Car2work

Shared Mobility Concept to Connect Commuters with Workplaces

Robert Regue, Neda Masoud, and Will Recker

Over the past decade there has been a surge of shared-use mobility concepts that are redefining how people move in urban areas. In this context, a new shared-use mobility concept, Car2work, that fills the gap between the existing approaches by integration of those approaches with the transit network is proposed. Car2work differs from the traditional dynamic ridesharing approaches in the following ways: (*a*) it is designed for recurring trips; (*b*) the concept of drivers is dropped; instead, vehicles that carry at least one commuter are used; (*c*) commuters announce their trips in advance; and (*d*) multiple trips per commuter are allowed during the day. The main goal is to connect commuters with workplaces but to guarantee a trip home and offer some degree of flexibility. The proposed shared mobility system is modeled as a pure binary problem that is solved with an exact solution method. The solution method decomposes the original problem into a master problem and a subproblem, aggregating over the vehicles and reducing the number of decision variables and constraints. A link reduction strategy based on spatiotemporal constraints is also implemented to reduce the number of decision variables. Numerical experiments were performed for two scenarios. The first scenario included 10 commuters with two trips each, three workplaces, two transit stations, and 15 transfer points. The second scenario comprised 25 commuters with two trips each, four workplaces, two transit stations, and 31 transfer points. It is demonstrated that consideration of the transit network increases the matching rate and reduces vehicle costs.

Over the past decade there has been a surge of shared-use mobility concepts. As defined elsewhere, shared-use mobility is "an innovative transportation solution that enables users to have short-term access to transportation modes on an as-needed basis" (*1*).

Providers are taking different approaches to shared-use mobility alternatives. For example, ride-sourcing companies, such as Uber and Lyft, started by offering on-demand mobility but quickly launched Uberpool and Lyft Line to let their customers share rides. According to the Lyft chief executive officer, within 6 months Lyft Line became the most popular service that the company was offering in San Francisco, California (*2*). Other companies, such as Leap, Chariot, Bridj, and Via, offer an alternative to transit by providing on-demand, flexible private bus lines. ZipCar provides short-term car rentals; Carma and Zimride allow users to log their trips so that other users can find matches, and Scoop automatically creates carpools on a per trip basis.

Department of Civil and Environmental Engineering, Institute of Transportation Studies, University of California, Irvine, CA 92697-3600. Corresponding author: R. Regue, rreguegr@uci.edu.

Transportation Research Record: Journal of the Transportation Research Board, No. 2542, Transportation Research Board, Washington, D.C., 2016, pp. 102–110.
DOI: 10.3141/2542-12

Traditional vanpool and carpool services that coworkers themselves arrange can also be found.

Numerous benefits have been reported to result from such systems, including reductions in the rates of car ownership and vehicle use, increased network connectivity, and encouragement of the use of multimodality transportation (*1*). However, their long-term effects are yet to be understood. In addition, most, if not all, of the approaches presented above do not fully integrate with the existing transit network. As stated elsewhere, the effective integration of a ridesharing system with transit has the potential to increase the transit system coverage area, leading to societal and environmental benefits (*3*).

As a result, the authors propose a new shared-use mobility concept, Car2work, that has as its main goal connection of commuters with workplaces with leveraging of the line-haul capabilities of existing public transit and a guarantee of a trip back home. Car2work differs from the traditional dynamic ridesharing approaches, as it is designed for recurring trips with an emphasis on commuters, the concept of drivers is dropped and instead vehicles that carry at least one commuter are used, trips are announced in advance (hours before the trip is taken), and the users can request multiple trips for the same day. In addition, Car2work implements an automated all-or-nothing matching strategy that guarantees that all or none of the trips announced by the commuter are satisfied.

As such, Car2work integrates with the existing transit network and efficiently tackles the last mile problem that is a limiting characteristic of public transit.

When Car2work is compared with existing shared mobility services, Car2work falls between casual carpooling approaches, such as Carma or Zimride, and prearranged traditional vanpool systems. It offers more flexibility than traditional carpools, as no commitment to departure times or days of the week is required, but it is less flexible or casual than Carma or Zimride, in which users actively search for their rides. Addition of automation to the process reduces flexibility, but this reduced flexibility is compensated for by the initial targeting of recurrent (commuting) trips.

A motivation for this work comes from real-world observations (in Orange County, California) that in the types of regional development patterns that tend to dominate the postautomobile era, one of the main barriers for the use of public transit (and especially rail) is the connectivity between workplaces, homes, and rail transit stations. Rail transit offers line-haul rail service between outlying residential areas and concentrated employment centers, but this service is largely negated by the need for personalized mobility between transit stations, homes, and places of employment. An attempt to address this issue was Zev•Net, a corporate station car system launched by the University of California, Irvine, in April 2002, in cooperation with Toyota Motor Sales (*4*). However, its implementation was limited to a single station, and the full concept was never modeled from an operational perspective.

Car2work is modeled under a simulation framework that at its core relies on a variation of the peer-to-peer ride-matching problem presented elsewhere that is extended to accommodate the existing specifications (*5*). Commuters announce their trips, and the model finds the optimal trip plan, including transit connections and a guaranteed match for the return trip home. An exact solution method based on an aggregation–disaggregation algorithm is proposed.

The methodology is general enough to address a variety of scenarios, including the use of autonomous vehicles, different fleet splits, multiple transit modes, various commuter preferences, dynamic requests, and a short-term car rental service while vehicles are idling, although these scenarios are not considered here. In addition, this paper does not focus on a potential business model that will make the proposed system economically viable.

RELATED WORK

The peer-to-peer ride-matching problem is not new to the literature. Agatz et al. provided an extensive review of the dynamic ridesharing problem, detailing its characteristics and variants and the different strategies to solve the problem that have been proposed (*3*). Among the variants of the problem presented by Agatz et al., Car2work falls under the category of multiple riders, multiple drivers with the added multimodality and multihop component (*3*).

Similar problems described in the literature are the pickup and delivery problem with transfers (*6–8*) and the dial-a-ride problem with transfers (*9*). Shang and Cuff proposed a scheduling heuristic to solve the problem in the instance of a real case with 167 deliveries (*6*). The number of vehicles was also optimized, and transfers could occur at any location, for any item, and for any vehicle. Cortés et al. provided an extensive review of the existing literature of the pick-up and delivery problem, its extensions, and various solution approaches and proposed a new extension to handle transfers (*7*). The solution method was exact. It used a branch-and-cut method based on Benders' decomposition (*10*) and implemented combinatorial Benders' cuts (*11*). The largest instance proposed had six requests, two vehicles, and one transfer point. Masson et al. used an adaptive large neighborhood search to solve the pickup and delivery problem with transfers (*8*). All delivery points could be transfer points, and the larger instances had 106 requests, 24 delivery locations, and 24 transfer points and 193 requests, five delivery locations, and five transfer points. The same investigators extended the previous heuristic to solve the dial-a-ride problem with transfers (*9*). In all cases, it was shown that the transportation costs were reduced by the inclusion of transfer points. However, the added user inconvenience of transfers was not accounted for.

Transit concepts similar to the one presented here that aim to increase transit flexibility also exist, notably, flexible route transit systems (*12–14*), flexible taxi-pooling dispatching systems (*15*), or such variations of the dial-a-ride problem as the high-coverage point-to-point transit system (*16*). Herbawi and Weber proposed a multihop ride-matching problem with time windows in which drivers cooperate to bring riders to their destinations (*17*). The problem is solved by the use of a tailored genetic algorithm (*17*). With the exception of the mobility allowance shuttle transit described by Quadrifoglio et al. (*12*), which is solved by the introduction of logic cuts to a mixed-integer programming formulation on the basis of assumptions about user behavior, the examples presented above rely on the building of custom heuristics to find nearly optimal solutions. In most cases, such heuristics are used to respond to a rider's request in real time.

Car2work is a peer-to-peer ridesharing system similar to the systems described above, in the sense that it is inherently spatiotemporally sparse. This factor makes the use of the most efficient matching algorithms important, given the limited available resources. Although the use of heuristics in dial-a-ride problems is common and can yield good-quality solutions, this is not the case with the proposed system. One reason is that dial-a-ride problems have multiple drivers that work for the system and may perform pickups and drop-offs at any location and at any point in time. In addition, not all requests are concentrated in peak hours (as opposed to commuter trips), and therefore the possibility that multiple drivers will be idle when a request arrives is higher. In such a setting, a heuristic algorithm can provide a good driver assignment to a passenger, mostly on the basis of spatial proximity. The proposed system, however, uses a multitrip approach in which commuters can announce more than one trip—namely, the home-to-work and the work-to-home trips—and allows transfers (multihop) and multimodality transport. These conditions make the use of clustering heuristics impractical, as not only the spatiotemporal constraints of a single trip but also the trip connectivity constraints that in most cases span the entire day need to be met. In addition, since the demand for the proposed system is not dynamic—trips are announced well in advance—the matching does not need to happen in real time. Since detection of a match is not time sensitive, the use of heuristic algorithms will lead only to suboptimal solutions with no additional benefits. As a result, the problem is formulated as a binary problem, and it is solved by use of an aggregation–disaggregation algorithm that renders optimal solutions.

SYSTEM DEFINITION

Car2work is a mobility alternative designed for commuters with relatively regular commuting schedules. It differs from the traditional dynamic ridesharing problem described previously, in which the matching occurs on short notice, drivers are independent private entities, the system is designed for occasional or nonrecurring trips, and the trips are prearranged (*3*). The following are the main differences between dynamic ridesharing and Car2work:

1. Car2work's core is based on recurring (commuting) trips. However, it can be extended to nonrecurring, occasional trips, if such trips are added on top of existing routes, as in a dynamic ridesharing problem.
2. The concept of drivers is dropped and instead vehicles carry at least one commuter when traveling. The driver can be any commuter in the vehicle. Transit vehicles are treated differently, as they do not require a driver for the user to travel.
3. Commuters announce their trips in advance, and an automated all-or-nothing matching strategy is performed. All trips announced by the user need to be completed; otherwise, the traveler is not matched. A simple trip announcement consists of an origin, destination, earliest departure time, latest arrival time, and maximum deviation from the shortest travel time (or, alternatively, a maximum travel time budget). Commuters can announce either one or multiple trips.
4. Because of the possibility of multiple trips, the routing decision variables are indexed over trips and not commuters.

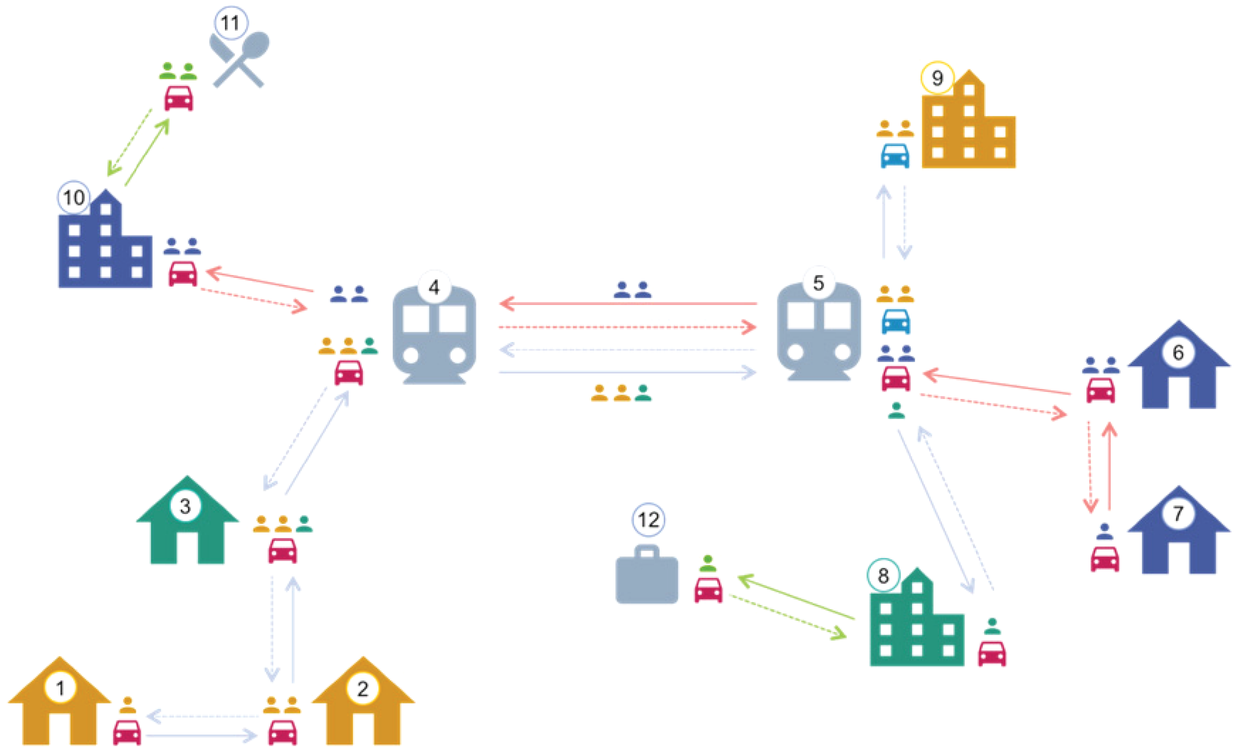

FIGURE 1 Car2work example.

Most of the points listed above can be relaxed to accommodate other types of operations. Such commuter preferences as willingness or ability to drive could be introduced, and it could be assumed that vehicles are autonomous and can travel without a driver or that not all trips announced by a user need to be accommodated.

As an example, Figure 1 shows a Car2work system with five users, three workplaces, and two transit stations. The color code indicates the commuter-to-workplace relationship, and the solid and dashed arrows represent the morning and evening commutes, respectively. A lunch and a personal business trip, corresponding to Nodes 11 and 12, respectively, are also depicted. For simplicity, this representation does not include time as a variable, and it is assumed that all the commuters have the same working schedule and preferences.

Commuters 1 and 7 have preassigned vehicles, and another vehicle is parked at Transit Station 5. The optimal solution to this problem with the objective of maximization of the number of served commuters suggests that Commuter 1 should leave home to pick up Commuter 2 and then Commuter 3 and drive to Transit Station 4. Commuters 1 and 2 work at the same workplace (Workplace 9), whereas Commuter 3 works at Workplace 8. At Transit Station 4, they all take the train toward Transit Station 5. Here, Commuters 1 and 2 take the vehicle parked to drive to their final destination, Workplace 9. Commuter 3 drives to Workplace 8 using a vehicle that Commuters 6 and 7 have parked at Transit Station 5 on their way to Workplace 10. Commuters 6 and 7 use the vehicle left by Commuters 1, 2, and 3 at Transit Station 4 to get to their workplace, Workplace 10. While the commuters are at their respective workplaces, two employees from Workplace 8 decide to use the vehicle for a lunch trip. Another employee from Workplace 8 has a business meeting at Location 12 and uses the vehicle left by Commuter 3. For the return trip home, the commuters undo what they did to get to work.

MODEL FORMULATION

The formulation presented is inspired by the peer-to-peer ride-matching problem defined elsewhere (*5*). The problem is formulated by the use of a time-expanded network and as a pure transshipment problem. Stations or nodes are homes, workplaces, transit station locations, or any other location announced by the commuters. A supply node (S_O) and a demand node (S_D) were added to the set of stations (S) in the network. Time (T) is discretized into T_n intervals of length dt between the earliest departure and the latest arrival times observed in the set of trips TS. The trip set is split into transit trips (TS_t) and commuter trips (TS_r). O_k and D_k represent the origin and destination stations, respectively, of trip $k \in TS$. Similarly, the set of vehicles V is defined and includes the vehicles available to the commuters (V_r), the transit vehicles (V_t), and a dummy vehicle (V_{dummy}). A dummy vehicle is introduced to ensure that commuters can linger during some time periods at a given station; by definition of the decision variables, a commuter must be in a vehicle at all times. This may occur with transferring situations or at the beginning or end of a trip. Each vehicle $v \in V$ has a capacity C_v. Commuters are represented by the set R. Finally, the set of links L is defined as the 4-tuple (s_i, t_i, s_j, t_j), representing a link from station s_i to station s_j departing at time interval t_i and arriving at time interval t_j. The travel time between stations s_i and s_j ($tt_{i,j}$) can be defined as $(t_j - t_i) \cdot dt$. The following subsets of L are further defined: L_t is the subset of transit links in which s_i and s_j are transit stations, L_k is the subset of feasible links for trip $k \in TS$, and L_v is the subset of feasible links for vehicle $v \in V$. The set L_{kv} is defined as $(L_k \cap L_v)$. The set of feasible links for the dummy vehicle is also defined as L'_v. The decision variables are

$$X_l^{kv} = \begin{cases} 1 & \text{if trip } k \text{ includes traveling on link } l \text{ with vehicle } v \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$X_l^v = \begin{cases} 1 & \text{if vehicle } v \text{ travels on link } l \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$Y_r = \begin{cases} 1 & \text{if commuter } r \text{ is matched} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The complete formulation of the mathematical problem is provided in Equations 4.0 to 4.11.

$$\max Z = \beta_1 \sum_{r \in R} Y_r - \beta_2 \sum_{v \in V_r} \sum_{\substack{l \in L_v: \\ i \neq j}} c_{i,j} X_l^v - \beta_3 \sum_{r \in R} \sum_{k \in TS_r} \sum_{v \in V} \sum_{l \in L_{kv}} c_{i,j} X_l^{kv} + \beta_4 \sum_{v \in V_r} \sum_{\substack{l \in L_v: \\ i = S_O, j = S_D}} X_l^v \tag{4.0}$$

where

Z = objective function value,
β_i = objective coefficients,
$c_{i,j}$ = cost of traveling on link (i, j),

and where Equation 4.0 is subject to

$$\sum_{\substack{t_i, s_i: \\ l=(s_i,t_i,s,t) \in L_v}} X_l^v = \sum_{\substack{t_j, s_j: \\ l=(s,t,s_j,t_j) \in L_v}} X_l^v \qquad \forall v \in V_r; \forall t \in T; \forall s \in S - \{S_O\} \tag{4.1}$$

$$\sum_{\substack{l=(s_i,t_i,s_j,t_j) \in L: \\ s_i = S_D, s_j = S - \{S_O\}}} X_l^v = 0 \qquad \forall v \in V_r \tag{4.2}$$

$$\sum_{l=(S_D,t_i,S_O,t_j) \in L} X_l^v = 1 \qquad \forall v \in V_r \tag{4.3}$$

$$X_{l'}^v = 1 \qquad \forall l' \in L'_v; \forall v \in V_{\text{dummy}} \tag{4.4}$$

$$X_l^v = 1 \qquad \forall l \in L_t; \forall v \in V_t \tag{4.5}$$

$$\sum_{v \in V} \sum_{\substack{l \in L_{kv}: \\ s_i = O_k}} X_l^{kv} - \sum_{v \in V} \sum_{\substack{l \in L_{kv}: \\ s_j = O_k}} X_l^{kv} = Y_r \qquad \forall r \in R; \forall k \in TS_r \tag{4.6}$$

$$\sum_{v \in V} \sum_{\substack{l \in L_{kv}: \\ s_j = D_k}} X_l^{kv} = Y_r \qquad \forall r \in R; \forall k \in TS_r \tag{4.7}$$

$$\sum_{v \in V'} \sum_{\substack{t_i, s_i: \\ l=(t_i,s_i,t,s) \in L \\ s_i \neq O_k, s_j \neq D_k}} X_l^{kv} = \sum_{v \in V'} \sum_{\substack{t_j, s_j: \\ l=(t,s,t_j,s_j) \in L \\ s_i \neq O_k, s_j \neq D_k}} X_l^{kv} \qquad \forall k \in TS_r; \forall t \in T; \forall s \in S - \{S_O, S_D\} \tag{4.8}$$

$$\sum_{\substack{k \in TS: \\ l \in L_{kv}}} X_l^{kv} \geq X_l^v \qquad \forall v \in V_r; \forall l = (s_i, t_i, s_j, t_j) \in L_v : s_i \neq s_j; s_i \neq S_O; s_j \neq S_O; s_i \neq S_D; s_j \neq S_D \tag{4.9}$$

$$\sum_{\substack{k \in TS: \\ l \in L_{kv}}} X_l^{kv} \leq C_v X_l^v \qquad \forall v \in V, \forall l \in L \tag{4.10}$$

$$X_l^v, X_l^{kv}, Y_r \in \{0, 1\} \qquad \forall v \in V, \forall l \in L, \forall r \in R, \forall k \in TS_r \tag{4.11}$$

A weighted multiobjective approach that maximizes the number of participants served (Z_1) and minimizes the total vehicle cost (Z_2) and total commuter cost (Z_3), as suggested elsewhere, was used (*18*). Furthermore, an extra term was added to minimize the fleet size (Z_4). Z_1 to Z_4 are the first to fourth terms of Equation 4.0, respectively. Given a fixed number of vehicles, those that remain unused travel directly on the link of cost zero that connects the supply and demand nodes, S_O and S_D, respectively. Inclusion of this link in the maximization of the objective function indirectly minimizes the total number of vehicles used. Determination of the optimal fleet size is a system design stage decision rather than an operational stage decision in which the exact number of vehicles may be known. The cost of traveling on link (i, j), $c_{i,j}$, is estimated by use of the travel time on the link. However, any other linear cost structure can be used. The cost of vehicles does not account for the idling time at stations, whereas in the case of commuters, the idling time is considered a part of the total travel time. The objective coefficients β_i are selected on the basis of the operational goals and the relative scale of each objective.

The constraint sets in Equations 4.1 to 4.5 deal with vehicle routing. The constraint set in Equation 4.1 imposes flow conservation on the vehicles available to the commuters V_r in all stations except supply node S_O. This forces all vehicles in V_r to depart the supply node. The constraints in Equations 4.2 and 4.3 rule out illogical flows between S_O and S_D, enforcing the constraint that vehicles cannot travel from S_D to any other station except S_O and that for every vehicle a link connecting S_D to S_O must exist. The constraint in Equation 4.4 handles dummy vehicles, and the constraint in Equation 4.5 enforces transit vehicle schedules.

The constraint sets in Equations 4.6 to 4.8 route commuters. The constraint in Equation 4.6 performs the vehicle–trip–commuter matching. If the net outflow of any origin station O_k of all trips $k \in TS_r$ is unity, commuter r is matched and all other trips in TS_r also need to occur. Similarly, if a commuter is matched, all of the destination stations of TS_r need to be reached. Together, these two constraint sets enforce the all-or-nothing matching strategy. Net flow is also used, which means that station O_k could be revisited. However, the set of links L_{kv} includes only the feasible links for that particular trip k that are constrained by spatiotemporal constraints that limit revisiting. The constraint in Equation 4.8 is the transshipment constraint on the commuters.

The constraint sets in Equations 4.9 and 4.10 are connectivity constraints between commuters and vehicles. The constraint set in Equation 4.9 ensures that all vehicles carry at least one commuter, and the constraint set in Equation 4.10 guarantees that the vehicle capacities are not violated. The formulation presented above did not include a constraint on the maximum number of vehicle transfers allowed per commuter, as this information is not readily available from the decision variables and would require introduction of a new set of decision variables. Furthermore, a distinction that depends on whether the trip includes transit or not needs to be made, given that a transit trip would automatically account for at least two vehicle transfers, unless the transit station is the workplace.

Any attempt to solve the proposed mathematical problem by the use of traditional solvers leads to impractical solution times, even for small instances of the problem. As a result, an aggregation–disaggregation algorithm is proposed.

SOLUTION APPROACH

To solve the problem in Equations 4.0 to 4.11, an iterative aggregation–disaggregation algorithm is proposed. The underlying idea is to decompose the problem into a master problem and a subproblem. In the master problem, the decision variable X_l^{kv} from the original formulation that takes care of the trip–vehicle–link assignments over v is aggregated and a new variable, X_l^{kv}, is defined as

$$X_l^k = \sum_{v \in V} X_l^{kv} = \begin{cases} 1 & \text{if trip } k \text{ includes traveling on link } l \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

By use of this aggregation, the number of decision variables and constraints in the original problem is reduced, which makes the problem more tractable. The aggregation leads to a reduction of $(|V| - 1) \cdot |L| \cdot |K|$ in the number of binary variables and $(|V_r| - 1) \cdot |L_v| + (|V| - 1) \cdot |L|$ in the number of constraints. In addition, by use of this aggregation procedure, the set of feasible links does not proportionally increase with the number of commuters or trips, since it is likely that multiple commuters share the same links in their paths.

After the aggregate problem is solved and the links on which each commuter travels are determined, the vehicle–commuter assignment needs to be recovered by solution of another problem, the subproblem. In the master problem, it is ensured that the aggregate capacity (i.e., the sum of capacities of all the vehicles that travel on that link) of each link is not violated. Since in the subproblem commuters are assigned to vehicles, the possibility exists that a solution that respects each individual vehicle's capacity does not exist. In such instances, the subproblem can become infeasible and a new master problem solution needs to be generated.

The iterative aggregation–disaggregation algorithm steps are shown in Figure 2. To start, the master problem is solved and its solution is used to solve a linear relaxation of the subproblem. If the subproblem is infeasible, a Benders' feasibility cut is found and added to the master problem. If the subproblem is feasible, the integrality of the solution is checked. If all variables are binary, the optimal solution is available; otherwise, the subproblem is solved with the integrality constraints. If the solution is feasible, the optimal solution is available; otherwise, a logical constraint is added to the master problem to eliminate this noninteger solution from the pool of feasible solutions. Iteration is performed until the convergence criteria of the algorithm are met.

Given that a constraint was added to the master problem for iteration, the algorithm is finite as the feasible region is shrinking. Therefore, either an optimal solution is reached or the entire feasible region is eliminated (implying that the problem is infeasible). If the solution to the subproblem relaxation is feasible but not binary, the subproblem is solved a second time, but this time the integrality constraints are enforced. If the solution turns out to be integral, the optimal solution is claimed to have been obtained, since the subproblem is a feasibility problem; that is, it does not have an objective function. Therefore, any integral solution that can satisfy the set of constraints in the subproblem is also optimal to the subproblem.

The following subsections define the master problem, the subproblem, and the strategy used to generate feasible links.

Master Problem

In the master problem, the decision variable that takes care of the trip–vehicle–link assignment (X_l^{kv}) is aggregated over the set of vehicles $v \in V$. As a result, the particular vehicle in which each commuter is traveling is unknown; rather, it is known that a commuter performing trip k will be traveling on a link l. The trip–vehicle assignment is retrieved in the subproblem, which is detailed in the following subsection. Equations 6.0 to 6.11 show the proposed formulation for the master problem. As in the original problem, in the master problem the four objectives described above are retained. The constraints in Equations 6.1 to 6.5 are the same as those for the original problem. The constraints in Equations 6.6 to 6.8 are equivalent to those in the original problem without the sum over v. The constraint sets in Equations 6.9 and 6.10 ensure that a link with trips must have vehicles and that the link capacity is not exceeded, respectively. The constraint in Equation 6.10, as in the original problem, is defined over every vehicle, enforcing vehicle capacity; here, under the aggregation procedure, the aggregate capacity of the link is enforced.

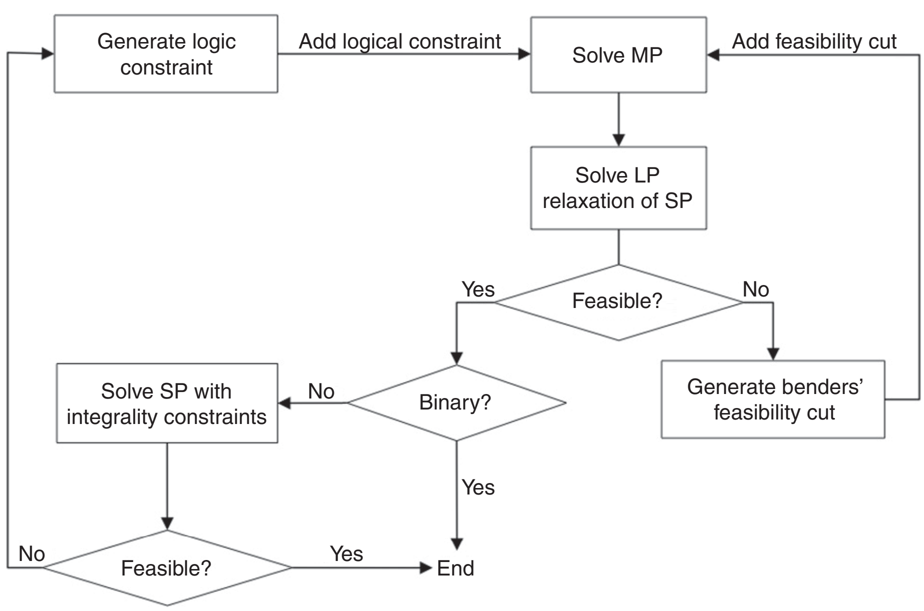

FIGURE 2 Aggregation–disaggregation iterative algorithm steps (MP = master problem; SP = subproblem; LP = linear problem).

$$\max Z = \beta_1 \sum_{r \in R} Y_r - \beta_2 \sum_{v \in V_r} \sum_{\substack{l \in L_v: \\ i \neq j}} c_{i,j} X_l^v - \beta_3 \sum_{r \in R} \sum_{k \in TS_r} \sum_{l \in L_k} c_{i,j} X_l^k + \beta_4 \sum_{v \in V_r} \sum_{\substack{l \in L_v: \\ i = S_O, j = S_D}} X_l^v \tag{6.0}$$

subject to

$$\sum_{\substack{t_i, s_i: \\ l=(s_i,t_i,s,t) \in L_v}} X_l^v = \sum_{\substack{t_j, s_j: \\ l=(s,t,s_j,t_j) \in L_v}} X_l^v \qquad \forall v \in V_r; \forall t \in T; \forall s \in S - \{S_O\} \tag{6.1}$$

$$\sum_{\substack{l=(s_i,t_i,s_j,t_j) \in L: \\ s_i = S_D, s_j = S - \{S_O\}}} X_l^v = 0 \qquad \forall v \in V_r \tag{6.2}$$

$$\sum_{l=(S_D,t_i,S_O,t_j) \in L} X_l^v = 1 \qquad \forall v \in V_r \tag{6.3}$$

$$X_{l'}^v = 1 \qquad \forall l' \in L'_v; \forall v \in V_{\text{dummy}} \tag{6.4}$$

$$X_l^v = 1 \qquad \forall l \in L_t; \forall v \in V_t \tag{6.5}$$

$$\sum_{\substack{l \in L_k: \\ s_i = O_k}} X_l^k - \sum_{\substack{l \in L_k: \\ s_j = O_k}} X_l^k = Y_r \qquad \forall r \in R; \forall k \in TS_r \tag{6.6}$$

$$\sum_{\substack{l \in L_k: \\ s_j = D_k}} X_l^k = Y_r \qquad \forall r \in R; \forall k \in TS_r \tag{6.7}$$

$$\sum_{\substack{t_i, s_i: \\ l=(t_i,s_i,t,s) \in L_k \\ s_i \neq O_k, s_j \neq D_k}} X_l^k = \sum_{\substack{t_j, s_j: \\ l=(t,s,t_j,s_j) \in L_k \\ s_i \neq O_k, s_j \neq D_k}} X_l^k \qquad \forall k \in TS_r; \forall t \in T; \forall s \in S - \{S_O, S_D\} \tag{6.8}$$

$$\sum_{\substack{v \in V_r: \\ l \in L_v}} X_l^v \leq \sum_{\substack{k \in TS: \\ l \in L_k}} X_l^k \qquad \forall l = (s_i, t_i, s_j, t_j) \in L: s_i \neq s_j; s_i \neq S_O; s_j \neq S_O; s_i \neq S_D; s_j \neq S_D \tag{6.9}$$

$$\sum_{\substack{k \in TS: \\ l \in L_k}} X_l^k \leq \sum_{\substack{v \in V: \\ l \in L_v}} C_v X_l^v \qquad \forall l = (s_i, t_i, s_j, t_j) \in L: s_i \neq s_j; s_i \neq S_O; s_j \neq S_O; s_i \neq S_D; s_j \neq S_D \tag{6.10}$$

$$X_l^v, X_l^k, Y_r \in \{0, 1\} \tag{6.11}$$

Subproblem

The subproblem is shown in Equations 7.0 to 7.4. The subproblem is a feasibility problem, and the objective function is constant, as all the terms in the original objective function can be written in terms of the variables defined in the master problem, which thus become constant values on the subproblem. The parameters $U_l^k = X_l^{k*}$ and $U_l^v = X_l^{v*}$ are also defined, where X_l^{k*} and X_l^{v*} are the optimal solutions to the master problem.

The constraint in Equation 7.1 retrieves the vehicle–trip assignment. The constraint in Equation 7.2 imposes the condition that the vehicles available to the commuters (V_r) cannot ride alone, and the constraint in Equation 7.3 is the vehicle capacity constraint.

$$\min Z_{\text{SP}} = 0 \tag{7.0}$$

where Z_{SP} is Z for the subproblem and Equation 7.0 is subject to

$$\sum_{\substack{v \in V: \\ l \in L_{kv}}} X_l^{kv} = U_l^k \qquad \forall k \in TS, \forall l = (s_i, t_i, s_j, t_j) \in L_k: s_i \neq s_j \tag{7.1}$$

$$\sum_{\substack{k \in TS: \\ l \in L_{kv}}} X_l^{kv} \geq U_l^v \qquad \forall v \in V_r, \forall l = (s_i, t_i, s_j, t_j) \in L_v: s_i \neq s_j, s_i \neq S_O, s_j \neq S_O, s_i \neq S_D, s_j \neq S_D \tag{7.2}$$

$$\sum_{\substack{k \in TS: \\ l \in L_{kv}}} X_l^{kv} \leq C_v U_l^v \qquad \forall v \in V, \forall l = (s_i, t_i, s_j, t_j) \in L_v: s_i \neq s_j, s_i \neq S_O, s_j \neq S_O, s_i \neq S_D, s_j \neq S_D \tag{7.3}$$

$$X_l^{kv} \in \{0, 1\} \tag{7.4}$$

Link Reduction Strategy

Each trip is constrained by the physical location of the origin and destination points and such temporal constraints imposed by the commuter as the earliest departure time (ED), the latest arrival time (LA), and the maximum travel time budget (tt_B). Figure 3 depicts the definition of the departure and arrival time windows between any given pair of nodes s_i and s_{i+1} and the time windows that are used to find feasible links between these two nodes, including travel links and lingering links.

For each trip announced, N shortest paths P_k are found such that either N is larger than a maximum number of paths allowed, N_M (e.g., $N_M = 1{,}000$), or the travel time of the nth path is larger than the travel time budget (tt_B) defined by the commuter for that particular trip k.

Given a feasible path p_n in P_k, for each node s_i in p_n, the earliest arrival time ED_i and the latest arrival time LA_i to that node can be further defined on the basis of the minimum travel time between nodes and the commuter travel time budget for the entire trip. Within the time window in which the commuter can reach a node, the commuter may travel to the next node or linger, as long the commuter does not violate the time constraints to reach the next node. In a particular case, two consecutive nodes, s_i and s_{i+1}, belong to the set S_t. If this occurs, the transit trips between s_i and s_{i+1} that depart during the available time window are added. In addition, the number of nodes visited during a trip is limited to maxNodes. This process is repeated for the N paths in

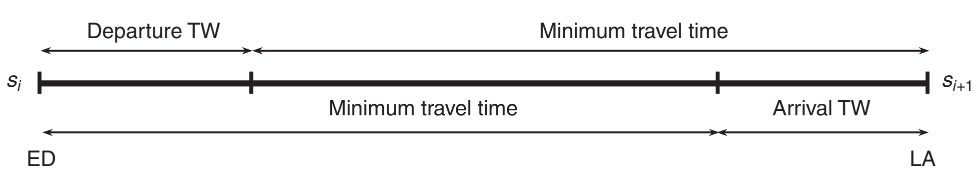

FIGURE 3 Time window (TW) definition.

P_k. The final set of links L_k for trip k is the union of the traveling and lingering links. The logic used to generate L_k is shown below.

for each trip k in TS_r:
find N shortest paths $P_k = \{p_1, \dots, p_n, \dots, p_N\}$ until $N > N_M$ or $tt(p_n) > tt_B$
for each p_n in P_k:
if length(p_n) > maxNodes:
remove p_n from P_k
continue
else:
for each node s_i in $p_n = \{s_1, \dots, s_i, \dots, s_M\}$:
add travel links $L(s_i, t_i(:), s_{i+1}, t_{i+1}(:))$ to L_k
add lingering links $L(s_i, t_i(:), s_i, t_i(:) + 1)$ to L_k

The set of links on which vehicles v can travel, denoted by L_v, is the union of the set of trip links, L_k, $\forall k \in K$; the set of idling links, L_{id} (which is formed on the basis of the set of nodes that are visited by that vehicle); and links L_O and L_D, which connect supply and demand nodes S_O and S_D together, respectively, and to all other nodes. The sets L_O and L_D are generated by use of the same concepts of available time windows in which vehicles can reach nodes; an extra link, $L(S_O, t_0, S_D, t_0)$, was added to account for the unused vehicles. To impose transit schedules, for each transit vehicle the set of links corresponding to that transit schedule was defined. Although in the current approach the same set of links L_v is assigned to all vehicles in V_r, one could heuristically force vehicles in V_r to travel on only a subset of links on the basis of the spatial distribution of homes and workplaces. For example, following from the example described in Figure 1, the range of one vehicle could be reduced to all links that connect nodes {1,2,3,4,10,11}.

NUMERICAL TESTS

Data

Two different randomly generated scenarios were used. The first scenario (SC1) has 10 commuters with two trips each, three workplaces, and two transit stations. Given that each location can be a transfer point, this scenario has a total of 15 transfer points. Locations are GPS coordinates randomly sampled from a region near the Santa Ana Metrolink Transit Station and the Irvine Metrolink Transit Station, both of which are located in Orange County, California. Workplaces are located at the station opposite to where the commuter resides to simulate the potential of rail transit to service commute trips under the proposed system. The travel time matrix is computed by use of the great circle distance between two coordinates.

The second scenario (SC2) has 25 commuters with two trips each, four workplaces, and two transit stations. This scenario has 50 requests and 31 transfer points. A larger instance that could also be solved to optimality, SC3-1, was also tested. SC3-1 has 30 users (60 requests), but transit was not considered.

For each scenario, different instances with various values of the system parameters were tested. Transit speed, transit frequency, commuter distribution, and overlapping trips were modified. The list of parameters is provided in Table 1, in which the first value listed is the default value. To encourage transit use, a higher average speed of 25 mph was set on transit links, whereas 15 mph was assumed for regular vehicles. Two trips were defined for each user: home to work and work to home. The travel time budget was assumed to be either 10% or 20% more than the travel time of the shortest path.

TABLE 1 Parameter Settings

Parameter	Values	Units
Time interval (dt)	5	Minutes
Maximum number of paths allowed (N_M)	1,000	
Maximum number of nodes in path (maxNodes)	10	
Travel time budget (tt_B)	1.1, 1.2	
Commuter vehicle speed	15	Miles per hour
Transit vehicle speed	25, 5	Miles per hour
Transit frequency	5, 15	Minutes
Balanced requests	1, 0	
Commuter vehicle capacity (C_v)	5	
Transit vehicle capacity (C_v)	100	
Commuter distribution	Clustered, random	
β_1	3 • ceil(UB/100) • 100	
β_2	1	
β_3	1, 0	
β_4	0, $\beta_1/2$	
Gurobi setting: mipgap	Default	
Gurobi setting: mipgapabs	Default	
Solver setting: timelim	9,000	Seconds

Two different parameters were also introduced to measure the impact of the spatial distribution of homes and the temporal distribution of trips. Setting of balanced requests to unity indicates that the overlap of all commuter trips according to departure time windows was perfect. Otherwise, when balanced requests were set to zero, the trip departure times were randomly set within a predefined time window. Setting of the commuter distribution to clustered means that all commuters that work at the same workplace are near the same transit station. If it is set to random, commuters are randomly assigned to workplaces.

In the setting of the utility coefficients (β_i), a strategy that prioritizes the matching of users rather than the cost and vehicle usage minimization was used. The underlying idea is to set β_1 larger than the upper bound (UB) on the commuters' and vehicles' travel time objective.

Results

Initial data inputs were built by the use of MATLAB, and for the iterative algorithm, AMPL with Gurobi as a solver was used.

Table 2 summarizes the results of various instances for each scenario. The number of vehicles was fixed to four for SC1 and to 10 for SC2. These values correspond to 40% of the total number of commuters. The introduction of the transit network reduced the total vehicle cost and increased the number of matched users. For example, in SC2, the number of matched users increased from 14 to 18, an increase that was achieved even with the suboptimal solution obtained when transit was considered. In terms of the transit frequency, in SC1 an increase in transit frequency did not affect commuter travel times. For CPU time, the use of transit increased the computational time by a factor of 4 for SC1, and computational time was also affected by transit fre-

TABLE 2 Experimental Results

Instance	Balanced Requests[a]	Commuter Distribution	Transit Use[a]	Transit Frequency	tt_B	CPU Time	Number of Vehicles Used	Number of Matched Users	Number of Links	Z_1	Z_2	Z_3	Z_4
SC1-1	1	Clustered	1	5	1.1	123	4	10	4,503	−8,762	−9,000	41	197
SC1-2	1	Clustered	0	5	1.1	32	4	10	4,382	−8,686	−9,000	82	232
SC1-3	1	Clustered	1	15	1.1	36	4	10	4,439	−8,764	−9,000	38	198
SC1-4	0	Clustered	1	15	1.1	737	4	10	6,076	−8,741	−9,000	54	205
SC2-1	0	Random	0	5	1.1	5,329	10	14	7,270	−20,694	−21,000	139	167
SC2-2	0	Random	1	5	1.1	9,060[b]	10	18	7,370	−26,620	−27,000	109	271
SC2-3	0	Clustered	0	5	1.1	545	10	13	8,598	−15,305	−15,600	134	161
SC2-4	0	Clustered	1	5	1.1	9,063[b]	10	18	8,696	−21,241	−21,600	111	248
SC2-5	1	Clustered	0	15	1.1	1,044	10	14	6,514	−16,511	−16,800	105	184
SC2-6	1	Random	0	15	1.1	9,042[b]	10	15	6,054	−22,107	−22,500	167	226
SC3-1	0	Clustered	0	5	1.1	863	10	14	10,947	−20,744	−21,000	110	146

NOTE: Z_1 = number of participants; Z_2 = time; Z_3 = time; Z_4 = number of vehicles.
[a]1 = true, 0 = false.
[b]The optimal solution was not found.

quency. This increase was due to the increase in the number of links and the fact that every transit trip added a new vehicle into the problem, which increased the number of decision variables. For SC2-2 and SC2-4, both of which included transit trips, an optimal solution could not be found in less than 9,000 s, but the solutions found after 9,000 s were better than the equivalent alternative with no transit.

Another important factor to consider is the travel time budget. If the travel time budget is set to a larger value, the solution time increases. This increase is due to the larger number of feasible links. A general pattern that can be observed in the solutions involving transit is for a vehicle to pick up commuters, drop them at the transit station, and wait for the incoming commuters to bring them to the desired workplace. This represents an ideal scenario in which all commuters have the same schedule and are grouped together by workplace.

CONCLUSIONS AND FUTURE WORK

Building on the concept of shared mobility, the authors have proposed a new mobility system that has as its main goal the connection of commuters with workplaces and a guarantee of a trip home and that is integrated with the existing transit network. It differs from the traditional dynamic ridesharing approaches because of the focus on nonrecurring trips and the automated all-or-nothing matching strategy.

A formulation of the problem was presented as a pure binary problem that is solved by use of an aggregation–disaggregation algorithm that renders optimal solutions. The original problem was decomposed into a master problem and a subproblem, in which in the master problem the variable that assigns trips–vehicles–links was aggregated over the vehicles, which reduced the number of decision variables and constraints. To recover the initial solution, a feasibility problem was solved. With this approach, various instances of the problem could be solved in a reasonable amount of time. The numerical results indicate that integration of transit increases the matching rate and reduces vehicle costs. Even though in some larger instances in which transit was considered an optimal solution could not be found in less than 9,000 s, the suboptimal solution was better than the optimal solution obtained without consideration of the transit network.

Future efforts should focus on a reduction of solution times and simulation of the mobility impacts of a large-scale implementation of the concept. A heuristic could be proposed to solve the master problem, or, in a preprocessing step, vehicles could be preassigned to commuters, which would fix some of the decision variables. Transfer points could be limited to a subset of the stations, such as workplaces or transit stations only. For a large-scale implementation of the concept, on the basis of observations from the numerical tests, a cluster first, route second approach, in which commuters are grouped on the basis of spatiotemporal proximity before the optimization problem is solved, could be proposed.

ACKNOWLEDGMENTS

This research has been made possible, in part, by grants from the University of California Transportation Center, the Balsells-Generalitat de Catalunya Fellowship, and the Fundación Caja Madrid Fellowship. Their support is gratefully acknowledged.

REFERENCES

1. Shaheen, S. A., and N. Chan. *Mobility and the Sharing Economy: Impacts Synopsis.* Transportation Sustainability Research Center, University of California, Berkeley, 2015.
2. Terdiman, D. Lyft CEO Says Lyft Line Now Accounts for Majority of Rides in San Francisco. *VentureBeat,* 2015. http://venturebeat.com/2015/03/16/lyft-ceo-says-lyft-line-now-accounts-for-majority-of-rides-in-san-francisco/. Accessed March 2015.
3. Agatz, N., A. Erera, M. Savelsbergh, and X. Wang. Optimization for Dynamic Ride-Sharing: A Review. *European Journal of Operational Research,* Vol. 223, No. 2, 2012, pp. 295–303.
4. *Irvine Transportation Network.* City of Irvine, Irvine, Calif. http://www.cityofirvine.org/cityhall/pw/itn_new/transit/routes/irvine.asp. Accessed June 2014.
5. Masoud, N., and R. Jayakrishnan. A Decomposition Algorithm to Solve the Multi-Hop Peer-to-Peer Ride-Matching Problem. Presented at 94th Annual Meeting of the Transportation Research Board, Washington, D.C., 2015.
6. Shang, J. S., and C. K. Cuff. Multicriteria Pickup and Delivery Problem with Transfer Opportunity. *Computers & Industrial Engineering,* Vol. 30, No. 4, 1996, pp. 631–645.

7. Cortés, C. E., M. Matamala, and C. Contardo. The Pickup and Delivery Problem with Transfers: Formulation and a Branch-and-Cut Solution Method. *European Journal of Operational Research,* Vol. 200, No. 3, 2010, pp. 711–724.
8. Masson, R., F. Lehuede, and O. Peton. An Adaptive Large Neighborhood Search for the Pickup and Delivery Problem with Transfers. *Transportation Science,* Vol. 47, No. 3, 2013, pp. 344–355.
9. Masson, R., F. Lehuede, and O. Peton. The Dial-a-Ride Problem with Transfers. *Computers & Operations Research,* Vol. 41, 2014, pp. 12–23.
10. Benders, J. F. Partitioning Procedures for Solving Mixed-Variables Programming Problems. *Numerische Mathematik,* Vol. 4, No. 1, 1962, pp. 238–252.
11. Codato, G., and M. Fischetti. Combinatorial Benders' Cuts for Mixed-Integer Linear Programming. *Operations Research,* Vol. 54, No. 4, 2006, pp. 756–766.
12. Quadrifoglio, L., M. M. Dessouky, and F. Ordóñez. Mobility Allowance Shuttle Transit (MAST) Services: MIP Formulation and Strengthening with Logic Constraints. *European Journal of Transportation and Infrastructure Research,* Vol. 185, No. 2, 2008, pp. 387–391.
13. Li, X., and L. Quadrifoglio. Feeder Transit Services: Choosing Between Fixed and Demand Responsive Policy. *Transportation Research Part C,* Vol. 18, No. 5, 2010, pp. 770–780.
14. Qiu, F., W. Li, and J. Zhang. A Dynamic Station Strategy to Improve the Performance of Flex-Route Transit Services. *Transportation Research Part C,* Vol. 48, 2014, pp. 229–240.
15. Lee, K.-T., D.-J. Lin, and P.-J. Wu. Planning and Design of a Taxipooling Dispatching System. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1903,* Transportation Research Board of the National Academies, Washington, D.C., 2005, pp. 86–95.
16. Cortés, C. E., and R. Jayakrishnan. Design and Operational Concepts of High-Coverage Point-to-Point Transit System. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1783,* Transportation Research Board of the National Academies, Washington, D.C., 2002, pp. 178–187.
17. Herbawi, W., and M. Weber. Modeling the Multihop Ridematching Problem with Time Windows and Solving It Using Genetic Algorithms. Presented at 2012 IEEE 24th International Conference on Tools with Artificial Intelligence, Athens, Greece, 2012.
18. Agatz, N., A. Erera, M. Savelsbergh, and X. Wang. *Sustainable Passenger Transportation: Dynamic Ride-Sharing.* ERIM Report Series Research in Management. Erasmus Research Institute of Management, Rotterdam, Netherlands, 2010.

The Standing Committee on Emerging and Innovative Public Transport and Technologies peer-reviewed this paper.