Service Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Frontiers in Service Science: Ride Matching for Peer-to-Peer Ride Sharing: A Review and Future Directions

Amirmahdi Tafreshian, Neda Masoud, Yafeng Yin

informs
http://pubsonline.informs.org/journal/serv

SERVICE SCIENCE
Vol. 12, No. 2-3, June–September 2020, pp. 44–60
ISSN 2164-3962 (print), ISSN 2164-3970 (online)

Frontiers in Service Science: Ride Matching for Peer-to-Peer Ride Sharing: A Review and Future Directions

Amirmahdi Tafreshian,[a] Neda Masoud,[a] Yafeng Yin[a]

[a] Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, Michigan 48109

Contact: atfresh@umich.edu, https://orcid.org/0000-0003-1175-0707 (AT); nmasoud@umich.edu, https://orcid.org/0000-0002-6526-3317 (NM); yafeng@umich.edu, https://orcid.org/0000-0003-3117-5463 (YY)

Received: March 23, 2020
Revised: May 7, 2020
Accepted: May 18, 2020
Published Online in Articles in Advance: September 25, 2020

https://doi.org/10.1287/serv.2020.0258

Abstract. As a consequence of the sharing economy attaining more popularity, there has been a shift toward shared-use mobility services in recent years, especially those that encourage users to share their personal vehicles with others. To date, different variants of these services have been proposed that call for different settings and give rise to different research questions. Peer-to-peer (P2P) ride sharing is one such service that provides a platform for drivers to share their personal trips with riders who have similar itineraries. Unlike ride-sourcing services, drivers in P2P ride sharing have their own individual trips to make and are not driving for the sole purpose of serving rider requests. Unlike traditional carpooling, P2P ride sharing can serve on-demand and one-time trip requests. P2P ride sharing has been identified as a sustainable mode of transportation that results in several individual and societal benefits. The core of a P2P ride-sharing system is a ride-matching problem that determines ride-sharing plans for users. This paper reviews the major studies on the operations of P2P ride-sharing systems, with a focus on modeling and solution methodologies for matching, routing, and scheduling. In this paper, we classify ride-sharing systems based on their operational features and review the existing methodologies for each class. We further discuss a number of important directions for future research.

History: S. Benjaafar served as the editor-in-chief for this article. This paper was accepted through the *Service Science*: Frontiers review process.
Funding: N. Masoud was partially supported by grants from the U.S. DOT Center for Connected and Automated Transportation [Grant 69A3551747105], the National Science Foundation [Grant 1831347], and the Michigan Institute for Data Science. Y. Yin was partially supported by the National Science Foundation [Grants CMMI-1740865 and CMMI- 1854684].

Keywords: peer-to-peer ride sharing • dynamic ride sharing • carpooling • ride-matching problem

1. Introduction

The sharing economy is a business model that leverages peer-to-peer (P2P) transactions to generate value. The past decade has witnessed widespread access to the internet and the proliferation of smartphones, leading to an unprecedented growth in businesses that are built on the foundation of the sharing economy. The mobility market has implemented the sharing economy model in many forms, including but not limited to car sharing, ride sourcing, carpooling, taxi sharing, ride sharing, bike sharing, and scooter sharing (Shaheen and Cohen 2007, Shaheen et al. 2010, Shaheen and Chan 2016, Benjaafar et al. 2017, Benjaafar and Hu 2020). Among these shared-use mobility services, car sharing, ride sourcing, carpooling, taxi sharing, and P2P ride sharing are based on sharing a vehicle. These services introduce a number of benefits, the most notable of which is enhanced mobility as a result of the reduced cost of transportation. However, they may leave different footprints on the transportation system and the environment.

Car sharing is a shared-use mobility service that allows its registered users to rent a car from a nearby location for a short period of time (usually a few hours) (Shaheen et al. 2018). It differs from the conventional car rental services in that the entire process of reservation, pickup, and drop-off is self-served, and pickup/drop-off locations are more widely spread across the network, thereby making the service more accessible to customers. Car-sharing services typically operate under two different business models: one-way (or free-floating) car sharing (Weikl and Bogenberger 2013, Correia et al. 2014, Boyacı et al. 2015, He et al. 2017) and two-way (or round-trip) car sharing (Nourinejad and Roorda 2015, Chang et al. 2017, Ströhle et al. 2019). In the former case, the pickup and drop-off location of the vehicle may be different, whereas in the latter case, customers are required to return the vehicle to the same location from which it is picked up. Car sharing is fundamentally different from other shared-use mobility services that are centered on vehicle sharing, as its users take on the task of driving and do not share rides. Therefore, increased vehicle occupancy and utilization are not at the center of the expected benefits from this form of service.

Ride sourcing is an emerging shared mobility service in which the scheduled or on-demand trip requests are served by private vehicle owners (Zha et al. 2016, 2018a; Clewlow and Mishra 2017; Braverman et al. 2019; Xu et al. 2020). Drivers in ride-sourcing services serve as independent contractors with flexible, self-determined schedules. Vehicles in ride-sourcing systems, such as Uber and Lyft, may add empty miles to the transportation network as a result of cruising and deadheading. Cruising accounts for empty miles driven by a ride-sourcing vehicle when it roams the network, waiting for its next assignment, and deadheading captures the empty miles that a ride-sourcing vehicle has to travel to its next pickup location. There are variations of ride-sourcing services that are specifically designed to increase vehicle occupancy by incorporating pooling. Such variations are referred to as ride pooling, ride splitting, or ride sharing, and they may curb the empty miles of ride-sourcing vehicles to some extent; however, unless the pooling strategy is paired with an efficient matching algorithm and implemented widely, its positive impacts may remain limited (Jacob and Roet-Green 2018).

Ride sharing is fundamentally different from other shared-use mobility systems. Car sharing and ride-sourcing services, and the overwhelming majority of their variants, focus on sharing a vehicle among users. Ride sharing takes sharing to the next level, where both the vehicle and the ride are shared between two or more users of the system. Carpooling is one of the first realizations of ride sharing that facilitates sharing rides for recurrent trips among groups of peers with the same origin and/or destination or other commonalities. In traditional carpooling, also referred to as work-to-home or home-to-work carpooling, typically the origin or the destination of all trips resides in a single location, and users with similar itineraries use a single vehicle to complete their trips (Baldacci et al. 2004, Xia et al. 2015, Chou et al. 2016). Traditional carpooling often requires long-term commitment from participants and accommodates prearranged, recurrent trips. An informal version of carpooling has been practiced largely by companies to incentivize their employees to share their commutes. In order to share the common costs of transportation, users in each group often take turns serving their peers using their personal vehicles. Casual carpooling (or slugging) is another variant of this service that requires no commitment and usually occurs ad hoc at meeting points along the drivers' routes (Kelly 2007, Mote and Whitestone 2011, Shaheen et al. 2016, Cui et al. 2019). These meeting points are usually determined to allow users to take the most advantage of gaining access to high-occupancy-vehicle (HOV) lanes. The inherent uncertainty in the availability of riders at meeting points and the typical unwillingness of drivers to take detours cast a shadow of doubt on the self-sustainability of such services in practice.

Taxi sharing is another variant of ride-sharing systems that allows for pooling rides. In such systems, a predetermined number of taxis are dispatched from a single depot or multiple depots to satisfy on-demand requests (Ma et al. 2013, Hosni et al. 2014, Santi et al. 2014, Jung et al. 2016, Alonso-Mora et al. 2017). The decision making in such systems is composed of an assignment problem, where taxis are assigned to batches of riders with similar itineraries, and a rebalancing problem that routes idle taxis toward regions with a high rate of unserved riders. The main purpose of designing shared-taxi services is to alleviate a shortcoming commonly observed in traditional taxi systems (i.e., low utilization of available seats). However, similar to ride-sourcing services, this form of service may suffer from increasing vehicle miles traveled (VMT) because of deadheading.

In the last decade, the ubiquity of the internet and the proliferation of smart phones have given rise to P2P ride-sharing systems that offer nonrecurrent, on-demand services. A P2P ride-sharing system is a shared-use mobility service that provides matching, routing, and scheduling for a group of peer travelers with compatible itineraries (Agatz et al. 2011). A P2P ride-sharing system consists of an online platform in which participants register their trips ahead of time (a few hours to a few seconds prior to their trips), and a system operator matches riders and drivers and devises their itineraries.

The P2P ride-matching problem is a generalization of the dial-a-ride problem (DARP). DARP was originally designed to model paratransit systems (Healy and Moll 1995, Madsen et al. 1995, Fu 2002); however, it has evolved through several years to model the operation of ride-sourcing systems (Jaw et al. 1986, Cordeau 2006, Wang and Yang 2019). However, the P2P ride-matching problem has a fundamental difference from ride-sourcing and shared-taxi services, in that P2P ride-sharing system drivers are also customers who are willing to share their rides while completing their own trips. As a result, they typically have tight time windows and are not available during the entire time horizon. P2P ride-sharing differs from traditional or casual carpooling, as it does not require a long-term arrangement and recurrence. These distinctive characteristics have three major implications in developing matching methods for on-demand P2P ride-sharing systems: (i) for a P2P ride-sharing system to thrive, a high level of service should be provided to *both* riders and drivers; (ii) a ride-sharing system can provide the best outcomes by determining a strategically identified ratio of riders and drivers, which will depend on the system's matching strategy; and (iii) the tighter driver time windows in a ride-sharing system call for developing customized algorithms that can provide exact, or high-quality, solutions for the P2P matching problem. Given the close definition of the P2P ride-sharing problem with DARP, in this paper we also cover a number of studies in carpooling whose methodologies can be applied (with some modifications) to a static P2P ride-sharing system.

P2P ride-sharing has been shown to offer several societal and individual benefits. First, it substantially reduces the number of single-occupancy vehicles on roads, which may ameliorate the perennial traffic congestion, especially in large metropolitan areas or during the peak hours (Agatz et al. 2010, Stiglic et al. 2016). Second, it could meet the increasing demand for mobility without adding to, or even curbing, the environmental footprint of the transportation sector by utilizing the empty spaces in traveling vehicles, rather than recruiting drivers whose sole purpose is to transport passengers. Third, it can potentially reduce the cost of transportation, as the associated costs are shared among the participants sharing a ride (Chan and Shaheen 2012).

This survey reviews the recent papers that propose ride-matching algorithms for day-to-day operation of P2P ride-sharing systems. More specifically, we study different variants of the ride-matching problem, their formulations, and their solution methodologies. Note that our survey reviews the papers that focus on operational decisions made by the P2P ride-sharing system operator, including the matching, routing, and scheduling of users. Therefore, this survey does not include the study of economics (Wang et al. 2018a, Zha et al. 2018b, Bimpikis et al. 2019) and/or user equilibrium analyses (Wang et al. 2018b) involved in such systems. Compared with the existing surveys (Agatz et al. 2012, Chan and Shaheen 2012, Furuhata et al. 2013, Ordóñez and Dessouky 2017), the contributions of this paper are threefold: (i) it includes the related work that has appeared since then, (ii) it delves into details of the proposed algorithms, and (iii) it identifies new directions for research. As such, Section 2 introduces both the common and customized features of P2P ride-sharing systems that have been covered in the literature. In Section 3, we review the solution methods that are developed for different types of ride-matching problems. Finally, some missing aspects of the existing studies that can serve as potential directions for future work are discussed in Section 4.

2. Problem Definition

A P2P ride-sharing system is an online platform that operates over a transportation network during a time horizon. The system consists of a set of passengers (riders) and a set of drivers, denoted by $\mathcal{R}$ and $\mathcal{D}$, respectively. Together, they form the set of participants, $\mathcal{P}$. A participant $p \in \mathcal{P}$ shares with the system the origin and destination of his or her trip, denoted by u_p and v_p, respectively, and his or her travel time window, denoted by $w_p = [t_p^{\text{dep}}, t_p^{\text{arr}}]$, where t_p^{dep} and t_p^{arr} denote the participant's desired earliest departure time and latest arrival time. Despite these common features, ride-sharing systems have been characterized by a variety of customized features that will be discussed in more detail in the following subsections.

2.1. System Type

The ride-sharing system can be designed in various ways. Initially, most ride-sharing systems were static systems, in which all participants were required to provide their trip information in advance (Baldacci et al. 2004, Calvo et al. 2004, Xia et al. 2015, Chou et al. 2016). Over time, the emergence of advanced telecommunication technologies as well as high-performance computation turned the dynamic systems into a viable option in practice. In a dynamic ride-sharing system, participants enter the system over time and expect a response from the system immediately or at short notice. Hence, unlike the static version, not all trip information is available at time 0. In order to handle this issue, many studies adopt a rolling-horizon framework to optimize the ride-matching problem under imperfect information (Agatz et al. 2011, 2012; Nourinejad and Roorda 2016; Masoud and Jayakrishnan 2017a; Tafreshian and Masoud 2020). In this framework, a static version of the problem is reoptimized frequently given the available trip announcements. Furthermore, all matched trips will not be finalized and can enter the next optimization problem (as long as they do not expire) to account for the potential trips that become available in the future. Conducting several experiments under different scenarios of the matching objective function (Najmi et al. 2017) determined the best policy regarding postponing trips. They suggest that postponing confirmation of trips as late as possible is the best policy to maximize vehicle miles travelled savings. Also, Agatz et al. (2012) suggest more frequent optimizations under the as-early-as-possible policy and less frequent optimizations in case of the as-late-as-possible policy.

A few studies adopt an event-based approach to optimize the dynamic system (Di Febbraro et al. 2013, Pelzer et al. 2015, Masoud and Jayakrishnan 2017b). In this approach, the arrival of a passenger triggers the ride-matching problem to reoptimize. This approach is faster but may result in inferior outcomes, because considering a few requests at a time decreases the likelihood of successfully pooling the requests. Needless to say, delaying decision making to collect multiple requests results in higher response times, which could reduce the perceived level of service (LoS).

2.2. Trip Time Windows

As mentioned earlier, having a rather tight time window is a common feature of all P2P ride-sharing systems. A ride-sharing system may require that the participants specify both t_p^{dep} and t_p^{arr}, or only one of them, having the other one set by the operator based on a target LoS criterion. Most of the existing ride-matching problems treat time windows as hard constraints and ensure that both the pickup and drop-off times fall into the ranges specified by the time windows (Baldacci et al. 2004; Agatz et al. 2011; Amey 2011; Masoud and Jayakrishnan 2017a, b; Lloret-Batlle et al. 2017; Tafreshian and Masoud 2020). However, some studies allow for small violations from the specified windows (Di Febbraro et al. 2013, Pelzer et al. 2015, Cangialosi et al. 2016). Such studies minimize the deviations from the specified time windows in the objective function of the matching problem. Although such relaxation may result in a higher rate of matching, it may also produce solutions with a lower LoS.

2.3. Objective Function

The ride-matching problem determines the matches between rider and driver participants by optimizing an objective function. This objective function could vary between minimizing cost (Baldacci et al. 2004), maximizing the total number of served passengers (Masoud and Jayakrishnan 2017a), and maximizing the total VMT savings (Agatz et al. 2012, Nourinejad and Roorda 2016, Tafreshian and Masoud 2020), among other objectives. A subset of studies consider more than one objective. Multiple objectives can be applied simultaneously by optimizing a linear combination of them (Baldacci et al. 2004; Herbawi and Weber 2011a, 2012; Chou et al. 2016; Masoud and Jayakrishnan 2017b; Bei and Zhang 2018) or in a hierarchical manner (Stiglic et al. 2015, 2016).

2.4. Matching Type

Based on the number of pickup/drop-off stops by ride-sharing vehicles and the number of rider transfers between vehicles, ride-matching problems can be divided into four categories: one-to-one, one-to-many, many-to-one, and many-to-many. One-to-one matching is the simplest form of matching used in a P2P ride-sharing system. In one-to-one matching, each vehicle transports a single passenger (or multiple passengers if they share the same origin and destination locations), and passengers are not allowed to transfer between vehicles. Because of its higher levels of convenience for both drivers and riders and its computational scalability, this problem has been widely studied in the literature. One-to-one matching problems can arise under different system configurations—namely, fixed role, flexible role, and guaranteed ride back. In a fixed role-matching problem, participants register their trips into the system with predetermined roles of riders or drivers (Agatz et al. 2010, 2011; Amey 2011; Najmi et al. 2017; Tafreshian and Masoud 2020). In a flexible role problem, a subset of participants may be able to take either the rider or the driver role (Agatz et al. 2010, Amey 2011, Tafreshian and Masoud 2020). In both configurations, the matches are made for a single trip. For a ride-sharing system that targets commuters, guaranteeing a ride back would make the system desirable to a set of users who may prefer to drive their personal vehicles if such a guarantee is not made. The one-to-one matching problem with a guaranteed ride back only matches individuals who can receive both rides (Agatz et al. 2010, Lloret-Batlle et al. 2017). Note that although commuter trips may be the most prominent beneficiaries of a ride-sharing system that guarantees two trips per a single rider, such a system may be desirable for riders who need to have guaranteed trip chains, even when their trips of interest do not share the same origin or destination locations.

More complex forms of ride matching—namely, one-to-many, many-to-one, and many-to-many—are introduced to exploit the full capacity of available seats in P2P ride-sharing systems. In the one-to-many scheme, drivers can serve multiple passengers, but passengers are not allowed to transfer between vehicles (Baldacci et al. 2004, Stiglic et al. 2015, Nourinejad and Roorda 2016). Traditional carpooling is an example of a static ride-sharing system that implements a one-to-many matching algorithm. In the many-to-one scheme, however, drivers' stops are limited to two (for one pickup and one drop-off), but passengers can transfer between vehicles (Masoud and Jayakrishnan 2017b). Finally, a many-to-many scheme constitutes the most general matching problem in which the number of vehicle stops may exceed two, and passengers may transfer between multiple vehicles (Agatz et al. 2010, Masoud and Jayakrishnan 2017a). This general many-to-many matching scheme may be used in multimodal transportation systems in which ride sharing is integrated with transit or other types of transportation systems. Because of its high computational complexity, some studies fix drivers' routes in many-to-many matching problems (Herbawi and Weber 2011a, Ghoseiri 2012). However, Masoud and Jayakrishnan (2017a) show that fixing drivers' routes can tremendously decrease the matching rate.

2.5. Operational Scheme

The task of optimizing a ride-matching problem was originally assumed to be done by a central operator who matches riders and drivers based on a global view of the entire system (Agatz et al. 2011, Amey 2011, Ghoseiri 2012). Not only does this scheme render the matching optimization problems intractable; it makes the system vulnerable to faults or cyberattacks, as in centralized ride sharing, there exists a single point of failure. With the advent of cloud computing and multiprocessor machines, some studies have diverged from the assumption of a centralized system by devising decentralized optimization schemes (Winter and Nittel 2006, Nourinejad and Roorda 2016, Najmi et al. 2017, Tafreshian and Masoud 2020). Although these approaches do not typically guarantee global optimality with respect to the available information, they give rise to a distributed optimization plan that seems necessary for real-time operations of such systems in practice.

2.6. Travel Times

In almost all studies, network travel times are assumed to be known constants. However, Long et al. (2018) were the first to address travel time uncertainty in a P2P ride-sharing system. In their study, they assumed stochastic time-independent travel times that follow a general distribution with a lower bound. They further extended their methodology to incorporate time-dependent travel times. Also, Wang et al. (2016a), Lloret-Batlle et al. (2017), and Masoud et al. (2019) considered a ride-sharing system in which travel times are different based on the load on vehicle because of the possibility of using HOV lanes. A number of studies model the ride-matching problem on a time-expanded network, on which the travel time between any origin-destination pair can be different depending on the departure time (Agatz et al. 2012; Masoud and Jayakrishnan 2017a, b). Finally, a group of studies embed the shortest-path calculations inside their real-time matching algorithms (Herbawi and Weber 2011a, b; Ta et al. 2017; Thangaraj et al. 2017; Xu et al. 2018).

3. Methodology

In this section, we review the ride-matching methodologies in a number of major studies in the P2P ride-sharing literature and its related fields. These methods are presented in the next four subsections in order of increased complexity. As such, we start with the simplest form of the ride-matching problem and end with multimodal ride matching. For the first three subsections, we further group the methods and algorithms proposed in the literature based on whether they were exact/approximation methods or heuristics. Figures 2 and 3, respectively, provide an overview of the studies with exact/approximation and heuristic methods. We finalize this section by presenting a brief discussion of the presented methodologies.

3.1. One-to-One Ride Matching

One-to-one ride-matching problems have a distinct advantage in that they can all be modeled on graphs and formulated as well-known optimization problems for which polynomial-time exact or high-quality approximation or heuristic methods exist. In the next two subsections, we discuss how these graphs can be built and solved efficiently.

3.1.1. Exact/Approximation Algorithms. Figure 1 shows the three graphs on which different configurations of one-to-one matching methods can be modelled. Let us denote the set of nodes on these graphs by V and the set of edges by E.

Figure 1(a) shows the bipartite graph on which the one-to-one matching problem with fixed roles can be modeled. This graph consists of two disjoint sets of nodes: one corresponding to the set of riders and one to the set of drivers. On this bipartite graph, an edge between a rider and a driver node exists if their trip characteristics allow for them to share a ride. A simple set of inequalities can check the existence of graph edges for any rider-driver pair—an edge between rider i and driver j exists if driver j can serve rider i within the rider's time window ($\max\{t_j^{\text{dep}} + t_{u_j,u_i}, t_i^{\text{dep}}\} + t_{u_i,v_i} \leq t_i^{\text{arr}}$) and then make it to his or her own destination within his or her own time window ($\max\{t_j^{\text{dep}} + t_{u_j,u_i}, t_i^{\text{dep}}\} + t_{u_i,v_i} + t_{v_i,v_j} \leq t_j^{\text{arr}}$). The weight w_{ij} of each

Figure 1. Graph Representation of One-to-One Matching Problems

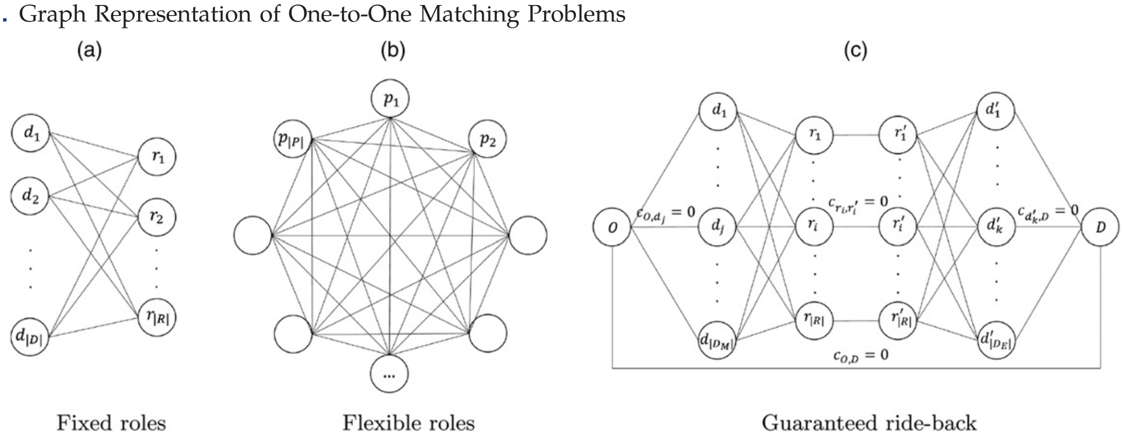
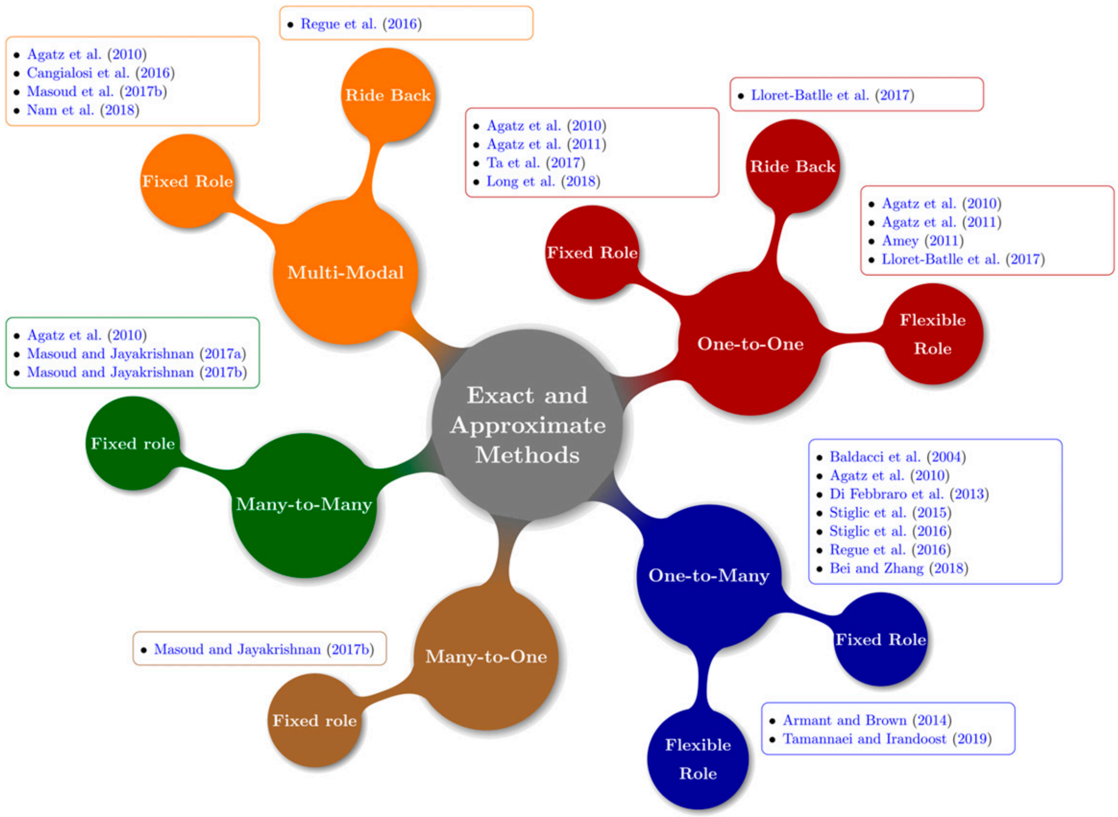


Figure 2. Studies with an Exact or Approximation Method



edge (i, j) in this graph could be set based on the objective function of the system. A solution to the max-weighted bipartite matching problem formulated on this graph would determine how riders and drivers should be assigned together. This optimization problem is presented in model (1). The decision variable x_{ij} in model (1) takes the value 1 if rider i is matched with driver j and the value 0 otherwise. Equation (1a) maximizes the weighted sum of selected edges. Inequalities (1b) and (1c) ensure that each driver and rider are matched with at most one rider and driver, respectively. Constraint (1d) relaxes the binary constraint that is implied by the definition of the decision variable x_{ij}, as a result of the totally unimodular structure of the constraint sets (1b) and (1c) and the integer right-hand-side vector in model (1).

$$\max \quad \sum_{(i,j)\in E} \sum w_{ij} x_{i,j} \tag{1a}$$

$$\text{s.t.} \quad \sum_{\substack{i\in\mathcal{R}:\\(i,j)\in E}} x_{i,j} \le 1 \qquad \forall\, j \in \mathcal{D}, \tag{1b}$$

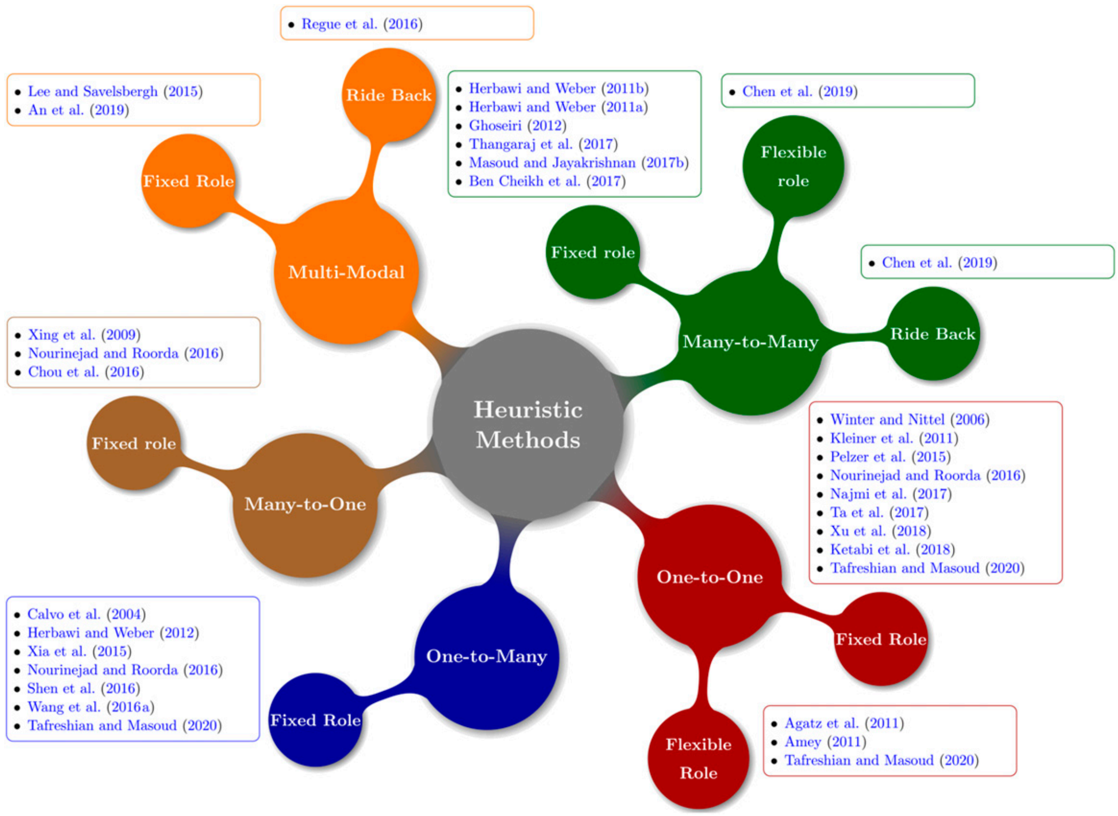
$$\sum_{\substack{j\in\mathcal{D}:\\(i,j)\in E}} x_{ij} \le 1 \qquad \forall\, i \in \mathcal{R}, \tag{1c}$$

$$x_{ij} \ge 0 \qquad \forall\, (i,j) \in E. \tag{1d}$$

Although this LP can be quickly solved using any conventional optimization software, there exists a large variety of exact and approximation algorithms for this problem with polynomial worst-case running time bounds. These algorithms may be of interest in large-scale implementations or because of their simplicity. In the case of maximum cardinality bipartite matching (i.e., where all w's are 1), Hopcroft and Karp (1973) proposed an iterative algorithm based on augmenting paths with running time complexity of $\mathcal{O}(E\sqrt{V})$. Also, a simplified version of this algorithm yields a $(1-\epsilon)$-approximation with running time complexity of $\mathcal{O}(E\epsilon^{-1})$. For the weighted bipartite matching problem, where w_{ij}'s may be different from 1, Fredman and Tarjan (1987) proposed an $\mathcal{O}(V^2 \log V + VE)$ algorithm by introducing the Fibonacci heaps to delete items from a priority queue. For integer weights, Duan and Su (2012) proposed a scaling algorithm with worst-case running time of $\mathcal{O}(E\sqrt{V}\log V)$, where V is the largest integer weight. Moreover, Gabow and Tarjan (1989) showed that a slight modification of their exact scaling algorithm yields an $(1-\epsilon)$-approximation with running time complexity of $\mathcal{O}(E\sqrt{V}\log(V/\epsilon))$.

Figure 1(b) shows the general graph on which the matching problem with role flexibility can be modelled. Each node in this graph corresponds to a

Figure 3. Studies with a Heuristic Method



participant in the ride-sharing system. An edge (i, j) between nodes i and j exists if participant i can serve as the driver for participant j, or vice versa. The same set of inequalities that were used in the case of matching with fixed roles can be used to determine the feasibility of a match between any rider-driver pair. In a pair (i, j) where both rider-driver and driver-rider role assignments are feasible, the assignment that leads to the higher edge weight can be selected as the superior assignment, based on which the edge weight is determined. In cases where there are participants who are limited to a single role, the feasibility inequalities and the edge weight will be determined according to the announced role.

Model (2) shows the mathematical formulation for the general matching problem formulated on the graph in Figure 1(b). The decision variable x_{ij} in this model takes the value 1 if participants i and j are matched and takes the value 0 otherwise. The objective function (2a) maximizes the weight of the selected matches. Constraint set (2b) ensures that each participant is assigned at most one match, and constraint (2c) enforces the decision variables to take only 0 and 1 values:

$$\max \quad \sum_{(i,j)\in E}\sum w_{ij}x_{ij} \tag{2a}$$

$$\text{s.t.} \quad \sum_{\substack{j\in\mathcal{P}:\\(i,j)\in E}} x_{ij} \le 1 \qquad \forall\, i \in \mathcal{P}, \tag{2b}$$

$$x_{ij} \in \{0, 1\} \qquad \forall\, (i, j) \in E\,. \tag{2c}$$

The blossom algorithm proposed by Edmonds (1965a, b) is the best-known algorithm for the general matching, with the running time complexity of $\mathcal{O}(V^2E)$. However, Gabow (1990) introduced a data structure that showed that this algorithm can be implemented in $\mathcal{O}(V^2 \log V + VE)$. In the case of integer weights, the scaling algorithm proposed by Gabow and Tarjan (1991) yields the running time complexity of $\mathcal{O}(E\sqrt{V \log V}\log(VN))$. They further show that this algorithm yields a running time of $\mathcal{O}(E\sqrt{V})$ for maximum cardinality matching in general graphs. Similar to the bipartite case, a simplified implementation of this scaling algorithm provides a $(1-\epsilon)$-approximation for the cardinality matching problem whose running time complexity is $\mathcal{O}(E\epsilon^{-1})$. Finally, Duan and Pettie (2014) introduced a novel scaling algorithm that leads to a $(1-\epsilon)$-approximation for the maximum weighted general matching with running time complexities of $\mathcal{O}(E\epsilon^{-1}\log\epsilon^{-1})$ for arbitrary weights and $\mathcal{O}(E\epsilon^{-1}\log N)$ for integer weights. It is worth mentioning that the trivial greedy algorithm (iteratively pick the edge of the largest weight and remove its incident edges) runs in $\mathcal{O}(E\log V)$ and yields a 1/2-approximation.

Figure 1(c) shows the graph on which the ride-sharing problem with a guaranteed ride back can be modeled. This graph has two bipartite graphs, structurally similar to the one in Figure 1(a), as subgraphs; these subgraphs represent the matches in the morning and in the evening. Additional edges are introduced to ensure that a rider is only considered matched if both his or her morning and evening trips can be served (Lloret-Batlle et al. 2017). Let us denote the set of riders in the morning and evening by $\mathcal{D}_M$ and $\mathcal{D}_E$, respectively. Note that these two sets may overlap but need not be identical. The bipartite subgraph on the left side of Figure 1(c) represents matches for the morning period; an edge (i,j) between driver $i\in\mathcal{D}_M$ and rider $j\in\mathcal{R}$ exists if the feasibility inequalities are satisfied. The bipartite subgraph on the right-hand side of this figure contains the set of riders $\mathcal{R}$, with the same ordering as the left-hand side subgraph and the set of evening drivers $\mathcal{D}_E$. An edge (i,j) between rider $i\in\mathcal{R}$ and driver $j\in\mathcal{D}_E$ exists if the feasibility inequalities are satisfied. Unlike the two previous configurations of one-to-one matching, the weight of an edge (i,j) in the graph in Figure 1(c) is the cost associated with having the edge in the solution. A zero-cost edge connects a rider in the first subgraph with its duplicate in the second subgraph. Finally, additional zero-cost edges connect an origin node O to the set of morning drivers, the set of evening drivers to a destination node D, and the node O to the node D. Let us define $\mathcal{K}=\min\{|\mathcal{D}_M|,|\mathcal{D}_E|,|\mathcal{R}|\}$, and let us define μ_{ij} as the capacity of edge (i,j). The edge (O,D) in this figure has capacity of $\mathcal{K}$, whereas all other edges have unit capacity.

A capacitated min-cost flow problem formulated on the graph in Figure 1(c) can provide the solution to the matching problem with a guaranteed ride back. The corresponding mathematical model is displayed in model (3). The decision variable x_{ij} indicates the units of flow on link (i,j). The objective function (3a) minimizes the total system cost. Constraints (3b) and (3c) send $\mathcal{K}$ units of flow out of the origin node O and to the destination node D, respectively. Constraint (3d) is the flow balance constraint. Finally, constraint set (3e) ensures that the units of flow on the edges are nonnegative and do not exceed edge capacities. Note that, similar to the problem of one-to-one matching with fixed roles, this problem enjoys a totally unimodular structure in the constraint set coefficients and an integer right-hand-side vector that prevents the need for enforcing integrality constraints on the decision variables:

$$\min \quad \sum_{(i,j)\in E}\sum c_{ij}x_{ij} \tag{3a}$$

$$\text{s.t.} \quad \sum_{j\in\{\mathcal{D}_M\cup D\}} x_{Oj}=\mathcal{K}, \tag{3b}$$

$$\sum_{i\in\{\mathcal{D}_E\cup O\}} x_{iD}=\mathcal{K}, \tag{3c}$$

$$\sum_{j\in V} x_{ij}=\sum_{k\in V} x_{jk} \qquad \forall j\in V-\{O\cup D\}, \tag{3d}$$

$$0\le x_{ij}\le \mu_{ij} \qquad \forall\ (i,j)\in E. \tag{3e}$$

The min-cost flow problem can be efficiently solved using the network simplex algorithm with running time complexity of $\mathcal{O}(\min\{(V^2E\log VC, V^2E^2\log V)\})$ due to Orlin (1997), where C is the maximum capacity of edges. An issue with this algorithm is that the running time depends on the maximum capacity. As such, Orlin (1993) proposed a strongly polynomial algorithm that is a refined version of the scaling technique proposed by Edmonds and Karp (1972). The running time complexity of this algorithm is $\mathcal{O}(E\log V(E+V\log V))$, which is independent of capacity and demand.

3.1.2. Heuristic Algorithms. For a small- or medium-scale system, different versions of one-to-one ride matching can be easily solved using conventional optimization packages. However, when a P2P ride-sharing system operates over a large network with high penetration rates, one may need to rely on decentralized approaches that may not guarantee optimality yet provide high-quality solutions in a timely manner. As such, for solving large-scale bipartite matching, Agatz et al. (2012) suggested a clustering approach based on the pickup/drop-off locations of trips. After clustering, all trips whose pickup and drop-off locations fall into the same cluster form a smaller subproblem. The resulting subproblems are independent and, hence, can be solved in parallel. This approach, however, leaves out several trips whose origin and destination do not fall in the same cluster. Najmi et al. (2017) introduced a clustering approach similar to that of Agatz et al. (2012). However, in their work, trips whose origins and destinations fall in different clusters will be included in both subproblems. In this case, subproblems are not independent anymore. Hence, after solving the subproblems in parallel, a follow-up problem must be solved for those trips that have been matched in more than one subproblem.

Ta et al. (2017) considered a variant of one-to-one matching in which the shared route percentage (i.e., the ratio of the shared route's distance to the driver's total travel distance) must be larger than a prespecified constant. They proposed an efficient approximate method to maximize the shared route percentage based on road network partitions with error bound guarantees. Ketabi et al. (2018) developed an efficient spectral clustering algorithm based on a similarity score that accounts for both spatial and temporal similarities. They further used this clustering approach to separate the trips and find the matching between riders and drivers. Tafreshian and Masoud (2020) proposed a polynomial-time graph partitioning technique that clusters trips into multiple subproblems based on a proximity measure. Their partitioning algorithm imposes a level of uniformity between subproblem sizes to minimize the solution time of ride-matching subproblems in a parallelized setting. They further provided a worst-case bound on the computation reduction as a result of partitioning. They also applied their methodology to the cases where role flexibility or multiple passengers are allowed.

Other classes of methods have been used to heuristically solve one-to-one matching problems. Agatz et al. (2011) used an iterative LP rounding heuristic to obtain a high-quality solution, though not necessarily optimal, for the flexible role-matching problem when the corresponding IP is too large. Kleiner et al. (2011) developed an agent-based algorithm based on the sealed-bid second-price auction mechanism. Their algorithm also provides a trade-off between vehicle miles traveled savings (VMTS) and matching rate. Besides an exact IP formulation, Nourinejad and Roorda (2016) proposed another agent-based algorithm that works based on a single-shot, first-price Vickrey auction. In this auction, drivers initially bid on the riders in their list, riders then pick the most suitable bid, and finally, drivers confirm the deal. In order to reduce the drivers' choice sets, they must only bid on riders whose pickup locations are close to their paths. They tailored their agent-based algorithm to cover multihop itineraries for riders and to cover multiple passengers (at most two) per vehicle.

3.2. One-to-Many Ride Matching

In this category, we review the P2P ride-sharing papers that allow multiple passengers per a driver's trip. As mentioned earlier, carpooling is a special case of the static one-to-many ride-matching problem. Therefore, our literature review in this section covers a number of major studies on carpooling.

3.2.1. Exact/Approximation Algorithms. As mentioned earlier, carpooling problems can be categorized as a one-to-many ride-matching problem. Baldacci et al. (2004) considered a home-to-work variant of this problem in which all trips share the same destination. They present a mixed integer program (MIP) based on the multicommodity flow problem as well as a set partitioning problem. They adopt a column generation approach to solve the set partitioning problem. They describe a dynamic programming procedure to generate feasible paths. The novelty of their method, however, comes from a bounding procedure that allows them to evaluate the paths efficiently and obtain a tight lower bound. Moreover, they devised a heuristic method to compute a valid upper bound. Tamannaei and Irandoost (2019) considered another variant of the home-to-work carpooling in which drivers are not known ahead of time and will be determined through optimization. They formulated this problem as an MIP, for which they developed a refined branch-and-bound (B&B) algorithm. They further introduced a beam search algorithm to find near-optimal solutions for large-scale networks.

Di Febbraro et al. (2013) formulated the one-to-many ride-matching as an MIP that models a dynamic ride-sharing system inside a discrete-event simulation framework. The arrival of a new customer triggers this optimization to be solved using CPLEX. Their formulation does not impose any constraint that prevents violating riders' time windows. However, they adopt an objective function that penalizes such violations. Another variant of this problem was studied by Armant and Brown (2014), where they developed an MIP formulation for the one-to-many ride-matching problem with flexible roles. They solved the problem using CPLEX with the addition of a set of constraints that avoid symmetrical solutions. Using numerical experiments, they showed the effectiveness of these constraints when compared with the regular branch-and-cut algorithm. By introducing the concept of meeting points, Stiglic et al. (2015) proposed a framework that matches a driver to possibly multiple riders without increasing the number of stops. They determine feasible matches in a subproblem by enumerating all possible scenarios. For multiple riders, they use a novel technique to avoid enumerating all possible cases. They adopted a lexicographic goal programming approach to maximize the VMTS for a maximal matching.

Because of the complexity of the one-to-many matching problem, approximate methods on this problem are very scarce. Bei and Zhang (2018) developed a 2.5-approximation algorithm for ride sharing with two riders per driver. They present a two-phase greedy algorithm. First, it groups $2n$ riders into n pairs based on shortest-path distances between riders, and then it assigns drivers to these n pairs based on the shortest-path distances between vehicles and the closet rider in the pair. The running time complexity of this algorithm is $\mathcal{O}(n^3)$. They do not impose a hard constraint on the travelers' time windows.

3.2.2. Heuristic Algorithms. Calvo et al. (2004) proposed a heuristic algorithm for the home-to-work carpooling problem. Their method consists of two steps: path construction and local search. For path construction, they use a greedy algorithm based on regret minimization. The local search step includes an iterative procedure composed of extraction and insertion in the neighborhood of the previous solution. Their model also makes an attempt to capture the impact of congestion by allowing different travel times during different times of the day. Xia et al. (2015) presented an IP to solve home-to-work carpooling with one vehicle serving at most C customers. They presented two heuristics based on simulated annealing and tabu search to solve this problem. These heuristics start by initial arbitrary carpooling routes and try to improve them by either changing the visited stations by a route or reordering them.

Herbawi and Weber (2012) formulated the dynamic one-to-many ride-matching problem as a pickup and delivery problem with time windows. Their method divides the day into a set of time periods. For each period, the ride-matching problem is solved using a genetic algorithm (GA) given all available requests. Upon the arrival of a new request, they execute an insertion heuristic similar to that of Jaw et al. (1986) to modify the solution obtained by the GA in real time. They also keep unmatched requests for solving the GA in the next period. For the same problem, Pelzer et al. (2015) introduced a partitioning-based matchmaking algorithm. They initially take three steps to divide the road network into a number of partitions. The arrival of a traveler invokes the matchmaking algorithm to find all travelers who are currently in that partition and whose destination partitions are in same corridor. Instead of imposing hard constraints on time windows, they choose to match travelers with higher VMTS and lower detour time. Shen et al. (2016) addressed the real-time one-to-many ride-matching problem using a filter-and-refine framework. In the filter stage, they group trips based on the spatiotemporal location of requests and vehicles. This helps them reduce the size of the matching problem. They further use a kinetic tree structure to store the current and potential schedules of drivers. In the refine stage, they exploit the information from the filter step to match riders and drivers efficiently.

3.3. Multihop Ride Matching

This subsection provides a survey of studies that allow customers to transfer between vehicles while completing their trips. This includes both many-to-one and many-to-many ride matching.

3.3.1. Exact/Approximation Algorithms. Agatz et al. (2010) were the first to formulate an MIP on a spatiotemporal network for the many-to-many ride-matching problem, but they relied on optimization packages to solve it. In their formulation, they replaced home locations with stations in the vicinity to accommodate riders who are not willing to reveal their home location because of security and/or privacy issues. Masoud and Jayakrishnan (2017a) developed another MIP formulation for the many-to-many problem based on a time-expanded network. They proposed an exact decomposition algorithm to solve this problem with the objective of maximizing the matching rate. This algorithm is an iterative process that solves a set of smaller, independent subproblems in parallel at each iteration. Until convergence, the subproblem solutions are not feasible. However, they manage to compute a lower and upper bound at each iteration, which enables them to stop the algorithm earlier with a high-quality feasible solution based on their lower bound. They further introduced a multistep preprocessing algorithm that significantly reduces the solution search space without compromising optimality. Masoud and Jayakrishnan (2017b) propose an exact method for a real-time many-to-one ride-sharing system in which customers expect an immediate response from the system. As such, they consider serving customers on a first-come, first-served (FCFS) basis. More specifically, the arrival of each customer triggers a multiobjective optimization problem that determines the rider's itinerary based on a single or multiple drivers' path. They develop an exact method based on dynamic programming to solve the optimization problem over the reduced search space. They further suggest a ride exchange policy to alleviate the impact of serving customers in an FCFS order (Masoud et al. 2017a). To the best of our knowledge, the last two studies are the only ones that developed an exact algorithm for the multihop ride-matching problem based on a rolling-horizon and an event-based approach, respectively.

3.3.2. Heuristic Algorithms. Xing et al. (2009) were among the first who considered transfer in ride-sharing systems. They developed a multiagent-based simulation framework for a highly dynamic ride-sharing system. Their method uses a number of agents—namely, PassengerAgent, DriverAgent, NodeAgent, and RoutingAgent—to explore the possibility of serving riders using one or two drivers. However, their framework assumes a capacity of one for vehicles. Chou et al. (2016) formulated a carpooling problem as a discrete optimization problem to maximize a multi-objective function that includes both system-level conditions and the participants' expectations. They developed an algorithm based on stochastic set-based particle swarm optimization (PSO).

Herbawi and Weber (2011a) studied a variant of the multihop ride-sharing problem in which drivers have fixed routes and time schedules. They formulated this problem as finding shortest paths on a spatiotemporal graph with the objective of minimizing costs, waiting times, and the number of transfers. They suggested a multiobjective ant colony optimization to solve the problem. By comparing its performance to a multiobjective GA in Herbawi and Weber (2011b), they found it crucial to apply a local search improvement on the final solution. Ghoseiri (2012) modeled the dynamic multihop ride-sharing problem as an MIP. Aside from scheduling, routing, and matching, their formulation further considers a number of personal preferences. They limited the number of transfers to two and proposed a heuristic solution that solves the problem by decomposing it into a three-level hierarchical decision-making problem. In the first level, they schedule only riders who can be served without transferring. For the next two levels, after updating the drivers' schedules from the previous level, they make decisions about riders who can be served with one (in level 2) or two (in level 3) transfers. Ben Cheikh et al. (2017) proposed a real-time multiobjective GA. By introducing some fixed points along drivers' routes, they design a dynamic chromosome coding that enables the system to transfer riders between vehicles en route. More recently, Chen et al. (2019) consider a dynamic many-to-many ride-sharing system with a ride back guarantee for promoting ride sharing among the employees of a community of companies. Their proposed system further includes meeting points for sharing a ride and role flexibility for the participants. Based on the included features, this is the most comprehensive model among the P2P ride-sharing systems that have been proposed so far. They formulate this problem as an MIP and develop a greedy heuristic that yields a high-quality solution in a timely manner. Their proposed heuristic sorts customers based on their potential for being a rider and greedily assigns others to serve them. They further show the merits of their algorithm in the existence of external mobility services.

3.4. Multimodal Ride Matching

In spite of several benefits that P2P ride-sharing systems offer to both users and the society in general, some studies question the self-sustainability of such systems in the long run (Furuhata et al. 2013). One of the main discussed issues resides in the fact that the existence of enough drivers to serve the transportation needs of riders is not guaranteed, at least during some periods of the day. To tackle this issue, a few studies in the literature recommended integrating P2P ride-sharing systems with other means of transportation in the hope of increasing the LoS. Lee and Savelsbergh (2015), for instance, proposed a dynamic ride-sharing system with the addition of a few dedicated drivers. For this system, they presented an IP to minimize the costs while guaranteeing a minimum rate of served passengers. They assumed dedicated drivers can serve multiple riders while ad hoc drivers can only serve one rider. In order to solve the IP, they presented a heuristic solution composed of a construction step and an improvement step. In the construction step, they generate a pool of initial solutions by first maximizing the served riders in the absence of dedicated drivers and then inserting unmatched riders greedily in the routes of dedicated drivers. Next, they adopted a neighborhood search to improve the initial solutions by changing the schedule of dedicated drivers and/or swiping the type of driver for a rider. Cangialosi et al. (2016) introduced a generalized ride-sharing system that improves the performance of a multihop ride-matching problem with the help of one-way car-sharing vehicles and the transit system (bus, taxi, etc.). They presented a multiobjective MIP that maximizes the matching rate while minimizing the inconvenience factor for passengers. They solved this formulation using the conventional B&B in CPLEX. For serving recurrent commuter trips, Regue et al. (2016) proposed a system that integrates static multihop ride sharing with the public transit system. They proposed a pure binary program that optimizes a multiobjective function over a time-expanded network and allows for a ride back guarantee. They proposed an exact solution methodology based on an aggregation/disaggregation algorithm. More precisely, their formulation can be decomposed into a master and a subproblem, between which the solution iterates until convergence. In the master problem, they aggregate the vehicle-commuter-link assignment variables with respect to vehicles, and the subproblem recovers the matching between commuters and vehicles. Masoud et al. (2017b) extended their presented methodology in Masoud and Jayakrishnan (2017a) to combine multihop ride sharing with metro transit system. Using a real-world case study, they showed the effectiveness of such systems in addressing the first-mile/last-mile problem faced by public transit systems. Nam et al. (2018) designed a transit feeder system that routes riders to their destinations using a variety of transportation options—namely, P2P ride sharing, bike sharing, walking, and transit. They model their proposed system using a super network that consists of multiple layers, one for each mode. The novelty of their work comes from the fact that they introduced multiple connectors that link riders' starting points to multiple transfer nodes. Using the multiple-connector scheme, they are able to maximize the matching rate while finding shorter travel-time itineraries. Finally, An et al. (2019) examine the integration of shared autonomous vehicles into a P2P ride-sharing system. They adopt a sequential framework in which they first find the riders that can be optimally served by private ride-share vehicles. Next, they find the minimum number of shared autonomous vehicles to meet the demand of unserved trips. For solving the ride-matching problem, they use the dynamic programming approach proposed by Masoud and Jayakrishnan (2017b).

3.5. Takeaways

Table 1 lists all the reviewed papers along with their key features in chronological order. This table demonstrates that in recent years, there has been a shift toward more complex forms of ride matching (i.e., multiple passengers, transfers, and modes) with a focus on real-time operational schemes. This shift is directed by the fact that although they are more difficult to optimize, more complex systems can offer much higher benefits in terms of both profit and sustainability. In general, allowing passengers to transfer between vehicles and for vehicles to have multiple passengers on board increases the occupancy and utilization rate of vehicles as well as the likelihood of matching. However, modeling more complex ride-sharing systems can render real-time solution methods, which are of interest in dynamic, large-scale systems, intractable. As such, there is significant potential for developing high-quality approximation and/or heuristic methods. Following this direction, recent efforts have been made to switch from centralized to decentralized solution schemes, where large-scale optimization problems can be divided into smaller subproblems, sometimes with specific structures, and solved efficiently.

4. Directions for Future Work

Despite several efforts made in the past to incorporate more realistic scenarios into P2P ride-sharing systems, we believe there are a number of potential directions to further extend and explore the prospective benefits of P2P ride-sharing systems. In the following subsections, some of these directions are briefly introduced.

4.1. Incorporating Role Flexibility and Ride Back Guarantees

A quick look at Table 1 reveals that a very limited number of papers have incorporated role flexibility and/or ride back guarantee features in their methodologies, although both these extensions can potentially make P2P ride sharing a more appealing option to the public. On one hand, role flexibility is an effective way to increase the matching rate, especially during nonpeak hours where the system may experience a high rate of imbalance between riders and drivers. On the other hand, ride back guarantees can increase the LoS for customers as they can safely avoid using their personal vehicles because they have a guaranteed ride back home. Furthermore, the concept of a "guaranteed ride back home" can be generalized to encompass any chain of activities that a user would like to complete during a period of time.

4.2. Fast Algorithms with Bounds for Ride-Matching Allocation Problems

Another important missing aspect from the existing literature is exact or high-quality and fast approximate methods for ride matching. Although heuristic methods can be very effective in providing fast solutions from an operational point of view, they cannot be effectively integrated into design of mechanisms. As mobility providers move toward offering more individualized services, mechanism design is attracting more attention in the transportation field (Ropke and Cordeau 2009, Gomes 2014, Cachon et al. 2017, Guda and Subramanian 2019). Mechanism design allows for inquiring private information from users (e.g., their willingness to pay for service) and allocating and pricing resources based on this private information. A well-designed mechanism has several attractive properties, including incentive compatibility (i.e., ensuring that users would not benefit from misreporting their private information), individual rationality (i.e., ensuring users would benefit from participating in the system and hence would voluntarily do so), and budget balance (i.e., ensuring that we would not have to inject money into the system). Having exact solutions for the matching allocation problem would make it possible to devise prices that guarantee subsets of these properties would hold. For more complex problems for which efficient and exact algorithms may not exist, these properties can be guaranteed within bounds if there exist bounds on the allocation matching solutions. This makes approximation algorithms, or other types of algorithms for which allocation bounds exist, appealing.

4.3. Moving from Reactive to Proactive Dynamic Ride Matching

Another research direction, especially for real-time operation of large-scale ride-sharing systems, is using high-quality predictions of demand to enhance system throughput. High-quality demand prediction models can allow the system to dispatch the fleet proactively, cutting down the passenger waiting time and improving their LoS, while at the same time increasing the utilization rate of vehicles. Using nonmyopic methods has already shown promising results in other types of shared-use mobility services (Sayarshad and Chow 2015, Chen and Hu 2019, Özkan and Ward 2020). More specifically, it can help

Table 1. Summary of P2P Ride-Sharing Papers

Study	Dynamic	Decentralized	Multi-Objective	Role-Flexibility	Multi-Passenger	Multi-Hop	Ride-Back	Multi-Modal	Methodology
Baldacci et al. (2004)			✓		✓				MIP+set partitioning+column generation
Calvo et al. (2004)			✓		✓				path construction+local search
Winter and Nittel (2006)	✓	✓							local communication+negotiation
Xing et al. (2009)	✓		✓			✓			multi-agent based simulation
Kleiner et al. (2011)	✓	✓	✓						sealed-bid second price auction
Agatz et al. (2010)	✓								bipartite matching
	✓			✓					general matching
	✓		✓		✓				MIP formulation
	✓		✓		✓	✓			MIP formulation
	✓		✓		✓	✓		✓	MIP formulation
Agatz et al. (2011)	✓								bipartite matching
	✓			✓					general matching+CPLEX/iterative LP rounding
Amey (2011)				✓					general matching+CPLEX/greedy heuristic
Herbawi and Weber (2011b)	✓				✓	✓			multi-objective GA+insertion heuristic
Herbawi and Weber (2011a)	✓				✓	✓			Ant colony optimization+local search
Herbawi and Weber (2012)	✓		✓		✓				MIP+GA+real-time insertion
Ghoseiri (2012)	✓		✓		✓	✓			MIP+hierarchical heuristic
Di Febbraro et al. (2013)	✓		✓		✓				MIP+simulation+B&B
Armant and Brown (2014)	✓			✓	✓				MIP+modified B&B
Pelzer et al. (2015)	✓	✓	✓						partition-based match making
Stiglic et al. (2015)	✓		✓			✓			refined enumeration+lexicographic optimization
Xia et al. (2015)					✓				IP+Simulated Anealing/Tabu Search
Lee and Savelsbergh (2015)	✓				✓			✓	MIP+path construction+neighborhood search
Nourinejad and Roorda (2016)	✓	✓							IP+single-shot first price auction
	✓	✓			✓				enumeration+single-shot first price auction
	✓	✓				✓			single-shot first price auction
Chou et al. (2016)			✓			✓			IP+stochastic set-based PSO
Shen et al. (2016)	✓	✓	✓		✓				filter-and-refine+route planning
Stiglic et al. (2016)	✓		✓		✓				schedule/detour flexibility+lexicographic optimization
Cangialosi et al. (2016)	✓		✓		✓	✓		✓	MIP+CPLEX
Regue et al. (2016)			✓			✓	✓	✓	Binary program+aggregation/disaggregation
Wang et al. (2016a)			✓		✓				IP+insertion+improvement heuristic+Tabu Search
Najmi et al. (2017)	✓	✓							clustering+bipartite matching
Ta et al. (2017)		✓							exact/approximate route planning and matching
Thangaraj et al. (2017)	✓	✓			✓	✓			clustering+online route planning and matching
Masoud and Jayakrishnan (2017a)	✓				✓	✓			Binary program+exact decomposition algorithm
Masoud and Jayakrishnan (2017b)	✓		✓		✓	✓			exact Dynamic programming
Lloret-Batlle et al. (2017)				✓			✓		VCG mechanism+min-cost flow+Network Simplex
Ben Cheikh et al. (2017)	✓		✓		✓	✓			multi-criterion GA with dynamic coding
Bei and Zhang (2018)			✓		✓				approximation algorithm
Long et al. (2018)			✓						bipartite matching+time uncertainty+Monte-Carlo
Xu et al. (2018)	✓	✓	✓						route planning+Simulated Annealing GA
Masoud et al. (2017b)	✓		✓		✓	✓		✓	first-mile, last-mile service+multi-hop matching
Nam et al. (2018)	✓		✓		✓	✓		✓	multi-modal route planing and matching
Ketabi et al. (2018)	✓	✓							spectral clustering+greedy matching
Tamannaei and Irandoost (2019)			✓	✓	✓				MIP+refined B&B+beam search algorithm
An et al. (2019)			✓		✓	✓		✓	sequential optimization+Dynamic programming
Chen et al. (2019)	✓		✓	✓	✓	✓	✓	✓	MIP+greedy heuristic
Tafreshian and Masoud (2020)	✓	✓							graph partitioning+bipartite matching
	✓	✓		✓					graph partitioning+general matching
	✓	✓			✓				graph partitioning+B&B

Note. VCG, Vickrey-Clarke-Groves.

the system substantially reduce the search space of the feasible matches and thereby decrease the response time. Moreover, finding even an exact solution in a rolling horizon framework cannot guarantee the optimality of matching over the whole time horizon under stochastic demand (Agatz et al. 2011). Using historical data may allow the system to improve its matching policies to obtain higher-quality solutions with respect to all (i.e., both the realized and forthcoming) trips. Finally, it should be pointed out that a low-quality demand prediction can lead to dispatching policies that perform poorly, even compared with myopic approaches. As such, it is important to develop routing and dispatching policies that are robust to low-quality predictions and provide recourse actions.

4.4. Optimal Policy for the Rolling-Horizon Framework

A large number of studies in the literature adopt a rolling-horizon approach to incorporate static algorithms into dynamic systems. With a larger rolling horizon, the system can collect a larger pool of participants—hence producing a higher matching rate. On the other hand, a longer horizon also implies higher waiting times for users (both riders and drivers) to learn about the status of their requests—thereby a lower LoS. In highly dynamic systems where the request arrival times are very close to the latest desired departure times of users, longer horizons could even lead to lost demand. As such, there is an obvious dependency between the operational efficiency of the system and the LoS of users, which is partly affected by the length of the optimization horizon. This interdependency calls for incorporating behavioral models into the routing, scheduling, and pricing optimization problems and for capturing the interaction between the supply and demand curves.

When a ride-sharing system has high modal share, then the matching solutions would also impact network travel times and therefore the LoS of system users, highlighting the importance of developing network-aware methodologies that can capture such endogenous relationships.

4.5. Electrification and Autonomy

Experts believe that autonomous vehicles will largely be electrified for a number of technical reasons that revolve around the structural compatibility of autonomous driving with electric propulsion systems (Ersal et al. 2020). It is also believed that autonomous vehicles will be deployed by shared mobility providers because of their larger price tags, their ability to reposition themselves, and environmental concerns (Masoud and Jayakrishnan 2016, 2017c). As such, ride sharing provides a suitable platform to host electric autonomous vehicles. However, the integration of autonomy and electrification into ride-sharing systems gives rise to a number of technical challenges in operating these systems. Specifically, matching, routing, scheduling, and pricing in these systems should take into account the level of charge of vehicles and the distribution of charging stations in the network, adding to the complexity of these problems (MirHassani and Ebrazi 2013, Schneider et al. 2014, Desaulniers et al. 2016, Sweda et al. 2017).

The prospect of a future electrified transportation system has resulted in a number of alternative charging strategies that enable vehicles to be charged while driving. One such strategy is providing dedicated charging lanes that would enable electric vehicles to charge their batteries from the electric grid while driving (Alonso et al. 2014, Li and Mi 2014, Parvez et al. 2014, Stüdli et al. 2014). The possibility of charging a vehicle while traveling on a selected set of links changes the structure of the ride-matching problems and calls for customized routing algorithms. Vehicle-to-vehicle power transfer is another recently proposed technology that enables an electric car to charge its battery en route by connecting to another electric vehicle (Sanchez-Martin et al. 2012, Wang et al. 2016b, Abdolmaleki et al. 2019). The ride-matching optimization problems that arise in this context grow even more complex than the case with dedicated charging lanes. This is because the ride-matching problems have to model scenarios in which not only can a vehicle receive electric power while traveling on a link but also the supply of electricity needs to be provided by other electric vehicles that need to be routed so as to share (part of) their paths with the recipient vehicles. This charging strategy would necessitate developing not only new algorithms but also pricing mechanisms based on utility functions of the suppliers and recipients of power in this exchange. Such a charging platform provides the opportunity for a shared mobility provider to use its own fleet as power suppliers, giving rise to more complex problems.

4.6. Integrating P2P Ride Sharing into Mobility-as-a-Service

With mobility-as-a-service (MaaS) becoming a more popular mode of transportation, an interesting research direction is to investigate integrating P2P ride sharing into MaaS operations, the implications of such integration on the MaaS LoS, as well as the empty miles imposed on the transportation infrastructure (Qi and Shen 2019). More specifically, the following two research questions are of interest: (1) the implications of different P2P matching settings (i.e., one-to-one, one-to-many, many-to-one, and many-to-many) on the empty miles driven of MaaS and (2) regulations and policies specific to each matching setting to curb the MaaS empty miles driven.

References

Abdolmaleki M, Masoud N, Yin Y (2019) Vehicle-to-vehicle wireless power transfer: Paving the way toward an electrified transportation system. *Transportation Res. Part C Emerging Tech.* 103(June):261–280.

Agatz N, Erera A, Savelsbergh M, Wang X (2010) Sustainable passenger transportation: Dynamic ride-sharing. ERIM Report ERS-2010-010-LIS, Erasmus Institute of Management, Erasmus University Rotterdam, Rotterdam, Netherlands.

Agatz N, Erera AL, Savelsbergh MW, Wang X (2011) Dynamic ridesharing: A simulation study in metro Atlanta. *Procedia Soc. Behav. Sci.* 17(9):532–550.

Agatz N, Erera A, Savelsbergh M, Wang X (2012) Optimization for dynamic ride-sharing: A review. *Eur. J. Oper. Res.* 223(2): 295–303.

Alonso M, Amaris H, Germain JG, Galan JM (2014) Optimal charging scheduling of electric vehicles in smart grids by heuristic algorithms. *Energies* 7(4):2449–2475.

Alonso-Mora J, Samaranayake S, Wallar A, Frazzoli E, Rus D (2017) On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment. *Proc. Natl. Acad. Sci. USA* 114(3):462–467.

Amey A (2011) A proposed methodology for estimating rideshare viability within an organization, applied to the MIT community. *Proc. Transportation Res. Board 90th Annual Meeting*, (Transportation Research Board, Washington, DC), 1–16.

An S, Nam D, Jayakrishnan R (2019) Impacts of integrating shared autonomous vehicles into a peer-to-peer ridesharing system. *Procedia Comput. Sci.* 151:511–518.

Armant V, Brown KN (2014) Minimizing the driving distance in ride sharing systems. *2014 IEEE 26th Internat. Conf. Tools Artificial Intelligence* (IEEE, Piscataway, NJ), 568–575.

Baldacci R, Maniezzo V, Mingozzi A (2004) An exact method for the car pooling problem based on Lagrangean column generation. *Oper. Res.* 52(3):422–439.

Bei X, Zhang S (2018) Algorithms for trip-vehicle assignment in ridesharing. *Proc. 32nd AAAI Conf. Artificial Intelligence* (AAAI Press, Palo Alto, CA), 3–9.

Ben Cheikh S, Tahon C, Hammadi S (2017) An evolutionary approach to solve the dynamic multihop ridematching problem. *Simulation* 93(1):3–19.

Benjaafar S, Hu M (2020) Operations management in the age of the sharing economy: What is old and what is new? *Manufacturing Service Oper. Management* 22(1):93–101.

Benjaafar S, Bernhard H, Courcoubetis C (2017) Drivers, riders and service providers: The impact of the sharing economy on mobility. *Proc. 12th Workshop on the Econom. of Networks, Systems and Computat.* (ACM, New York), 1–6.

Bimpikis K, Candogan O, Saban D (2019) Spatial pricing in ridesharing networks. *Oper. Res.* 67(3):744–769.

Boyacı B, Zografos KG, Geroliminis N (2015) An optimization framework for the development of efficient one-way car-sharing systems. *Eur. J. Oper. Res.* 240(3):718–733.

Braverman A, Dai JG, Liu X, Ying L (2019) Empty-car routing in ridesharing systems. *Oper. Res.* 67(5):1437–1452.

Cachon GP, Daniels KM, Lobel R (2017) The role of surge pricing on a service platform with self-scheduling capacity. *Manufacturing Service Oper. Management* 19(3):368–384.

Calvo RW, de Luigi F, Haastrup P, Maniezzo V (2004) A distributed geographic information system for the daily car pooling problem. *Comput. Oper. Res.* 31(13):2263–2278.

Cangialosi E, Di Febbraro A, Sacco N (2016) Designing a multimodal generalised ride sharing system. *IET Intelligent Transport Systems* 10(4):227–236.

Chan ND, Shaheen SA (2012) Ridesharing in North America: Past, present, and future. *Transportation Rev.* 32(1):93–112.

Chang J, Yu M, Shen S, Xu M (2017) Location design and relocation of a mixed car-sharing fleet with a CO_2 emission constraint. *Service Sci.* 9(3):205–218.

Chen Y, Hu M (2020) Pricing and matching with forward-looking buyers and sellers. *Manufacturing Service Oper. Management*, 22(4):717–734.

Chen W, Mes M, Schutten M, Quint J (2019) A ride-sharing problem with meeting points and return restrictions. *Transportation Sci.* 53(2):401–426.

Chou SK, Jiau MK, Huang SC (2016) Stochastic set-based particle swarm optimization based on local exploration for solving the carpool service problem. *IEEE Trans. Cybernetics* 46(8):1771–1783.

Clewlow RR, Mishra GS (2017) Disruptive transportation: The adoption, utilization, and impacts of ride-hailing in the United States. Research Report UCD-ITS-RR-17-07, Institute of Transportation Studies, University of California, Davis, Davis.

Cordeau JF (2006) A branch-and-cut algorithm for the dial-a-ride problem. *Oper. Res.* 54(3):573–586.

Correia GHDA, Jorge DR, Antunes DM (2014) The added value of accounting for users' flexibility and information on the potential of a station-based one-way car-sharing system: An application in Lisbon, Portugal. *J. Intelligent Transportation Systems* 18(3):299–308.

Cui S, Li K, Yang L, Wang J (2019) Slugging: Casual carpooling for urban transit. Working paper, Georgetown University, Washington, DC.

Desaulniers G, Errico F, Irnich S, Schneider M (2016) Exact algorithms for electric vehicle-routing problems with time windows. *Oper. Res.* 64(6):1388–1405.

Di Febbraro A, Gattorna E, Sacco N (2013) Optimization of dynamic ridesharing systems. *Transportation Res. Record* 2359(1):44–50.

Duan R, Pettie S (2014) Linear-time approximation for maximum weight matching. *J. ACM* 61(1):1–23.

Duan R, Su HH (2012) A scaling algorithm for maximum weight matching in bipartite graphs. Rabani Y, ed. *Proc. 23rd Annual ACM-SIAM Sympos. Discrete Algorithms* (Society for Industrial and Applied Mathematics, Philadelphia), 1413–1424.

Edmonds J (1965a) Maximum matching and a polyhedron with 0,1-vertices. *J. Res. Natl. Bureau Standards* 69B(1–2):125–130.

Edmonds J (1965b) Paths, trees, and flowers. *Canadian J. Math.* 17:449–467.

Edmonds J, Karp RM (1972) Theoretical improvements in algorithmic efficiency for network flow problems. *J. ACM* 19(2):248–264.

Ersal T, Kolmanovsky I, Masoud N, Ozay N, Scruggs J, Vasudevan R, Orosz G (2020) Connected and automated road vehicles: State of the art and future challenges. *Vehicle Systems Dynam.* 58(5):672–704.

Fredman ML, Tarjan RE (1987) Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM* 34(3):596–615.

Fu L (2002) Scheduling dial-a-ride paratransit under time-varying, stochastic congestion. *Transportation Res. Part B Methodological* 36(6):485–506.

Furuhata M, Dessouky M, Ordóñez F, Brunet ME, Wang X, Koenig S (2013) Ridesharing: The state-of-the-art and future directions. *Transportation Res. Part B Methodological* 57(November):28–46.

Gabow HN (1990) Data structures for weighted matching and nearest common ancestors with linking. *Proc. 1st Annual ACM-SIAM Sympos. Discrete Algorithms* (Society for Industrial and Applied Mathematics, Philadelphia), 434–443.

Gabow HN, Tarjan RE (1989) Faster scaling algorithms for network problems. *SIAM J. Comput.* 18(5):1013–1036.

Gabow HN, Tarjan RE (1991) Faster scaling algorithms for general graph matching problems. *J. ACM* 38(4):815–853.

Ghoseiri K (2012) Dynamic rideshare optimized matching problem. Unpublished PhD thesis, University of Maryland, College Park.

Gomes R (2014) Optimal auction design in two-sided markets. *RAND J. Econom.* 45(2):248–272.

Guda H, Subramanian U (2019) Your Uber is arriving: Managing on-demand workers through surge pricing, forecast communication, and worker incentives. *Management Sci.* 65(5):1995–2014.

He L, Mak HY, Rong Y, Shen ZJM (2017) Service region design for urban electric vehicle sharing systems. *Manufacturing Service Oper. Management* 19(2):309–327.

Healy P, Moll R (1995) A new extension of local search applied to the dial-a-ride problem. *Eur. J. Oper. Res.* 83(1):83–104.

Herbawi W, Weber M (2011a) Ant colony vs. genetic multiobjective route planning in dynamic multi-hop ridesharing. *2011 IEEE 23rd Internat. Conf. Tools Artificial Intelligence* (IEEE, Piscataway, NJ), 282–288.

Herbawi W, Weber M (2011b) Evolutionary multiobjective route planning in dynamic multi-hop ridesharing. Merz P, Hao JK, eds. *Eur. Conf. Evolutionary Comput. Combin. Optim.* (Springer, Berlin), 84–95.

Herbawi W, Weber M (2012) The ridematching problem with time windows in dynamic ridesharing: A model and a genetic algorithm. *2012 IEEE Congress Evolutionary Comput.* (IEEE, Piscataway, NJ), 1–8.

Hopcroft JE, Karp RM (1973) An $n5/2$ algorithm for maximum matchings in bipartite graphs. *SIAM J. Comput.* 2(4):225–231.

Hosni H, Naoum-Sawaya J, Artail H (2014) The shared-taxi problem: Formulation and solution methods. *Transportation Res. Part B Methodological* 70(December):303–318.

Jacob J, Roet-Green R (2018) Ride solo or pool: Designing price-service menus for a ride-sharing platform. Working Paper 3008136, Simon Business School, University of Rochester, Rochester.

Jaw JJ, Odoni AR, Psaraftis HN, Wilson NH (1986) A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Res. Part B Methodological* 20(3):243–257.

Jung J, Jayakrishnan R, Park JY (2016) Dynamic shared-taxi dispatch algorithm with hybrid-simulated annealing. *Comput. Aided Civil Infrastructure Engrg.* 31(4):275–291.

Kelly KL (2007) Casual carpooling—Enhanced. *J. Public Transportation* 10(4):119–130.

Ketabi R, Alipour B, Helmy A (2018) Playing with matches: Vehicular mobility through analysis of trip similarity and matching. *Proc. 26th ACM SIGSPATIAL Internat. Conf. Adv. Geographic Inform. Systems* (ACM, New York), 544–547.

Kleiner A, Nebel B, Ziparo VA (2011) A mechanism for dynamic ride sharing based on parallel auctions. Walsh T, ed. *Proc. 22nd Internat. Joint Conf. Artificial Intelligence*, Vol. 1 (AAAI Press, Menlo Park, CA), 266–272.

Lee A, Savelsbergh M (2015) Dynamic ridesharing: Is there a role for dedicated drivers? *Transportation Res. Part B Methodological* 81-(Part 2, November):483–497.

Li S, Mi CC (2014) Wireless power transfer for electric vehicle applications. *IEEE J. Emerging Selected Topics Power Electronics* 3(1):4–17.

Lloret-Batlle R, Masoud N, Nam D (2017) Peer-to-peer ridesharing with ride-back on high-occupancy-vehicle lanes: Toward a practical alternative mode for daily commuting. *Transportation Res. Record* 2668(1):21–28.

Long J, Tan W, Szeto W, Li Y (2018) Ride-sharing with travel time uncertainty. *Transportation Res. Part B Methodological* 118-(December):143–171.

Ma S, Zheng Y, Wolfson O (2013) T-share: A large-scale dynamic taxi ridesharing service. *2013 IEEE 29th Internat. Conf. Data Engrg.* (IEEE, Piscataway, NJ), 410–421.

Madsen OB, Ravn HF, Rygaard JM (1995) A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives. *Ann. Oper. Res.* 60(1):193–208.

Masoud N, Jayakrishnan R (2016) Formulations for optimal shared ownership and use of autonomous or driverless vehicles. *Proc. Transportation Res. Board 95th Annual Meeting* (Transportation Research Board, Washington DC), 1–17.

Masoud N, Jayakrishnan R (2017a) A decomposition algorithm to solve the multi-hop peer-to-peer ride-matching problem. *Transportation Res. Part B Methodological* 99(May):1–29.

Masoud N, Jayakrishnan R (2017b) A real-time algorithm to solve the peer-to-peer ride-matching problem in a flexible ridesharing system. *Transportation Res. Part B Methodological* 106-(December):218–236.

Masoud N, Jayakrishnan R (2017c) Autonomous or driver-less vehicles: Implementation strategies and operational concerns. *Transportation Res. Part E Logist. Transportation Rev.* 108(December):179–194.

Masoud N, Lloret-Batlle R, Jayakrishnan R (2017a) Using bilateral trading to increase ridership and user permanence in ridesharing systems. *Transportation Res. Part E Logist. Transportation Rev.* 102(June):60–77.

Masoud N, Nam D, Yu J, Jayakrishnan R (2017b) Promoting peer-to-peer ridesharing services as transit system feeders. *Transportation Res. Record* 2650(1):74–83.

Masoud S, Son YJ, Masoud N, Jayakrishnan J (2019) Impact of traffic conditions and carpool lane availability on peer to peer ridesharing demand. Working paper, University of Arizona, Tucson.

MirHassani S, Ebrazi R (2013) A flexible reformulation of the refueling station location problem. *Transportation Sci.* 47(4):617–628.

Mote JE, Whitestone Y (2011) The social context of informal commuting: Slugs, strangers and structuration. *Transportation Res. Part A Policy Practice* 45(4):258–268.

Najmi A, Rey D, Rashidi TH (2017) Novel dynamic formulations for real-time ride-sharing systems. *Transportation Res. Part E Logist. Transportation Rev.* 108(December):122–140.

Nam D, Yang D, An S, Yu JG, Jayakrishnan R, Masoud N (2018) Designing a transit-feeder system using multiple sustainable modes: Peer-to-peer (P2P) ridesharing, bike sharing, and walking. *Transportation Res. Record* 2672(8):754–763.

Nourinejad M, Roorda MJ (2015) Carsharing operations policies: A comparison between one-way and two-way systems. *Transportation* 42(3):497–518.

Nourinejad M, Roorda MJ (2016) Agent based model for dynamic ridesharing. *Transportation Res. Part C Emerging Tech.* 64-(March):117–132.

Ordóñez F, Dessouky MM (2017) Dynamic ridesharing. Batta R, Peng J, eds. *Leading Developments from INFORMS Communities*, Tutorials in Operations Research (INFORMS, Catonsville, MD), 212–236.

Orlin JB (1993) A faster strongly polynomial minimum cost flow algorithm. *Oper. Res.* 41(2):338–350.

Orlin JB (1997) A polynomial time primal network simplex algorithm for minimum cost flows. *Math. Programming* 78(2):109–129.

Özkan E, Ward AR (2020) Dynamic matching for real-time ride sharing. *Stochastic Systems* 10(1):29–70.

Parvez M, Mekhilef S, Tan NM, Akagi H (2014) Model predictive control of a bidirectional AC-DC converter for V2G and G2V applications in electric vehicle battery charger. *2014 IEEE Transportation Electrification Conf. Expo* (IEEE, Piscataway, NJ), 1–6.

Pelzer D, Xiao J, Zehe D, Lees MH, Knoll AC, Aydt H (2015) A partition-based match making algorithm for dynamic ridesharing. *IEEE Trans. Intelligent Transportation Systems* 16(5):2587–2598.

Qi W, Shen ZJM (2019) A smart-city scope of operations management. *Production Oper. Management* 28(2):393–406.

Regue R, Masoud N, Recker W (2016) Car2work: Shared mobility concept to connect commuters with workplaces. *Transportation Res. Record* 2542(1):102–110.

Ropke S, Cordeau J-F (2009) Branch and cut and price for the pickup and delivery problem with time windows. *Transportation Sci.* 43(3):267–286.

Sanchez-Martin P, Sanchez G, Morales-España G (2012) Direct load control decision model for aggregated EV charging points. *IEEE Trans. Power Systems* 27(3):1577–1584.

Santi P, Resta G, Szell M, Sobolevsky S, Strogatz SH, Ratti C (2014) Quantifying the benefits of vehicle pooling with shareability networks. *Proc. Natl. Acad. Sci. USA* 111(37):13290–13294.

Sayarshad HR, Chow JY (2015) A scalable non-myopic dynamic dial-a-ride and pricing problem. *Transportation Res. Part B Methodological* 81(Part 2, November):539–554.

Schneider M, Stenger A, Goeke D (2014) The electric vehicle-routing problem with time windows and recharging stations. *Transportation Sci.* 48(4):500–520.

Shaheen S, Chan N (2016) Mobility and the sharing economy: Potential to facilitate the first-and last-mile public transit connections. *Built Environ.* 42(4):573–588.

Shaheen SA, Cohen AP (2007) Growth in worldwide carsharing: An international comparison. *Transportation Res. Record* 1992(1): 81–89.

Shaheen SA, Chan ND, Gaynor T (2016) Casual carpooling in the San Francisco Bay Area: Understanding user characteristics, behaviors, and motivations. *Transport Policy* 51(October):165–173.

Shaheen S, Cohen A, Jaffee M (2018) Innovative mobility: Carsharing outlook. UC Berkeley policy brief, University of California, Berkeley, Berkeley.

Shaheen SA, Guzman S, Zhang H (2010) Bikesharing in Europe, the Americas, and Asia: Past, present, and future. *Transportation Res. Record* 2143(1):159–167.

Shen B, Huang Y, Zhao Y (2016) Dynamic ridesharing. *SIGSPATIAL Special* 7(3):3–10.

Stiglic M, Agatz N, Savelsbergh M, Gradisar M (2015) The benefits of meeting points in ride-sharing systems. *Transportation Res. Part B Methodological* 82(December):36–53.

Stiglic M, Agatz N, Savelsbergh M, Gradisar M (2016) Making dynamic ride-sharing work: The impact of driver and rider flexibility. *Transportation Res. Part E Logist. Transportation Rev.* 91-(July):190–207.

Ströhle P, Flath CM, Gärttner J (2019) Leveraging customer flexibility for car-sharing fleet optimization. *Transportation Sci.* 53(1):42–61.

Stüdli S, Crisostomi E, Middleton R, Shorten R (2014) Optimal real-time distributed V2G and G2V management of electric vehicles. *Internat. J. Control* 87(6):1153–1162.

Sweda TM, Dolinskaya IS, Klabjan D (2017) Optimal recharging policies for electric vehicles. *Transportation Sci.* 51(2):457–479.

Ta N, Li G, Zhao T, Feng J, Ma H, Gong Z (2017) An efficient ride-sharing framework for maximizing shared route. *IEEE Trans. Knowledge Data Engrg.* 30(2):219–233.

Tafreshian A, Masoud N (2020) Trip-based graph partitioning in dynamic ridesharing. *Transportation Res. Part C Emerging Tech.* 114(May):532–553.

Tamannaei M, Irandoost I (2019) Carpooling problem: A new mathematical model, branch-and-bound, and heuristic beam search algorithm. *J. Intelligent Transportation Systems* 23(3):203–215.

Thangaraj RS, Mukherjee K, Raravi G, Metrewar A, Annamaneni N, Chattopadhyay K (2017) Xhare-a-ride: A search optimized dynamic ride sharing system with approximation guarantee. *2017 IEEE 33rd Internat. Conf. Data Engrg.* (IEEE, Piscataway, NJ), 1117–1128.

Wang H, Yang H (2019) Ridesourcing systems: A framework and review. *Transportation Res. Part B Methodological* 129(November): 122–155.

Wang X, Agatz N, Erera A (2018a) Stable matching for dynamic ride-sharing systems. *Transportation Sci.* 52(4):850–867.

Wang X, Dessouky M, Ordonez F (2016a) A pickup and delivery problem for ridesharing considering congestion. *Transportation Lett.* 8(5):259–269.

Wang X, Yang H, Zhu D (2018b) Driver-rider cost-sharing strategies and equilibria in a ridesharing program. *Transportation Sci.* 52(4):868–881.

Wang M, Ismail M, Zhang R, Shen X, Serpedin E, Qaraqe K (2016b) Spatio-temporal coordinated V2V energy swapping strategy for mobile PEVS. *IEEE Trans. Smart Grid* 9(3):1566–1579.

Weikl S, Bogenberger K (2013) Relocation strategies and algorithms for free-floating car sharing systems. *IEEE Intelligent Transportation Systems Magazine* 5(4):100–111.

Winter S, Nittel S (2006) Ad hoc shared-ride trip planning by mobile geosensor networks. *Internat. J. Geographical Inform. Sci.* 20(8):899–916.

Xia J, Curtin KM, Li W, Zhao Y (2015) A new model for a carpool matching service. *PLoS One* 10(6):e0129257.

Xing X, Warden T, Nicolai T, Herzog O (2009) SMIZE: A spontaneous ride-sharing system for individual urban transit. Braubach L, van der Hoek W, Petta P, Pokahr A, eds. *Proc. 7th German Conf. Multiagent System Tech.* (Springer, Berlin), 165–176.

Xu Z, Yin Y, Ye J (2020) On the supply curve of ride-hailing systems. *Transportation Res. Part B Methodological* 132(February): 29–43.

Xu J, Zhang Y, Xing C, Zhang G (2018) A real-time ride-sharing matching framework using simulated annealing genetic algorithm. *Proc. 30th Internat. Conf. Software Engrg. Knowledge Engrg.* (KSI Research, Philadelphia), 250–255.

Zha L, Yin Y, Du Y (2018a) Surge pricing and labor supply in the ride-sourcing market. *Transportation Res. Part B Methodological* 117-(Part B, November):708–722.

Zha L, Yin Y, Xu Z (2018b) Geometric matching and spatial pricing in ride-sourcing markets. *Transportation Res. Part C Emerging Tech.* 92(July):58–75.

Zha L, Yin Y, Yang H (2016) Economic analysis of ride-sourcing markets. *Transportation Res. Part C Emerging Tech.* 71(October):249–266.