Transportation Research Part C 152 (2023) 104191

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Dial-a-ride problem with modular platooning and en-route transfers

Zhexi Fu, Joseph Y.J. Chow *

C2SMART University Transportation Center, Department of Civil & Urban Engineering, NYU Tandon School of Engineering, USA

ARTICLE INFO

Keywords:
Modular vehicle
Dial-a-ride problem
Vehicle platooning problem
Variable capacity
Synchronized En-route Transfers
Pickup and delivery problem with transfer

ABSTRACT

Modular vehicles (MV) possess the ability to physically connect/disconnect with each other and travel in platoon with less energy consumption. A fleet of demand-responsive transit vehicles with such technology can serve passengers door to door or have vehicles deviate to platoon with each other to travel at lower cost and allow for en-route passenger transfers before splitting. A mixed integer linear programming (MILP) model is formulated to solve this "modular dial-a-ride problem" (MDARP). A heuristic algorithm based on Steiner-tree-inspired large neighborhood search is developed to solve the MDARP for practical scenarios. A set of small-scale synthetic numerical experiments are tested to evaluate the optimality gap and computation time between exact solutions of the MDARP using commercial software and the proposed heuristic. Large-scale experiments are conducted on the Anaheim network with up to 75 vehicles and 150 requests considering 378 candidate join/split nodes to further explore the potentials and identify the ideal operation scenarios of MVs. The results show that MV technology can save up to 52% in vehicle travel cost, 41% in passenger service time, and 29% in total cost against existing on-demand mobility services in the scenarios tested. Results suggest that MVs best benefit from platooning by serving "enclave pairs" as a hub-and-spoke service.

1. Introduction

Conventional mass transit and demand-responsive mobility systems use vehicles with fixed capacity and cannot adapt effectively to the temporal and spatial demand variations with a satisfactory level of service (Dakic et al., 2021). During peak hours and in areas with a high volume of demand requests, low-capacity vehicles may exacerbate passenger wait time and in-vehicle time due to detours. On the contrary, operating high-capacity vehicles may lead to low vehicle occupancy and unnecessary energy consumption during off-peak hours and in low demand density areas. As for demand-responsive transit (DRT) services like paratransit, their low efficiency in service throughput (e.g, each vehicle only serving several passengers at a time) may cause more traffic congestion problems in the city.

The emerging modular vehicle (MV) technology, such as NEXT Future Transportation (2022), may be able to address the mismatch problems between heterogeneous demand and fixed vehicle capacity mentioned above (Guo et al., 2018; Caros and Chow, 2020; Chen et al., 2019; Chen et al., 2020; Dakic et al., 2021). The MV technology allows vehicles to connect and disconnect with each other in motion so that (1) they can join and travel in a platoon for less energy consumption, and (2) to reposition on-board passengers between vehicles (en-route transfer) at the same time. With these two distinct advantages over such existing microtransit systems as Via and

* Corresponding author.
E-mail address: joseph.chow@nyu.edu (J.Y.J. Chow).

https://doi.org/10.1016/j.trc.2023.104191
Received 22 December 2022; Received in revised form 25 May 2023; Accepted 26 May 2023
Available online 3 June 2023

MOIA, MVs can expand their vehicle capacity according to the demand during peak hours and separate individually to provide on-demand services during off-peak hours or at lower density areas. Guo et al. (2018) showed the value of having such flexibility in adjusting vehicle size, while Caros and Chow (2020) found that it can save on both operation cost and user disutility over current MOD systems in simulation tests using demand data from Dubai, depending on the operational structure. An illustrative diagram is shown in Fig. 1 to demonstrate the operation of MVs.

There are three major challenges to overcome to optimize the demand-responsive routing of these vehicles. First, since MVs may join and leave a platoon at any location and time, tracking the status of each vehicle in both spatial and temporal dimensions is necessary to ensure the synchronization of the docking and undocking process of a platoon. For vehicles that travel in the same platoon, their departure time at the preceding location and arrival time at the following location should be identical. Next, passengers could be relocated and transferred between MVs while they travel in platoons. The second challenge is to search and identify passenger en-route transfers that further improve the total cost. The third challenge comes with the changes in platoon size. A longer platoon with more MVs can be regarded as a single new bus-like platform with expanded capacity. However, if any vehicle leaves the platoon, the on-board carrying capacity of this platoon varies accordingly. Thus, the variable capacity is an important feature and a challenge of MVs.

While Fu and Chow (2022) proposed a method to route vehicles considering synchronized spatial–temporal transfers, it does not handle the benefits and challenges from vehicle platooning. In this study, we propose a mixed integer linear programming (MILP) model for a dial-a-ride problem with modular platooning, or a "modular dial-a-ride problem" (MDARP). The MILP formulation tracks the vehicle platoon status, captures the passenger en-route transfer, and address the variable capacity feature of MV platoons. To solve large-scale MDARPs, we propose a heuristic based on Steiner-tree-inspired large neighborhood search to construct, search, and improve MV platoons from an existing non-platooning routing solution.

2. Literature review

An illustrative example is presented first to demonstrate the differences and benefits in the operation of modular vehicles against existing mobility-on-demand services. Then, we provide a comprehensive literature review on relevant studies in the past. We define the MDARP. Modular vehicles can be physically connected with each other to travel in platoon with zero following gaps, which leads to reduction in air resistance and energy consumption. Platoon savings can be further made under an automated vehicle fleet setting where drivers are not needed for each vehicle (Tirachini and Antoniou, 2020). Meanwhile, passengers with similar destinations can be relocated and transferred between the connected MVs to optimize their delivery paths. Thus, the MDARP is a complex combination and extension of multiple sub-problems, such as the dial-a-ride problem with transfers (DARPT) and the vehicle platooning problem (VPP). To the best of our knowledge, the MDARP is a new problem that has not been studied in the literature.

2.1. Problem illustration

This section demonstrates the problem settings and potential benefits of using MVs through an illustrative example. Two operation policies are considered: (1) a solo (S) mode (the non-transfer non-platooning operation) and (2) a modular (M) mode. The solo mode operates as a conventional DRT service where passengers board a vehicle at a pickup location and alight at a destination location without any transfers or platooning. In this case, each individual vehicle is regarded as an independent unit and the capacity of each vehicle cannot be combined. As for the modular mode, vehicles are allowed to join and connect as a platoon and thus move together to save vehicle travel cost, in terms of fuel savings from less air drag. Moreover, connected MVs allow passengers to transfer en-route, i.e. be relocated between platooned vehicles such that requests with similar destinations can be grouped together for delivery after splitting.

Consider 3 requests (each with a paired pickup and drop-off locations) and 2 vehicles on a 24-node undirected graph (each link can go on both directions) shown in Fig. 2. This is a coarser representation of an infrastructure network, where each node is a pickup or drop-off location, or a candidate platoon join or split location, and links are shortest paths that do not include any of those nodes. The pick-up locations of 3 requests are at nodes {8, 4, 5}, and their corresponding drop-off locations are at nodes {20, 19, 24}. Vehicles are

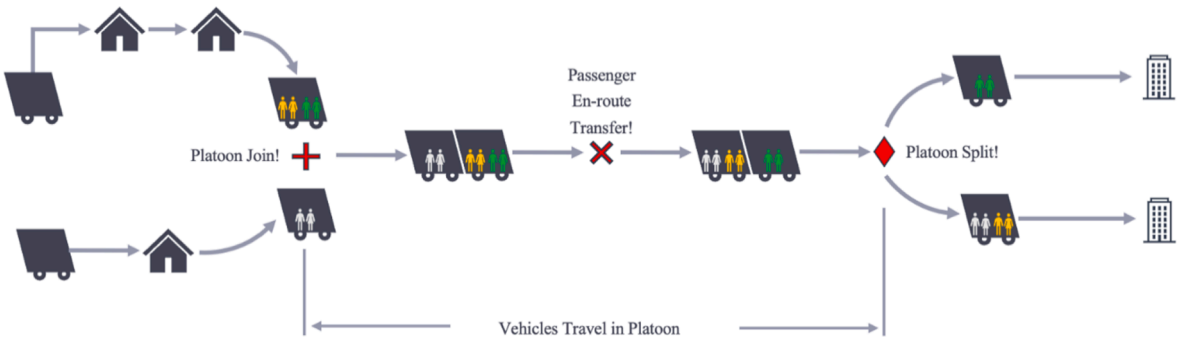


Fig. 1. Modular vehicle diagram: platoon join/split, passenger en-route transfer.

initially located at nodes {1, 6}, without any specified depot. For the simplicity of the problem illustration, each request consists of 2 passengers and each vehicle has a maximum capacity of 4 passengers on-board. We assume that all requests and vehicles are ready at time $T = 0$ and the time used for the platoon join/split operation is ignored (assumed to be zero). Except links $(9, 15)$ and $(10, 16)$, the travel costs on each link are set to 1.

The objective is to minimize the weighted sum of vehicle travel cost and passenger service time (the objective weight of passenger service time relative to vehicle travel cost is set to 1 in this example without any loss of generality). The average vehicle travel cost can be reduced by $\eta = 10\%$ for each additional platooned vehicle along the platooned paths. The passenger service time is measured by the time difference between the passenger drop-off time and the time when travel request is submitted, which includes the passenger wait time and in-vehicle travel time. In addition, since vehicles may arrive at the platoon join/split location at different times (spatial–temporal synchronization), the passenger service time also accounts for this possible platoon join/split delay time.

For the solo mode, the optimal solution can be observed as shown in Fig. 3(a). Vehicle 1 is assigned to serve request 1 only, whereas vehicle 2 is assigned to serve requests 2 and 3. This optimal solution has a total cost of 140.

In the modular mode, the solution is shown in Fig. 3(b). Vehicle 1 is assigned to pick up request 1, and vehicle 2 is assigned to pick up requests 2 and 3, which is the optimal assignment for request pickups. The vehicles travel in platoon through nodes [9, 15, 21]. Vehicle 1 arrives at the platoon join node {9} earlier than vehicle 2, which causes a time delay to customer 1. The platoon join delay time impacts the passengers carried by the vehicle who arrives first. Along the path where vehicles travel in platoon, request 2 is transferred from vehicle 2 to vehicle 1. It is assumed that transfers can be made within the length of a link. After the platoon split, vehicle 1 completes the delivery of requests 1 and 2, whereas vehicle 2 only finishes the delivery of request 3. The results are summarized in Table 1. The total cost equals to 133.8 in the modular mode, which is better than the optimal solution from solo mode. The relative cost difference between solo (S) and modular (M) mode is calculated, resulting in total cost reductions of 4.43% against the solo mode.

2.2. Prior research

There are several studies on the impacts and efficiencies of vehicle platoons. Tsugawa et al. (2016) showed the effectiveness of platooning in energy saving and transportation capacity due to short gaps between vehicles through truck platooning experiments and projects. Nguyen et al. (2019) focused on the ride comfort and travel delay. Sethuraman et al. (2019) studied the bus platooning effects in an urban environment. Since the type of adjacent vehicle influences the energy saving of each vehicle, Lee et al. (2021) focused on the platoon formation strategy with heterogeneous vehicles and found that a bell-shaped platoon pattern is generally effective for energy saving.

The vehicle platooning problem (VPP) has been investigated as a network flow problem where flows may be assigned to seemingly longer paths between origin–destination (OD) pairs because they can platoon with other vehicles flows to increase cost savings. Larsson et al. (2015) developed a MILP formulation for the truck platooning problem and presented several heuristic algorithms for large-scale instances to minimize the total fuel consumption. With the same objective, Luo et al. (2018) proposed a coordinated platooning MILP model that integrates the speed selection and platoon formation/dissolution into the problem formulation. They also proposed a heuristic decomposed approach and tested on a grid network and the Chicago-area highway network. Boysen et al. (2018) investigated an identical-path truck platooning problem to explore various aspects that could impact the efficiency of platoons, such as the platoon formation process, inter-vehicle distance, maximum platoon length and tightness of time windows. Bhoopalam et al. (2018) presents a comprehensive overview on relevant truck platooning studies. However, all these previous studies only focus on the minimization of vehicle energy consumption within the network flow optimization problem. There is no consideration of user costs involving the pickup and delivery of requests as a vehicle routing problem.

The combination of VPP with DARP requires searching for the join/split locations and maximizing the platoon paths and savings,

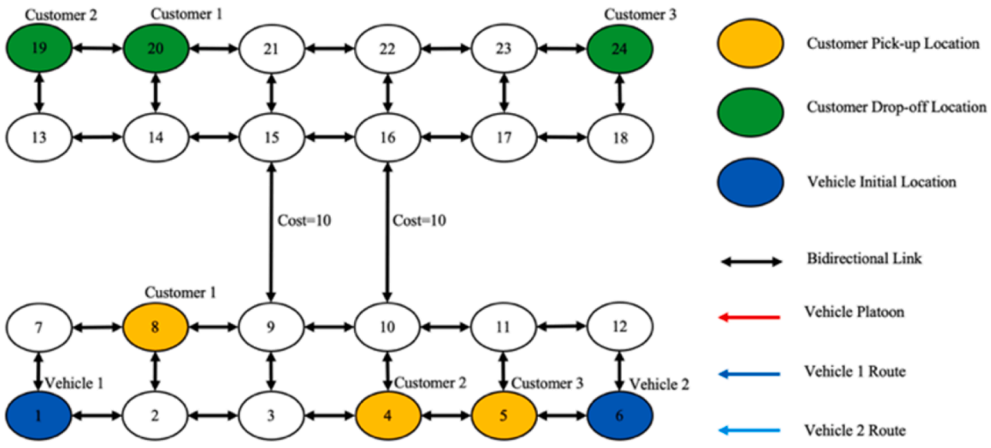


Fig. 2. Illustrative example: initial locations.

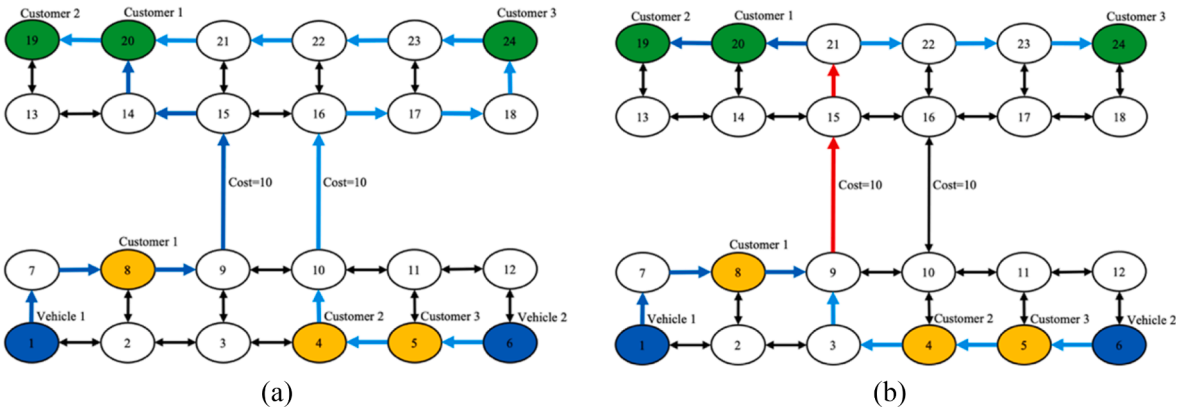


Fig. 3. Routes for (a) S1, (b) S2.

Table 1
Summary of illustrative example results.

Operation Mode	Assignment	Vehicle Travel Cost	Passenger Service Time	Total Cost
Solo (S)	*Veh* 1 → *Req* 1 *Veh* 2 → *Req* 2, 3	36	104	140*
Modular (M)	*Veh* 1 → *Req* 1 *Veh* 2 → *Req* 2, 3 *Pla* over {9, 15, 21}	31.8 (-11.67%)	102 (-1.92%)	133.8* (-4.43%)

Notes: (1) *Veh* for vehicle, *Req* for request, *Pla* for platoon.
(2) * for optimal solution in each operation mode.
(3) percentages shown in () are cost differences against solo mode, calculated as. $\frac{M-S}{S} \times 100\%$

which is a more complex extension of DARP. The presence of platooning means that a complete graph structure (i.e. conventional structure used in vehicle routing problems) would not work because platooning decisions depend on knowing path proximities to each other.

The second major challenge in the modular vehicle routing problem comes with the en-route transfer feature, where passengers could freely move and relocate between connected vehicles for more flexible and optimized routings. MDARP, which includes en-route transfers, can be seen as a variant of the DARPT or, more broadly, pickup and delivery problem with transfers (PDPT). A PDPT makes transfers at inactive stops or hubs, where passengers finish the entire transfer movement at a specific location. Cortés et al. (2010) considered transfers at a pre-specified static location, where each transfer node r is split into two separate nodes, a start node $s(r)$ and finish node $f(r)$. They formulated a link-based MILP model and proposed a branch-and-cut solution method based on Benders Decomposition to solve the problem. Rais et al. (2014) proposed a MILP model for the PDPT problem at candidate transfer locations with and without time windows. However, their model does not explicitly track the vehicle time along its route and thus is not able to optimize the time used for transfer. Fu and Chow (2022) extended the model from Rais et al. (2014) and proposed a set of new constraints that can track the vehicle arrival time at every stop along its path. A vehicle is allowed to wait at transfer locations for passenger transfers within a maximum time limit. Additionally, another recent study by Pierotti and Theresia van Essen (2021) developed two MILP models, one in continuous time and one in discrete time, for the DARPT. However, modeling of en-route transfers *within platoon vehicles in motion* has not been addressed in the literature in the context of DARP.

Third, since MVs can be assembled together in a platoon and treated as an entire new vehicle with expanded capacity, traditional capacity constraints might not be applicable for the operation of MVs. The on-board capacity of a platoon is the sum of all carrying limits of MVs in that platoon. Passengers served by a platoon of MVs belong to the entire platoon rather than individual vehicles. As a consequence, the platoon capacity changes dynamically with the join and split of vehicles, which requires constantly tracking the spatial and temporal status of vehicles.

The literature on this variable capacity feature has been discussed in transit operations. For example, Chen et al. (2019, 2020) proposed both discrete and continuous methods that considers the variable capacity used in the operation of modular vehicles. However, their study only focuses on the joint design of dispatch headway and vehicle capacity on fixed route shuttle systems. Dakic et al. (2021) proposed an optimization model for the flexible bus dispatching system that jointly determines the optimal configuration of modular bus units and their dispatch frequency at each bus line. They use a three-dimensional macroscopic fundamental diagram (3D-MFD) to capture the dynamics of traffic congestion and complex interactions between the modes at the network level, assuming that the MVs impact the congestion level and the study area is homogenous. Tian et al. (2022) studied the optimal planning of public

transit services with modular vehicles by determining the optimal location and capacity of stations where modular units can be assembled and dissembled. They formulate the problem as a mixed-integer nonlinear program (MINLP) and apply surrogate model-based optimization approaches for large-size problems. Similar studies (Dai et al., 2020; Zhang et al., 2020; Pei et al., 2021; Shi and Li, 2021; Wu et al., 2021; Li and Li, 2022) also focus on the operation of modular vehicles for public transit services. Liu et al. (2021) presented a novel operational design to allow MVs to visit customers freely outside of a predetermined order of checkpoints for flex-route transit services. Requests can be picked up and dropped off at either checkpoints or within a zone, and passenger transfers are also allowed at checkpoints in their study. Although they consider dispatching and optimizing the number of MVs from a terminus together to serve dynamic demands, they still treat a platoon of MVs as a group of individual vehicles simply moving together, rather than as an entire platform with expanded capacity as defined in our study (see Section 3.5). Moreover, platooning benefits and join/split synchronizations along the operation were not considered in their study. Additionally, Lin et al. (2022) presented a paradigm for the bi-modality feature of modular vehicles, which integrates transit services with last-mile logistics to serve both passenger and freight with the same fleet.

So far, all relevant studies on the MV technology focus on individual or only combinations of vehicle platooning problem, pickup and delivery problem with transfers, or variable capacity for fixed or flexible public transit services. Moreover, we treat a platoon of connected MVs as an entire vehicle with expanded capacity, which is different from most existing studies on the definition of variable/dynamic capacity. Overall, this study integrates the vehicle platooning problem with request pickup and delivery, considers passenger en-route transfers during vehicle platooning, and addresses the variable capacity feature of MVs for on-demand routing all together.

The key literature of MDARP is summarized in Table 2.

In summary, to fill the research gaps in the concept of MV technology, we propose a MILP model for the MDARP and a heuristic algorithm to solve it. The heuristic builds on the trends in the routing with transfer literature: many rely on neighborhood search heuristics (Mitrović-Minić and Laporte, 2006; Drexl, 2012; Masson et al., 2013; Dumez et al., 2021), similar to other routing problems (e.g. Larsson et al., 2015). The major contributions of our paper are summarized as follows:

(1) We formulate a mathematical model for the dial-a-ride problem that integrates vehicle platooning with request pickup and delivery, considers passenger en-route transfers during vehicle platooning, and addresses the variable capacity feature of platoons at the same time.
(2) A heuristic based on Steiner tree-inspired local neighborhood search for join/split locations is proposed to solve the MDARP for large-scale problems.
(3) We conduct a set of small- and large-scale numerical experiments to validate the feasibility of MVs. Results show that using MVs can save up to 52% in vehicle travel cost, 41% in passenger service time, and 29% in total cost against existing MOD services, depending on the operational setting.

Table 2
Summary of key literature.

Vehicle platooning problem		
Study	***Methodology***	***Major Contributions***
Larsson et al. (2015)	MILP model and two-phase heuristic algorithm	Considered the truck platooning problem with routing, speed-dependent fuel consumption, and platooning formation/split decisions.
Boysen et al. (2018)	MILP model and efficient algorithms	Presented a basic scheduling problem for the platoon formation process along a single path.
Luo et al. (2018)	MILP model and decomposed heuristic approach	Integrated and solved multiple speed selections, scheduling, routing, and platoon formation/dissolution into coordinated platooning problems.
Pickup and delivery problem with transfers		
Study	***Methodology***	***Major Contributions***
Cortés et al. (2010)	MILP model and branch-and-cut solution method based on Benders Decomposition	Formulated the PDPT problem with a static transshipment facility and splitable transfers.
Rais et al. (2014)	MILP model	Formulated the PDPT problem with a number of candidate transfer locations and time window constraints on directed networks.
Fu and Chow (2022)	MILP model and two-phase heuristic algorithm	Proposed a new formulation for the PDPT problem to optimize the temporal and spatial synchronization of transfers.
Modular vehicles		
Study	***Methodology***	***Major Contributions***
Chen et al. (2019)	MILP model and dynamic programming algorithm	Used discrete modeling method to jointly design the dispatch headways and vehicle capacities of MAVs.
Dakic et al. (2021)	Methodological framework	Jointly optimize the configuration of modular bus units and their dispatch frequency at each bus line.
Liu et al. (2021)	MILP model and a two-stage solution framework with dynamic programming and heuristic	Novel operational design to allow MVs to pick up and drop off customers freely outside of a predetermined order of checkpoints for flex-route transit services and passenger transfers are also allowed at checkpoints.
Tian et al. (2022)	MINLP with linearization and surrogate model based optimization approaches	Location and capacity optimization of the docking/undocking stations for new modular-vehicle transit services.

3. Mathematical model

We now present our proposed MILP model for MDARP. Since this study mainly focuses on static problem settings, we assume the information of vehicles and requests are given before departure. The traffic condition on the network such as travel cost and travel time remains unchanged during the operation period. And we do not modify the assignments and routes of vehicles and requests once after departure. However, we have employed a dummy depot design for vehicle destination, as further introduced in Section 3.1, which can be applicable for vehicle relocation and re-optimization of the operation under dynamic problem settings. Additional modifications such as the state of vehicles with on-board passengers should be considered as well. The pre-departure information includes a fleet of vehicles with their initial locations and available time for service, and a set of requests with their pick-up and drop-off locations and request submission time.

The objective is to find the optimal dispatch assignments of vehicles to requests and corresponding routes at minimum total cost. The total cost is measured by the weighted sum of vehicle travel cost and passenger service time. Vehicles are allowed to operate in platoon and thus save the vehicle travel cost from lower air resistance. The passenger service time is calculated by the difference between the passenger drop-off time and the time when travel request is submitted, which might include the wait time, in-vehicle travel time and possible platoon delay time.

The model requires the use of an undirected graph instead of a complete graph structure because of the need to quantify proximities between vehicle paths for platoon join/split synchronization. Using an undirected graph could reduce the number of decision variables. For a network with N nodes, it requires a total number of $C_2^N \times 2$ variables (any pair of 2 nodes on the network and both directions) to represent all platoon join/split possibilities for a complete graph, which is much larger than using the number of arcs for an undirected graph in most cases (e.g., $C_2^{24} \times 2 = 552$ vs 76 arcs for Sioux Falls network, $C_2^{378} \times 2 = 142,506$ vs 796 arcs for Anaheim network). There are different options to pursue in the direction of using an undirected graph. There are multicommodity flow formulations of vehicle routing problems on a directed graph (e.g. Garvin et al., 1957; Letchford and Salazar-González, 2015) including those on a time-expanded network (e.g. Mahmoudi and Zhou, 2016). The latter discretizes time into intervals. Two previous studies (Rais et al., 2014; Fu and Chow, 2022) instead have continuous time arrival but the design on an undirected graph prevents cyclic vehicle routes.

To allow the same vehicle to revisit the same location more than twice, we propose a multi-layer network structure so that we can still track continuous arrival time but also allow cyclic vehicle routes. For example, on the graph shown in Fig. 4(a), suppose there are two customers with pickup and drop-off at (2,4) and (3,5), and a vehicle initially located at node 1. With traditional subtour elimination constraints such as in Rais et al. (2014) and Fu and Chow (2022), the vehicle would not be able to pass twice at node 3, and thus not be able to find a feasible solution. In order to let the vehicle traverse the same location repetitively, we employ a multi-layer network structure by adding duplicate layers of network, as shown in Fig. 4(b). It has the advantage of the time-expanded network without the restriction of discretized time and only requires layers as needed instead of duplicating one layer for every time interval. With this multi-layer structure, an n-layer network would allow a maximum of n revisits at the same location for each vehicle. In other words, for each node, the number of duplicate nodes is the same as the maximum number of revisits allowed. This can be interpreted as a similar but less restrictive form of a time-expanded network.

Here we present more details about the design of multi-layer network with the example. In Fig. 4(b), layer 1 is the original network and includes nodes {1,2,3,4,5}, whereas layer 2 is a duplicate network and includes nodes {1′,2′,3′,4′,5′}. Each set of duplicate nodes (e.g., node 1 and 1′) are connected from the upper layer (layer 1) to the lower layer (layer 2) with zero cost. If there are n layers, each set of duplicate notes would be connected from layer 1 to layer 2 to layer n in sequence. Vehicle starts the service from the top layer (layer 1) and can only move from higher layers to lower layers through the link between each set of duplicate nodes. The duplicate node represents equivalent meanings for pickup and drop-off as the original node on the network. For example, the customer request

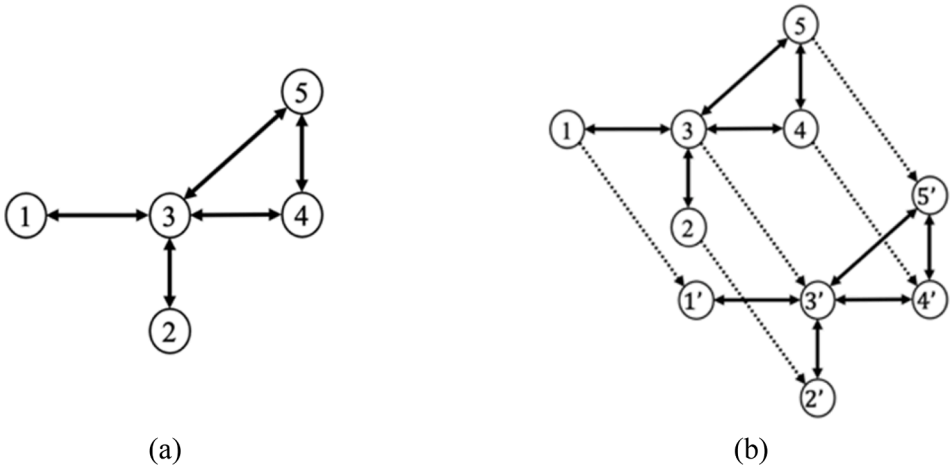


Fig. 4. (a) Original undirected network (b) modified multi-layer network.

$(2, 4)$ can be picked up at either node 2 or node 2′, and dropped off at either node 4 or node 4′, but not at both duplicate nodes at the same time. Thus, with the same example on our modified multi-layer network structure, the optimal vehicle route in terms of node number becomes $1 \rightarrow 3 \rightarrow 2 \rightarrow 2' \rightarrow 3' \rightarrow 4' \rightarrow 5'$, which achieves the multi-visit feature with the same optimal cost. A vehicle picks up requests at node 3 and 2, moves from layer 1 to layer 2 through the link from node 2 to node 2′, then traverses the node 3′ to drop off requests at node 4′ and 5′. This way, the vehicle path can be explicitly shown in terms of unique nodes and links, and allows us to track the vehicle arrivals in continuous time (in Section 3.2) for platoon join/split synchronization later.

3.1 Basic MILP formulation

The MDARP is defined on an undirected graph $G(N, A)$ over an operational time horizon $[0, T]$. N is a set of nodes, consisting of vehicle initial and destination locations, request pickup and drop-off locations, and platoon join/split locations. A is the set of undirected arcs, where A_i^+ and A_i^- represent the set of inbound and outbound arcs at node i, respectively. The notation is shown in Table 3.

Let K be the set of vehicles available to serve customers at time $t_k^V \in [0, T]$. For a vehicle $k \in K$, we use s_k and e_k to denote its starting and ending locations. The vehicle ending location is assigned to a dummy depot, which can be considered as the designated zone for idle vehicles and initial location for future services. Under dynamic and online settings, vehicles ended up at the dummy depot is an indicator of availability for service and then can be used in future implementations for serving new requests. This way, the model could be applied for dynamic and online scenarios with some modifications to track state variables and used together with other algorithms to decompose large problem instances into a few sub-problems suitable for the model (see Berbeglia et al., 2010; Ma et al., 2019; Luo et al., 2018). We assume by default that every node on the network is directly connected to e_k with zero travel cost. If a vehicle is not assigned with any requests, it would directly move to the dummy depot with no cost and be ready for future requests. In practice, we could have more than one dummy depot on the graph to represent different major stations, hubs and even electric vehicle charging facilities with different relocation costs so that the model can also be integrated to consider vehicle relocation problem. For simplicity, e_k is left out and not shown on the graph. In addition, reoptimization would have to consider passengers on-board the vehicle as part of its set of state variables. Overall, this study only focuses on static problem settings, but the formulation of model could be helpful for the future research in dynamic and online applications of MVs.

Let R be the set of customer pick-up and drop-off requests. For a customer request $r \in R$, q_r denotes the number of passengers for the request, o_{rl} denotes the pick-up location on network layer $l \in L$, and d_{rl} denotes the corresponding drop-off location on network layer $l \in L$. The basic MILP model is shown in constraints (1) – (8).

Table 3
Model notations.

Notations	Definitions
Parameters	
β	objective weight for passenger service time relative to vehicle travel cost
d_{ij}	travel distance for arc $ij \in A$
τ_{ij}	travel time for arc $ij \in A$
L	number of network layers
K	set of vehicles ready for service
s_k	starting location of vehicle $k \in K$
e_k	ending location of vehicle $k \in K$
c_k	capacity of vehicle $k \in K$
t_k^V	ready time for service of vehicle $k \in K$
R	set of customer requests
o_{rl}	pick-up location of request $r \in R$ on layer $l \in L$
d_{rl}	drop-off location of request $r \in R$ on layer $l \in L$
q_r	number of passengers in request $r \in R$
t_r^Q	request submission time of request $r \in R$
u	maximum platoon length
η	platoon cost saving rate
M	large positive constant
Decision variables	
X_{ijk}	1 if vehicle k traverses arc ij, and 0 otherwise
Y_{ijkr}	1 if vehicle k carries request r onboard and traverses arc ij, and 0 otherwise
S_{kri}	1 if vehicle k is assigned to pick up and/or drop off request r at node i, and 0 otherwise
T_{ik}^V	time at which vehicle k arrives at node i
T_{ir}^Q	time at which request r arrives at node i
T_{ik}^U	dwell time of vehicle k at node i
P_{ijkm}	1 if vehicle k travels in platoon with vehicle m on arc ij, and 0 otherwise
P_{ijk}^N	number of vehicles that travel in platoon with vehicle k on arc ij
Dummy variables	
Z_{ijk}	non-negative dummy variable for ensuring the continuity of T_{ik}^V
U_{ijk}	non-negative dummy variable for ensuring the continuity of T_{ik}^U
W_{kri}	non-negative dummy variable for measuring the pick-up and/or drop-off time of S_{kri}
C_{ijkm}	non-negative dummy variable that captures the total number of passengers carried by vehicle m when vehicle k and m travel in platoon on arc ij

$$Min: \sum_{k\in K}\sum_{(i,j)\in A} d_{ij}(X_{ijk} - \eta P^N_{ijk}) + \beta \sum_{r\in R} q_r \left(\sum_{l\in L} T^Q_{d_{rl}r} - t^Q_r \right) \tag{1}$$

Subject to

$$\sum_{(i,j)\in A_i^-} X_{ijk} = 1, \forall k \in K, i = s_k \tag{2}$$

$$\sum_{(j,i)\in A_i^+} X_{jik} = 1, \forall k \in K, i = e_k \tag{3}$$

$$\sum_{(i,j)\in A_i^-} X_{ijk} - \sum_{(j,i)\in A_i^+} X_{jik} = 0, \forall k \in K, \forall i \in N \backslash \{s_k, e_k\} \tag{4}$$

$$\sum_{l\in L}\sum_{k\in K}\sum_{(i,j)\in A_i^-} Y_{ijkr} - \sum_{l\in L}\sum_{k\in K}\sum_{(j,i)\in A_i^+} Y_{jikr} = 1, \forall r \in R, i = o_{rl} \tag{5}$$

$$\sum_{l\in L}\sum_{k\in K}\sum_{(j,i)\in A_i^+} Y_{jikr} - \sum_{l\in L}\sum_{k\in K}\sum_{(i,j)\in A_i^-} Y_{ijkr} = 1, \forall r \in R, i = d_{rl} \tag{6}$$

$$\sum_{k\in K}\sum_{(i,j)\in A_i^-} Y_{ijkr} - \sum_{k\in K}\sum_{(j,i)\in A_i^+} Y_{jikr} = 0, \forall r \in R, \forall i \in N \backslash \{o_{rl}, d_{rl}\} \tag{7}$$

$$\sum_{k\in K} Y_{ijkr} \le 1, \forall r \in R, \forall (i,j) \in A \tag{8}$$

$$X_{ijk} \in \{0,1\}, \forall k \in K, \forall (i,j) \in A \tag{9}$$

$$Y_{ijkr} \in \{0,1\}, \forall k \in K, \forall r \in R, \forall (i,j) \in A \tag{10}$$

$$S_{kri} \in \{0,1\}, \forall k \in K, \forall r \in R, \forall i \in \{o_{rl}, d_{rl}\} \tag{11}$$

$$T^V_{ik} \ge 0, \forall k \in K, \forall i \in N \tag{12}$$

$$T^Q_{ir} \ge 0, \forall r \in R, \forall i \in N \tag{13}$$

$$T^U_{ik} \ge 0, \forall k \in K, \forall i \in N \tag{14}$$

$$P_{ijkm} \in \{0,1\}, \forall k, m \in K, k \ne m, \forall (i,j) \in A \tag{15}$$

$$P^N_{ijk} \in [0, u-1], \forall k \in K, \forall (i,j) \in A \tag{16}$$

$$Z_{ijk} \ge 0, \forall k \in K, \forall (i,j) \in A \tag{17}$$

$$U_{ijk} \ge 0, \forall k \in K, \forall (i,j) \in A \tag{18}$$

$$W_{kri} \ge 0, \forall k \in K, \forall r \in R, \forall i \in \{o_{rl}, d_{rl}\} \tag{19}$$

$$C_{ijkm} \ge 0, \forall k, m \in K, k \ne m, \forall (i,j) \in A \tag{20}$$

Objective function (1) minimizes the total of vehicle travel cost and passenger service time, weighted by a constant parameter β in general form. The exact amount of weight β needs to be calibrated to real data if implementing empirically and it could vary from scenario to scenario, depending on the operator's objective, elasticity of demand, operation policy, and other factors. The cost savings from vehicle platooning are deducted by a constant rate of η from the vehicle travel cost, as shown in the first term of objective function. We assume that the average travel cost of each vehicle linearly decreases as the increase of platoon length (further discussed in Section 3.6). The second term of the objective function measures the passenger service time, which is calculated by the difference between the passenger drop-off time and the time when the travel request is submitted. Thus, the passenger service time might include the wait time before pickup, in-vehicle travel time and possible platoon delay time. Constraints (2) and (3) ensure that each vehicle leaves its initial location and ends its trip at the ending location. If a vehicle is not used for service, it will be assigned to the ending location without any cost. Constraints (4) maintain the vehicle flow conservation at any node except its initial and ending locations. Constraints (5) and (6) ensure that each request is picked up and dropped off by exactly one vehicle on only one level of the multi-layer network. We assume all vehicle initial locations are placed on the top network layer and then vehicles traverse the network to serve

requests. However, requests might be picked up and dropped off by different vehicles on different network layers, which is the reason that constraints (5) and (6) consider over all vehicles and network layers. Constraints (7) maintain the passenger flow conservation at any node except its pick-up and drop-off locations. Constraints (8) enforce each request to be served by only one vehicle at a time. Integral and non-negativity constraints are defined in Eqs. (9) to (20).

3.2 Continuity of vehicle arrival time constraints

To ensure the spatial–temporal synchronization between vehicles and support the optimization of passenger service time, it is necessary to track the exact travel time of each vehicle along the path. Thus, we adopt the constraints (21) – (22) as proposed and proven in Fu and Chow (2022) to ensure the continuity of vehicle arrival time. For each vehicle $k \in K$, constraints (21) ensure that vehicles can only start the service at its earliest available time $t_k^V \in [0, T]$ from the initial location s_k. As for Eq. (22), we track the exact arrival time along the path where the vehicle traverses on the undirected graph. Here X_{ijk} and X_{jik} cannot equal to 1 at the same time. Otherwise, either node i or node j would be visited twice by the same vehicle, which is infeasible. This type of subtour is eliminated by Eq. (29) in this section. If both X_{ijk} and X_{jik} equal to 0, constraints (22) would be ignored. If either X_{ijk} or X_{jik} equals to 1, for example $X_{ijk} = 1$ and $X_{jik} = 0$ here, Eq. (22) become $T_{ik}^V + \tau_{ij} + T_{ik}^U = T_{jk}^V$, where the arrival time at node j exactly equals to the sum of the arrival time at node i, dwell time at node i, and the travel time through arc ij.

$$T_{ik}^V \geq t_k^V, \forall k \in K, i = s_k \tag{21}$$

$$\left(X_{ijk} + X_{jik}\right)\left(T_{ik}^V + \tau_{ij}X_{ijk} + X_{ijk}T_{ik}^U\right) = \left(X_{ijk} + X_{jik}\right)\left(T_{jk}^V + \tau_{ji}X_{jik} + X_{jik}T_{jk}^U\right),$$

$$\forall (i,j) \in A, \forall k \in K \tag{22}$$

Note that Eq. (22) contain multiple nonlinear combinations of variables. From Glover (1975), consider a continuous variable Z where $Z = Ax$, with A as a continuous variable bounded by $[0, M)$ and x as a binary variable. Z can be replaced by inequalities (23) – (25), where M is a large constant number. Here we consider $(X_{ijk} + X_{jik})$ as x and $(T_{ik}^V + \tau_{ij}X_{ijk} + X_{ijk}T_{ik}^U)$ or $\left(T_{jk}^V + \tau_{ji}X_{jik} + X_{jik}T_{jk}^U\right)$ as A. Similarly, $X_{ijk}T_{ik}^U$ and $X_{jik}T_{jk}^U$ can also be regarded as $Z = Ax$ and replaced by equivalent inequalities.

$$Z \leq Mx \tag{23}$$

$$Z \leq A \tag{24}$$

$$Z \geq A - M(1 - x) \tag{25}$$

Therefore, the equivalent linearized constraints of Eq. (22) are now shown in constraints (26) – (32). We replace the left- and right-hand side of Eq. (22) with Z_{ijk} and Z_{jik} as in constraints (26), respectively. Constraints (27) – (29) are the equivalent inequalities for Z_{ijk} and constraints (30) – (32) are used to replace $X_{ijk}T_{ik}^U$ with U_{ijk}. In Eq. (29), if both X_{ijk} and X_{jik} equal to 1 at the same time, the constraints would be equivalent to $Z_{ijk} \geq M$, which exceeds the upper limit of Z_{ijk} and leads to an infeasible solution. Thus, the subtour for both X_{ijk} and X_{jik} equal to 1 is eliminated here.

$$Z_{ijk} - Z_{jik} = 0, \forall k \in K, \forall (i,j) \in A \tag{26}$$

$$Z_{ijk} \leq \left(X_{ijk} + X_{jik}\right)M, \forall k \in K, \forall (i,j) \in A \tag{27}$$

$$Z_{ijk} \leq T_{ik}^V + \tau_{ij}X_{ijk} + U_{ijk}, \forall k \in K, \forall (i,j) \in A \tag{28}$$

$$Z_{ijk} \geq T_{ik}^V + \tau_{ij}X_{ijk} + U_{ijk} - \left[1 - \left(X_{ijk} + X_{jik}\right)\right]M, \forall k \in K, \forall (i,j) \in A \tag{29}$$

$$U_{ijk} \leq X_{ijk}M, \forall k \in K, \forall (i,j) \in A \tag{30}$$

$$U_{ijk} \leq T_{ik}^U, \forall k \in K, \forall (i,j) \in A \tag{31}$$

$$U_{ijk} \geq T_{ik}^U - \left(1 - X_{ijk}\right)M, \forall k \in K, \forall (i,j) \in A \tag{32}$$

3.3 Proposed passenger pickup and drop-off time constraints

The passenger pickup and drop-off time is measured by the arrival time of the assigned pickup and delivery vehicle. Under certain circumstances, other vehicles may also traverse the same pickup and delivery location of a specific request, so it is necessary to identify the designated pickup and delivery vehicle for each request. Therefore, constraints (33) and (34) are proposed to capture the customer-vehicle pickup and delivery assignments, respectively. The decision variable $S_{kri} = 1$ if vehicle k is assigned to pick up or drop off request r at node i, and 0 otherwise.

$$S_{kri} = \sum_{(i,j)\in A_i^-} Y_{ijkr} - \sum_{(j,i)\in A_i^+} Y_{jikr}, \forall k \in K, \forall r \in R, \forall i \in \{o_{rl}\} \tag{33}$$

$$S_{kri} = \sum_{(j,i)\in A_i^+} Y_{jikr} - \sum_{(i,j)\in A_i^-} Y_{ijkr}, \forall k \in K, \forall r \in R, \forall i \in \{d_{rl}\} \tag{34}$$

Constraints (35) are proposed to measure the corresponding pickup and delivery time for each customer request, where the equivalent linearized inequalities are shown in constraints (36) – (38). We use dummy variable $W_{kri} = S_{kri}T_{ik}^V$ to represent the product of the binary customer-vehicle assignment and the continuous vehicle arrival time using the same substitution method in Eqs. (23) – (25).

$$T_{ir}^Q = \sum_{k\in K} W_{kri}, \forall r \in R, \forall i \in \{o_{rl}, d_{rl}\} \tag{35}$$

$$W_{kri} \le S_{kri}M, \forall k \in K, \forall r \in R, \forall i \in \{o_{rl}, d_{rl}\} \tag{36}$$

$$W_{kri} \le T_{ik}^V, \forall k \in K, \forall r \in R, \forall i \in \{o_{rl}, d_{rl}\} \tag{37}$$

$$W_{kri} \ge T_{ik}^V - (1 - S_{kri})M, \forall k \in K, \forall r \in R, \forall i \in \{o_{rl}, d_{rl}\} \tag{38}$$

3.4. Proposed vehicle platoon constraints

For any two vehicles k and m ($k \neq m$) traveling in a platoon over arc ij, their departure time at node i and arrival time at node j are guaranteed to be the same by constraints (39) – (42). The decision variable $P_{ijkm} = 1$ if vehicle k and m travel in platoon over the arc ij, and 0 otherwise. Constraints (43) ensure that $P_{ijkm} = 1$ if and only if both vehicles travel on the same arc ij.

$$\left(T_{ik}^V + T_{ik}^U\right) - \left(T_{im}^V + T_{im}^U\right) \le M\left(1 - P_{ijkm}\right), \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{39}$$

$$\left(T_{im}^V + T_{im}^U\right) - \left(T_{ik}^V + T_{ik}^U\right) \le M\left(1 - P_{ijkm}\right), \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{40}$$

$$T_{jk}^V - T_{jm}^V \le M\left(1 - P_{ijkm}\right), \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{41}$$

$$T_{jm}^V - T_{jk}^V \le M\left(1 - P_{ijkm}\right), \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{42}$$

$$2P_{ijkm} \le X_{ijk} + X_{ijm}, \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{43}$$

Constraints (44) are used to capture the number of vehicles traveling with vehicle k on the arc ij, while constraints (45) limit the maximum allowed number of vehicles in a platoon.

$$P_{ijk}^N \le \sum_{m\in K, m\neq k} P_{ijkm}, \forall k \in K, \forall (i,j) \in A \tag{44}$$

$$P_{ijk}^N \le u - 1, \forall k \in K, \forall (i,j) \in A \tag{45}$$

3.5. Proposed variable capacity constraints

Two challenges exist in formulating the capacity constraints of MVs. First, during the operation, each individual MV may join and leave the platoon at any time and location, which leads to frequent changes of the onboard carrying capacity. This requires the model to track the status of each vehicle, whether it is traveling in platoon or by itself. Second, since MVs can be physically connected as a longer bus-like platform and passengers are free to move between connected MVs, a platoon consisting of multiple connected vehicles should be treated as ONE entire unit with an expanded capacity, rather than a group of individual and independent vehicles that simply moving together.

To clarify, we use the term of “expanded capacity” to refer to the maximum carrying capability of one unit, which could be an individual independent vehicle or a platoon of connected MVs. The total capacity of a platoon of MVs remain the same as the sum of all individual vehicles. However, under certain circumstances, even as the total carrying capacity stays the same, a platoon of connected MVs can still serve more requests than the same number of individual and independent vehicles (higher service throughput) due to requests with group sizes greater than one. For example, consider two vehicles (each with capacity of 4) and three requests (each with 3, 3, 2 number of passengers). If we use traditional capacity constraints to just make sure the capacity limit of each individual vehicle in the platoon is not violated, we can only serve at most two requests at the same time. Either combination of two requests would have more than 4 passengers in total, which exceeds the capacity limit of 4 in any individual vehicle. Thus, each individual and independent vehicle can only serve at most one request at a time. However, if these two MVs are connected and treated as one entire unit with expanded capacity as introduced, they would now have a maximum carrying capacity limit of 8 (the total capacity does not change, still $4 + 4 = 8$). In this case, the new “vehicle” made of two MVs can serve all three requests at the same time ($3 + 3 + 2 = 8$). This

higher service throughput benefit also makes sense in practice, where buses and light rails might be able to carry more passengers than individual vehicles like taxis with the same required road space.

Traditional vehicle capacity constraints in the literature only limit the number of carried passengers on each individual vehicle, which is not suitable to address the above-mentioned challenges. In our model, constraints (46) are proposed to accommodate the variable capacity changes regarding to platoon length. The left-hand side of constraints (46) indicates the total number of passengers in the platoon that vehicle k involves, while the right-hand side of constraints (46) is the total on-board capacity of the platoon. If vehicle k travels by itself on arc ij, constraints (46) would become the traditional vehicle capacity constraints as $\sum_{r\in R} q_r Y_{ijkr} \le c_k X_{ijk}$. In addition, constraints (46) also ensure that a request served by any vehicle can only happen when that vehicle also travels on the same arc. Here, in Eq. (46), we use another dummy variable $C_{ijkm} = P_{ijkm}\left(\sum_{r\in R} q_r Y_{ijmr}\right)$ to calculate the number of passengers carried by vehicle m if vehicle k and vehicle m travel in platoon on the same arc ij. The equivalent linearized constraints for the dummy variable C_{ijkm} are shown in Eqs. (47) – (49).

$$\sum_{r\in R} q_r Y_{ijkr} + \sum_{m\in K, m\neq k} C_{ijkm} \le c_k X_{ijk} + \sum_{m\in K, m\neq k} c_m P_{ijkm}, \forall k \in K, \forall (i,j) \in A \tag{46}$$

$$C_{ijkm} \le P_{ijkm} M, \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{47}$$

$$C_{ijkm} \le \sum_{r\in R} q_r Y_{ijmr}, \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{48}$$

$$C_{ijkm} \ge \sum_{r\in R} q_r Y_{ijmr} - \left(1 - P_{ijkm}\right) M, \forall k, m \in K, k \neq m, \forall (i,j) \in A \tag{49}$$

3.6. Proposed model extensions and summary

Platoon saving rates. In contrast to traditional truck platooning studies in literature, MVs possess lighter mass and are physically connected with zero inter-vehicle gap between each other. In a platoon with three trucks (Tsugawa et al., 2016), the lead vehicle saves the least fuel from platooning (up to about 9% with 5 m gap), while the following vehicles in the middle of platoon benefit the most from energy savings (up to about 23% with 5 m gap), and the fuel saving of the tail vehicle does not change much regardless of the inter-vehicle gap (about 15%). Similar conclusions can also be found in Song et al. (2021). Thus, it might be safe to assume that, if the number of vehicles in a platoon is large enough, the average cost savings among all vehicles would approach but never reach the saving rate of middle vehicles (upper bound). Moreover, Song et al. (2021) also showed that the average fuel cost nonlinearly decreases with the total number of vehicles in a platoon. However, in this study, we do not distinguish the vehicle position in platoon, such as the lead vehicle, following vehicle, and tail vehicle. Instead, we average the cost savings among all vehicles for the operator cost and assume that the average travel cost linearly decreases with the platoon length for convenience as shown in Eq. (50). Since the design of MVs has 0 m gaps between vehicles in a platoon, we might expect that the average energy consumption among all MVs decreases at similar or even better rates than truck platoons as the platoon length increases because of economies of scale.

The vehicle travel cost term in the objective function (1) shown in Section 3.1 can be further modified and substituted by Eq. (50), where η_1 represents an initial platoon saving rate if platooning occurs and η_2 represents additional saving bonus if with more vehicles in the platoon. A new binary decision variable, P^V_{ijk}, is needed and defined in Eq. (51) to indicate whether vehicle k travels in platoon on arc ij or not.

$$\sum_{k\in K} \sum_{(i,j)\in A} d_{ij} \left(X_{ijk} - \eta_1 P^V_{ijk} - \eta_2 \left(P^N_{ijk} - 1 \right) \right) \tag{50}$$

$$\sum_{m\in K, m\neq k} P_{ijkm} \le P^V_{ijk} M, \forall k \in K, \forall (i,j) \in A \tag{51}$$

Hard time windows. Throughout this study, we try to solve the optimal routes for the MDARP and explore the potentials of MVs by penalizing the request service time in the objective function without applying hard time windows for pickups and deliveries. However, we do have constraints to ensure that passengers can only be served after they submit the request. Since we have already provided constraints to measure the passenger pickup and delivery time, constraints (52) and (53) can be easily applied if hard time windows are required for the pickup and delivery.

$$a^o_r \le T^Q_{ir} \le b^o_r, \forall r \in R, \forall i \in \{o_r\} \tag{52}$$

$$a^d_r \le T^Q_{ir} \le b^d_r, \forall r \in R, \forall i \in \{d_r\} \tag{53}$$

Passenger en-route transfers. As one of the key features of MV, passengers can be relocated between vehicles when they are traveling in the same platoon to optimize their delivery routes. For any platoon of MVs traveling on arc ij, we assume that passengers can be transferred at any time when they move from node i to node j. Since passenger flows are conserved at each node in our proposed MILP model, here we could add a binary decision variable F_{rikm} (1 if request r is transferred from vehicle k to vehicle m at node i, and 0 otherwise) to capture the passenger en-route transfers. For any request r transferred from vehicle k to m over the arc ij, we would have

either $F_{rikm} = 1$ or $F_{rjkm} = 1$. Constraints (54) are used to identify the passenger en-route transfers and constraints (55) – (56) ensure that $F_{rikm} = 1$ if and only if both $\sum_{(j,i)\in A_i^+} Y_{jikr}$ and $\sum_{(i,j)\in A_i^-} Y_{ijmr}$ equal to 1. Furthermore, constraints (57) ensure that the passenger en-route transfer captured at node i between vehicles k and m can only happen when they both travel in the same platoon and pass through node i.

$$\sum_{(j,i)\in A_i^+} Y_{jikr} + \sum_{(i,j)\in A_i^-} Y_{ijmr} \le F_{rikm} + 1, \forall r \in R, \forall i \in N, \forall k, m \in K, k \ne m \tag{54}$$

$$F_{rikm} \le \sum_{(j,i)\in A_i^+} Y_{jikr}, r \in R, \forall i \in N, \forall k, m \in K, k \ne m \tag{55}$$

$$F_{rikm} \le \sum_{(i,j)\in A_i^-} Y_{ijmr}, r \in R, \forall i \in N, \forall k, m \in K, k \ne m \tag{56}$$

$$F_{rikm} \le \sum_{(i,j)\in A_i^-} P_{ijkm} + \sum_{(j,i)\in A_i^+} P_{jikm}, \forall r \in R, \forall i \in N, \forall k, m \in K, k \ne m \tag{57}$$

In summary, the full MILP model for MDARP consists of Eqs. (1) – (21) and (26) – (49). As this simplifies to a DARP when the set of vehicle platoon, passenger en-route transfers and variable capacity constraints are relaxed from the model, the problem is already NP-hard and thus requires efficient heuristics to solve problems of practical size.

4. Proposed heuristic algorithm

We propose a route improvement heuristic based on modifying feasible solo (S) mode (non-transfer non-platoon) solution (which can be obtained using any existing DARP algorithm).

Based on the routes from solo mode, we iteratively seek to improve them with modular (M) mode solutions. For convenience, we use the word "platoon" to describe feasible MV routes, where a platoon might consist of multiple MVs with different origins and destinations but at least share one common segment along their paths.

The main idea is to partition the general problem into multiple subproblems. In Section 4.1, the solo mode routes are deconstructed and then reconstructed between pairwise individual vehicles to find feasible two-vehicle platoons. In Section 4.2, we first iteratively merge between feasible MV platoons. If there is any new platoon created, we continue to explore merging until no platoons can be joined with each other. Then, remaining individual vehicles are iteratively inserted into platoons found previously to extend the common platoon paths and maximize the cost savings. Thus, the proposed heuristic algorithm consists of two major parts: (1) a Steiner-tree-inspired neighborhood search algorithm to modify the solo mode routes and find two-vehicle MV platoons, and (2) an improvement heuristic to iteratively merge between feasible MV platoons and then insert individual vehicles to platoons to maximize the common platoon path and save more cost. Since we constantly modify and try to improve the solution, the proposed heuristic could also be applied to dynamic scenarios with some modifications, such as the tracking of vehicle and request status, and the option to change existing vehicle routes with passengers onboard.

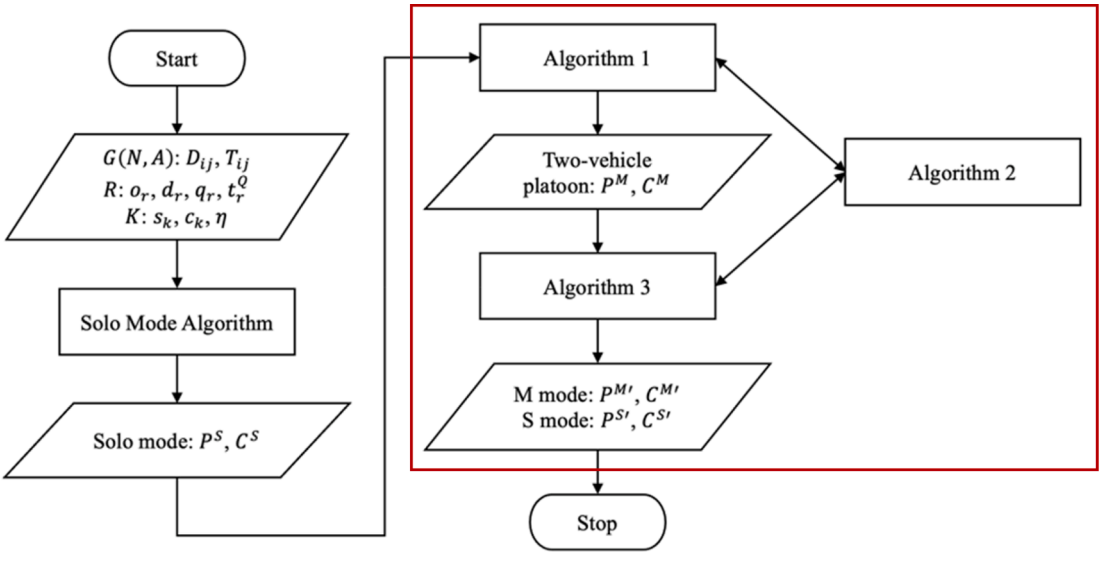


Fig. 5. Framework of overall heuristic algorithm (in the red box) integrated with a DARP algorithm. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Unlike the MILP model using link costs on an undirected graph to find the optimal routes, we use pre-processed shortest-path travel distance (SPD) and time (SPT) matrices between pairwise nodes *as a complete graph* in the heuristic. As a result, the multi-layer structure from the MILP model is no longer needed in our proposed heuristic algorithm. In addition, the output vehicle routes only need to be expressed in terms of starting and ending vehicle locations, request pickup/drop-off locations, and platoon join/split locations, rather than all explicit nodes along the vehicle route.

The overall framework of proposed heuristic is summarized in Fig. 5.

4.1 Two-vehicle platoon

Algorithm 1 is used to search for two-vehicle platoons and passenger en-route transfers between individual vehicles. The main idea of searching for platoon join/split location was inspired from the Steiner Tree problem, especially the case with four terminals and two Steiner points (Beasley, J., 1992; Zachariasen, M., 1999; Bhoopalam et al., 2018). Based on the large neighborhood search (LNS) algorithm, Algorithm 1 iteratively destroys and reconstructs the solo mode vehicle routes to search for platoon possibilities. Passengers can be re-assigned to a different vehicle and their pickup and drop-off sequences can be changed as well. However, since MVs traveling in platoon can enlarge the on-board carrying limit, vehicle capacity constraint is temporarily ignored when reconstructing the routes but it is still guaranteed when checking the solution feasibility later. Next, for any pair of reconstructed routes, we check every segment along the new vehicle routes, search for potential platoon join and split locations, force them to divert and travel in platoon between the join and split nodes, and then calculate the corresponding cost. If the platoon deviation satisfies capacity constraints and time window constraints, we continue to iteratively extend the shared common platoon path in the routes to maximize the platoon length and cost savings. Lastly, we again destroy and rebuild the delivery paths for passengers to consider en-route transfers within those platoons.

Algorithm 1. Two-vehicle platoon formation.

Input:	Graph $G(N,A)$, set of R with o_r, d_r, q_r and $t_r^Q = a_r^o$, set of K with s_k and c_k. Platoon saving rate η. $\varsigma = 1$.
Initialization:	SPD matrix D_{ij} and SPT matrix T_{ij} for any nodes $i,j \in N$. Solo mode routes P^S and costs C^S.
1.	**For** any $k,m \in K, k \neq m$ **do**
2.	Deconstruct solo mode paths p_k^S, $p_m^S \in P^S$ into sets of stops.
3.	Randomly reconstruct new sets of routes as P_{km}^M, ignoring c_k, c_m and t_r^Q.
4.	**For** each segment in routes $p_k^M, p_m^M \in P_{km}^M$ **do**
5.	Search for J and S from N and to generate candidate platoons P_{km}^C (*Algo 2*).
6.	Calculate new costs of P_{km}^C with η, while satisfying c_k, c_m and t_r^Q.
7.	**For** each $p_{km}^C \in P_{km}^C$ **do**
8.	**While** $\varsigma = 1$ **do**
9.	Search for additional J prior to start and S after end of existing platoon (*Algo 2*).
10.	Calculate new costs with modified routes.
11.	Record the extension if improvement found, otherwise $\varsigma = 0$.
12.	Deconstruct extended p_{km}^C after end of platoon into sets of stops.
13.	Randomly reconstruct new sets of passenger delivery routes between k,m as ϕ.
14.	Calculate new costs of ϕ with passenger en-route transfers. Set $\varsigma = 1$.
15.	Rank and select platoons by descending order of savings against C^S as $p_k^M, p_m^M \in P^M$ with associated $c_k, c_m \in C^M$. Remove selected $k,m \in K$ from P^S and C^S.
Output:	Two-vehicle modular mode routes P^M and costs C^M, updated P^S and C^S.

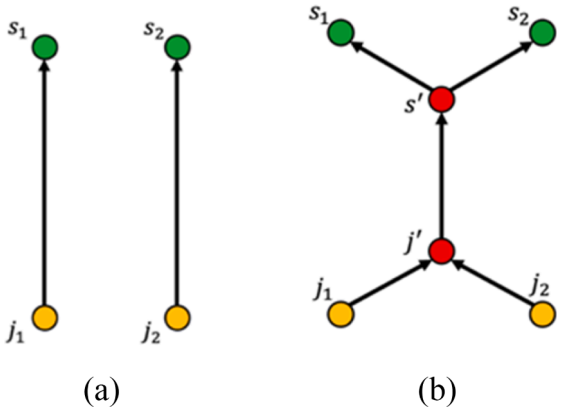


Fig. 6. Platoon join/split location search illustration.

Note: *Algo* for Algorithm.

For the ideal platoon join and split locations, they may already exist in current vehicle routes or in the neighborhood along the vehicle paths, which requires a neighborhood search algorithm in the latter case. As shown in Fig. 6(a), given two route segments $j_1 \rightarrow s_1$ from vehicle 1 and $j_2 \rightarrow s_2$ from vehicle 2 (consecutive visiting nodes along the vehicle route), both j_1 and j_2 are then considered as candidate platoon join locations, while both s_1 and s_2 are considered as candidate split locations.

We then use a neighborhood search method inspired from the Steiner Tree problem, as presented in Algorithm 2, to find and return a number of potential join ($j_1, j_2 \in J$) and split ($s_1, s_2 \in S$) nodes in the neighborhood. The ideal platoon join/split nodes should be located in-between the candidate join/split nodes with minimum connecting distance and corresponding difference to reduce the vehicle detour cost as much as possible. The cost coefficient φ is used to make trade-offs between these two measurements. For convenience, we set the value of $\varphi = 1$ throughout this study. Next, we iterate over each combination of candidate platoon join ($j' \in J$) and split ($s' \in S$) nodes , to divert the vehicle routes and force them to travel in platoon along the segment $j' \rightarrow s'$. We calculate the new cost of modified vehicle routes to identify the maximum feasible platoon length among different combinations of j' and s' (Fig. 6(b)).

Algorithm 2. *Platoon join/split location search.*

Input:	Nodes $n_1, n_2 \in N$, SPD matrix D_{ij}.
	Maximum number of returned nodes N_{max}, and cost coefficient φ.
Initialization:	Cost $\delta_i = M$ for $i \in N$.
1.	**For** each $i \in N \backslash \{n_1, n_2\}$ **do**
2.	Record and update the cost $\delta_i = (D_{n_1 i} + D_{n_2 i}) + \varphi \lvert D_{n_1 i} - D_{n_2 i} \rvert$.
3.	Rank and select N_{max} nodes by descending orders of δ_i as J or S.
Output:	Candidate platoon join J or split S locations.

To demonstrate the process of identifying and selecting two-vehicle platoon, we use the same illustrative example in Section 2.1 here. Note that the following illustration does not intend to list all detailed steps but to show the logic of our proposed heuristic algorithm for approaching the solution. Given one of the reconstructed solo mode routes (in terms of visiting nodes with pickup and delivery) of

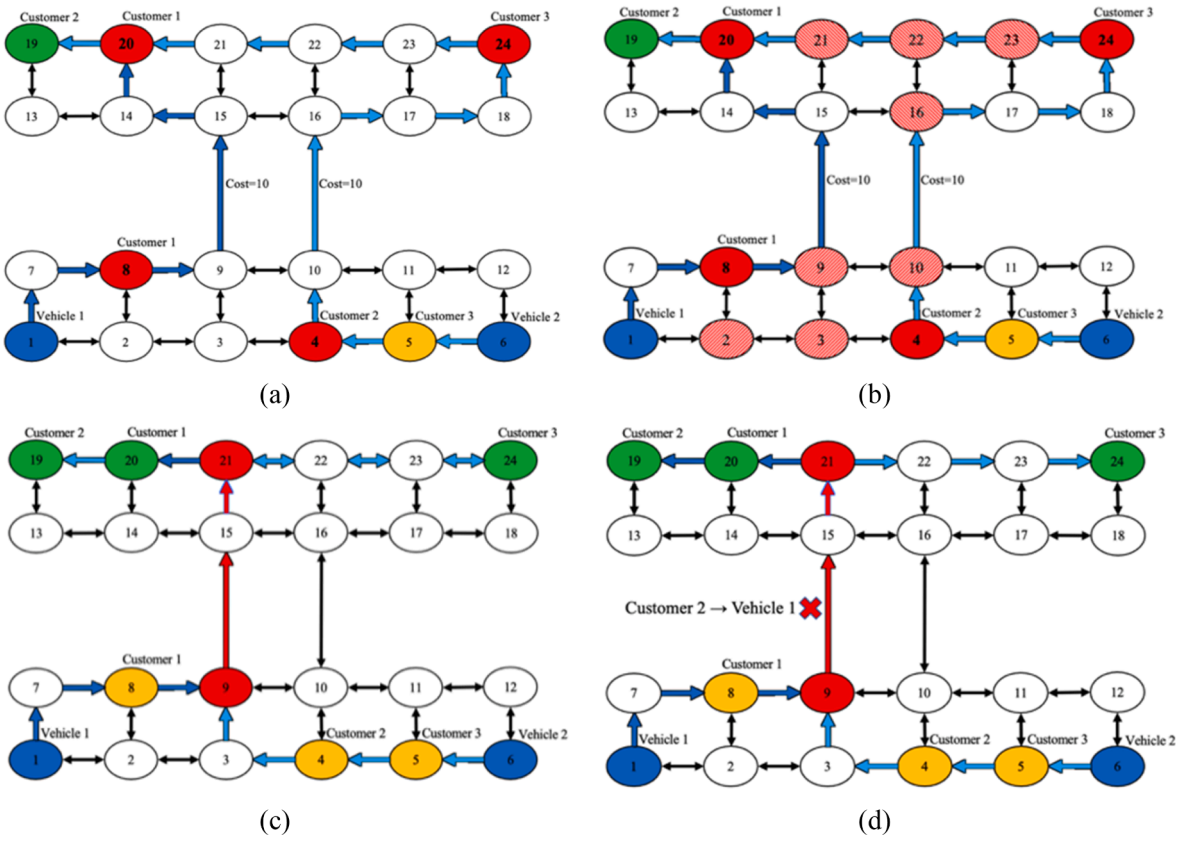


Fig. 7. Example for finding two-vehicle platoon.

vehicle 1 as 1 → 8 → 20 and vehicle 2 as 6 → 5 → 4 → 24 → 19 (Fig. 7(a)), we would then search between every pair of route segments along the vehicle 1 and 2′s paths. For instance, when we try the segment 1 → 8 from vehicle 1 and the segment 6 → 5 from vehicle 2, the potential join (including nodes 1 and 6) and split (including nodes 8 and 5) locations would be distributed among the nodes {1,2,3,4,5,6} and nodes {8,9,10,11,2,3,4,5} according to Algorithm 2, respectively. If we set $N_{max} = 2$ and only return two potential locations, we would then select nodes {3,4} for join and nodes {3,10} for split. These nodes are preferred since they are located in-between the route segment start {1,6} and end {8,5} nodes with minimum detour distance for both vehicles. Then, we would insert the potential join and split nodes and modify the vehicle 1 path as 1 →{join at 3 or 4}→{split at 3 or 10}→ 8 → 20 and vehicle 2 path as 6 →{join at 3 or 4}→{split at 3 or 10}→ 5 → 4 → 24 → 19. Obviously, we could not find a feasible platoon for this pair of route segments, because the segments 1 → 8 and 6 → 5 are too far from each other while the travel distances themselves are too short to reduce cost sufficiently from platooning.

After iterating and testing every pair of route segments between these two vehicles (Algorithm 1 steps 1–6), we find that segments 8 → 20 from vehicle 1 and 4 → 24 from vehicle 2 can form a successful two-vehicle platoon. As shown in Fig. 7(b), with nodes {8,4} as candidate join locations and set $N_{max} = 4$, Algorithm 2 returns additional join nodes $\{2,3,9,10\} \in J$. Similarly, consider nodes {20,24} as candidate split locations and then potential split nodes $\{16, 21, 22, 23\} \in S$ are returned (Fig. 7(b)). By iterating the nodes in sets of J and S, we calculate the new cost of modified vehicle routes after insertion and find that platoon segment 9 → 21 generates the most savings from solo mode (Fig. 7(c)). Now, the vehicle 1 path becomes 1 → 8 →{join at 9}→{split at 21}→ 20 and the vehicle 2 path becomes 6 → 5 → 4 →{join at 9}→{split at 21}→ 24 → 19. As for following steps 7–11 in Algorithm 1, we again search for additional join and split nodes to try to further extend the already found platoon segment if possible. In this example, the platoon between nodes {9,21} is already the longest possible platoon solution. But in some cases where two vehicles share multiple consecutive visiting nodes along their paths, steps 7–11 in Algorithm 1 would keep searching and extending the platoon part until there is no improvement found.

Up to this step, vehicle 2 is still responsible to deliver requests 2 and 3, which does not consider en-route transfer and is not optimal yet. Referring to Algorithm 1 steps 12–14, we reconstruct the delivery nodes {20,24,19} of vehicle 1 and 2 after the platoon split and search for en-route transfers that could further reduce the cost. In this case, we find that relocating request 2 from vehicle 2 to vehicle 1 over the platoon segment 9 → 21 could reach even more cost savings, as shown in Fig. 7(d). Thus, the final optimal path of vehicle 1 is 1 → 8 →{join at 9}→{split at 21}→ 20 → 19 and vehicle 2 is 6 → 5 → 4 →{join at 9}→{split at 21}→ 24.

4.2 Multi-vehicle platoon joining

In the previous section, we propose a customized neighborhood search algorithm to find the minimal platoon formation consisting of only two vehicles. To search and construct multi-vehicle platoons, Algorithm 3 first iteratively merges platoons together from the results of Section 4.1, and then insert remaining individual vehicles into feasible platoons found previously to explore all the join possibilities.

Algorithm 3. *Multi-vehicle platoon search.*

Input:	Algorithm 1 inputs and outputs, maximum platoon length u, $\varsigma = 1$.
1.	**While** $\varsigma = 1$ **do**
2.	**For** any $p_i^M, p_j^M \in P^M$ **do**
3.	Identify the *LCPS* l_i in p_i^M and l_j in p_j^M.
4.	**For** each segments in l_i and l_j **do**
5.	Identify the *LCPS* l_{ij} between selected segments of l_i and l_j.
6.	**If** l_{ij} exists **then**
7.	Merge p_i^M and p_j^M over l_{ij}, and calculate new costs while satisfying u.
8.	Search for platoon extension and en-route transfers (as *Algo 1, lines 8–14*).
9.	Rank and select the platoon joins with most savings than C^M and update P^M as $P^{M'}$.
10	**If** no platoon join can be found **then** $\varsigma = 0$
11.	**For** any $\{k \in K \mid k \notin P^{M'}\}$ and $p_m^M \in P^{M'}$ **do**
12.	Insert $p_k^S \in P^S$ into p_m^M and calculate the new cost while satisfying u.
13.	Search for platoon extension and en-route transfers (as *Algo 1, lines 8–14*).
14.	Rank and select the individual vehicle insertions into $P^{M'}$ with costs $C^{M'}$ and update P^S as $P^{S'}$ with costs $C^{S'}$.
Output:	Solo mode $P^{S'}$ with $C^{S'}$ and modular mode $P^{M'}$ with $C^{M'}$.

Note: *LCPS* = longest common platoon segments.

We keep searching for join between platoons while there still exists any platoon that has not been explored yet or new platoon join is just created. If any new platoon is created, we re-iterate the join between all platoons again. At each iteration, by choosing any pair of platoons, we first identify the longest common platoon segments (LCPS) for each platoon group (lines 2 to 3). Then, the heuristic iterates over each segment between the two LCPS and search for their common platoon paths again (lines 4 to 5). Note that our proposed heuristic algorithm solves the MDARP based on shortest-path distance and time, which might leave out intermediate nodes along the routes and leads to difficulty in identifying the LCPS. To tackle this, all the nodes along the shortest path of each segment are listed and then used to identify the LCPS. Once a common platoon path is found, the heuristic merges these two platoons together over the shared segments and re-calculates the new cost to ensure the capacity constraints and platoon length limit are satisfied (lines 6 to

7). Then, similar to Algorithm 1, we continue to search for platoon extension and passenger en-route transfers (line 8). If there is no further possibility to merge between platoons, remaining individual vehicles are iteratively inserted into existing platoons (lines 11 to 13). Algorithm 3 returns the vehicle routes for modular mode and remaining unmatched individual vehicle routes from solo mode.

5. Numerical experiments

Numerical experiments are conducted in this section to (1) evaluate the computational performance and (2) explore the potential benefits of modular vehicle technology. We first implement small-scale tests on a simple network. The optimal solutions obtained from the MILP model and the corresponding computation times are compared with our proposed heuristic algorithm. To solve the MILP model, we used Gurobi 8.1.1 optimization software as the commercial solver, running on a 64-bit Windows 8.1 personal computer with the Intel Core i7-6700 K CPU and 40 gigabyte RAM. For the parameter settings of Gurobi, we only set the runtime limit to be 2 h with all other parameters left as default. To further explore the potential benefits and optimal operation scenarios of MVs, large-scale and more practical instances are tested on the Anaheim network (378 nodes and 796 arcs, after removing centroids) with our proposed heuristic algorithm. The network information can be found in Github (Stabler, 2022). For convenience, the initial feasible solo mode solutions are obtained using a basic insertion heuristic (see Fu and Chow, 2022) without any loss of generality in the effectiveness of the proposed heuristic.

Overall, from the set of numerical experiments, our proposed MILP model can be solved exactly using the commercial software Gurobi for up to 4 vehicles and 4 requests within a 2-hour computation time limit on a small-scale network with 13 nodes and 40 links, which again indicates the need for an efficient heuristic for practical instances. The required computation time of the commercial solver increases quickly with the consideration of passenger service time and the number of vehicles and requests. However, the commercial solver might be able to handle a larger problem size if it mainly concentrates on the operator's perspective. Moreover, to reduce the solution space and improve the capability for more practical applications, the model could be changed to apply to directed networks with pre-defined candidate routes. Vehicles and requests with similar spatial–temporal characteristics could be clustered together and treated as a new identity rather than separate and independent individuals. By comparison, our proposed heuristic algorithm could handle a much more complex instance with up to 75 vehicles and 150 requests on the Anaheim network with 378 nodes and 796 arcs with the same 2-hour time limit.

5.1. Small-scale test

The goal of the small-scale test is to evaluate the computational efficiency of a commercial solver for the MILP model and the heuristic, as well as the optimality of the heuristic algorithm under different scenarios. Two operation policies, solo mode and modular mode, are compared. The optimal solutions found using a benchmark MILP commercial solver with branch-and-bound/-cut methods are compared with our heuristic algorithm. Since the MDARP with different vehicle and passenger locations requires enormous computation power to solve exactly, we are only able to handle examples involving at most 4 vehicles and 4 requests within 2 h computation time limit.

Three scenarios (S) are tested on the network shown in Fig. 8. Each scenario has a different combination of vehicles and requests: $(S1)|K| = 2, |R| = 3$, $(S2)|K| = 3, |R| = 4$, and $(S3)|K| = 4, |R| = 4$. There are 13 nodes and 40 links on the network, where the travel cost is labeled next to each link in both directions. For the MILP model, two layers are used. For convenience, we assume that the travel distance is in units of miles and the vehicle travel speed to be 1 mile/min for all tests. As a result, the travel cost in distance and time are in the same value for each link. Small-scale test instances, including vehicle initial locations, passenger pick-up and drop-off locations, and number of passengers in each request, can be found on Github (Fu, 2022). The vehicle capacity is set to be 4 and the request submission time is set to be 0 in all small-scale tests. Various combinations of objective weight values are chosen to evaluate the trade-offs between operator cost and passenger cost. Since request on average consists of more than 2 passengers, the weight values vary from $\beta = 1$, $\beta = 1/2$, $\beta = 1/4$, $\beta = 1/6$ to $\beta = 0$. In addition, we set the platoon saving rate η to be 5% and 10%.

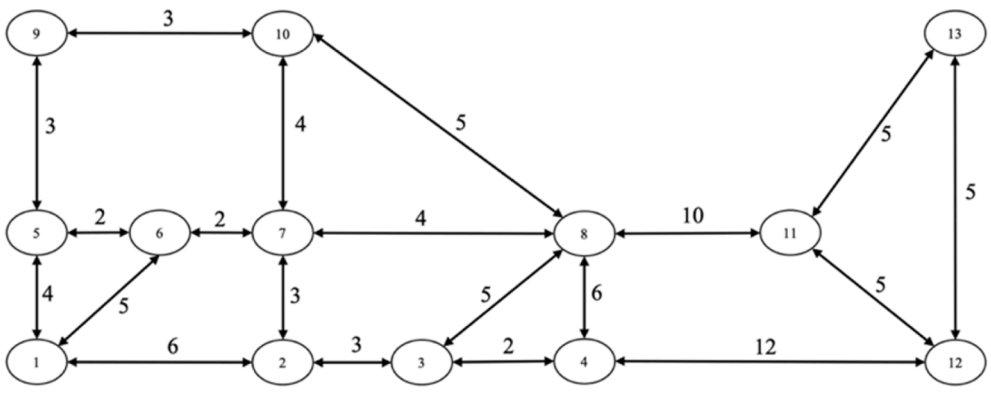


Fig. 8. Small-scale test network.

The computation accuracy is summarized in Fig. 9 for various objective weight values. We consider the optimal objective value obtained from solo mode and solved from the MILP model as the baseline (the 100%, light grey column without shading in Fig. 9). The optimality gap between MILP solution and heuristic algorithm (HA) can be observed by comparing them for the same scenario respectively: solo mode, modular mode with $\eta = 5\%$, and modular mode with $\eta = 10\%$.

HA performs best when considering passenger service time in the objective function (average of 0.17% when $\beta = 1$). Overall, our proposed HA only shows an average of 0.6% optimality gap from the objective value obtained by MILP solver. The cost savings and benefits from MVs can be observed by comparing the MILP objective values across different scenarios. Without considering the passenger service time, MV saves about 10% when $\eta = 5\%$ and about 15% when $\eta = 10\%$ against the solo mode. When the cost on passenger side is also included, the savings from MV mode decrease to about 4.1% when $\eta = 5\%$ and 6.6% when $\eta = 10\%$. From all observations, we find that MVs save more cost when the operator side gets more attention and weight, i.e. passengers are more inelastic to travel cost.

The computation time for solving MILP model and heuristic algorithm under different operation modes and scenarios are summarized in Table 4. The required computation time of the MILP model increases dramatically with the consideration of passenger service time under MV mode, whereas our proposed heuristic algorithm remains stable for all cases (Table 4(a)). This suggests that the MILP model might be beneficial if focusing more on the operator's perspective in practice. From Table 4(b), the computation time of MILP model increases by more than 10 times over the three scenarios with different vehicle and request numbers. By contrast, the heuristic algorithm only requires a few seconds before obtaining the solution and outperforms the MILP solver as the problem size increases. The problem size in this small-scale test only goes from $|K| = 2, |R| = 3$ in $S1$, $|K| = 3, |R| = 4$ in $S2$, to $|K| = 4, |R| = 4$ in $S3$ on a 13-node, 40-link network. The results from Table 4 show that the MILP model is not practical for realistic scenarios. Overall, our proposed heuristic algorithm can reach an average optimality gap of 0.57% with only a fraction of the required computation time against the MILP model from the set of small experiments.

5.2. Large-scale tests

To explore the potential benefits and identify the ideal operation scenarios of modular vehicle technology, we apply our heuristic algorithm to instances on the Anaheim network, as shown in Fig. 10 with 378 nodes and 796 arcs (zone centroids 1–38 and linked arcs are removed from the network), to conduct more realistic large-scale numerical experiments. All 378 nodes are considered potential join/split nodes as well pickup or drop-off locations. The shortest-path travel distance and travel time matrices are calculated beforehand and imported at the beginning of our algorithms for each pair of nodes. The unit of travel distance on each link is in miles and the unit of free-flow travel time on each link is in minutes.

Eight random instances are drawn each from a set of 40 scenarios (10 different problem sizes and 4 different spatial distributions) resulting in a total of 320 instances, from which the output measures are used to fit a linear regression model to characterize the relationship of the different parameters on the cost reduction from solo mode.

The large-scale experiment settings are summarized in Table 5. A total number of ten different problem sizes are defined ranging in terms of the number of vehicles and requests from $K = 5, R = 8$ to $K = 25, R = 50$. For each size, we consider four spatial distribution scenarios for the vehicle initial locations, request pickup and drop-off locations: (1) uniformly randomly generated over the network; (2–4) clustered around randomly selected node with 3, 5, or 10 neighborhoods, respectively. We also consider five temporal distribution scenarios for the request submission times: (1) all demands submit ride requests at $T = 0$; (2–5) uniformly randomly generated over [0,1], [0,4], [0,8], or [0,16] mins.

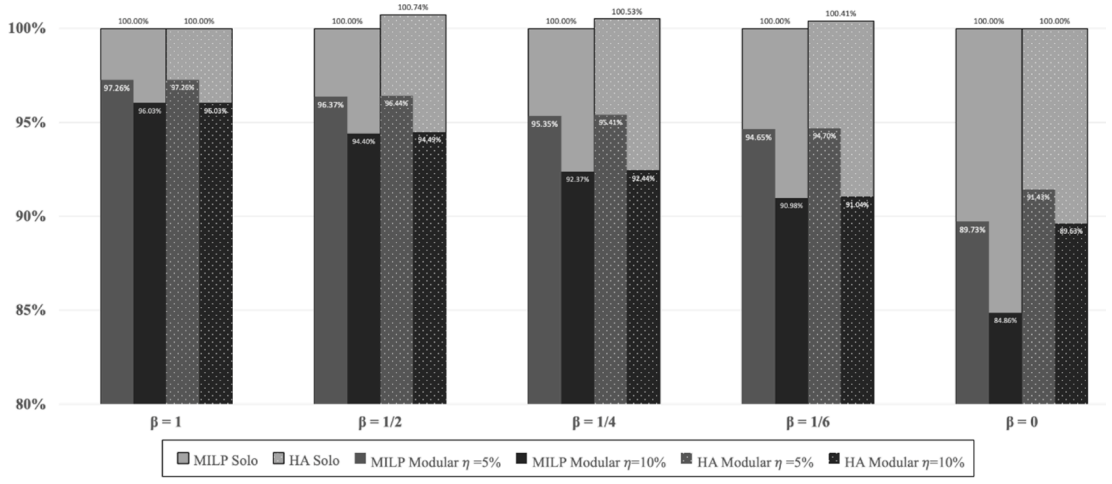


Fig. 9. Comparison of objective values under different operations and solution algorithm.

Table 4
Computation time over various scenarios, (a) Objective weight..β

Method	Mode	$\beta = 1$	$\beta = 1/2$	$\beta = 1/4$	$\beta = 1/6$	$\beta = 0$
MILP	Solo	7.03	4.31	3.08	1.75	0.08
	Modular $\eta = 5\%$	2748.98	1752.21	720.16	696.47	20.01
	Modular $\eta = 10\%$	2671.54	2097.99	1120.92	755.44	12.53
HA	Solo	0.01	0.01	0.01	0.01	0.01
	Modular $\eta = 5\%$	15.53	15.36	14.36	14.61	13.58
	Modular $\eta = 10\%$	15.40	15.07	14.56	14.45	13.91
(b) Vehicle and request numbers						
Method	**Mode**	$S1$	$S2$	$S3$		
MILP	Solo	0.86	5.11	3.78		
	Modular $\eta = 5\%$	5.41	332.78	3224.51		
	Modular $\eta = 10\%$	8.09	226.47	3760.50		
HA	Solo	0.00	0.00	0.00		
	Modular $\eta = 5\%$	9.58	12.50	21.98		
	Modular $\eta = 10\%$	9.62	12.39	22.02		

Note: all computation times are shown in seconds.

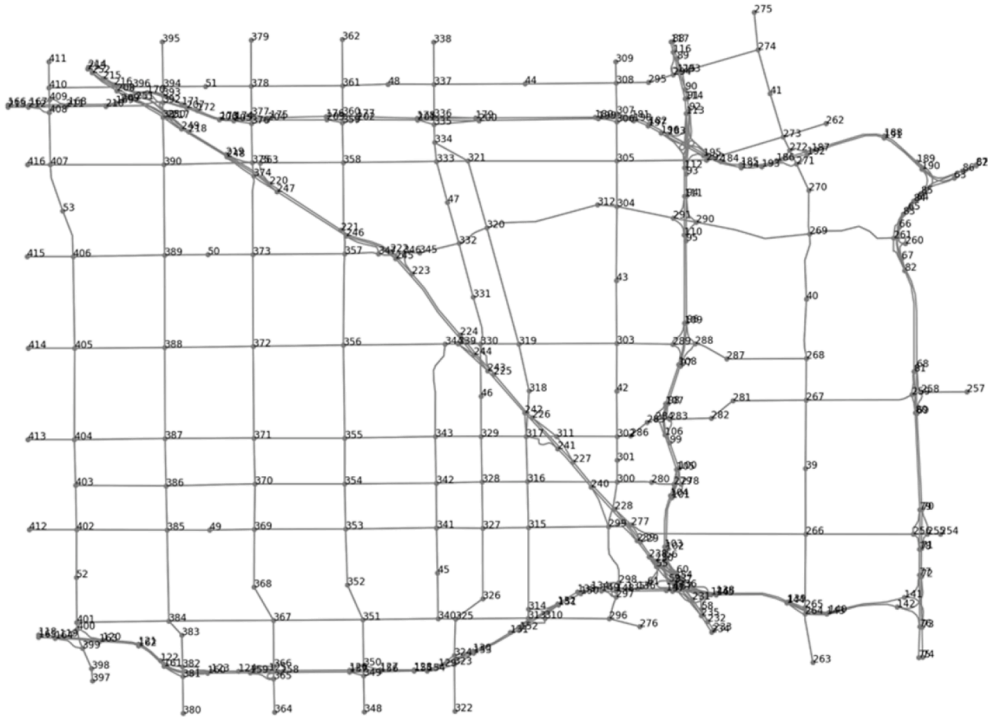


Fig. 10. Anaheim network with 378 nodes and 796 arcs.

5.2.1. Summary of results of the 320 samples

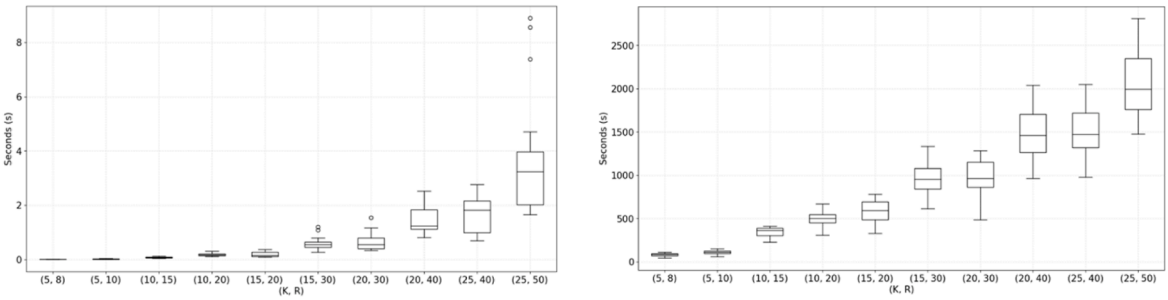
The other parameters are randomly generated among the each of the 40 scenarios to build the 320 samples. We assume that all vehicles are ready for service at $T = 0$. For the objective weight values between vehicle travel distance and request service time, we randomly select between $\beta = 1$ and $\beta = 1/3$. The maximum allowed platoon length and vehicle capacity limit are randomly selected among [4,5,6,7]. For the MV platoon saving rate η, we randomly select among [5%,6%,7%,8%,9%,10%]. The number of passengers in each request is randomly selected from $[1, 2, 3, 4]$. The set of 320 generated instances can be found on Github (Fu, 2022).

Computation times for solo (S) mode and modular (M) mode over the ten problem sizes (in terms of vehicle and request numbers) are shown in Fig. 11. The required computation time for solo mode remains within 10 s for all tests. As for modular mode, the computation time steadily increases with the problem size while our proposed heuristic algorithm can still handle 25 vehicles and 50 requests ($K = 25, R = 50$) within 1 h. By contrast, the MILP model supported by commercial solver Gurobi can barely deal with 4 vehicles and 4 requests on a 13-node network within 2 h.

The impact of spatial and temporal distribution of vehicles and requests on various performance metrics have been summarized in Table 6(b). The percentages presented in the rows of vehicle travel cost, request service time and total cost of Table 6 are computed as follows,

Table 5
Large-scale experiment settings.

Instance-specific	
Vehicle and request numbers (K,R)	(5,8), (5,10), (10,15), (10,20), (15,20), (15,30), (20,30), (20,40), (25,40), (25,50)
Objective weights	$\beta = 1$ or $\beta = 1/3$
Maximum platoon length	Randomly selected in $[4,5,6,7]$
Vehicle capacity	Randomly selected in $[4,5,6,7]$
Platoon saving rate (η)	Randomly selected in $[5\%, 6\%, 7\%, 8\%, 9\%, 10\%]$
Individual-specific	
Vehicle and request locations (Spatial distribution)	1) Uniformly randomly generated over the network (U) 2) Clustered around randomly selected 3 neighbors (C3) 3) Clustered around randomly selected 5 neighbors (C5) 4) Clustered around randomly selected 10 neighbors (C10)
Request submission times (Temporal distribution)	1) All at $T = 0$ 2) Uniformly generated over $[0,1]$ 3) Uniformly generated over $[0,4]$ 4) Uniformly generated over $[0,8]$ 5) Uniformly generated over $[0,16]$
Passenger number in each request	Randomly selected in $[1,2,3,4]$



(a) Solo mode (b) Modular mode

Fig. 11. Computation time of solo and modular mode.

Table 6
Average performance metrics, (a) Summary statistics.

Percentage of cost differences	Mean	Standard deviation	Min	Max
Vehicle travel cost	−5.75%	12.22%	−52.04%	+43.13%
Request service time	−2.86%	6.07%	−40.88%	+10.58%
Total cost	−3.55%	4.61%	−29.40%	0.00%
Note: Number of observations = 320				

(b) Average performance metrics grouped by spatial and temporal distribution

Metrics	Spatial distribution	Temporal distribution							
	U	C10	C5	C3	[0,16]	[0,8]	[0,4]	[0,1]	0
Vehicle travel cost	+0.73%	−3.27%	−5.34%	−14.98%	−0.05%	−5.21%	−6.8%	−6.53%	−7.57%
Request service time	−1.21%	−2.98%	−3.82%	−3.44%	−8.59%	−2.38%	−2.82%	−1.52%	−1.06%
Total cost	−0.77%	−2.90%	−4.13%	−6.34%	−6.43%	−3.29%	−3.84%	−2.75%	−2.44%
Number of platoons (per 100 vehicles)	5.92	16.79	16.83	16.21	19.41	14.09	12.6	13.7	12.38
En-route transfers (per 100 requests)	2.38	3.04	3.23	3.47	8.32	3.39	2.79	1.38	1.77
Vehicles in platoon (%)	11.83%	45.06%	50.58%	62.39%	54.37%	40.57%	40.24%	41.28%	40.25%
Platoon size (avg.)	2	2.68	3.01	3.85	2.8	2.88	3.19	3.01	3.25

$$percentageofcostdifferences = \frac{Modularmode - Solomode}{Solomode} \times 100\%,$$

where a negative sign means that the modular mode performs better and the cost is less than the corresponding solo mode. From the large-scale results, we find that more clustered spatial distribution would lead to more cost savings in all aspects: vehicle travel distance, request service time, and total cost. It is also more likely to have increased platoon numbers, more en-route transfers, higher involvement of platooning vehicles, and longer platoon sizes. As for the temporal distribution, denser demand would lead to higher savings in vehicle travel cost and longer platoon sizes. However, it is preferred to have low demand density from the perspective of request service time, total cost and en-route transfer opportunities.

Among all randomly generated large-scale instances on the Anaheim network, the highest savings can reach up to 52.04% in vehicle travel cost, 40.88% in request service time, and 29.40% in total cost (Table 6(a)), which reveals the promising potentials of modular vehicle technology even under practical scenarios.

5.2.2. Quantifying effects of spatial/temporal clustering with regression

To further explore the ideal operation scenarios and identify the key parameters that impact the efficiency of modular vehicles, we estimated a multiple linear regression on the 320 samples to quantify the effects of different parameters on the total cost differences between solo and modular mode (dependent variable), where a more negative difference means modular improves more in cost. The overall regression results are summarized in Table 7(a). The $R^2 = 0.29$, which suggests that the relationship is probably more nonlinear with other effects not controlled for. However, this should not discount the statistical significance of the coefficients that we did control for.

Detailed individual contributions of statistically significant predictors are summarized in Table 7(b). Here *Spatial_C*10, *Spatial_C*5, and *Spatial_C*3 are binary independent variables and when *Spatial_C*10 = 0, *Spatial_C*5 = 0, and *Spatial_C*3 = 0, it refers to the uniform spatial distribution. Similarly, for the temporal distribution, we use *Temporal*_16 = 0, *Temporal*_8 = 0, *Temporal*_4 = 0, and *Temporal*_1 = 0 to represent the case that all request submission times are at $T = 0$. From the multiple linear regression results, we find that more clustered spatial distribution and less dense temporal distribution would lead to more savings in total cost. Moreover, smaller vehicle capacity would account for larger total cost savings, because the gap in carrying capability and service throughput between individual vehicle and connected MV platoon is more significant than larger vehicle, thus leading to higher cost in solo mode and more cost savings in modular mode. On the contrary, we do not find statistically significant effects from maximum platoon length, instance size, or saving rates on the cost savings relative to solo mode. This suggests that scaling to larger instances would not necessarily benefit or worsen platooning operations.

The results in Table 7 confirm that clustered trip patterns are more conducive to modular platooning benefits, which suggest that platooning can be applicable to providing service to dense residential areas to connect them to the local CBD. In such operations, vehicles can operate in solo mode to collect and distribute passengers and platoon together to take the long-haul trip. This type of operation is labeled as a hub-and-spoke design for MVs in Caros and Chow (2020), where it is shown to be the best design for maximizing consumer surplus compared to other operations like door-to-door microtransit or running only first/last mile access. In other words, MVs can maximize their platooning benefits by identifying major residential neighborhoods or enclaves and serving those via hub-and-spoke to the CBD and vice versa.

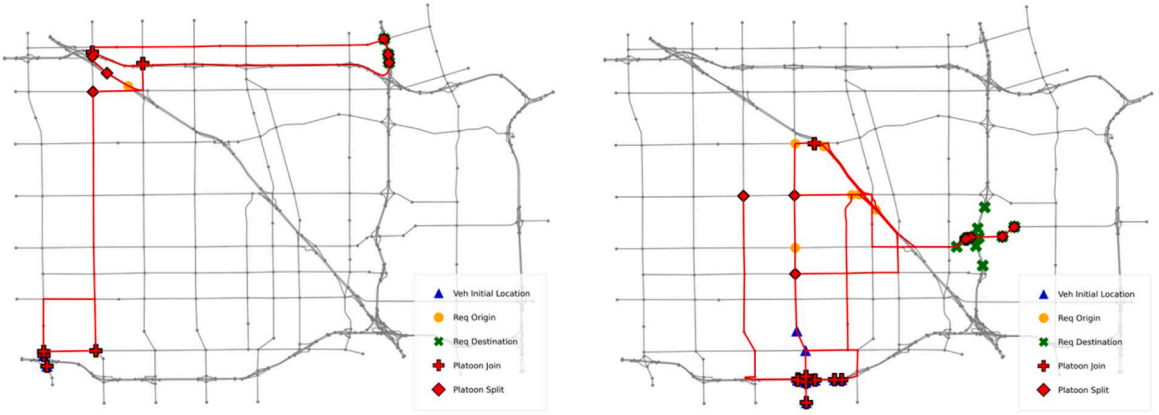
5.2.3. Illustration of largest instance under different spatial clustering

To further explore the potential of our proposed algorithm, we have conducted additional tests and found that our algorithm is able to handle up to 75 vehicles and 150 requests on the Anaheim network within the 2-hour time limit. Two large-scale examples, both with 75 vehicles and 150 requests, are visualized in Fig. 12(a) with spatial distribution *C*3, and Fig. 12(b) with spatial distribution *C*10. Vehicle initial locations, request origins and destinations, platoon join and split locations are marked on the network. For simplicity, we only highlight the paths where modular vehicles travel in platoon.

Table 7
Multiple linear regression results. (a) Model regression statistics.

Multiple R	R^2	Adjusted R^2	Standard Error	Observations
0.54	0.29	0.27	0.04	320
(b) Detailed independent variables				
Independent Variables	**Coefficients**	**Standard Error**	**t Stat**	**P-value**
*Spatial_C*10	-2.1×10^{-2}	6.28×10^{-3}	−3.34	9.36×10^{-4}***
*Spatial_C*5	-3.28×10^{-2}	6.25×10^{-3}	−5.25	2.83×10^{-7}***
*Spatial_C*3	-5.4×10^{-2}	6.24×10^{-3}	−8.65	2.69×10^{-16}***
*Temporal*_16	-3.44×10^{-2}	7.48×10^{-3}	−4.6	6.24×10^{-6}***
*Temporal*_8	-1.35×10^{-2}	8.28×10^{-3}	−1.63	0.103*
*Temporal*_4	-1.46×10^{-2}	6.41×10^{-3}	−2.28	0.024**
*Temporal*_1	-3.22×10^{-3}	6.28×10^{-3}	−0.51	0.608
Veh_Cap	6.68×10^{-3}	2.05×10^{-3}	3.27	1.2×10^{-3}***
Constant	-3.46×10^{-2}	1.36×10^{-2}	−2.55	1.13×10^{-2}**

Note: *** Significant at 0.01; ** Significant at 0.05; * Significant at 0.1. "Veh_Cap" stands for vehicle capacity.



(a) Clustered spatial distribution (C3) (b) Clustered spatial distribution (C10)

Fig. 12. Largest experiment examples: $K = 75, R = 150$.

6. Conclusion

With the ability to connect and disconnect as platoons and thus expand and reduce its capacity limit according to demand, MVs have the potential to reduce both operator cost and passenger costs in demand-responsive transit, the so-called dial-a-ride problem. The concept of MVs has attracted much research attention in recent years. However, all relevant studies on the innovative modular vehicle technology focus on individual or only combinations of vehicle platooning problem, pickup and delivery problem with transfers, or variable capacity for fixed or flexible public transit services. None of existing literature has addressed all these three major challenges together for the operation of MVs with modular platooning and en-route transfers.

In order to find the optimal assignments and routes of MVs, we first formulate a MILP model for the MDARP that integrates vehicle platooning with request pickup and delivery, considers passenger en-route transfers during vehicle platooning, and addresses the variable capacity feature of platoons at the same time. The weighted objective values of vehicle travel cost and passenger service time are optimized in the MILP model. The vehicle docking and undocking process for platoons as well as the passenger transfers are strictly synchronized between vehicles on both temporal and spatial dimensions. The model requires an undirected graph structure with multiple layers.

Since the MDARP can be simplified to a DARP which is known to be NP-hard, a Steiner tree-inspired local neighborhood search algorithm is proposed to solve large-scale problems. Based on initial feasible solo mode solutions, the proposed heuristic algorithm consists of two major steps: (1) modify the solo mode solutions and find two-vehicle MV platoons, (2) merge between feasible MV platoons and then insert individual vehicles to platoons to find multi-vehicle platoons. To validate the performance of our proposed heuristic algorithm, small-scale experiments show that our proposed heuristic algorithm can reach an average optimality gap of 0.57% with only a fraction of the required computation time against the MILP model. To further explore the potential benefits and identify the ideal operation scenarios of MVs, a set of large-scale experiments are implemented on a 378-node Anaheim network. To understand the role of different factors in benefits for platooning, we randomly construct 320 large-scale instances on this network and estimate a linear regression model with the output data. Results reveal that more clustered spatial distribution, less dense temporal distribution, and smaller vehicle capacity would lead to more savings in the total cost, while such factors as maximum platoon length, instance size, and saving rates are not statistically significant. Depending on the operational settings, using MVs can save up to 52% in vehicle travel cost, 41% in passenger service time, and 29% in total cost against existing mobility-on-demand services.

There are several future research directions that we can continue our work on. First of all, the operation performance of MVs needs to be tested and verified under more dynamic, uncertain and realistic settings with simulation-based evaluation methods. For example, we can apply a re-optimization process in which we run the model and algorithm in a rolling horizon way (e.g., every 5–15 mins), rather than trying to solve for an entire whole day with thousands of requests together. In that case, we use a dummy depot to allow vehicles to finish services at any location on the network for relocation purposes and reuse for future incoming demands. More realistic treatments can include districting the region into zones, defining hubs, or employing clustering algorithms. Second, since the modular vehicle concept is most likely deployed with electric vehicles, operators might need to consider the charging process in the operation of MVs. Third, since MVs are physically connected with each other, more innovative applications could be further explored based on features of MVs, such as mobile charging-as-a-service (Abdolmaleki et al., 2019), and integrated service of logistics and passengers (Hatzenbühler et al., 2022).

Author Contributions.

All Authors, ZF and JYJC, confirm contributions to the study conception and design, analysis and interpretation of results, and manuscript preparation of the paper. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors are partially supported by the C2SMART University Transportation Center (USDOT #69A3551747124) and NSF CMMI-2022967. The extended abstract of this paper was presented at the 15th International Conference on Advanced Systems in Public Transport (CASPT 2022).

References

Abdolmaleki, M., Masoud, N., Yin, Y., 2019. Vehicle-to-vehicle wireless power transfer: [aving the way toward an electrified transportation system. Transport. Res. Part C: Emerg. Technol. 103, 261–280.
Beasley, J., 1992. A heuristic for Euclidean and rectilinear Steiner problems. Eur. J. Oper. Res. 58 (2), 284–292.
Berbeglia, G., Cordeau, J.F., Laporte, G., 2010. Dynamic pickup and delivery problems. Eur. J. Oper. Res. 202, 8–15.
Bhoopalam, A.K., Agatz, N., Zuidwijk, R., 2018. Planning of truck platoons: A literature review and directions for future research. Transp. Res. B Methodol. 107, 212–228.
Boysen, N., Briskorn, D., Schwerdfeger, S., 2018. The identical-path truck platooning problem. Transp. Res. B Methodol. 109, 26–39.
Caros, N.S., Chow, J.Y.J., 2020. Day-to-day market evaluation of modular autonomous vehicle fleet operations with en-route transfers. Transportmetrica B: Transport Dynamics 9 (1), 109–133.
Chen, Z., Li, X., Zhou, X., 2019. Operational design for shuttle systems with modular vehicles under oversaturated traffic: Discrete modeling method. Transp. Res. B Methodol. 122, 1–19.
Chen, Z., Li, X., Zhou, X., 2020. Operational design for shuttle systems with modular vehicles under oversaturated traffic: Continuous modeling method. Transp. Res. B Methodol. 132, 76–100.
Cortés, C.E., Matamala, M., Contardo, C., 2010. The pickup and delivery problem with transfers: Formulation and a branch-and-cut solution method. Eur. J. Oper. Res. 200 (3), 711–724.
Dai, Z., Liu, X.C., Chen, X., Ma, X., 2020. Joint optimization of scheduling and capacity for mixed traffic with autonomous and human-driven buses: A dynamic programming approach. Transportation Research Part C: Emerging Technologies 114, 598–619.
Dakic, I., Yang, K., Menendez, M., Chow, J.Y.J., 2021. On the design of an optimal flexible bus dispatching system with modular bus units: Using the three-dimensional macroscopic fundamental diagram. Transp. Res. B Methodol. 148, 38–59.
Drexl, M., 2012. Synchronization in Vehicle Routing—A Survey of VRPs with Multiple Synchronization Constraints. Transp. Sci. 46 (3), 297–316.
Dumez, D., Lehuédé, F., Péton, O., 2021. A large neighborhood search approach to the vehicle routing problem with delivery options. Transportation Research Part B-Methodological 144, 103–132.
Fu, Z., Chow, J.Y.J., 2022. The pickup and delivery problem with synchronized en-route transfers for microtransit planning. Transport. Res. Part E: Logist. Transport. Rev. 157, 102562.
Fu, Z. (2022). Modular vehicle small- and large-scale test instances, https://github.com/BUILTNYU/Modular_Vehicle, last accessed Nov 28, 2022.
Garvin, W.W., Crandall, H.W., John, J.B., Spellman, R.A., 1957. Applications of linear programming in the oil industry. Manag. Sci. 3 (4), 407–430.
Glover, F., 1975. Improved Linear Integer Programming Formulations of Nonlinear Integer Problems. Manag. Sci. 22 (4), 455–460.
Guo, Q.W., Chow, J.Y., Schonfeld, P., 2018. Stochastic dynamic switching in fixed and flexible transit services as market entry-exit real options. Transportation Research Part C: Emerging Technologies 94, 288–306.
Hatzenbühler, J., Jenelius, E., Gidófalvi, G., & Cats, O. (2022). Modular Vehicle Routing for Combined Passenger and Freight Transport. *arXiv preprint arXiv: 2209.01461*.
Larsson, E., Sennton, G., Larson, J., 2015. The vehicle platooning problem: Computational complexity and heuristics. Transportation Research Part C: Emerging Technologies 60, 258–277.
Lee, W.J., Kwag, S.I., Ko, Y.D., 2021. The optimal eco-friendly platoon formation strategy for a heterogeneous fleet of vehicles. Transp. Res. Part D: Transp. Environ. 90, 102664.
Letchford, A.N., Salazar-González, J.J., 2015. Stronger multi-commodity flow formulations of the capacitated vehicle routing problem. Eur. J. Oper. Res. 244 (3), 730–738.
Li, Q., Li, X., 2022. Trajectory planning for autonomous modular vehicle docking and autonomous vehicle platooning operations. Transport. Res. Part E: Logist. Transport. Rev. 166, 102886.
Lin, J., Nie, Y., Kawamura, K., 2022. An Autonomous Modular Mobility Paradigm. IEEE Intell. Transp. Syst. Mag. 2–10.
Liu, X., Qu, X., Ma, X., 2021. Improving flex-route transit services with modular autonomous vehicles. Transport. Res. Part E: Logist. Transport. Rev. 149, 102331.
Luo, F., Larson, J., Munson, T., 2018. Coordinated platooning with multiple speeds. Transportation Research Part C: Emerging Technologies 90, 213–225.
Ma, T.-Y., Rasulkhani, S., Chow, J.Y., Klein, S., 2019. A dynamic ridesharing dispatch and idle vehicle repositioning strategy with integrated transit transfers. Transport. Res. Part E: Logist. Transport. Rev. 128, 417–442.
Mahmoudi, M., Zhou, X., 2016. Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: A dynamic programming approach based on state-space-time network representations. Transp. Res. B Methodol. 89, 19–42.
Masson, R., Lehuédé, F., Péton, O., 2013. An Adaptive Large Neighborhood Search for the Pickup and Delivery Problem with Transfers. Transp. Sci. 47 (3), 344–355.
Mitrović-Minić, S., Laporte, G., 2006. The Pickup And Delivery Problem With Time Windows And Transshipment. INFOR: Information Systems and Operational Research 44 (3), 217–227.
Nguyen, T., Xie, M., Liu, X., Arunachalam, N., Rau, A., Lechner, B., Busch, F., Wong, Y.D., 2019. Platooning of autonomous public transport vehicles: the influence of ride comfort on travel delay. Sustainability 11 (19), 5237.
Pei, M., Lin, P., Du, J., Li, X., Chen, Z., 2021. Vehicle dispatching in modular transit networks: A mixed-integer nonlinear programming model. Transport Res E-Log 147, 102240.
Pierotti, J., Theresia van Essen, J., 2021. MILP models for the dial-a-ride problem with transfers. EURO Journal on Transportation and Logistics 10, 100037.
Rais, A., Alvelos, F., Carvalho, M., 2014. New mixed integer-programming model for the pickup-and-delivery problem with transshipment. Eur. J. Oper. Res. 235 (3), 530–539.
Sethuraman, G., Liu, X., Bachmann, F.R., Xie, M., Ongel, A., Busch, F., 2019. Effects of bus platooning in an urban environment. In: In 2019 IEEE Intelligent Transportation Systems Conference (ITSC). IEEE, pp. 974–980.
Shi, X., Li, X., 2021. Operations Design of Modular Vehicles on an Oversaturated Corridor with First-in. First-out Passenger Queueing. *Transportation Science* 55 (5), 1187–1205.
Song, M., Chen, F., Ma, X., 2021. Organization of autonomous truck platoon considering energy saving and pavement fatigue. Transportation Research Part D-Transport and Environment 90, 102667.
Stabler, B. (2022). Transportation networks for research, https://github.com/bstabler/TransportationNetworks, last accessed Nov 10, 2022.

Tian, Q., Lin, Y.H., Wang, D.Z., Liu, Y., 2022. Planning for modular-vehicle transit service system: Model formulation and solution methods. Transportation Research Part C: Emerging Technologies 138, 103627.
Tirachini, A., Antoniou, C., 2020. The economics of automated public transport: Effects on operator cost, travel time, fare and subsidy. Econ. Transp. 21, 100151.
Next Future Transportation. (2022). Home: Next Future Transportation, https://www.next-future-mobility.com/, last accessed Nov 10, 2022.
Tsugawa, S., Jeschke, S., Shladover, S.E., 2016. A review of truck platooning projects for energy savings. IEEE Trans. Intell. Veh. 1, 68–77.
Wu, J., Kulcsár Selpi, & Qu, X., B., 2021. A modular, adaptive, and autonomous transit system (MAATS): An in-motion transfer strategy and performance evaluation in urban grid transit networks. Transp. Res. A Policy Pract. 151, 81–98.
Zachariasen, M., 1999. Local search for the Steiner tree problem in the Euclidean plane. Eur. J. Oper. Res. 119 (2), 282–300.
Zhang, Z., Tafreshian, A., Masoud, N., 2020. Modular transit: Using autonomy and modularity to improve performance in public transportation. Transport. Res. Part E: Logist. Transport. Rev. 141, 102033.