출된 해를 분석해 볼 때 서로 다른 종류의 트럭의 운영적 특성에는 뚜렷한 차이가 드러나지 않았다.

앞서 제시한 본 연구의 성과를 개선하기 위하여 다음과 같은 문제 상황에 대한 추가연구를 수행하고자 한다. 우선, 본 문제에서 고객의 주문에 대해 분할배달(split delivery) 조건이 가능한 상황을 고려하는 것이 유의미할 수 있다. 본 연구의 실험결과에서와 같이 Multi-depot case의 경우 가용트럭의 수 및 물류활동의 효율성 측면에서 Single-depot case보다 우수한 성과를 보여주었지만, 고객의 주문이 평균적인 수준보다 훨씬 크거나 작은 경우 물류활동의 효율성 악화에 영향을 미칠 개연성이 있다. 이러한 문제에 대한 대안으로 분할배달이 고려되는 Multi-depot case 차량경로문제를 고려할만한 가치가 있다. 아울러 차량운행 시간제약을 고려하여 이동거리 혹은 제품 상하차에 필요한 소요시간의 불확실성을 고려하여 본 연구에서 제기한 문제를 개선할 필요가 있다.

Acknowledgement

This work was supported by Inha University Research Grant.

References

- [1] Baldacci, R., Christofides, N., and Mingozzi, A., An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts, *Mathematical Programming*, 2008, Vol. 115, No. 2, pp. 351-385.
- [2] Baldacci, R., Toth, P., and Vigo, D., Recent advances in vehicle routing exact algorithms, *4OR : A Quarterly Journal of Operations Research*, 2007, Vol. 5, No. 4, pp. 269-298.
- [3] Cho, B.-M., Kim, J.-G., and Lee, D.-H., A two-stage heuristic for period vehicle routing : Minimizing the fleet size, *Journal of Society of Korea Industrial and Systems Engineering*, 2008, Vol. 31, No. 3, pp. 90-100.
- [4] Dantzig, G.B. and Ramser, J.H., The Truck Dispatching Problem, *Management Science*, 1959, Vol. 6, No. 1, pp. 80-91.
- [5] Fukasawa, R., Longo, H., Lysgaard, J., Aragao, M., Reis, M., Uchoa, E., and Werneck, R., Robust branch-and-cut-and-price for the capacitated vehicle routing problem, *Mathematical Programming*, 2006, Vol. 106, No. 3, pp. 491-511.
- [6] Goel, A. and Gruhn, V., Large neighborhood search for rich VRP with multiple pickup and delivery locations, In *Operations Research Proceedings 2005*, Haasis, H.D., Kopfer, H., and Schönberger, J. (Eds.), *Operations Research Proceedings*, 2005, pp. 367-372.
- [7] Hartl, R.F., Hasle, G., and Janssens, G.K., Special issue on rich vehicle routing problems, *Central European Journal of Operations Research*, 2006, Vol. 14, No. 2, pp. 103-104.
- [8] Kim, H.J., Kim, J.W., Yang, W.S., Lee, K.Y., and Kwon, O.S., Effects of Truck-shared Delivery : A Case of a Korean Mixed Feed Company, *Journal of International Logistics and Trade*, 2016, Vol. 14, No. 2, pp. 165-173.
- [9] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P., Optimization by simulated annealing, *Science*, 1983, Vol. 220, pp. 671-680.
- [10] Lahyani, R., Khemakhem, M., and Semet, F., Rich Vehicle Routing Problems : From a Taxonomy to Definition, *European Journal of Operational Research*, 2015, Vol. 241, No. 1, pp. 1-14.
- [11] Laporte, G., Fifty years of vehicle routing, *Transportation Science*, 2009, Vol. 43, No. 4, pp. 408-416.
- [12] Lee, S.-C. and Yu, J.-C., Improved VRP & GA-TSP model for multi-logistics center, *Journal of the Korea Academia-Industrial Cooperation Society*, 2007, Vol. 8, No. 5, pp. 1279-1288.
- [13] Pellegrini, P., Application of Two Nearest Neighbor Approaches to a Rich Vehicle Routing Problem, Technical Report, 2005.
- [14] Rieck, J. and Zimmermann, J., A new mixed integer linear model for a rich vehicle routing problem with docking constraints, *Annals of Operations Research*, 2010, Vol. 181, No. 1, pp. 337-358.
- [15] Ropke, S. and Pisinger, D., A unified heuristic for a large class of vehicle routing problems with backhauls, *European Journal of Operational Research*, 2006b, Vol. 171, No. 3, pp. 750-775.

- [16] Ropke, S. and Pisinger, D., An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows, *Transportation Science*, 2006a, Vol. 40, No. 4, pp. 455-472.
- [17] Santillan, C.G., Reyes, L.C., Rodriguez, M.L.M., Barbosa, J.J.G., Opez, O.C.L., Zarate, G.R., and Hernandez, P., Variants of VRP to optimize logistics management problems, In Zezzatti, O., Ochoa, C.A., Chira, C., Hernandez, A., and Basurto, M. (Eds.), *Logistics Management and Optimization through Hybrid Artificial Intelligence Systems*, 2012, pp. 207-237.
- [18] Shaw, P., Using constraint programming and local search methods to solve vehicle routing problems, In : Proceedings of the 4th International Conference on Principles and Practice of Constraint Programming (CP-98) (Lecture Notes in Computer Science), Maher, M. and Puget, J.F. (Eds.), 1998, pp. 417-431.
- [19] Solomon, M.M., Algorithms for the vehicle routing and scheduling problems with time window constraints, *Operations Research*, 1987, Vol. 35, No. 2, pp. 166-324.
- [20] Suman, B. and Kumar, P., A survey of simulated annealing as a tool for single and multiobjective optimization, *Journal of the Operational Research Society*, 2006, Vol. 57, No. 10, pp. 1143-1160.
- [21] Vidal, T., Crainic, T.G., Gendreau, M., and Prins, C., Heuristics for multi-attribute vehicle routing problems : A survey and synthesis, *European Journal of Operational Research*, 2013, Vol. 231, No. 1, pp. 1-21.
- [22] Wen, M., Clausen, J., and Larsen, J., Rich Vehicle Routing Problems and Applications, [Ph.D Thesis], [Lyngby, Denmark] : Management, Technical University of Denmark, 2010.
- [24] Wen, M., Krapper, E., Larsen, J., and Stidsen, T.K., A multilevel variable neighborhood search heuristic for a practical vehicle routing and driver scheduling problem, *Networks*, 2011, Vol. 58, No. 4, pp. 311-322.
- [23] Wen, M., Larsen, J., Clausen, J.F., and Laporte, G., Vehicle routing with cross-docking, *Journal of the Operational Research Society*, 2008, Vol. 60, No. 12, pp. 1708-1718.
- [25] Zeng, L., Ong, H.L., and Ng, K.M., A generalized crossing local search method for solving vehicle routing problems, *Journal of the Operational Research Society*, 2007, Vol. 58, No. 4, pp. 528-532.

ORCIDDong-Hoon Son | <http://orcid.org/0000-0003-1313-3315>Hwa-Joong Kim | <http://orcid.org/0000-0003-4976-4813>

<Appendix>

앞서 밝혔던 바와 같이 본 연구에서 인용하는 차량경로문제는 Kim et al.[8]이 제안한 것이다. 본 연구는 독자의 편의를 위하여 선행연구에서 제시된 수리모형을 다시 제시하고자 한다.

집합

- I 모든 사료생산공장의 집합
- J 모든 사료제품 구매고객의 집합
- J_i 사료생산공장 i 로부터 제품배송 서비스를 받는 사료제품 구매고객의 집합
- K_j 고객 j 에게 제품배송 서비스를 하는 사료생산공장이 담당하는 모든 고객의 집합
- T 다회 방문에 대한 순서 인덱스 집합
- V 모든 트럭의 집합

파라미터

- c_{ij} 고객 i 에서 j 까지의 물류비용
- d_j 고객 j 의 사료제품 수요
- f_j 고객 j 의 주문을 담당하는 생산공장
- i_v 트럭 v 가 속한 사료생산공장
- i_v^+ 트럭 v 가 최초로 대기하는 사료생산 공장(물리적으로 공장 i_v 와 동일함)
- m_v 트럭 v 의 선적공간의 수
- q_v 트럭 v 가 탑재할 수 있는 최대중량

결정변수

- X_{vij}^t 만약 트럭 v 가 t 번째 운송 중에 사료생산공장/고객 i 로부터 j 까지 이동할 경우 1, 그렇지 않으면 0
- Y_{vj}^t 만약 트럭 v 가 t 번째 운송 중에 고객 j 가 사료생산공장 i 로부터 서비스를 받는다면 1, 그렇지 않으면 0
- Z_{vij}^t 트럭 v 가 t 번째 운송 중에 사료생산공장/고객 i 로부터 j 까지 이동할 때 사용하는 선적공간의 수

본격적으로 수리모형을 밝히기 앞서 보조 집합 $I_v^- = I \setminus \{i_v\}$, $I_v^+ = I_v^- \cup \{i_v^+\}$ 을 정의한다. 또한 $\lceil \cdot \rceil$ 는 \cdot 보다 작지 않은 가장 작은 정수를 의미한다. 선행연구에서 제시된 모형을 언급하면 다음과 같다.

$$\text{Minimize } \sum_{v \in V} \sum_{i \in I} \sum_{j \in J_i} \sum_{t \in T} c_{ij} X_{vij}^t + \sum_{v \in V} \sum_{i \in J} \sum_{j \in I_v^+ \cup K_i} \sum_{t \in T} c_{ij} X_{vij}^t$$

subject to

$$\sum_{v \in V} \sum_{t \in T} X_{vfj}^t + \sum_{v \in V} \sum_{i \in K_j} \sum_{t \in T} X_{vij}^t = 1 \quad \forall j \in J \tag{1}$$

$$X_{vfh}^t + \sum_{i \in K_h} X_{vih}^t = \sum_{j \in I_v^+ \cup K_h} X_{vhj}^t \quad \forall v \in V, h \in J, t \in T \tag{2}$$

$$\sum_{j \in J_i} X_{vih}^t = \sum_{j \in J_h} X_{vhj}^t \quad \forall v \in V, h \in I_v^-, t \in T \tag{3}$$

$$\sum_{j \in J_i} X_{vij}^t = \sum_{i \in J} X_{vii_v^+}^t \quad \forall v \in V, t \in T \tag{4}$$

$$\sum_{v \in V} \sum_{t \in T} Y_{vfj}^t = 1 \quad \forall j \in J \tag{5}$$

$$\sum_{j \in J_i} \left\lceil \frac{d_j}{q_v/m_v} \right\rceil Y_{vij}^t \leq m_v \quad \forall v \in V, i \in I, t \in T \quad (6)$$

$$\sum_{i \in K_j \cup \{f_j\}} X_{vij}^t = Y_{vfj}^t \quad \forall v \in V, j \in J, t \in T \quad (7)$$

$$Z_{vfh}^t - \sum_{i \in k_h} Z_{vih}^t - \sum_{j \in i_v^+ \cup k_h} Z_{vhj}^t = \left\lceil \frac{d_h}{q_v/m_v} \right\rceil Y_{vfh}^t \quad \forall v \in V, h \in J, t \in T \quad (8)$$

$$Z_{vij}^t \leq m_v X_{vij}^t \quad \forall v \in V, i \in I, j \in J, t \in T$$

$$\forall v \in V, i \in J, j \in K_i^+, t \in T \quad (9)$$

$$Z_{vij}^t \leq 0 \quad \forall v \in V, i \in J, j \in I_v^+, t \in T \quad (10)$$

$$X_{vij}^t \in \{0, 1\}, Z_{vij}^t \geq 0 \quad \forall v \in V, i \in I, j \in J, t \in T$$

$$\forall v \in V, i \in J, j \in I_v^+ \cup K_i^+, t \in T \quad (11)$$

$$Y_{vij}^t \in \{0, 1\} \quad \forall v \in V, i \in I, j \in J, t \in T \quad (12)$$

주어진 수리모형의 목적함수는 총물류비로 구성되어 있으며 이를 최소화하는 것을 목적으로 한다. 제약식 (1)은 각 고객들의 주문이 임의의 트럭으로부터 반드시 충족되어야 한다는 것을 의미한다. 제약식 (2)~(4)는 트럭의 흐름을 유지(flow conservation)하기 위한 조건으로 구성되어 있다. 제약식 (5)는 각 고객은 트럭이 가능한 운송횟수 내에 지정된 생산공장에서 주문을 받아야 한다는 제약이다. 제약식 (6)은 트럭에 제품을 선적할 시 주어진 선적공간의 수를 초과할 수 없다는 제약이며, 제약식 (7)은 X_{vij}^t 와 Y_{vfj}^t 두 결정변수의 관계를 규정한다. 제약식 (8)은 고객들에게 제품을 전달하는 과정에서 제품이 선적된 공간의 수의 변화를 계산한다. 제약식 (9)와 제약식 (10)은 트럭의 선적공간에 대한 제약을 정의하며 나머지 제약식들은 결정변수가 취할 수 있는 값을 제약하고 있다.