

다목적 차량경로문제를 위한 발견적 해법

A Heuristic for Multi-Objective Vehicle Routing Problem

강경환*, 이병기**, 이영훈*

*연세대학교 정보산업공학과 서울시 서대문군 신촌동 134
(optimal@yonsei.ac.kr, youngh@yonsei.ac.kr)

**동부일렉트로닉스 시스템 기술파트 경기도 부천시 원미구 도당동 222-1
(byungki.lee@dsemi.com)

Abstract

This paper is concerned with multi-objective vehicle routing problem(VRP), in which objective of this problem is to minimize the total operating time of vehicles and the total tardiness of customers. A mixed integer programming formulation and a heuristic for practical use are suggested. The heuristic is based on the route-perturbation and route-improvement method(RPRI). Performances are compared with other heuristic appeared in the previous literature using the modified bench-mark data set. It is shown that the suggested heuristic give good solution within a short computation time by computational experiment.

1. 서론

차량경로문제(VRP)의 표준 형태는 Danzig and Ramser(1959)에 의해 최초로 제안되었으며, 최근 연구추세는 고객의 서비스수준 향상과 실제 현실적인 고려요소의 적용차원에서 다음과 같은 몇 가지 형태로 연구되어왔다. 첫 번째는 VRPTW(VRP with time window)로서 고객의 서비스요구 하한시간과 상한시간이 있는 문제로서, Homberger and Gehring(1998)은 진화알고리즘을 적용한 해법을 제시하였으며, Taillard *et al.*(1997), Berger and Barkaoui(2003)은 메타휴리스틱을 이용한 방법을 연구하였다. 시간제약이 있는 외관원문제에 대해 Malandraki and Dial(1996)은 동적계획법에 의한 해법을 제시하였으며, 김병인(2005)은 반복적 삽입 알고리즘을 연구하였다. 두 번째는 복수의 차고지를 가지는 MDVRP(Multi-Depot VRP)로서, Fagerholt(1999), Wu *et al.*(2002) 등에 의해 연구되었다. 차량경로문제의 또 다른 형태는 PDVRP(Pick up and Delivery VRP)로서 고객이 배송 또는 수거를 요구하는 형태이다. Mosheiov(1998)가 경로분할을 이용한 해법을 제시하였다. 이밖에도 차량의 적재용량이 상이한

HVRP(Heterogenous VRP), 고객, 수요, 이동시간이 확률적으로 주어지는SVRP(Stochastic VRP), 차량운용을 위한 계획기간이 다기간으로 확장된 PVRP(Periodic VRP)등이 연구되어 왔다.

대부분의 VRP 연구들이 총 운행거리(시간)를 최소화하는 목적함수를 가지고 있지만, 최근에는 고객이 요구하는 서비스 시간대를 준수하는 것이 중요하게 부각되었다(Malmborg, 1996). 다수의 고객이 요구하는 모든 서비스 시간대를 준수(hard time window)하는 차량 운행계획을 수립하는 것이 현실적으로 어렵기 때문에 고객의 서비스 요구 시간대를 어겼을 경우 벌금비용을 추가하는 모델(soft time window)들이 연구되었다.

본 연구에서 제시하는 다목적 차량경로문제(Multi-objective VRP)는 다음과 같은 제약조건과 목적함수를 가진다. 단일의 중앙차고지에는 적재용량이 동일한 K 개의 차량들이 있으며, 이 차량들은 지리적으로 산재되어 있는 N 개의 고객들의 수요를 충족시킨 후 중앙 차고지로 복귀한다. 각 고객의 수요는 알려져 있으며, 납기요구 시간이 주어지고, 서비스에 필요한 상, 하차 시간이 소요된다. 각 고객은 한대의 차량에 의해서만 서비스를 받는다. 목적함수는 차량의 총 운행시간과 총 납기지연시간(Tardiness)의 합을 최소화 하는 것이다. 차량의 운행시간과 납기지연시간의 가중치는 의사결정자의 판단에 의해 주어진다. 납기지연은 식 (1)과 같다.

$$\text{Tardiness} = \text{Max}(0, \text{arrival time of vehicle} - \text{due time of customer}) \quad (1)$$

제안된 발견적 해법은 RPRI(Route Perturbation and Route Improvement)으로서, 첫 번째 단계는 경로변환(route-perturbation)단계로서, 초기해에 의한 경로상에서, 고객들 중에서 부적합도(unfitness)가 가장 큰 다수개의 후보고객을 선별하고, 선별된 다수개의 후보고객의 입장에서 적합도(fitness)가 가장 좋은 차량으로 각각 이주시키는 단계이다. 이 단계를 통해 2개의 경로에서 고객집합이 수정된다. 즉 이 주전의 경로에서는 선별된 고객이 제외되며, 이주

가 이루어진 경로는 기존의 고객집합에서 선별된 하나의 고객이 추가된다. 두 번째 단계는 경로개선(route-improvement) 단계로서, 이주를 통해 수정된 2개의 경로를 대상으로 지역탐색을 통해 해를 개선한다. 지역탐색 방법으로 경로내 해의 이동방법(intra-route change)중 효과적인 것으로 알려진 2-opt와 Or-opt 방법을 적용하였다. 현재 해에서 생성되는 이웃해의 수는 선별된 후보 고객 수 만큼 존재하며, 이중 가장 좋은 해를 제공하는 이웃해로 해의 이동이 이루어진다. 이 과정은 종료 조건시까지 반복되며 타부 서치를 통해 제어된다.

본 연구가 가지는 차별성은 기존의 경로간 해의 이동방법(inter-route change; relocate, exchange, cross method)이 고객의 특성을 고려하지 않고 이루어진 반면, 제안된 발견적 해법은 각 고객이 가지는 특성, 즉 현재해에서의 납기지연 정도, 고객의 위치, 이웃한 차량의 정보 등을 이용하여 경로변환을 통해 가장 해가 좋아질 가능성이 높은 고객을 가장 적합한 차량으로 이주시킴으로서 좋은 해에 빠른 시간에 수렴하는 장점을 가진다는 것이다. 2장에서는 최적화 수리모형을 제시하고, 3장에서는 발견적 해법, 4장에서는 실험결과를 보이고, 5장에서는 결론을 맺는다.

2. 최적화수리모형

다목적 차량경로문제의 최적화 수리모형은 Malandraki and Daskin(1992), Park(2000)에 의해 제시되었으며, 본 연구에서 제시하는 최적화 수리모형은 새롭게 설계되었다.

결정변수

$x_{k,i,j}$: k차량에 의해 고객 i, j지점사이를 운행하면 1, 그렇지 않으면 0.

$y_{k,i}$: k차량이 i고객지점 도착시의 납기지연시간.

u_i : 서비스순서, 부분경로(sub-tour)방지를 위한 임시 결정변수.

$z_{k,i}$: k차량의 i고객지점 도착시간.

상수변수

N : 고객수.

t_{ij} : 고객 i, j지점사이의 이동시간.

d_i : 고객 i의 수요량.

V : 차량의 적재용량.

L : 차량한대가 서비스할 수 있는 최대 고객수

m_i : 고객 i의 납기요구 시간.

s_i : 고객 i에서의 상, 하차 시간.

w_1 : 가중치

formulation

$$\text{Min } w_1 \left(\sum_{k,i,j} t_{i,j} x_{k,i,j} \right) + (1-w_1) \left(\sum_{k,i} y_{k,i} \right) \quad (2)$$

s.t

$$\sum_{k,i} x_{k,i,j} = 1 \quad \forall j = 2, \dots, N \quad (3)$$

$$\sum_{k,j} x_{k,i,j} = 1 \quad \forall i = 2, \dots, N \quad (4)$$

$$\sum_{j=2} x_{k,1,j} \leq 1 \quad \forall k = 1, \dots, K \quad (4)$$

$$\sum_i d_i \sum_j x_{k,i,j} \leq V \quad \forall k = 1, \dots, K \quad (5)$$

$$\sum_i x_{k,i,p} - \sum_j x_{k,p,j} = 0 \quad \forall p = 1, \dots, N-1, k = 1, \dots, K \quad (6)$$

$$x_{k,i,i} = 0 \quad \forall k = 1, \dots, K \quad i = 1, \dots, N \quad (7)$$

$$y_{k,i} \geq \sum_j x_{k,i,j} (z_{k,j} - m_j) \quad \forall k = 1, \dots, K \quad j = 1, \dots, N \quad (8)$$

$$z_{k,1} = 0 \quad \forall k = 1, \dots, K \quad (9)$$

$$\sum_i x_{k,i,j} (z_{k,i} + s_i + t_{i,j}) = z_{k,j} \quad \forall k = 1, \dots, K \quad j = 2, \dots, N \quad (10)$$

$$u_i - u_j + Lx_{k,i,j} \leq L-1 \quad \forall k = 1, \dots, K \quad i, j = 2, \dots, N (i \neq j) \quad (11)$$

목적함수 (1)은 차량의 총 운행시간과 총 납기지연 시간을 최소화한다. 제약조건 (2), (3)은 각 고객은 1대의 차량에 의해서만 서비스를 받음을 의미한다. 제약조건 (4)는 차고지의 모든 차량에 대해 multi-trip을 허용하지 않음을 의미한다. 즉 각 차량이 차고지에서 출발할 수 있는 횟수는 1보다 작거나 같다. (5)는 차량의 적재용량 제약조건이며 (6)은 차량흐름의 연속성을 의미한다. 제약조건 (7)은 각 차량은 동일 고객 간을 서비스할 수 없게 만든다. 제약조건 (8)은 고객의 납기지연을 구하기 위한 조건이며, 제약조건 (9)는 차고지에서 차량이 출발하는 시간에 대한 초기화조건이다. 제약조건 (10)은 차량이 각 고객에 도착한 시간을 정의한다. 제약조건 (11)은 부분경로 방지를 위한 제약조건으로, Kulkarni and Bhawe (1985)에 의해 제안되었다. 이 제약조건은 기존의 Danzig and Ramser(1959)의 부분경로 방지 제약조건이 수가 고객의 수가 증가함에 따라 지수적으로 증가하는 반면, $O(n^2)$ 의 추가적인 제약조건이 필요하다(Bektas, 2006). 제시된 최적화 수리모형은비선형계획법으로 모델링되었으며, 최적해를 구하기 위해서는 상당한 계산시간을 요구한다. 따라서 3장에서는 짧은 계산시간에 좋은 해를 제공하는 발견적 해법을 제시한다.

3. 발견적 해법(Route Perturbation and Route Improvement)

발견적 해법은 초기해를 구하는 것부터 시작한다. 초기해는 Gillett and Miller(1974)의 스위핑 방법을 기반으로 각 차량에 할당할 고객을 우선 결정한다. 즉 씨앗고객을 임의적으로 선정하고 차량의 적재용량이 허용할 때까지 차고지를 중심으로 시계방향, 혹은 반시계방향으로 고객들을 추가하여 클러스터링을 구성한다. 이와 같은 과정은 차량의 적재용량을 초과하지 않을 때 까지 반복되며, 차량의 적재용량을 초과할 경우 추가적인 차량을 할당한다. 이렇게 구성된 각 차량의 클러스터링에서 경로내 해의 이동방법은 2-opt, Or-opt를 적용하여 초기해를 구한다. 초기해는 다수개의 후보해 중에서 가장우수한 해를 초기해로 선정한다. 이때 차량수를 최소

화 하기위해 식 (12)를 통해 최소차량수를 구하고, 후보해중에서 최소차량수보다 많은 차량을 이용하는 후보해는 제거한다.

$$\text{Min_vehicle} = \text{Min integer} \geq (\text{total customer's demand} / \text{vehicle capacity}) \quad (12)$$

해의 개선은 2단계의 반복적 알고리즘을 통해 이루어지며, 타부 서치를 통해 탐색했던 곳으로 다시 돌아오는 현상을 방지한다. 타부 서치와 같은 메타휴리스틱에서 이웃해의 생성방법은 계산시간과 해의 질에 중요한 영향을 미친다. 따라서 좋은 이웃해를 생성하는 방법에 대해 다양한 연구가 진행되어왔으며, 본 논문에서는 경로변환과 경로개선방법을 통해 좋은 이웃 해를 생성하였다.

3.1 경로변환(Route-Perturbation)

경로변환은 각 차량이 서비스하는 고객의 집합을 바꾸는 과정이다. 이 과정은 각 차량에 의해서 서비스를 받는 고객 중 하나를 선정하여, 다른 차량이 서비스하는 고객의 집합으로 이주시킴으로서 이루어진다. 이때 어떤 고객을 선정하여, 어떤 차량으로 이주시킬 것인가 하는 문제에 대한 어려움이 있다. 본 논문에서는 부적합도란 개념을 도입하여 이주시킬 고객을 선정하고, 어떤 차량으로 이주시킬 것인가 하는 문제는 적합도란 개념을 도입하여 해결하였다. 이와 같은 과정은 각 차량과 고객의 위치정보, 납기지연정보를 이용한다.

Step 1) 부적합도(unfitness) 계산

각 고객의 부적합도는 식 (13)을 통해 구한다. 즉 동일 차량에 의해 서비스를 받는 고객의 집합 중에서 위치적으로 많이 이격되어있고, 납기지연이 큰 고객이 선정된다. 이는 지역적 부적합도(regional unfitness)와 납기지연 부적합도(tardiness unfitness)의 합으로 계산된다.

Notation

θ_{ij} : 차고지를 중심으로 한 고객 i 와 고객 j 의 각도.

If $\theta_{ij} > 180$, $\theta_{ij} = 360 - \theta_{ij}$

$F_{i,k}$: 고객 i 를 서비스하는 차량 k 에 의해 서비스를 받는 고객의 집합

UF_i : 고객 i 의 총 부적합도

UFR_i : 고객 i 의 지역적 부적합도

UFT_i : 고객 i 의 납기지연 부적합도

w_1 : 가중치

$$UF_i = w_1 UFR_i + (1 - w_1) UFT_i \quad \forall i \quad (13)$$

$$UFR_i = \sum_{j \in F_{i,k}} \theta_{ij} \quad \forall i \quad (14)$$

$$UFT_i = \max(0, z_{k,i} - m_i) \quad \forall i \quad (15)$$

식 (14)의 UFR_i 는 고객 i 와 같은 차량내 서비스를 받는 다른 고객간의 차고지를 중심으로 한 각도의 합을 의미한다. UFR_i 가 크다는 것은 해당 차량이 형성하는 경로 상에서 그 고객이 다른 고객들에 비

해 상대적으로 이격되어 있음을 의미한다. 식 (15)는 고객의 납기지연을 계산하기 위한 것이다. UFT_i 가 큰 고객은 다른 차량의 경로 상에 이주시킴으로써 해의 개선할 수 여지가 많다. 이와 같은 과정을 통해 부적합도가 큰 순서대로 총 이웃해 후보수만큼 선정한다.

Step 2) 적합도(fitness) 계산

Step1)에서 이주대상으로 선정된 고객은 적합도가 가장 좋은 차량으로 이주하게 된다. 적합도 계산은 식 (16)을 통해 구한다. 즉 고객의 위치를 기준으로 지역적으로 적합도가 좋은 차량, 각 차량의 총 납기지연이 작은 차량으로 이주하게 된다.

Notation

a_j : 차고지를 중심으로 x 축과 고객 i 각도.

$FT_{i,k}$: 이주고객 i 의 차량 k 에 대한 총 적합도

$FTR_{i,k}$: 이주고객 i 의 차량 k 에 대한 지역적 적합도

FTT_k : 차량 k 의 납기지연 적합도

SF_k : 차량 k 에 의해 서비스를 받는 고객의 집합

c_n : SF_k 에 속한 고객의 수

$$FT_{i,k} = w_1 FTR_{i,k} + (1 - w_1) FTT_k \quad \forall i, k \quad (16)$$

$$FTR_{i,k} = \left| \sum_{j \in SF_k} a_j / c_n - a_i \right| \quad \forall i, k \quad (17)$$

$$FTT_k = w_1 \left(\sum_{k,i,j \in SF_k} x_{k,i,j} \right) + (1 - w_1) \left(\sum_{k,i \in SF_k} y_{k,i} \right) \quad \forall i, j, k \quad (18)$$

식 (17)은 이주될 고객 i 의 입장에서 각 차량의 지역적 적합도를 계산하기 위한 것이다. 고객 i 의 차고지를 중심으로 한 각도와 각 차량이 방문하는 경로상의 다른 고객들의 차고지를 중심으로 한 각도의 평균값을 계산하는 것으로, 그 차이가 가장 작은 차량으로 이주시키는 것이 상대적으로 해의 개선의 여지가 많다. $FT_{i,k}$ 가 작다는 것은 고객 i 와 차량 k 가 구성하는 경로가 지역적으로 가깝다는 것이다. 식 (18)은 각 차량의 현재 목적함수를 계산하기 위한 것으로, FTT_k 가 작다는 것은 추가적으로 다른 고객을 서비스하더라도 여유가 있다는 의미이다. 이와 같은 과정을 통해 이주고객 i 는 총 적합도가 $FT_{i,k}$ 가 가장 작은 차량으로 이주하게 된다.

3.2 경로개선(Route-Improvement)

경로변환을 통해 이주고객이 발생함으로써 각 차량이 서비스하는 고객의 집합이 달라지며, 경로개선은 이주된 고객이 속해있던 기존차량의 경로와 새로 이주한 차량의 경로를 대상으로 이루어진다. 경로개선방법은 2-opt, Or-opt방법을 적용하였다.

3.3 타부서치

경로변환과 경로개선은 종료 조건시까지 반복적으로 이루어진다. 이 과정은 타부 서치를 통해 제어되며, 본 논문에서 사용된 타부 서치는 일반적 타부 서치를 준용하였다. 타부 리스트는 FIFO(First In First Out)에 의해 업데이트되며, 종료조건은 최대 반복횟수로 선정하였다. 타부서치의 과정은 아

래와 같다.

<Tabu search heuristic steps>

- Step 0. Initialization
 - Tabu List size, $\max_iteration(t)$ and $\max_number_non_update(t')$ are specified.
 - Initial_Sol := Solution of sweeping and 2-opt/Or-opt method
 - Current_Sol := Initial_Sol
 - Best_Sol := Initial_Sol
 - Counter (t) := 0 and Counter (t')=0
- Step 1. $t := t + 1, t' := t' + 1$
- Step 2. Generating neighborhood solutions
 - 2.1. Evaluate unfitness of customer
 - 2.2. Select n number of customer by decreasing order of unfitness
 - 2.2. Evaluate fitness of vehicle for the selected customers
 - 2.3. Check tabu list and vehicle capacity
If not in tabu list and not violate vehicle capacity, Transfer the selected customer from existing route to new route.
Otherwise, remove from neighborhood candidate
 - 2.4. Improve the two route such as existing and new route of selected customer by 2opt/Or-opt
 - 2.5 Current_Sol := best neighborhood solution. Go to Step 3.
- Step 3. Tabu List update by FIFO
- Step 4. Evaluation Current_Sol with Best_Sol
 - 4.1. If Current_Sol < Best_Sol,
Best_Sol := Current_Sol and $t'=0$.
go to Step 1.
 - 4.2. Otherwise, go to Step 1.
- Step 5. Stopping criterion
 - 5.1. If $t > \max_iteration$ or $t' > \max_number_non_update$
Stop.

타부리스트는 과거에 이주되었던 고객, 이주전의 차량정보, 이주후의 차량정보의 3가지 정보를 타부리스트 크기만큼 보유한다. 향후 이주하기로 선정된 고객과 선정된 차량정보, 이주할 차량정보를 타부리스트와 비교하여 최종 이주를 결정함으로써 재방문 현상을 방지한다. 타부리스트의 적용은 아래의 절차를 통해 이루어진다.

Notation

- Tabu_list[i].customer : i 번째 타부리스트의 고객
- Tabu_list[i].from_route : i 번째 타부리스트의 고객의 이주전의 차량정보
- Tabu_list[i].to_route : i 번째 타부리스트의 고객의 이주후의 차량정보

Existing route : 이번 iteration에서 선정된 고객이 속한 차량정보

New route : 이번 iteration에서 선정된 고객이 이주할 차량정보

<Tabu list criteria>

- For $i = 1; i \leq \text{Tabu list size}, i++$
 - If (Tabu_list[i].customer = selected customer)
 - If (Tabu_list[i].from_route = existing route) && (Tabu_list[i].from_route = new route)
 - Don't transfer, break;
 - Else if (Tabu_list[i].from_route = new route) && (Tabu_list[i].from_route = existing route)
 - Don't transfer, break;
 - Else if transfer;

4. 실험결과

제안된 발견적 해법의 성능은 기존연구에서 제시된 다양한 발견적 해법의 성능과 비교하였다. 기존연구의 해법은 construction 알고리즘과 improvement 알고리즘의 조합으로 구성하였다. Construction 알고리즘은 Clark and Wright(1964)의 saving method를 적용하여 초기해를 구하였으며, improvement 알고리즘은 경로간 개선방법(inter-route improvement)인 Relocate, Cross, Exchange 방법과 경로내 개선방법(intra-route improvement)인 2-opt, Or-opt 방법을 적용하였다. Improvement 알고리즘은 해의 개선이 더 이상 이루어지지 않을 때까지 반복하게 된다. 이러한 방법들은 Lin(1965), Lin and Kernighan(1973), Timon et al.(2004) 등에 의해 연구되었다. 기존의 해법은 FCSI(First Construction Second Improvement) 알고리즘으로 칭하고, ILOG사의 Dispatcher 3.3 version을 이용하여 모델링하였다.

실험에 사용된 데이터는 Solomon(1988)의 데이터를 수정하여 적용하였다. Solomon의 데이터 중 R101~R110, RC101~RC108의 총 18개 데이터를 이용하였으며, 이 데이터는 차량적용량은200, 각 고객의서비스 타임은 10, 하나의 차고지, 100개의 고객을 가지고 있다. 각 고객의 위치는 x,y 좌표 (0, 99) 사이에 분포되어있다. Solomon의 데이터 중에서 수정된 부분은 가용 차량수는 식 (12)를 이용하여 최소의 차량을 운용하도록 하였다. Solomon데이터는 time-window의 상, 하한인 ready time과 due time을 가지고 있다. 본 논문의 고객 납기요구 시간은 ready time과 due time의 합으로 설정하였다. 타부 서치를 위한 파라미터 세팅은 초기해의 후보 수는 30, 이웃해 후보 수는 30, 타부리스트 사이즈는 3, 종료조건인 최대 반복회수(t)는 1,000, best가 개선되지 않는 최대회수(t')는 100으로 설정하였다. 차량 운행시간최소화 목적의 가중치인 w_1 은 0.9, 0.5, 0.1의 세 가지 case에 대해 실험하였다.

표 1은 실험결과를 보여준다. 거의 대부분의 경우에서 RPRI해법이 FCSI해법보다 우수함을 알 수 있다. 평균적으로 목적함수 측면에서는 4.5%정도의 개선이 있으며, 총 차량운행시간 측면에서는 4.7%,

표 1. RPRI와 FCSI의 성능비교 결과

		w ₁ =0.9	w ₁ =0.5	w ₁ =0.1
MR101	RPRI	(1706, 1856, 359, 643)	(997, 1975, 19, 556)	(204, 2039, 0, 644)
	FCSI	(1761, 1916, 368, 1518)	(1130, 2215, 44, 926)	(214, 2139, 0, 2370)
MR102	RPRI	(1718, 1883, 233, 376)	(983, 1956, 11, 816)	(228, 2054, 25, 185)
	FCSI	(1883, 1987, 944, 132)	(1047, 2078, 17, 1610)	(231, 2268, 4, 2285)
MR103	RPRI	(1694, 1869, 119, 222)	(982, 1921, 43, 1277)	(232, 1989, 36, 351)
	FCSI	(1758, 1918, 323, 842)	(1016, 2031, 0, 803)	(251, 2294, 24, 1515)
MR104	RPRI	(1687, 1860, 126, 251)	(982, 1896, 68, 643)	(206, 1915, 16, 945)
	FCSI	(1746, 1900, 360, 1066)	(1047, 2009, 86, 865)	(219, 2023, 18, 2164)
MR105	RPRI	(1699, 1855, 296, 359)	(981, 1950, 13, 380)	(199, 1963, 3, 481)
	FCSI	(1770, 1935, 286, 1288)	(1102, 2144, 60, 1405)	(224, 2197, 5, 2212)
MR106	RPRI	(1714, 1869, 315, 301)	(976, 1941, 12, 432)	(203, 2032, 0, 166)
	FCSI	(1878, 1987, 899, 193)	(1055, 2095, 15, 1066)	(237, 2318, 6, 1162)
MR107	RPRI	(1698, 1870, 158, 246)	(951, 1888, 14, 1058)	(220, 2140, 6, 942)
	FCSI	(1726, 1896, 195, 658)	(1025, 2024, 26, 1797)	(251, 2198, 35, 1233)
MR108	RPRI	(1682, 1853, 145, 187)	(982, 1896, 68, 644)	(206, 1915, 16, 558)
	FCSI	(1746, 1900, 360, 714)	(1033, 2003, 63, 2079)	(227, 2022, 28, 1114)
MR109	RPRI	(1716, 1862, 401, 428)	(955, 1897, 13, 398)	(204, 2040, 0, 1273)
	FCSI	(1814, 1976, 354, 570)	(1089, 2160, 18, 953)	(208, 2074, 0, 919)
MR110	RPRI	(1696, 1862, 205, 282)	(967, 1928, 6, 289)	(191, 1906, 0, 340)
	FCSI	(1790, 1957, 284, 500)	(1007, 2008, 5, 394)	(198, 1984, 0, 505)
MRC101	RPRI	(1863, 2064, 58, 95)	(1042, 2078, 7, 91)	(213, 2125, 0, 24)
	FCSI	(1893, 2085, 172, 620)	(1105, 2199, 11, 160)	(220, 2196, 0, 188)
MRC102	RPRI	(1855, 2055, 59, 77)	(1036, 2070, 1, 32)	(214, 2106, 4, 168)
	FCSI	(1881, 2070, 177, 322)	(1048, 2093, 4, 297)	(229, 2292, 0, 134)
MRC103	RPRI	(1853, 2051, 76, 48)	(1041, 2077, 4, 24)	(208, 2077, 0, 38)
	FCSI	(1879, 2062, 230, 391)	(1073, 2146, 0, 400)	(219, 2187, 0, 331)
MRC104	RPRI	(1850, 2045, 97, 53)	(1051, 2062, 40, 50)	(272, 2143, 64, 31)
	FCSI	(1904, 2097, 173, 415)	(1049, 2098, 0, 550)	(234, 2202, 16, 502)
MRC105	RPRI	(1856, 2061, 12, 55)	(1028, 2055, 0, 96)	(241, 2181, 26, 62)
	FCSI	(1878, 2067, 180, 348)	(1080, 2157, 2, 333)	(218, 2179, 0, 154)
MRC106	RPRI	(1858, 2060, 40, 81)	(1038, 2076, 0, 38)	(208, 2079, 0, 23)
	FCSI	(1903, 2093, 186, 510)	(1102, 2192, 11, 358)	(224, 2244, 0, 81)
MRC107	RPRI	(1854, 2059, 7, 45)	(1027, 2055, 0, 15)	(208, 2077, 0, 27)
	FCSI	(1867, 2070, 44, 261)	(1094, 2181, 6, 121)	(214, 2135, 0, 83)
MRC108	RPRI	(1841, 2042, 34, 37)	(1022, 2044, 0, 10)	(207, 2065, 0, 20)
	FCSI	(1888, 2091, 55, 295)	(1074, 2147, 0, 90)	(211, 2112, 0, 135)
Average Gap*		(3.4, 2.6, 51.0, 64.4)	(5.9, 5.8, 13.0, 51.8)	(4.2, 5.7, -43.5, 63.2)

주(a, b, c, d) a: 목적함수, b: 총 차량운행시간, c: 총 납기지연, d: 계산시간

Gap*=(FCSI'value - RPRI'value)*100/FCSI'value

총 납기지연에서는 6.8%개선이 있었다. 계산시간측면에서는 59%의 개선을 보여주었다. 그림 1은 w_1 이 0.5일 경우의 목적함수와 총 차량운행시간, 총 납기지연, 계산시간을 나타낸 그래프이다. 목적함수와 총 차량운행시간 측면에서는 모든 예제에서 제안된 RPRI 방법이 기존의 FCSI방법보다 우수한 해를 보여주고 있으며, 총 납기지연 측면에서는 17개 예제의 총 납기지연이 RPRI방법이 320으로서 FCSI방법의 368보다 우수함을 보여주고 있다.

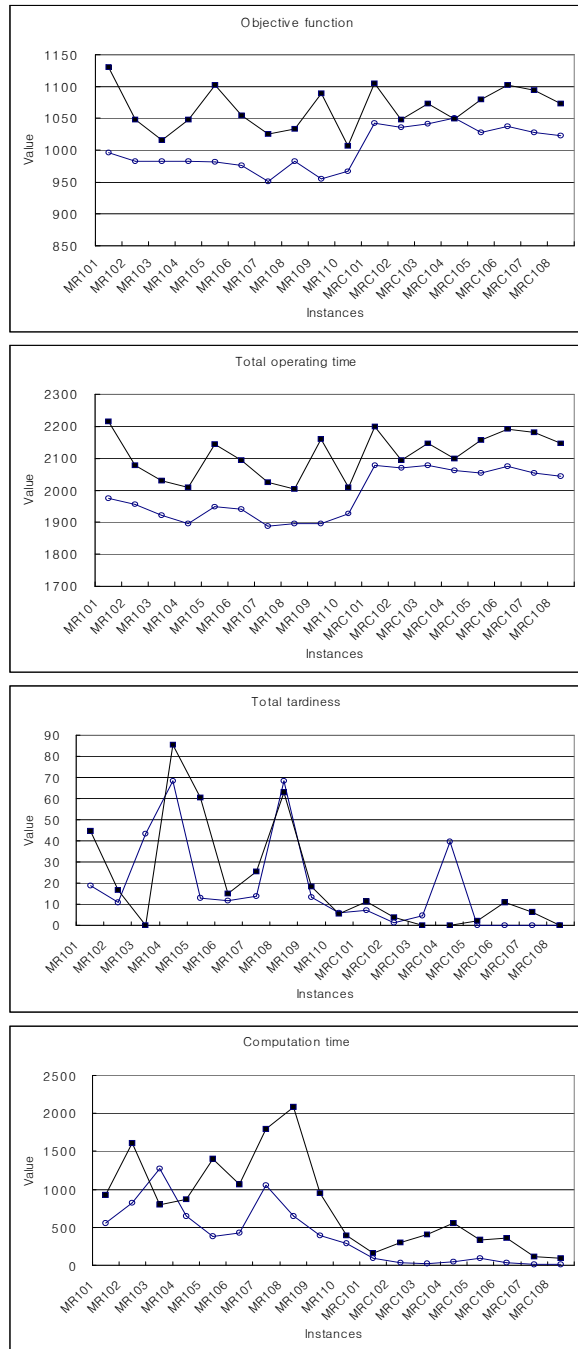


그림 1 RPRI와 FCSI의 성능비교($w_1=0.5$)

○ RPRI ■ FCSI

w_1 이 0.1일 경우, 총 납기지연 측면에서는 FCSI해법이 우수하였으나, 전체 납기지연시간측면에서는 평균적으로 RPRI가 7, FCSI가 5로서 큰 차이가 없으며, 각 예제의 총 납기지연시간이 0인경우가 많음을 고려할 때 RPRI해법의 성능이 나쁘다고 말할 수는 없다.

동일 예제에 대한 실험결과를 보면, 총 차량운행시간 최소화목적의 가중치인 w_1 이 변화함에 따라 총 차량운행시간과 총 납기지연이 trade-off됨을 알 수 있다.

5. 결론

본 논문에서는 총 차량운행시간 최소화과 총 납기지연 최소화라는 두 가지 목적을 함께 추구하는 차량경로문제를 다루었다. 차량 운행비용뿐만 아니라 고객의 서비스만족을 높인다는 측면에서 모델의 의미를 부여할 수 있으며, 이러한 두 가지 목적은 의사결정자의 판단에 따라 가중치를 부여할 수 있다. 제시된 최적화 수리모형은 계산시간의 한계로 큰 문제사이즈의 경우는 최적해를 제공해주지 못하며, 이에 따라 짧은 시간에 FCSI보다 4.5% 개선된 해를 제공하는 발견적 해법을 제시하였다. 제시된 발견적 해법은 경로변환과 경로개선의 단계로 이루어진다. 경로변환시 현재 해에서 의 각 고객이 가지는 특성을 이용하여 경로변환을 함으로서, 해가 좋아질 가능성이 높은 고객을 가장 적합한 차량으로 이주시킴으로서, 좋은 해에 짧은 계산시간에 수렴함을 실험을 통해 입증하였다.

참고문헌

김병인(2005). 시간제약이 있는 외판원 문제를 위한 발견적 해법. 대한산업공학회 2005 춘계학술대회 논문집.

Berger, J. and Barkaoui, M. (2004). A parallel hybrid genetic algorithm for the vehicle routing problem with time windows. Article in press, *Computers & Operations Research*, 31(12), 2037-2053.

Bektas, T.(2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34, 209-219.

Dantzig, G.B. and J.H.Ramser. (1959). The Truck Dispatching Problem. *Management Science*, 6(1), 80-91.

Fagerholt, K. (1999). Optimal fleet design in a ship routing problem. *International Transactions In Operational Research*. 6(5), 453-464.

Gillett, B and L. Miller. (1974). A Heuristic Algorithm for the Vehicle Dispatching Problem. *Operations Research*, 22, 340-349.

Homberger, J and Gehring, H. (2004). A two-phase hybrid metaheuristic for the vehicle routing problem with time windows. *European Journal of Operational Research*, In Press, Corrected Proof, Available online 24 March 2004.

- Kulkarni ,RV. and Bhave, PR.(1985). Integer programming formulations of vehicle routing problems. *European Journal of Operational Research*, 20, 58-67.
- Malandraki, C. and Daskin, M.S.(1992). Time dependent vehicle routing problems: formulations, properties and heuristic algorithms, *Transportation science*, 26(3), 185-200.
- Malandraki, C., and Dial, R. B. (1996). A restricted dynamic programming heuristic algorithm for the traveling salesman problem. *European Journal of Operational Research*, 90(1), 45-55.
- Malmberg, C. (1996). A genetic algorithm for servicelevel based vehicle scheduling, *European Journal of Operational Research*, 93, 121-134.
- Mosheiov, G. (1998). Vehicle Routing with Pick-up and Delivery : Tour-Partitioning Heuristics. *Computers and Industrial Engineering*, 34(3), 669-684.
- Lin, S. (1965). Computer solutions on the traveling salesman problem. *Bell Systems Technical Journal* ,44, 2245-2269.
- Lin, S., Kernighan, B. (1973). An efficient heuristic for the traveling salesman problem. *Operations Research* 21(2), 498-516.
- Park, Y. (2000). A solution of the bicriteria vehicle scheduling problems with time and area-dependent travel speeds. *Computers and Industrial Engineering*, 38, 173-187.
- Solomon, M.M.(1988). Time window constrained routing and scheduling problem: A survey, *Transportation Science*, 22(1), 1-13.
- Taillard, E., Badeau, P., Gendreau, M., Guertin, F. and Potvin, J.-Y. (1997). A parallel tabu search heuristic for the vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies*, 5(2), 109-122.
- Timon C. D., Eldon Y. L. and Defrose C. Dynamic vehicle routing for online B2C delivery. *Omega*, 2005;33(1):33-45.
- Wu, T., Low, C. and Bai, J.(2002), Heuristic solutions to multi-depot location-routing problem, *Computers & Operations Research*, 29, 1393-1415.