

정시배송가능 시간대(time window) 제시를 위한 다중정류소(multi-depot) 다회전(multi-trip) 차량경로문제: 온라인 할당 및 오프라인 재할당 접근법

이기주, 정동진, 채준재[†]
한국항공대학교 항공교통물류학과

Multi-depot Multi-trip Vehicle Routing Problem for Offering Available Delivery Time Windows: Online Assignment and Offline Reassignment Approach

Keyju Lee, Dongjin Jeong, Junjae Chae[†]
School of Air Transport, Transportation and Logistics, Korea Aerospace University

The last mile delivery efficiency is becoming prominent, as its demand is boosting. As the last mile delivery provider and customer are both interested in differentiated services, on-demand delivery and time-reserved delivery services are receiving attention. This research characterizes the problem into multi-depot multi-trip vehicle routing problem for offering available delivery time windows (MD&T VRPOTW). A mathematical formulation for the problem that includes previous schedule constraints and current-time constraints is provided. A strategy for reducing complexity is suggested. The problems were solved using CPLEX, and the complexity reduction was proven effective in online assignments. The experiments on offline re-assignment showed that over 10% of delivery distance (time) can be reduced when the schedules were re-assigned or re-optimized after about 10 delivery scheduled had been accumulated. This research and directed future research should contribute to the improvement of on-demand delivery and time-reserved delivery services.

Keywords: On-demand Delivery, Vehicle Routing, Vehicle Assignment, Vehicle Reassignment

논문접수일 : 2022.10.08.

심사완료일 : 2022.12.19.

게재확정일 : 2022.12.19.

본 연구는 국토교통부/국토교통과학기술진흥원의 지원으로 수행되었음(과제번호 22HCLP-C163182-02)

[†] Corresponding Author: jchae@kau.ac.kr

1. 서론

전세계 도시지역의 주요한 화두인 라스트마일(Boysen et al., 2021)은 물류 및 유통업계에서 상품이 소비자에게 전달되는 마지막 단계를 의미하며, 라스트마일배송의 속도, 품질, 편의성이 소비자에게 차별화된 경험을 제공하기 때문에 그 중요도가 커지고 있다(Korea Maritime Institute, 2022). 특히 코로나 19의 영향으로 전자상거래의 비중이 급속하게 증가하였으며(Bhatti et al., 2020), 이는 라스트마일배송에 대한 수요 증가로 이어졌다. 대한민국에서 코로나 19가 본격 확산된 2020년 택배물량은 전년 대비 약 21%의 급등세를 나타냈다(Park et al., 2021). 대한민국의 전자상거래는 소매시장의 37%를 차지하고 있으며, 음식배달과 즉시배송과 같은 서비스에 힘입어 향후 5년간 45%까지 증가할 것으로 전망된다(Morgan Stanley, 2022).

Dablanc et al. (2017)의 연구에 의하면 도시지역에서의 이동(mobility)환경은 Uber(2020)와 같이 스마트폰 앱을 기반으로 하는 주문형(on-demand) 운송 서비스로 변형되었고, 이러한 혁신은 도시지역의 화물운송 즉, 물류부문까지 재편할 것으로 예상된다. 해당 연구에서는 실시간 배송요청과 라스트마일배송에 많은 잠재적 개선 가능성이 존재함을 드러냈다. 국내에서도 고객이 원하는 시간대(time window)에 예약배송하는 서비스(Korea Logistics News, 2022; SK Telecom, n.d.-a) 또는 즉시배송 서비스(Kyunghyang, 2022; Maeil Business, 2021)와 같이 차별화된 배송서비스에 대한 관심과 수요가 증가하고 있다. 시간대예약배송 서비스와 즉시배송 서비스는 고객이 문을 열어줘야 하는 경우(가전제품 방문설치 및 점검, 가구의 배송과 조립 및 설치, 대형폐가전 방문수거 등) 및 고객과의 대면이 필요한 경우(자동차용품 방문장착 등)와 같이 고객이 배송지에 있을 예정인 상황(attended home delivery)에서 활용될 수 있다. 현재 이러한 서비스 대부분은 방문기사의 전화응대를 통해 계획되고 있으며, 이에 대한 효율적인 배차 솔루션을 제시할 수 있다면 자원관리의 개선과 운영비용 감소 효과를 기대할 수 있다.

본 연구에서는 이러한 문제형태를 정시배송가능 시간대(time window) 제시를 위한 다중정류소(multi-depot) 다회전(multi-trip) 차량경로문제(vehicle routing problem)로 정의한다. 차량경로문제는 거점에서 출발한 다수의 차량이 배송 등의 서비스를 요구하는 고객을 특정한 순서대로 방문하고 다시 거점으로 복귀하는 최소 비용의 경로를 결정하는 문제이다. 본 연구에서 제시하는 차량경로문제 유형은 다수의 거점(multi-depot)에 다수의 차량이 소속되어 있는 상황에서 임의의 위치에서 실시간으로 발생하는 고객의 배송요청에 대응한다. 각 차량의 기확정된 배송스케줄을 고려했을 때 고객위치로 배송 가능한 모든 시간대(time window)를 도출한 뒤 고객에게 제시(offer)하며, 고객은 제시된 시간대 중 하나를 선택할 수 있다(Figure 1). 각 차량은 소속된 정류소에서 출발하여 돌아올 때까지 여러 고객을 서비스할 수 있으나, 차량의 용량제약이 고려되어야 한다. 각 차량은 출차 및 회차를 반복하는 다회전(multi-trip) 배차가 가능하며, 마지막에는 소속된 정류소로 되돌아와야 한다. 본 연구의 수리모델 작성은 온라인(실시간) 할당의 특성인 빠른 응답시간 등을 반영하기 위해 고객의 배송요청과 차량스케줄이 순차적으로 누적되는 상황을 가정하였으며, 기확정된 할당은 유지되도록 제약하였다. 이러한 가정과 제약으로 배송요청이 누적되어감에 따라 계획이 지역최적해를 나타낼 수 있다. 따라서, 주기적 오프라인(비실시간) 재할당을 통해 전역최적해로 수정하기 위한 전략을 함께 제시한다.

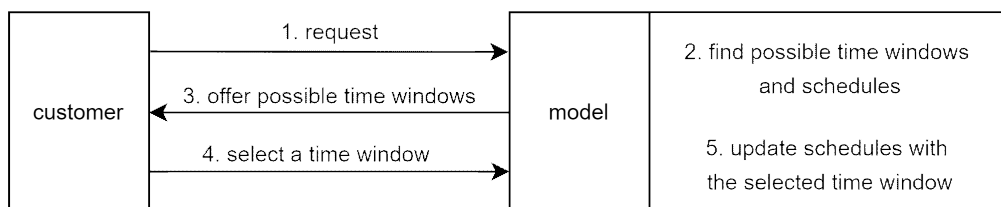


Figure 1. A simplified process of delivery request and offering possible time windows

2. 문헌연구

2.1. 관련 연구

Uber(2020)의 서비스로 대표되는 실시간 차량할당문제 즉, 승차공유(ride-sharing) 또는 차량호출(ride-hailing)에 대한 연구가 활발하게 이루어지고 있다. 해당 문제는 여정-차량(trip-vehicle) 할당문제(S. Li et al., 2020; Mori & Samaranayake, 2021) 또는 실시간배차(online vehicle dispatch)(Zhao et al., 2019) 문제로 정의된다. 여정-차량 할당문제에서의 여정(trip)은 승객이 설정한 기점에서 종점으로의 이동을 의미하며, 해당 여정을 가장 효율적으로 수행할 수 있는 차량에 할당하는 것을 목표로 한다. Alonso-Mora et al.(2017)의 연구에서는 여정-차량 할당문제를 다수 승객의 동승이 가능한 형태로 일반화했고, 실시간으로 큰 규모의 문제 해결을 위한 알고리즘을 제시했다. Zhou & Roncoli(2022)의 연구 또한 큰 규모의 문제에 대한 효율적인 솔루션을 제시했다. 그들의 연구에서는 여정-차량 할당문제를 더욱 현실적으로 발전시켜 도로상의 교통체증과 여정-차량 할당 자체에서 발생하는 내인성(endogenous) 교통체증을 함께 고려할 수 있도록 하였다.

여정-차량 할당문제와 유사한 형태로 음식배달경로(food/meal delivery routing) 문제(Reyes et al., 2018)가 있다. 해당 문제에서는 음식이 준비된 이후에 픽업이 가능하다는 조건이 추가적으로 고려되며, 배달완료까지 소요되는 시간을 제약하거나 초과시간을 최소화하기 위해 노력한다. 음식이 준비되는 시간을 확률적으로 고려한 연구(Ulmer et al., 2021), 장거리 배달 시의 환적(transfer)을 고려한 연구(J. Li et al., 2022) 등 다양한 파생연구가 이루어졌다. 음식배달경로 문제의 최적해를 실시간으로 도출하는 것은 현실적으로 어려우며 빠른 응답시간을 보장하기 위해 구조적(constructive) 알고리즘 연구(Wang et al., 2021), 딥강화학습을 활용한 연구(Jahanshahi et al., 2022) 등을 포함하여 기계학습된 모델을 활용한 솔루션이 제시되었다.

동적(dynamic)차량경로문제(Ojeda Rios et al., 2021; Pillac et al., 2013)는 확률적으로 존재하는 값들이나 시간이 지남에 따라 드러나게 되는 상황에 대응하기 위한 차량경로문제를 포괄하는 개념이며, 특히 최근 지리정보시스템 등의 기술발전으로 인해 주목받고 있다. 동적차량경로문제의 한가지 유형으로, 미리 계획된 차량경로를 실행하는 도중에 실시간으로 발생하는 새로운 주문들을 경로에 포함시키거나 새로운 차량을 할당하고자 하는 문제가 포함된다. 새로운 주문들은 예측된 경우(Predictive)와 예측되지 않은 경우(Unexpected)가 혼합될 수 있으며(Kucharska, 2019), 동적 할당에 의해 발생할 수 있는 비효율은 재최적화(re-optimization)를 과정을 통해 개선할 수 있다(AbdAllah et al., 2017).

시간대(time window)제약 차량경로문제(El-Sherbeny, 2010)에서는 각 고객이 배송을 원하는 시간대가 설정되어 있고(예를 들어, 17시~18시), 다수의 차량이 모든 고객의 배송시간대를 준수하거나, 미준수율을 최소화하여 배송을 완료하고자 한다. 해당 유형의 문제에 대해 정확한(exact) 알고리즘(Baldacci et al., 2012)과 메타-휴리스틱 알고리즘(Dixit et al., 2019) 등 다양한 접근법이 제시되었다. 또한, 시간대제약 차량경로문제와 명칭이 유사하지만 그 내용이 다른 시간대할당 차량경로문제(Spliet & Gabor, 2015)가 있다. 시간대할당 차량경로문제에서의 고객은 매일 또는 반복적으로 방문해야 하는 소매점이며, 고객이 시간대를 설정하는 것과는 반대로 공급자가 배송시간대를 약속하기 위한 문제이다. 시간대할당 차량경로문제는 할당할 고객배송시간을 연속적(continuous)이지 않은 이산(discrete)시간으로 가정한 연구(Spliet & Desaulniers, 2015), 제품군의 의존성을 고려한 연구(Neves-Moreira et al., 2018) 등 소매점 운영의 현실성을 보다 구체적으로 고려할 수 있도록 발전했으며, 정확한 해법으로 분기절단법(branch-and-cut) 알고리즘이 제시되었다(Dalmeijer & Spliet, 2018).

2.2. 차별점 및 연구 기여

앞서 분석한 관련 연구와의 비교를 통해, 아래 Table 1에 나타난 바와 같이 본 연구(정시배송가능 시간대 제시를 위한 다중정류소 다회전 차량경로문제, MD&T VRPOTW)의 차별점을 확인할 수 있다. MD&T VRPOTW에서 고객이 요청하는 것은 여정이 아닌 도달(arrival) 또는 배송(delivery)이며, 실시간 대응이 가능하다. 고객의 위치와 수요량은 미리 알 수 없고, 무작위로 발생하게 된다. 고객의 요청이 최대한 빨리 또는 즉시(on-demand) 충족되어야 하는 문제유형들과는 다르게 고객이 선택한 시간에 맞춰 정시(on-time)배송해야 한다. 만약, 고객에게 제시된 시간대 중 가장 이른 시간이 선택된다면 즉시배송의 형태에도 대응될 수 있다. MD&T VRPOTW는 배송가능 시간대를 고객에게 제시(offer)하기 위한 모델이며, 고객은 배송가능한 것으로 제시된 다수의 시간대 중에서 하나를 선택할 수 있다. 다중정류소가 고려되어 다수의 차량출발 위치가 존재하며, 각 차량은 소속 정류소를 출발하고 복귀하기를 반복하는 다회전 배차가 가능하다.

Table 1. Comparison of MD&T VRPOTW and other related works

| | Trip-vehicle assignment problem (TVAP) | Meal delivery routing problem (MDRP) | Dynamic VRP (DVRP) | VRP with time window (VRPTW) | Time window assignment VRP (TWAVRP) | Multi-depot/trip VRP for offering available delivery time windows (MD&T VRPOTW) |
|---------------------------|----------------------------------------|--------------------------------------|--------------------|------------------------------|-------------------------------------|---------------------------------------------------------------------------------|
| Request | Trip | Trip | Arrival | Arrival | Arrival | Arrival |
| Real-time | O | O | O | X | X | O |
| Unknown customer | O | O | X/O | X | X | O |
| On-demand /On-time | Demand | Demand | Demand | Time | Time | Demand/Time |
| Arrival time decision | Provider | Provider | Provider | Customer | Provider | Customer |
| # of Arrival time options | Single | Single | Single | - | Single | Multiple |
| Vehicle start locations | Multiple | Multiple | Single | Single | Single | Multiple |
| Vehicle return | X | X | O | O | O | O |
| Vehicle capacity | X | O | O | O | O | O |

3. MD&T VRPOTW 문제의 모형화

본 연구에서는 MD&T VRPOTW를 혼합정수계획법으로 구성하였다. 해당 문제는 V 가 노드의 집합, $A = \{(i, j) | i, j \in V, i \neq j\}$ 가 아크의 집합을 의미하는 그래프 $G = (V, A)$ 상에서 정의된다. 이외에 MD&T VRPOTW 문제 모형화를 위해 필요한 표기법(notation)을 아래 Table 2에 제시하였다. 두 노드 (i, j) 사이의 이동비용인 $c_{i,j}$ 는 차량의 이동 소요시간을 의미한다. 노드 집합은 $V = V_c \cup V_{dstart} \cup V_{dend} \cup V_{dcopy}$ 의 합으로 구성되며, 각 집합은 고객 노드 $V_c = \{1, \dots, n\}$, 출발정류소 노드 $V_{dstart} = \{n+1, \dots, n+w\}$, 도착정류소 노드 $V_{dend} = \{n+w+1, \dots, n+2w\}$, 복제정류소 노드 $V_{dcopy} = \{n+2w+1, \dots, n+2w+(n-1)w\}$ 로 표현된다. 해당 모델에서 w 는 다중정류소(multi-depot)의 개수를 의미한다. 출발정류소 노드와 도착정류소 노드는 물리적으로 동일한 지리좌표 상에 존재하지만, 수리모델 구성의 편의성을 위해 구분하였다. 복제정류소 노드 또한 출발 및 도착정류소와 동일한 지리좌표를 가지며, 복제정류소는 차량의 다회전(multi-trip) 배차를 모델링하기 위해 존재한다. 즉, 각 차량의 경로는 반드시 출발정류소 노드를 출발하여 마지막에는 도착정류소 노드에 도달해야하며, 그 사이에 고객노드와 복제정류소 노드를 포함할 수 있도록 구성된다.

각 노드 i 에 대한 수요량을 p_i , 서비스 소요시간을 s_i , 도착해야 하는 시간대의 시작시간(earliest)을 e_i , 도착해야 하는 시간대의 종료시간(latest)을 l_i 로 정의한다. 노드 i 가 고객 노드 집합 V_c 에 포함되지 않는 경우는 모두 정류소를 나타내므로 p_i 와 s_i 값이 모두 0으로 설정된다. 노드 i 가 V_{dstart} 에 포함된다면 e_i 와 l_i 값은 모두 0으로 설정하며, V_{dstart} 및 V_{dcopy} 에 포함된다면 e_i 는 0으로, l_i 는 차량의 업무종료시간으로 설정한다(예를 들어, 분 단위로 표현하되 10시간 근무한다면 600으로 설정). 차량은 각 정류소에 소속되어 있는 차량들의 합집합인 $K = K_1 \cup \dots \cup K_w = \{1, \dots, k\}$ 로 표기하며(K_1 은 1번 정류소에 소속된 차량 집합), 모든 차량은 동일한 용량인 Q 를 가진다.

해당 수리모델의 결정변수로는 $x_{i,j,k}$, $t_{i,k}$, $q_{i,k}$ 세 가지가 있다. 노드 (i, j) 로의 이동을 차량 k 가 수행하는 경우 $x_{i,j,k}$ 의 값은 1이며, 그렇지 않으면 0의 값을 가진다. $t_{i,k}$ 의 값은 노드 i 에 차량 k 가 도달했을 때의 시간을 나타내며, $q_{i,k}$ 는 노드 i 에 차량 k 가 도달했을 때 차량 k 에 적재된 용량을 나타낸다. MD&T VRPOTW 수리모델의 목적함수와 제약조건을 아래와 같이 표현한다.

Table 2. Notations used for modeling MD&T VRPOTW

| Index | Definition |
|-------------------|----------------------------------------------------------------------------------|
| i | index for nodes |
| j | index for nodes |
| w | index for multiple depots |
| k | index for vehicles |
| Set | Definition |
| V | set of all nodes, $V_c \cup V_{dstart} \cup V_{dend} \cup V_{dcopy}$ |
| V_c | set of customer nodes |
| V_{dstart} | set of starting depot nodes, $V_{dstart_1} \cup \dots \cup V_{dstart_w}$ |
| V_{dend} | set of ending depot nodes, $V_{dend_1} \cup \dots \cup V_{dend_w}$ |
| V_{dcopy} | set of copied depot nodes, $V_{dcopy_1} \cup \dots \cup V_{dcopy_w}$ |
| V_{dcopy_w} | set of copied depot nodes at depot w |
| K | set of all vehicles, $K_1 \cup \dots \cup K_w$ |
| K_w | set of vehicles at depot w |
| S' | set of (i, j, k) tuples of $x_{i,j,k} = 1$ in the previous solution (schedule) |
| Parameter | Definition |
| $c_{i,j}$ | cost (time) of traveling from node i to j |
| s_i | service time required at node i |
| p_i | amount of load to be delivered to node i |
| e_i | earliest possible time for arriving at node i |
| l_i | latest possible time for arriving at node i |
| Q | vehicle capacity for the amount of load |
| T | a number to denote the present time |
| M | a very large positive number |
| Decision Variable | Definition |
| $x_{i,j,k}$ | 1 if vehicle k travels from i to j , 0 otherwise |
| $t_{i,k}$ | arrival time of vehicle k at node i |
| $q_{i,k}$ | amount of available space on vehicle k after servicing node i |

$$\text{minimize } Z = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{i,j} \times x_{i,j,k} \quad (1)$$

$$\text{subject to } \sum_{i \in V - V_{dend}} \sum_{k \in K} x_{i,j,k} = 1, \forall j \in V_c \quad (2)$$

$$\sum_{j \in V - V_{dstart}} \sum_{k \in K} x_{i,j,k} = 1, \forall i \in V_c \quad (3)$$

$$\sum_{i \in V} x_{i,j,k} = \sum_{i \in V} x_{j,i,k}, \forall j \in V_c \cup V_{dcopy}, k \in K \quad (4)$$

$$\sum_{j \in V_c \cup V_{dend_w} \cup V_{dcopy_w}} x_{i,j,k} = 1, \forall i \in V_{dstart_w}, k \in K_w \quad (5)$$

$$\sum_{i \in V_c \cup V_{dstart_w} \cup V_{dcopy_w}} x_{i,j,k} = 1, \forall j \in V_{dend_w}, k \in K_w \quad (6)$$

$$x_{i,j,k} = 0, \forall i \in V_{dstart_w}, j \in (V_{dend} - V_{dend_w}) \cup (V_{dcopy} - V_{dcopy_w}), k \in K_w \quad (7)$$

$$x_{i,j,k} = 0, \forall i \in (V_{dstart} - V_{dstart_w}) \cup (V_{dcopy} - V_{dcopy_w}), j \in V_{dend_w}, k \in K_w \quad (8)$$

$$x_{i,j,k} = 0, \forall i \in V_{dstart_w}, j \in V, k \notin K_w \quad (9)$$

$$x_{i,j,k} = 0, \forall i \in V, j \in V_{dend_w}, k \notin K_w \quad (10)$$

$$x_{i,j,k} = 0, \forall i \in V_c, j \in V_{dend_w} \cup V_{dcopy_w}, k \notin K_w \quad (11)$$

$$x_{i,j,k} = 0, \forall i \in V, j \in V_{dstart}, k \in K \quad (12)$$

$$x_{i,j,k} = 0, \forall i \in V_{dend}, j \in V, k \in K \quad (13)$$

$$t_{i,k} + s_i + c_{i,j} \leq t_{j,k} + M \times (1 - x_{i,j,k}), \forall i \in V, j \in V_c \cup V_{dend} \cup V_{dcopy}, k \in K \quad (14)$$

$$t_{i,k} \geq 0, \forall i \in V, k \in K \quad (15)$$

$$t_{i,k} \geq e_i, \forall i \in V, k \in K \quad (16)$$

$$t_{i,k} \leq l_i, \forall i \in V, k \in K \quad (17)$$

$$q_{i,k} \geq 0, \forall i \in V, k \in K \quad (18)$$

$$q_{i,k} \leq Q, \forall i \in V, k \in K \quad (19)$$

$$q_{i,k} + p_j \leq q_{j,k} + M \times (1 - x_{i,j,k}), \forall i \in V, j \in V_c, k \in K \quad (20)$$

$$q_{i,k} + p_j \leq Q + M \times (1 - x_{i,j,k}), \forall i \in V, j \in V_c, k \in K \quad (21)$$

$$q_{i,k} + p_j + M \times (1 - x_{i,j,k}) \geq 0, \forall i \in V, j \in V_c, k \in K \quad (22)$$

$$\sum_{i \in V} x_{i,j,k} = 1, \forall (i,j,k) \in S' \quad (23)$$

$$t_{i,k} + M \times x_{i,j,k} \geq T, \forall (i,j,k) \in S' \quad (24)$$

수식(1)에 나타난 MD&T VRPOTW 수리모델의 목적함수는 각 차량의 운행거리(시간) 합계를 최소화하고자 한다. 제약조건 (2)와 (3)에 의하면 모든 고객 노드에 대해 정확히 한 번의 유입(inflow)과 유출(outflow)이 존재해야 한다. 즉, 모든 고객은 반드시 한 번씩 서비스되어야 한다. 제약조건 (4)는 고객 노드 및 복제정류소 노드에 대해 유입량과 유출량이 일치하도록 한다. 제약조건 (5)에 의해 각 차량은 반드시 자신이 소속된 출발정류소 노드로부터 출발하며, 제약조건 (6)에 의해 각 차량은 반드시 자신이 소속된 도착정류소 노드로 복귀해야 한다. 제약조건 (7)은 특정 출발정류소 노드로부터 출발한 차량이 타

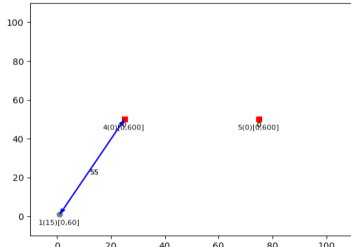
도착정류소 노드 또는 타 복제정류소 노드로 유입되지 않도록, 그리고 제약조건 (8)은 차량이 특정 도착정류소로 유입될 때 타 출발정류소 또는 타 복제정류소 노드로부터 접근한 것이 아니도록 제한한다. 제약조건 (9)는 차량이 소속되지 않은 출발정류소 노드로부터는 출발할 수 없도록, 제약조건 (10)은 차량이 소속되지 않은 거점으로는 복귀할 수 없도록 제한한다. 제약조건 (11)은 고객 노드로부터 도착정류소 노드 또는 복제정류소 노드로 유입될 때 차량의 정류소 소속이 일치해야 함을 나타낸다. 제약조건 (12)와 (13)은 출발정류소 노드로는 어떤 흐름도 유입되지 않도록, 도착정류소 노드로부터는 어떤 흐름도 유출되지 않도록 한다. 제약조건 (14)와 (15)는 서브투어제거(sub-tour elimination) 제약조건이며 Miller-Tucker-Zemlin(MTZ)의 구성방법(Desrochers & Laporte, 1991)을 활용하였다. 제약조건 (16)과 (17)은 고객 노드에 대해 설정된 도착시간대(시간구간)를 준수하도록 한다. 제약조건 (18)과 (19)는 차량이 경로상에서 가질 수 있는 적재량의 최소 및 최대량을 제약한다. 제약조건 (20)~(22)는 차량이 경로를 진행함에 따라 순차적으로 변경되는 적재량의 변화를 업데이트하기 위해 사용되었다. 제약조건 (23)과 (24)에 대해서는 이어지는 하위항목들에서 자세히 설명하도록 한다.

3.1. 기확정 스케줄 제약조건 (constraints for previous schedules)

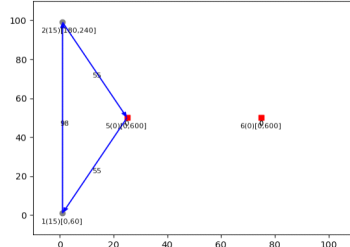
제약조건 (23)은 기확정 스케줄 제약조건을 나타낸다. 본 연구에서 제시하는 MD&T VRPOTW 모델은 실시간으로 들어오는 고객주문에 대한 스케줄링을 진행하며, 기확정된 스케줄을 누적시킨다. 이에 따라 해당 모델은 기확정 스케줄에 대한 제약조건을 포함하고 있다. 먼저, MD&T VRPOTW 모델을 통해 도출되는 차량의 경로 즉, $x_{i,j,k} = 1$ 인 (i,j,k) 조합에 대한 집합을 S 로 정의하고, 바로 직전 고객주문까지 도출된 차량의 경로 $x'_{i,j,k} = 1$ 인 (i,j,k) 조합에 대한 집합 S' 를 정의한다(첫 번째 고객의 주문은 직전 주문이 존재하지 않으므로 $S' = \emptyset$ 즉, 공집합으로 정의).

아래에 제시된 Figure 2(a)~(u)에서는 첫 번째 고객으로부터 스물한 번째 고객에 이르기까지 MD&T VRPOTW 모델에 의해 할당되고 생성된 배송경로를 예시로 나타내고 있다. 해당 예시에서는 두 개의 정류소가 있으며, 각 정류소에는 두 대의 가용차량이 존재하는 상황을 가정하였다. 각 차량의 경로는 색상(blue, green, red, cyan)으로 구분되며, 1회전에 최대 2개 고객을 처리할 수 있는 용량을 가진다. 각 노드의 번호 뒤 소괄호에는 해당 노드에서 소요되는 서비스시간 s_i 를, 대괄호에는 방문시간대인 e_i 와 l_i 를 기입하였다. 참고로 해당 예시에서는 100×100 크기의 좌표평면에 고객노드 위치를 임의로 생성하였으며, 두 정류소의 위치는 (30, 50)과 (70, 50)으로 가정하였다. 노드 간의 이동 소요시간은 좌표평면상의 직선거리를 사용하며, 고객이 선택 가능한 시간대를 0과 600 사이에서 60 길이의 구간으로 임의 설정하였다. 고객노드에서 소요되는 서비스시간은 모두 15로 설정하였다. 보다 구체적인 실험 데이터에 대한 설명은 본문 4.1절에 제시된 내용을 참조함으로써 대체한다.

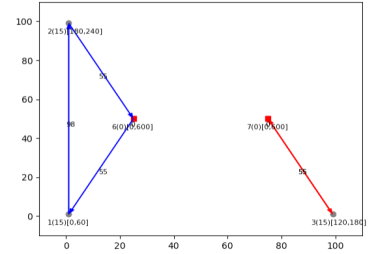
제약조건 (23)에 의하면, 차량에 일단 할당된 고객 노드는 추후 재할당되기 전까지 해당 차량에 할당된 것으로 강제된 상태로 스케줄이 누적된다. 만약 이전 스케줄에서 차량 k 가 고객 j 를 방문하는 것으로 결정되었다면($x_{i,j,k} = 1$), 이후에도 차량 k 는 고객 j 를 방문해야만 한다. 다만, 차량 k 가 고객 j 로 이동하기 위한 직전 노드는 반드시 기존의 i 일 필요는 없다. Figure 2(e)에서 녹색 차량의 경로는 [거점→고객(5)→거점]과 같이 표현될 수 있다. 이후 Figure 2(f)에서의 경로는 [거점→고객(6)→고객(5)→거점]으로 나타난다. 즉, 해당 차량은 기존에 고객(5)를 방문하기로 결정되었으므로 고객(5)를 반드시 방문해야 하지만, 고객(5)를 방문하기 직전 노드는 반드시 기존과 동일한 노드(거점)일 필요는 없다.



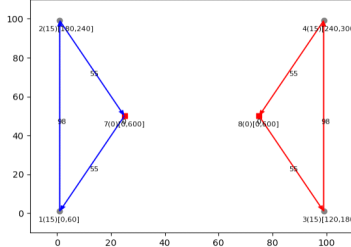
(a) node1 assigned to vehicle1



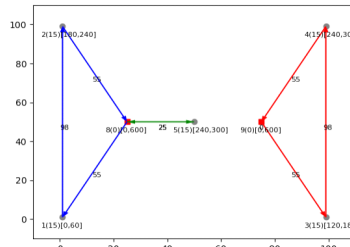
(b) node2 assigned to vehicle1



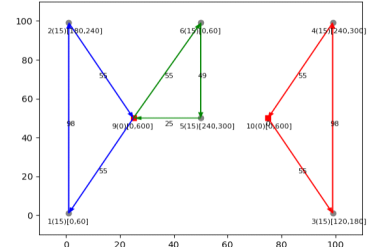
(c) node3 assigned to vehicle3



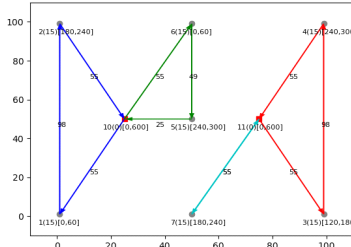
(d) node4 assigned to vehicle3



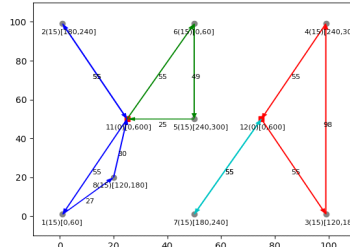
(e) node5 assigned to vehicle2



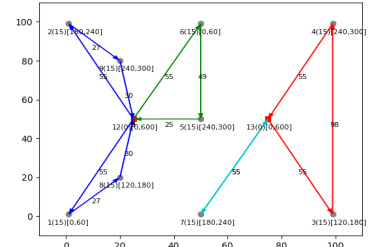
(f) node6 assigned to vehicle2



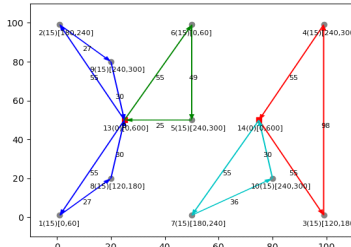
(g) node7 assigned to vehicle4



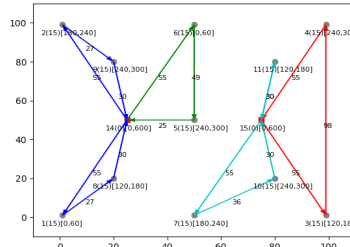
(h) node8 assigned to vehicle1



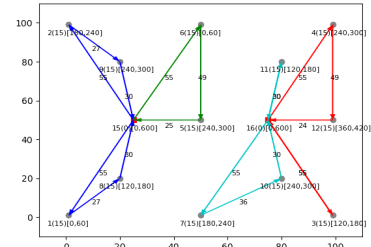
(i) node9 assigned to vehicle1



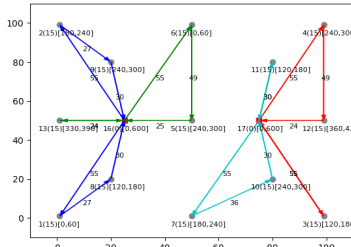
(j) node10 assigned to vehicle4



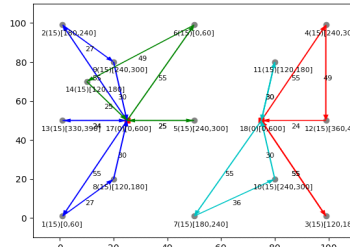
(k) node11 assigned to vehicle4



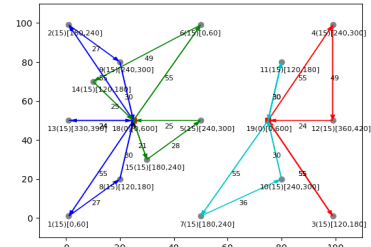
(l) node12 assigned to vehicle3



(m) node13 assigned to vehicle2



(n) node14 assigned to vehicle2



(o) node15 assigned to vehicle2

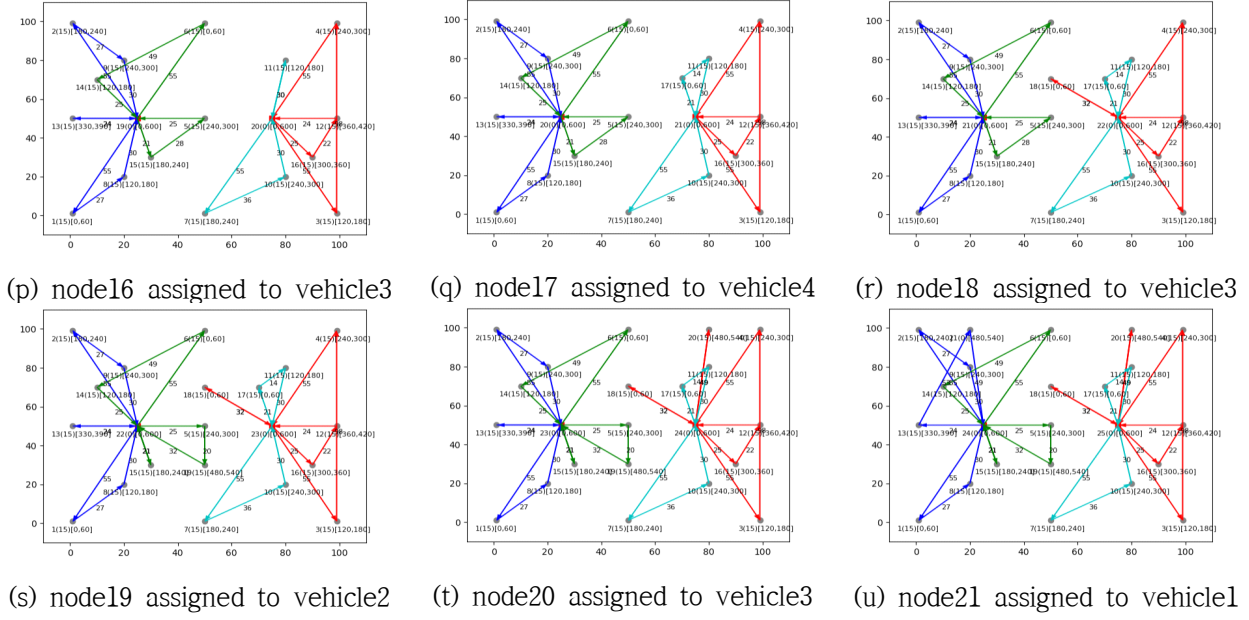


Figure 2. Example results of MD&T VRPOTW model's vehicle assignment with 2-depot, 2-vehicle per depot, and multi-trip assumptions (vehicles are distinguished by colors, service time required for each customer: 15, customer time window length: 60, full time window length: 600)

3.2. 현재시간 고려 제약조건 (constraints for current time consideration)

제약조건 (24)는 현재시간 고려 제약조건을 나타낸다. 현재의 고객주문의 요청에 대해, 시간상으로 현재의 시간을 의미하는 T 보다 과거에 운행되는 것으로 결정된 스케줄은 변경할 수 없다. 즉, 기존에 확정된 노드의 도착시간은 현재시간 이후로만 변경할 수 있도록 제약해야 하며, 이는 제약조건 (24)와 같이 표현할 수 있다. 아래의 Figure 3에서는 현재시간 T 의 값이 다를 때 서로 다른 결과가 도출될 수 있음을 예시하고 있다. Figure 3(a)는 현재 주문 고객(12) 이전까지 누적된 배송경로를 나타내고 있다. 현재 주문 고객(12)는 T 값이 100일 때, 차량(3)에 할당되는 것이 최적으로 나타나며, T 값이 200일 때에는 차량(3)에 할당하기 위한 스케줄을 생성할 수 없어 차량(4)에 할당된다. Figure 3(a)의 기확정 스케줄에서 $x_{3,4,3} = 1$ 이었기 때문에 새로운 스케줄 누적 시 $x_{3,4,3}$ 의 값을 0으로 변경하여 고객(3) 이후에 연결될 노드를 변경하고자 한다면, 제약조건 (24)에 의해 $t_{3,3} \geq T$ 를 만족해야 한다. 이때, 차량(3)이 고객(3)에 도착하는 시간을 나타내는 $t_{3,3}$ 값은 제약조건 (16)과 (17)에 의해 최소 120, 최대 180의 구간으로 제한된다. Figure 3(b)처럼 현재시간 T 값이 100인 경우 $x_{3,4,3}$ 값을 0으로 수정하더라도 $t_{3,3} = 120 \geq T = 100$ 과 같이 조건을 만족할 수 있으나, T 값이 200인 Figure 3(c)에서는 $x_{3,4,3}$ 값을 0으로 수정할 수 없다.

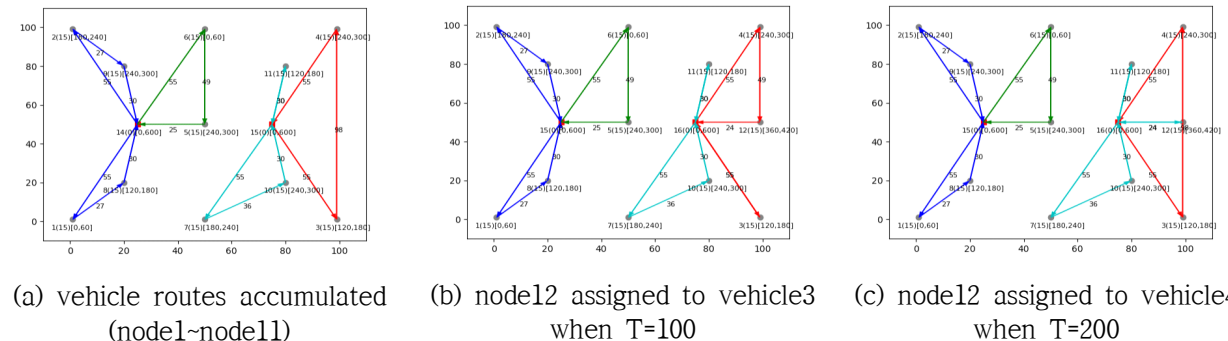


Figure 3. Example results of MD&T VRPOTW model's vehicle assignment with different present time

3.3. 오프라인 재할당

본 연구에서 제시된 MD&T VRPOTW 모델에서는 배송요청을 순차적으로 누적시킴에 따라 배송계획이 지역최적해를 나타낼 수 있기 때문에 주기적인 오프라인(비실시간) 재할당을 적용할 필요가 있다. 오프라인 재할당 결과는 제약조건 (23)을 제거한 MD&T VRPOTW 모델을 풀어냄으로써 제시할 수 있다. 아래의 Figure 4에서는 재할당을 통해 배송계획을 개선할 수 있음을 예시로서 보인다. Figure 4(a)는 재할당 없이 13개의 고객주문을 순차적으로 누적시킨 결과를 나타내며, Figure 4(b)는 재할당 이후의 결과를 나타낸다.

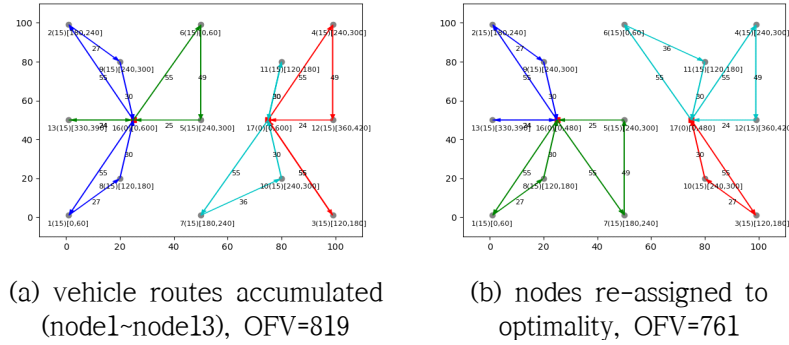


Figure 4. An example result of MD&T VRPOTW model's vehicle re-assignment

제약조건 (23)을 모델에서 제외하는 것은 문제의 복잡성을 증가시킨다. 이로 인해 해를 찾기 위해 소요되는 시간이 증가할 수 있으며, 제한시간 내에 풀어낼 수 있는 문제의 크기가 감소할 수 있다. 이어지는 실험에서는 재할당의 효과와 재할당 문제풀이의 한계 수준에 대한 분석을 포함하고 있다.

4. 실험

앞선 MD&T VRPOTW 수리모형 설명에 따르면, 해당 문제는 $(n-1)w$ 개수만큼의 복제정류소 노드 V_{dcopy} 를 포함하게 된다(n 은 고객 노드의 수, w 는 다중정류소의 수). 예를 들어, 다중정류소가 2개 존재하고, 과거에 10개의 고객주문이 누적된 상태라면, 11번째 주문에 대한 수리모형에서는 20개의 복제정류소 노드를 생성하게 된다. 복제정류소는 차량의 다회전배차(multi-trip)를 모델링하기 위해 필요하지만, 차량경로문제에서 노드의 개수는 직접적으로 차원(dimension) 즉, 복잡성에 영향을 미치기 때문에 불필요하게 복제된 노드가 있다면 문제에서 제외시키는 것이 합리적이다. 본 연구에서는 MD&T VRPOTW 모델에서 복제정류소의 개수를 아래와 같이 축소할 수 있음을 제시한다.

직전 고객의 주문까지 도출된 차량의 경로 S' 에서 차량이 소속된 정류소를 확인함으로써 각 정류소별로 할당된(assigned) 고객주문의 수로 정의된 a_w , 그리고 각 정류소별로 할당 및 연결(assigned and connected)된 고객주문의 수로 정의된 ac_w 를 계산할 수 있다. 아래의 Figure 5에 예시로 나타난 상황에서 정류소(1, 왼쪽)에 할당된 고객 노드의 개수가 5개($a_1 = 5$), 정류소(2, 오른쪽)에 할당된 고객 노드의 개수가 3개($a_2 = 3$), 정류소(1, 왼쪽)에 할당되었으면서 경로가 연속으로 연결된 고객 노드의 개수가 2개($ac_1 = 2$), 정류소(2, 오른쪽)에 할당되었으면서 경로가 연속으로 연결된 고객 노드의 개수가 1개($ac_2 = 1$)로 계산된다.

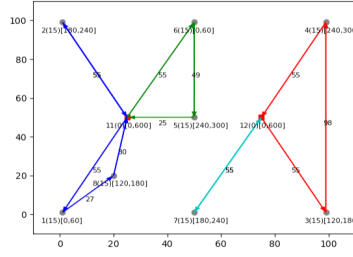


Figure 5. An example state for calculating a_w and ac_w values

Figure 5 상황 직후에 새로 들어온 고객주문에 대한 MD&T VRPOTW 모델에서 필요한 각 정류소 m 에 대한 복제정류소의 개수 $|V_{dcopy_m}|$ 는 아래의 식과 같이 표현할 수 있다.

$$|V_{dcopy_m}| = (n-1) - \left(\sum_{w \in W, w \neq m} a_w \right) - ac_m \quad (25)$$

한 대의 차량이 모든 고객 노드를 서비스하고, 한 번의 회차에 하나의 고객만을 서비스하는 최악의 경우(worst case)를 가정해보자. 이때에는 차량이 최초 출발정류소 노드를 출발하여 마지막 도착정류소에 도달하기 전까지 총 n 번의 회차를 완료하기 위해서 $(n-1)$ 개의 복제정류소를 필요로 한다. 하지만 본 연구에서 제시한 바와 같이 기확정 스케줄 제약조건을 통해 주문을 차량에 강제하게 된다면, $(n-1)$ 개보다 적은 수의 복제정류소만으로도 문제를 해결할 수 있다. 예를 들어, 이미 타 정류소에 기확정된 고객주문의 개수만큼(자신을 제외한 타 정류소에 할당된 고객주문 a_w 의 합) 복제정류소 개수를 줄일 수 있다. 추가적으로, 자신의 정류소에 기확정된 고객주문이라 하더라도 차량의 경로가 연속으로 연결된 경우 즉, ac_m 값만큼의 복제정류소를 생략할 수 있다.

본 연구에서 진행한 실험 중, 주문을 순차적으로 누적시키기 위한 실험(온라인 할당)에서는 수식 (25)에 표현된 방법을 통해 복제정류소의 개수를 감소시키는 전략을 이용하여 문제의 복잡성을 감소시켰으며, 이후 수리모델을 이용한 해를 도출하였다. 반면, 스케줄 재할당과 관련한 실험(오프라인 재할당)에서는 각 정류소별 복제정류소의 개수를 $(n-1)$ 개로 설정하였다. MD&T VRPOTW 모델의 솔루션 도출을 위해 CPLEX Optimization Studio를 활용하였으며, Intel(R) Core(TM) i5-11400 @2.60GHz 중앙처리장치(CPU)와 16GB의 메모리(RAM) 하드웨어가 활용되었다.

4.1. 실험 데이터

실험을 위해 총 30개의 에피소드를 생성하였다. 각 에피소드는 순차적으로 발생하는 고객주문 30개를 포함한다. 각 고객주문은 균일분포 $U \sim (0, 100)$ 에 의한 위도와 경도를 가진다. 고객은 선택 가능한 10개의 시간대에 대해 임의정렬된 선호도 우선순위를 가진다. 선택 가능한 시간대는 0분에서 600분까지의 시간을 60분 단위로 분할한 총 10개의 구간을 말한다. 고객은 MD&T VRPOTW 모델을 통해 제시되는 선택가능시간대 중에서 자신의 우선순위에 따라 첫 번째로 가능(feasible)한 시간대를 선택한다. 고객이 선택한 시간대에 해당하는 스케줄을 기존 스케줄로서 업데이트한다. 공통적으로 모든 고객의 주문량은 1, 각 차량의 용량은 2로 설정하였다. 하나의 정류소에는 두 대의 차량이 소속되어 있으며, 총 두 개의 정류소가 존재한다. 노드 간의 직선거리를 차량의 운행에 소요되는 시간으로 가정하였으며, 현재시간 T 는 0으로 가정하였다.

4.2. 온라인 할당 성능 분석

각 에피소드에 대해 고객주문과 그로 인한 차량스케줄을 누적하면서 MD&T VRPOTW 모델의 온라인 할당 성능에 대한 분석을 실시하였다. Figure 6에서는 요약된 분석결과를 확인할 수 있다. 제한된 시간 (60초) 내에 해결할 수 있었던 문제크기(고객 노드의 수 n)는 평균 16.6, 최대 25, 최소 9개로 나타났다. 해결 가능했던 문제크기의 분산은 7.97로 계산되며, 해결 가능한 문제크기의 95% 신뢰구간이 11.07~22.13으로 나타난다. Figure 6(a)에서는 실제 실험에서 확인된 도수분포와 정규 $N \sim (16.6, 7.97)$ 로 그려낸 분포를 함께 확인할 수 있다. 실시간 응답성을 고려하여 시간제한을 60초 이내로 설정하여 실험을 진행하였으며, 문제 크기에 따른 평균 소요시간을 Figure 6(b)에서 확인할 수 있다. Figure 6(b)의 평균 소요시간은 각 문제크기에 대한 30회의 실험에 대한 평균값이다. 이러한 평균 소요시간은 한 개 시간대에 대한 해를 찾는 데 필요한 시간을 의미하며, 10개 시간대의 실현가능성(feasibility)을 모두 확인하고 차량스케줄을 생성하여 고객에게 제공하기 위해서는 약 10배의 시간이 소요된다.

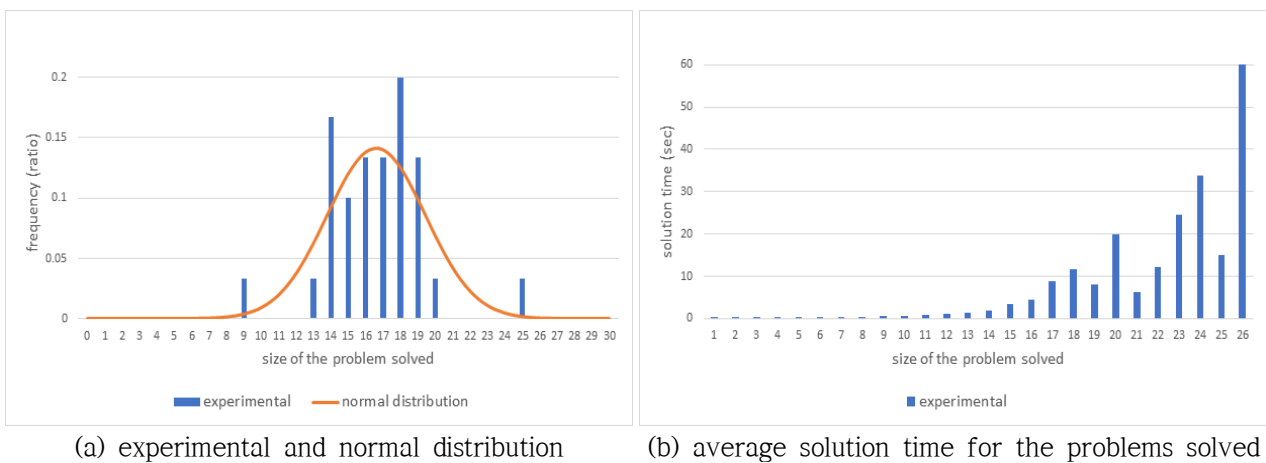


Figure 6. Experimental results of MD&T VRPOTW model's online vehicle assignment

4.3. 오프라인 재할당 성능 분석

오프라인 재할당에서는 제시된 MD&T VRPOTW 모델의 기확정 스케줄 제약조건 (23)을 제거해야 하며, 수식 (25)에 표현된 복제정류소를 감소시키는 전략을 사용하지 않는다. 앞서 생성한 동일한 에피소드를 사용하여 오프라인 재할당 실험을 진행하였으며, 분석결과는 Figure 7에 제시된 바와 같다. 재할당의 경우 제한시간이 600초로 설정되었다. 제한시간 내에 해결할 수 있었던 문제의 크기는 평균 11.5, 최대 14, 최소 6, 분산 3.36으로 확인되었으며, 95% 신뢰구간은 7.91~15.09로 계산되었다. Figure 7(a)에 실험을 통해 확인된 도수분포와 정규 $N \sim (11.5, 3.36)$ 가 함께 제시되었다. Figure 7(b)의 평균 소요시간은 각 문제크기에 대한 30회의 실험에 대한 평균값이다.

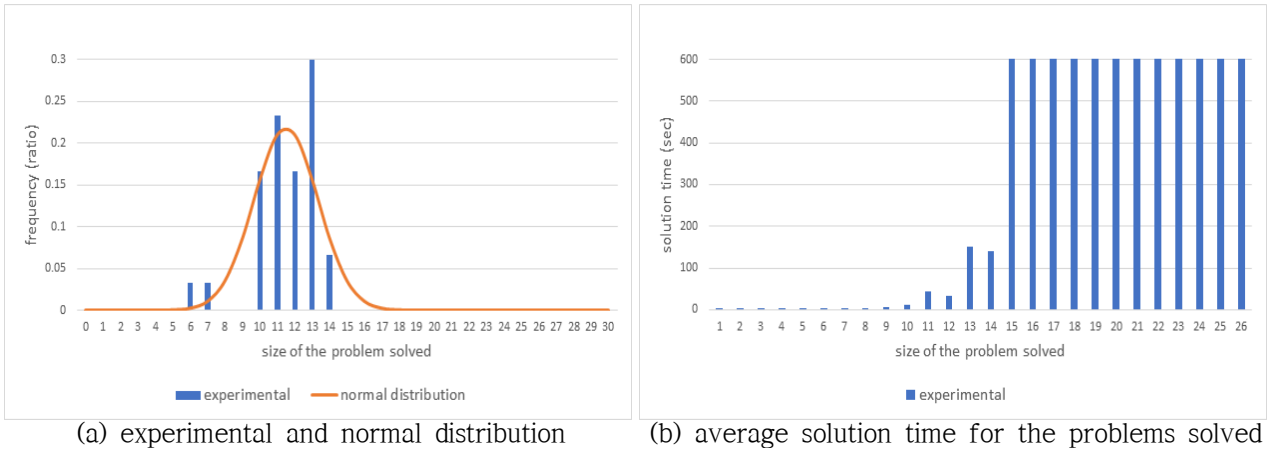


Figure 7. Experimental results of MD&T VRPOTW model's offline vehicle re-assignment

실험결과 분석을 통해 재할당의 효과를 확인할 수 있다. 실시간 할당 시 고객주문이 누적되는 과정에서, 기존 고객주문의 경우 앞서 할당되었던 차량으로 고정시키기 때문에 전체적인 관점에서의 최적상태가 오랜시간 유지되지 못한다. 재할당을 통해 스케줄을 최적의 상태로 개선할 수 있으며, 아래 Figure 8에서 그 효과를 확인할 수 있다. 재할당의 효과는 재할당으로 인한 차량의 운행거리(시간) 감소 비율로 나타낸다. 고객주문의 수가 많이 누적된 상태일수록 재할당 효과가 크며, 누적 고객주문 11개 수준부터는 재할당 효과가 평균 10%를 상회하는 것으로 확인된다. 이러한 결과는 본 실험설계의 가정사항 하에 도출된 것이며, 실제 적용되는 상황에서의 재할당 효과와 차이가 있을 수 있다.

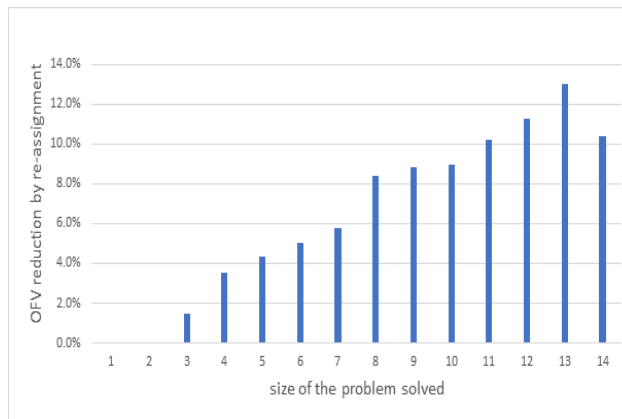


Figure 8. Experimental results on the effect of offline vehicle re-assignment

5. 논의

본 연구에서는 두 지리좌표간의 거리(시간)를 직선거리로 가정하였으며, 온전한($n \times n$) 비용매트릭스가 사전에 준비된 것으로 가정하였다. 그러나 실제 운영단계에서 배송시간을 준수하기 위해서는 이와 같은 가정을 하기 어려운 경우가 많다. 비교적 정확한 지리좌표간 거리(시간) 정보를 확보하기 위해 경로엔진을 활용하는 방안이 있다. 구글지도(Google Maps, n.d.)는 상업용 경로엔진의 예시이며, 국내에도 T-map(SK Telecom, n.d.-b)와 네이버지도(Naver Cloud Plaform, n.d.)와 같은 상업용 경로엔진 공급업체가 존재한다. 상업용 경로엔진은 일부 무료 사용량이 제공되기도 하지만, 거리(시간) 요청 건당 약 5원에서 10원 수준의 사용료가 발생하기 때문에 온전한 비용매트릭스를 모두 채우기에는 비용적인 부담이 크다. 무료로 사용 가능한 오픈소스 경로엔진(GraphHopper, 2022, Openrouteservice, 2022, OSRM, 2021, Valhalla, 2021) 활용을 고려해볼 수도 있으나, 오픈소스 경로엔진이 오픈소스 지도(OpenStreetMap, 2015)에 의존하기 때문에 교통

체증과 같이 시간에 따른 변화를 반영하기 어렵다는 치명적인 단점이 있다. 또한, 새로운 도로교통법규나 새로운 도로네트워크에 대한 정보를 반영하는데 오랜시간이 걸린다는 단점이 있다. 결과적으로, 실제 운영 단계에서 MD&T VRPOTW 배송모델이 적용되기 위해서는 상업용 경로엔진을 활용하되, 그 사용량을 전략적으로 감소시키기 위한 방법을 강구해야 한다.

실험에서 CPLEX를 통해 해결 가능한 MD&T VRPOTW 문제의 크기는 온라인 할당의 경우 평균 16.6개, 오프라인 재할당의 경우 평균 11.5개 수준에 불과한 것으로 확인했다. 현실적인 크기의 문제에 대응할 수 있는 빠르고, 비교적 정확한(exact) 알고리즘에 대한 연구가 요구되며, 훨씬 큰 규모의 문제까지 해결할 수 있는 휴리스틱 알고리즘에 대한 연구가 진행될 필요가 있다. 이를 통해 부수적으로 재할당 효과에 대한 분석을 더욱 현실적이고 정확하게 수행할 수 있을 것으로 기대된다. 따라서, MD&T VRPOTW의 확대 적용을 위한 알고리즘 연구가 필요하며, 연구를 진행함에 있어 앞서 언급한 상업적 경로엔진 사용량의 감소 전략이 함께 고려되어야 한다.

6. 결론

본 연구에서는 실시간 배송요청에 의한 시간대예약배송 서비스 또는 주문형 배송서비스(즉시배송)를 MD&T VRPOTW(정시배송가능 시간대 제시를 위한 다중정류소 다회전 차량경로문제)로 일반화한다. 문헌 연구를 통해 MD&T VRPOTW 문제유형이 기존의 차량-여정 할당문제, 음식배달경로문제, 동적차량경로문제, 시간대제약 차량경로문제, 시간대할당 차량경로문제 등과의 연구 차별점이 존재하는 것을 확인했다. MD&T VRPOTW 문제를 혼합정수계획법의 수리모형으로 구성하였으며, 특히 기확정 스케줄을 고려하기 위한 제약조건과 현재시간을 고려한 실시간성 제약조건을 조망하였다. MD&T VRPOTW 모델에서는 고객의 주문이 순차적으로 누적되어감에 따라 문제의 복잡도가 증가한다. 본 연구에서는 다회전배차(multi-trip)를 모델링하기 위해 필수적이거나 문제의 복잡성 증가에 큰 영향을 미치는 복제정류소의 개수를 축소하기 위한 전략을 함께 제시하였다.

상용 프로그램인 CPLEX를 활용하여 MD&T VRPOTW 수리모형에 대한 실험을 진행하였다. 실험은 온라인 할당(고객주문의 순차적 누적)과 오프라인 재할당(누적된 고객주문 재최적화)으로 구분하여 설계하였다. 온라인 할당 실험은 실시간 응답성을 고려하여 시간제한을 60초 이내로 설정했으며, 오프라인 재할당의 경우 600초로 제한하였다. 본 연구에서 문제의 복잡성을 완화하기 위해 제시한 복제정류소 개수 감소전략과 기확정 스케줄 제약조건은 온라인 할당 과정에 적용할 수 있다. 완화 전략이 적용되지 않은 오프라인 재할당의 결과를 온라인 할당 결과와 비교함으로써 완화 효과를 확인하였다. 오프라인 재할당의 경우 상용 프로그램을 통해 처리 가능한 고객주문의 수가 평균 11.5, 최대 14, 최소 6개로 나타났으며, 온라인 할당의 경우 평균 16.6, 최대 25, 최소 9개로 확인되었다. 실험에서는 온라인 할당 누적 고객주문의 수가 약 10개 이상일 때 오프라인 재할당을 적용하면 약 10% 이상의 차량운행 거리(시간) 감소효과가 나타나는 것으로 확인되었다.

MD&T VRPOTW 문제는 복잡성이 높아 상용 프로그램으로 현실에서 요구되는 크기의 문제를 해결하기에는 한계가 있을 것으로 판단된다. 이에 따라, 정확한(exact) 알고리즘 및 휴리스틱 알고리즘의 설계를 향후 연구 주제로 제시하고자 한다. 실용화를 위한 향후 연구에서는 현실적인 정시배송을 위해 경로엔진의 활용이 고려되어야 한다. 이때, 경로엔진 사용료 및 알고리즘 응답시간과 관련한 적절한 균형점을 제시할 필요가 있다.

배송서비스 차별화를 위한 노력이 새로운 서비스를 만들어내고 있으며, 새로운 서비스에는 많은 개선 가능성이 존재한다. 본 연구와 추가적인 향후 연구는 즉시배송 및 시간예약배송 서비스의 품질을 개선에 기여한다. 고객에게 폭넓은 시간선택 자유도를 제공함으로써 고객 만족도를 높일 수 있을 것으로 기대되며, 기존의 전화응대를 통한 배송(방문)계획 수립을 고객의 직접선택으로 대체함으로써 자원관리 효율성을 제고할 수 있을 것으로 기대된다. 본 연구의 수행범위에서는 이러한 효과를 수치적으로 제시하기 어려웠으나, 향후 구체적인 사례연구를 통해 확인하고자 한다.

참고문헌

- AbdAllah, A. M. F. M., Essam, D. L., & Sarker, R. A. (2017). On solving periodic re-optimization dynamic vehicle routing problems. *Applied Soft Computing Journal*, 55, 1-12. <https://doi.org/10.1016/j.asoc.2017.01.047>
- Alonso-Mora, J., Samaranayake, S., Wallar, A., Frazzoli, E., & Rus, D. (2017). On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proceedings of the National Academy of Sciences of the United States of America*, 114(3), 462-467. <https://doi.org/10.1073/pnas.1611675114>
- Baldacci, R., Mingozzi, A., & Roberti, R. (2012). Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research*, 218(1), 1-6. <https://doi.org/10.1016/j.ejor.2011.07.037>
- Bhatti, A., Akram, H., Basit, H. M., Khan, A. U., Naqvi, S. M. R., & Bilal, M. (2020). Ecommerce Trends During Covid-19. *International Journal of Future Generation Communication and Networking*, June, 1-25. <https://www.researchgate.net/publication/342736799%0Ahttps://www.contactpigeon.com/cp/resources/ebooks/ecommerce-trends-covid19.pdf>
- Boysen, N., Fedtke, S., & Schwerdfeger, S. (2021). Last-mile delivery concepts: a survey from an operational research perspective. *OR Spectrum*, 43(1), 1-58. <https://doi.org/10.1007/s00291-020-00607-8>
- Dablanc, L., Morganti, E., Arvidsson, N., Woxenius, J., Browne, M., & Saidi, N. (2017). The rise of on-demand 'Instant Deliveries' in European cities. *Supply Chain Forum*, 18(4), 203-217. <https://doi.org/10.1080/16258312.2017.1375375>
- Dalmeijer, K., & Spliet, R. (2018). A branch-and-cut algorithm for the Time Window Assignment Vehicle Routing Problem. *Computers and Operations Research*, 89, 140-152. <https://doi.org/10.1016/j.cor.2017.08.015>
- Desrochers, M., & Laporte, G. (1991). Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. *Operations Research Letters*, 10(1), 27-36. [https://doi.org/10.1016/0167-6377\(91\)90083-2](https://doi.org/10.1016/0167-6377(91)90083-2)
- Dixit, A., Mishra, A., & Shukla, A. (2019). Vehicle routing problem with time windows using meta-heuristic algorithms: A survey. *Advances in Intelligent Systems and Computing* (Vol. 741). Springer Singapore. https://doi.org/10.1007/978-981-13-0761-4_52
- El-Sherbeny, N. A. (2010). Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods. *Journal of King Saud University - Science*, 22(3), 123-131. <https://doi.org/10.1016/j.jksus.2010.03.002>
- Google Maps. (n.d.). Distance Matrix API. Retrieved June 9, 2022, from <https://developers.google.com/maps/documentation/distance-matrix/overview>
- GraphHopper. (2022). Graphopper Routing Engine. GraphHopper. <https://github.com/graphhopper>
- Jahanshahi, H., Bozanta, A., Cevik, M., Kavuk, E. M., Tosun, A., Sonuc, S. B., Kosucu, B., & Başar, A. (2022). A deep reinforcement learning approach for the meal delivery problem. *Knowledge-Based Systems*, 243, 108489. <https://doi.org/10.1016/j.knosys.2022.108489>
- Korea Logistics News. (2022). GS Shop, expected arrival date service "Delivery on customer's preferred date." <https://www.klnews.co.kr/news/articleView.html?idxno=304220>

- Korea Maritime Institute. (2022). Korea Maritime Institute International Logistics Weekly (Vol. 625).
- Kucharska, E. (2019). Dynamic Vehicle Routing Problem—Predictive and Unexpected Customer Availability. *Symmetry*. <https://doi.org/10.3390/sym11040546>
- Kyunghyang. (2022). “Eliminate delivery costs and reduce the minimum order amount” Competition for immediate delivery initiative in full swing. <https://m.khan.co.kr/economy/economy-general/article/202208021556011#c2b>
- Li, J., Yang, S., Pan, W., Xu, Z., & Wei, B. (2022). Meal delivery routing optimization with order allocation strategy based on transfer stations for instant logistics services. *IET Intelligent Transport Systems*, April, 1108–1126. <https://doi.org/10.1049/itr2.12206>
- Li, S., Li, M., & Lee, V. C. S. (2020). Trip-Vehicle Assignment Algorithms for Ride-Sharing. *Combinatorial Optimization and Applications*, LNCS 12577, 685–700. https://doi.org/10.1007/978-3-030-64843-5_9
- Maeil Business. (2021). Homeplus Express with “Immediate Delivery in 1 Hour.” <https://www.mk.co.kr/news/business/view/2021/08/749641/>
- Morgan Stanley. (2022). Global Ecommerce Growth Forecast 2022. <https://www.morganstanley.com/ideas/global-ecommerce-growth-forecast-2022>
- Mori, J. C. M., & Samaranayake, S. (2021). On the Request-Trip-Vehicle Assignment Problem. *SIAM Conference on Applied and Computational Discrete Algorithms (ACDA21)*, 693, 228–239. <https://doi.org/10.1137/1.9781611976830.21>
- Naver Cloud Plaform. (n.d.). Maps Application Services - NAVER Cloud Platform. Retrieved June 11, 2020, from <https://www.fin-ncloud.com/product/applicationService/maps>
- Neves-Moreira, F., Pereira da Silva, D., Guimarães, L., Amorim, P., & Almada-Lobo, B. (2018). The time window assignment vehicle routing problem with product dependent deliveries. *Transportation Research Part E: Logistics and Transportation Review*, 116(March), 163–183. <https://doi.org/10.1016/j.tre.2018.03.004>
- Ojeda Rios, B. H., Xavier, E. C., Miyazawa, F. K., Amorim, P., Curcio, E., & Santos, M. J. (2021). Recent dynamic vehicle routing problems: A survey. *Computers and Industrial Engineering*, 160(July). <https://doi.org/10.1016/j.cie.2021.107604>
- Openrouteservice. (2022). Openrouteservice: The open source route planner api. <https://github.com/GIScience/openrouteservice>
- OpenStreetMap. (2015). Setting up a local copy of the OpenStreetMap database. OpenStreetMap. https://wiki.openstreetmap.org/wiki/Setting_up_a_local_copy_of_the_OpenStreetMap_database,_kept_up_to_date_with_minutely_diffs
- OSRM. (2021). Open Source Routing Machine: High performance routing engine run on OpenStreetMap data. <https://github.com/Project-OSRM/osrm-backend>
- Park, H., So-Hyoung, K., Jeong, S.-J., & Sangbeom, S. (2021). The Changes in the Parcel delivery service according to the Spread of COVID-19 in Korea. *Journal of Transport Research*, 28(2), 51–66. <https://doi.org/10.34143/jtr.2021.28.2.51>
- Pillac, V., Gendreau, M., Guéret, C., & Medaglia, A. L. (2013). A review of dynamic vehicle routing problems. *European Journal of Operational Research*, 225(1), 1–11. <https://doi.org/10.1016/j.ejor.2012.08.015>

- Reyes, D., Erera, A. L., Savelsbergh, M. W. P., Sahasrabudhe, S., & O'Neil, R. J. (2018). The Meal Delivery Routing Problem. *Optimization Online*, 1-70.
http://www.optimization-online.org/DB_FILE/2018/04/6571.pdf
- SK Telecom. (n.d.-a). T Direct Shop, arrive at your desired time and place. Retrieved August 31, 2022, from <https://www.tworld.co.kr/poc/html/main/MA.html>
- SK Telecom. (n.d.-b). TMAP API. Retrieved June 11, 2020, from <https://openapi.sk.com/API/detail?svcSeq=4#pay>
- Spliet, R., & Desaulniers, G. (2015). The discrete time window assignment vehicle routing problem. *European Journal of Operational Research*, 244(2), 379-391. <https://doi.org/10.1016/j.ejor.2015.01.020>
- Spliet, R., & Gabor, A. F. (2015). The Time Window Assignment Vehicle Routing Problem. *Transportation Science*, 49(4), 721-731. <https://doi.org/http://dx.doi.org/10.1287/trsc.2013.0510>
- Uber. (2020). Ride with Uber - Request Rides 24/7. Uber. <https://www.uber.com/gb/en/ride/>
- Ulmer, M. W., Thomas, B. W., Campbell, A. M., & Woyak, N. (2021). The restaurant meal delivery problem: Dynamic pickup and delivery with deadlines and random ready times. *Transportation Science*, 55(1), 75-100. <https://doi.org/10.1287/TRSC.2020.1000>
- Valhalla. (2021). Valhalla: Open Source Routing Engine for OpenStreetMap.
<https://github.com/valhalla/valhalla>
- Wang, X., Wang, L., Wang, S., Chen, J. fang, & Wu, C. (2021). An XGBoost-enhanced fast constructive algorithm for food delivery route planning problem. *Computers and Industrial Engineering*, 152(December 2020), 107029. <https://doi.org/10.1016/j.cie.2020.107029>
- Zhao, K., Chen, W., & Lye, K. W. (2019). Online vehicle dispatch: From assignment to scheduling. *IEEE International Conference on Data Mining Workshops, ICDMW*, 2018-Novem, 608-611.
<https://doi.org/10.1109/ICDMW.2018.00094>
- Zhou, Z., & Roncoli, C. (2022). A scalable vehicle assignment and routing strategy for real-time on-demand ridesharing considering endogenous congestion. *Transportation Research Part C: Emerging Technologies*, 139(March). <https://doi.org/10.1016/j.trc.2022.103658>