CCAT | CENTER FOR CONNECTED AND AUTOMATED TRANSPORTATION

Project Start Date: 1/1/2017

Project End Date: 12/31/2018

February 2019

Enhancing network equilibrium models for capturing emerging shared-use mobility services

Neda Masoud, Assistant Professor
Yafeng Yin, Professor
University of Michigan Ann Arbor

M | UMTRI
The University of Akron
ILLINOIS CENTER FOR TRANSPORTATION
PURDUE UNIVERSITY
Washtenaw Community College
Central State University 1887

CCAT | CENTER FOR CONNECTED AND AUTOMATED TRANSPORTATION

DISCLAIMER

Funding for this research was provided by the Center for Connected and Automated Transportation under Grant No. 69A3551747105 of the U.S. Department of Transportation, Office of the Assistant Secretary for Research and Technology (OST-R), University Transportation Centers Program. The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

Suggested APA Format Citation:

Masoud, N. (2019). Enhancing network equilibrium models for capturing emerging shared-use mobility services. Final Report. USDOT CCAT Project No. 4. Identifier: http://hdl.handle.net/2027.42/162823

Contacts

For more information:

Neda Masoud
University of Michigan, Ann Arbor
2350 Hayward St., 2124 GG Brown Bldg.
Phone: (734) 764-8230
Fax: (734) 764-4292
nmasoud@umich.edu
http://www-personal.umich.edu/~nmasoud/

CCAT
University of Michigan Transportation Research
Institute 2901 Baxter Road
Ann Arbor, MI 48152
umtri-ccat@umich.edu
(734) 763-2498
www.ccat.umtri.umich.edu

Technical Report Documentation Page

1. Report No.

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle

Enhancing network equilibrium models for capturing emerging shared-use mobility services
Identifier: http://hdl.handle.net/2027.42/162823

5. Report Date

February 2019

6. Performing Organization Code

7. Author(s)
Principal Investigator: Neda Masoud (0000-0002-6526-3317)
Co- Principal Investigator: Yafeng Yin (0000-0003-3117-5463)
Co-Authors: Amirmahdi Tafreshian (0000-0003-1175-0707), Zhengtian Xu (0000-0001-5626-285X)

8. Performing Organization Report No.
CCAT Project No. 4

9. Performing Organization Name and Address
Center for Connected and Automated Transportation
University of Michigan Transportation Research Institute
2901 Baxter Road.Ann Arbor, MI 48109
University of Michigan Ann Arbor
2350 Hayward St., GG Brown Bldg.

10. Work Unit No. (TRAIS)

11. Contract or Grant No.
69A3551747105

12. Sponsoring Agency Name and Address

Federal Highway Administration
Office of Safety
1200 New Jersey Avenue SE
Washington, DC 20590

13. Type of Report and Period Covered
Final Report
Jan. 2017 to Dec. 2018

14. Sponsoring Agency Code
US DOT

15. Supplementary Notes

16. Abstract

Driven by the development of vehicle connectivity and automation, shared-use mobility services are expected to play a major role in meeting urban mobility needs. However, existing network equilibrium models cannot adequately model these emerging services, as these models are trip centric, assigning vehicular trips to transportation networks. With shared-use mobility, vehicular trips are the outcome of the interactions between service operators and travelers, a missing ingredient in the current network equilibrium analysis methodology. In this study, we will enhance the methodology by explicitly modeling the behaviors of both service operators and travelers. We will consider two implementations of shared-use mobility: one of a decentralized system in which vehicles choose which areas to serve based on their individually defined utility functions, and one of a centralized system in which a shared-use mobility service provider optimally assigns vehicles to requests based on a system-level objective function. The proposed models are expected to enhance the planning practice for shared-use mobility services.

17. Key Words

Shared-use mobility, network equilibrium

18. Distribution Statement

No restrictions

19. Security Classif. (of this report)

Unclassified

20. Security Classif. (of this page)

Unclassified

21. No. of Pages
46

22. Price

N/A

Table of Contents

List of Figures

List of Tables

List of Algorithms

1 Introduction

Recent advancements in information and vehicular technologies drive the wave of innovations in mobility services. Specifically, the number of smart mobile devices in the US has been rising steadily and a study suggests that nearly two-thirds of Americans now own at least one such device. These devices retrieve users's geolocations, enable ubiquitous communications, and allow instant peer-to-peer interaction, giving rise to various on-demand mobility services for goods and people, which bring together suppliers of resources (e.g., cars) and services (e.g., rides) with very low transaction costs. Connected and autonomous vehicle technology will further revolutionize urban and rural mobility and promote the shift from car ownership to sharing/subscription. Automated shared-use mobility services may eventually emerge where companies own a fleet of different types of automated vehicles and offer on-demand ride hailing services. Other types of mobility services will be likely catalyzed by new business models and venture capital investments. These mobility services are expected to play an increasingly important role in meeting mobility needs. It is critical to better design, plan and operate them.

Existing travel-demand forecasting models are limited in capturing the travel demand and system performance associated with them. One of the most critical challenges is that none of existing network assignment approaches can adequately model shared-use mobility services, as these approaches are trip centric, assigning vehicular trips to transportation networks. With shared-use mobility, vehicular trips are the outcome of the interactions between service operators and travelers, a missing ingredient in the current assignment methodology. In this study, we will enhance the methodology by explicitly modeling the behaviors of both service operators and travelers. More specifically, we consider two implementations of shared-used mobility: one of a decentralized system in which vehicles choose which areas to serve based on their individually defined utility functions, and one of a centralized system in which a shared-use mobility service provider optimally assigns vehicles to requests based on a system-level objective function. In both system architectures, the assignment of vehicles to regions/requests will be determined endogenously, and will be reflected in the origin-destination trip tables that are inputs to the assignment problem. In the assignment step, both shared-use and private vehicles will follow a user-optimum principle in choosing routes from their origins to destinations.

2 Equilibrium Analysis of Urban Traffic Networks with Ride-Sourcing

2.1 Base Model

Consider a road network $G(V, A)$ where V is the set of nodes and A is the set of links in the network. Each node $v \in V$ represents an isotropic zone or neighborhood. For the base model, we assume that the matching radius adopted by the e-hailing platform is relatively small such that customers who request rides online will only be matched to idle RVs nearby (mathematically, at the same node). Consequently, deadheading to pick up customers is negligible. Below we describe major components of the base model.

2.1.1 Customer Demand

Denote the node sets R, S respectively as the origins and destinations of customer demands, and let W be the set of origin-destination (OD) pairs. Define β^o and β^i respectively as the out-of-vehicle and in-vehicle value of time ($/h). Then, the travel cost C_{rs} between OD pair $(r, s) \in W$ is given by:

$$C_{rs} = F_{rs} + \beta^o w_r^c + \beta^i h_{rs}, \quad \forall (r, s) \in W$$

where h_{rs} is the equilibrium or shortest vehicular travel time between node r and s ; F_{rs} is the trip fare between the nodes and assumed to follow the structure of $F_{rs} = F_{rs}^0 + \beta^f h_{rs}$, where $\beta^f h_{rs}$ represents the time-based component with β^f ($\geqslant$0) characterizing the hourly surcharge and F_{rs}^0 denotes a constant for the rest time-irrelevant components; and w_r^c is the customer's average waiting time at node r .

Assuming the customer demand Q_{rs} to be a monotonically decreasing and convex function of the trip cost C_{rs} , we then have:

$$Q_{rs} = f_{rs}\left(\beta^o w_r^c + (\beta^i + \beta^f) h_{rs}\right), \quad \forall (r,s) \in W \tag{1}$$

where $f_{rs}' < 0$ and $f_{rs}'' \geqslant 0$. Moreover, $\lim\limits_{w_r^c \to +\infty} w_r^c \cdot f_{rs}(w_r^c \mid \mathbf{h}) \in (0,+\infty)$, implying that there are always finite numbers of customers waiting on each origin node no matter how long they shall wait.

2.1.2 Idle RV Supply

The idle RVs refer to the group of ride-sourcing vehicles who stay vacant waiting to be matched. Typically, idle RVs emerge at destination node set S where they drop off customers, and disappear at origin node set R where they get matched to new riders. Suppose the utility function of an idle RV cruising from node $s \in S$ to node $r \in R$ is prescribed as

$$U_{sr} = \overline{F}_r - \gamma \cdot (\overline{h}_r + h_{sr} + w_r^v), \quad \forall s \in S, r \in R \tag{2}$$

where $\overline{F}_r$ and $\overline{h}_r$ are the average fare and service time of the customer trips originating from node r ; w_r^v is the idle RV's average waiting time for matching or meeting at node r ; and γ denotes ride-sourcing drivers' value of time (\$/h). Specifically, $\overline{F}_r$ and $\overline{h}_r$ are given as follows,

$$\overline{F}_r = \frac{\sum\limits_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o) F_{rs}}{\sum\limits_{s:(r,s)\in W} (T_{rs}^o + \epsilon^0)}, \quad \overline{h}_r = \frac{\sum\limits_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o) h_{rs}}{\sum\limits_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o)}, \quad \forall r \in R \tag{3}$$

where T_{rs}^o is the occupied RV flow that serves customer demand from node r to s ; ϵ^0 is a small constant applied to ensure the feasibility of mapping when $\sum_{s:(r,s)\in W} T_{rs}^o$ in the denominator is zero. In brevity, the weighted averages of fare and service time converge to be unweighted when the flows $\{T_{rs}^o\}$ are very small. The rationale behind such a treatment is that when there are few flows sourced from one origin node, it is difficult for drivers to "learn" the actual distribution of flow destinations and then gives rise to the random perceptions for drivers. Nevertheless, this treatment is implemented for mathematical completeness, and we will prove later that the zero demand condition, which rarely happen in practice (Ban, 2019), will not arise in our system equilibrium.

Assume each idle RV after dropping off customers cruises towards a node $r \in R$ that maximizes its perceptual utility, and the perception error on the utility follows the Gumbel distribution. Then, the portion P_{sr} of idle RVs cruising to r among all those generated at node s is given as

$$P_{sr} = \frac{\exp(\theta U_{sr})}{\sum\limits_{k\in R} \exp(\theta U_{sk})}, \quad \forall s \in S, r \in R$$

where θ is a constant representing the degree of perceptual dispersion. This leads to the corresponding idle RV flow T_{sr}^v , written as

$$T_{sr}^v = \frac{\exp(\theta U_{sr})}{\sum\limits_{k\in R} \exp(\theta U_{sk})} \cdot \sum_{k:(k,s)\in W} T_{ks}^o, \quad \forall s \in S, r \in R \tag{4}$$

2.1.3 Intra-node Matching Between Hailing Customers And Idle RVs

As aforementioned, the base model considers only intra-node matching, i.e. customers can only be matched to idle RVs at the same node. Since each node features one isotropic zone, an aggregate matching function m_r can be used to characterize the matching frictions between unmatched RVs and customers (e.g., (Douglas, 1972)). In particular, we adopt the matching function suggested by (Yang, 2011) to capture the competition among drivers and customers over intra-node matching:

$$O_r^m = m_r(N_r^v, N_r^c), \quad \forall r \in R \tag{5}$$

where O_r^m represents the realized rate of matchings at node r ; N_r^v and N_r^c denote the number of idle RVs and hailing customers at node r, respectively. Under stationary states, the variables in the above matching function are also subject to the following relationships:

$$N_r^v = \left(\sum_{s \in S} T_{sr}^v \right) w_r^v \tag{6}$$

$$N_r^c = \left(\sum_{s:(r,s) \in W} Q_{rs} \right) w_r^c \tag{7}$$

$$O_r^m = \sum_{s:(r,s) \in W} T_{rs}^o = \sum_{s:(r,s) \in W} Q_{rs} = \sum_{s \in S} T_{sr}^v \tag{8}$$

2.1.4 Network Equilibrium Under Intra-node Matching

Given all the above relations regarding idle and occupied RV movements, we define the network equilibrium state that results from interactions between idle and occupied RVs, and the background regular traffic generated by non-sharing vehicles. Define W^b as the set of OD pairs for the regular traffic, and W^c as the complete set of OD pairs including those for idle RVs, occupied RVs as well as regular traffic, i.e. $W^c = \{(s,r) \mid s \in S, r \in R\} \cup W \cup W^b$. Let N denote the total number of RVs serving the network; and let T_{rs}^n denote the OD demand of regular traffic from node r to s , which in this study is assumed to be fixed for all $(r,s) \in W^b$. Then, the equilibrium link flow distribution $\{x_{ij}^{rs}\}$ solves the following system of equalities and inequalities:

Path Equilibrium between OD pairs

$$\left[t_{ij}(v_{ij}) - \rho_i^{rs} + \rho_j^{rs} \right] x_{ij}^{rs} = 0 \forall (i,j) \in A, (r,s) \in W^c \tag{9}$$

$$t_{ij}(v_{ij}) - \rho_i^{rs} + \rho_j^{rs} \geqslant 0 \forall (i,j) \in A, (r,s) \in W^c \tag{10}$$

$$v_{ij} = \sum_{(r,s) \in W^c} x_{ij}^{rs} \forall (i,j) \in A \tag{11}$$

$$x_{ij}^{rs} \geqslant 0 \forall (i,j) \in A, (r,s) \in W^c \tag{12}$$

$$T^{rs} = T_{rs}^v + T_{rs}^o + T_{rs}^n \forall (r,s) \in W^c \tag{13}$$

$$\sum_{i:(i,k) \in A} x_{ik}^{rs} - \sum_{j:(k,j) \in A} x_{kj}^{rs} = \begin{cases} -T^{rs}, & \text{if} k = r \\ T^{rs}, & \text{if} k = s \quad \forall (r,s) \in W^c \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

$$h_{rs} = \rho_r^{rs} - \rho_s^{rs} \forall (r,s) \in W^c \tag{15}$$

Customer Demand

$$Q_{rs} = f_{rs}\left(\beta^o w_r^c + (\beta^i + \beta^f) h_{rs} \right) \forall (r,s) \in W \tag{16}$$

$$F_{rs} = F_{rs}^{0} + \beta^{f} h_{rs} \forall (r,s) \in W \tag{17}$$

Idle RV movements

$$U_{sr} = \overline{F}_{r} - \gamma \cdot (\overline{h}_{r} + h_{sr} + w_{r}^{v}) \forall s \in S, r \in R \tag{18}$$

$$\overline{F}_{r} = \frac{\sum_{s:(r,s)\in W} (T_{rs}^{o} + \epsilon^{o}) F_{rs}}{\sum_{s:(r,s)\in W} (T_{rs}^{o} + \epsilon^{o})} \forall r \in R \tag{19}$$

$$\overline{h}_{r} = \frac{\sum_{s:(r,s)\in W} (T_{rs}^{o} + \epsilon^{o}) h_{rs}}{\sum_{s:(r,s)\in W} (T_{rs}^{o} + \epsilon^{o})} \forall r \in R \tag{20}$$

$$T_{sr}^{v} = \begin{cases} \dfrac{\exp(\theta U_{sr})}{\sum_{k\in R} \exp(\theta U_{sk})} \cdot \sum_{k:(k,s)\in W} T_{ks}^{o}, \ \forall s \in S, r \in R \\ 0, \ \forall (s,r) \in W^{c} \setminus \{(s,r) \mid s \in S, r \in R\} \end{cases} \tag{21}$$

Intra-node matching

$$\sum_{s:(r,s)\in W} T_{rs}^{o} = M_{r}\left(\left(\sum_{s\in S} T_{sr}^{v}\right) w_{r}^{v}, \left(\sum_{s:(r,s)\in W} Q_{rs}\right) w_{r}^{c}\right) \forall r \in R \tag{22}$$

$$\sum_{s\in S} T_{sr}^{v} = \sum_{s:(r,s)\in W} Q_{rs} \forall r \in R \tag{23}$$

$$T_{rs}^{o} = \begin{cases} Q_{rs} & \forall (r,s) \in W \\ 0 & \forall (r,s) \in W^{c} \setminus W \end{cases} \tag{24}$$

RV fleet conservations

$$\sum_{(r,s)\in W} T_{rs}^{o} h_{rs} + \sum_{(s,r):s\in S, r\in R} T_{sr}^{v} \cdot (h_{sr} + w_{r}^{v}) = N \tag{25}$$

where $\{\rho_{k}^{rs}\}$ are auxiliary variables, and $\{t_{ij}\}_{(i,j)\in A}$ denote link performance functions which increase monotonically on the corresponding link flows. The existence of equilibria for system (9)-(25) is proven in Appendix B, as a specialization of the inter-node matching system (33)-(56) introduced in the next section.

2.2 Modeling Inter-node Matching Between Customers And Idle RVs

In this section, we relax the intra-node matching assumption in the base model to consider the inter-node matching between customers and idle RVs, which appears to be a common practice by ride-sourcing platforms. Due to the spatial heterogeneity in the ride-sourcing markets, platforms frequently dispatch idle RVs from neighborhoods with excessive supply to those experiencing shortage. As a consequence, RVs are often matched to customers who are several miles away. Modifications should be made to handle this type of long-distance spatial matching.

One straightforward remedy is to enlarge each node to cover a relatively large area with internally balanced demand and supply. However, such an evading strategy may compromise the representativeness and accuracy of the established model, as the aggregate matching function is likely to be biased if intra-node heterogeneities are left out. More importantly, as the node "grows" larger, the intra-node traffic becomes substantial and comparable to the inter-node traffic, defeating the original intent of conducting a network equilibrium analysis.

We thus resort to modeling the long-distance matching among different nodes, while keeping the node itself to be isotropic and at the neighborhood scale. For each node $r \in R$, the spatial matching radius is sculptured by a set of nodes $M^c(r)$, to which the hailing customers at node r can potentially be matched. Let $L = \cup_{r \in R} M^c(r)$. Note that these matching sets are not mutually exclusive. Reversely, there is also another nested matching set $M^v(l) \subseteq R$ for idle RVs on each $l \in L$. In this study, we simply assume that all the nested sets $\{M^c(r)\}$ and $\{M^v(l)\}$ are exogenously predetermined, and then use T_{lr}^m to denote the rate of RVs matched from node $l \in L$ to $r \in M^v(l)$ [1].

2.2.1 Customer Demand Under Inter-node PickupsCustomers' travel costs now explicitly embody the inter-node pickup time:

$$C_{rs} = g_{rs} + \beta^o w_r^c + \beta^m \breve{h}_r + (\beta^i + \beta^f) h_{rs}, \quad \forall (r,s) \in W$$

where β^m denotes customers' value of the pickup time (\$/h), and $\breve{h}_r$ is the average inter-node pickup time for customers on node r:

$$\breve{h}_r = \frac{\sum\limits_{l \in M^c(r)} (T_{lr}^m + \epsilon^m) h_{lr}}{\sum\limits_{l \in M^c(r)} (T_{lr}^m + \epsilon^m)}, \quad \forall r \in R \tag{26}$$

The small constant ϵ^m features another treatment defined to ensure the feasibility of mapping (26) when all the flow T_{lr}^m s are zero. We note that the use of ϵ^m here is also innocuous, because the zero-flow condition essentially arises when w_r^c approaches infinity, under which any $\breve{h}_r$ valued from the convex hull formed by $\{h^{lr}\}$ becomes nil in comparison.

The customer demand Q_{rs} then subjects to a function reformulated as follows:

$$Q_{rs} = f_{rs}\left(\beta^o w_r^c + \beta^m \breve{h}_r + (\beta^i + \beta^f) h_{rs}\right), \quad \forall (r,s) \in W$$

2.2.2 Idle RVs' Search Target Zones

The movements of idle RVs are also different as drivers now choose their search target nodes based on all the potential matching outcomes, including the matches to passengers at other nodes in the matching set of a target node. So the utility function of an idle RV cruising from node $s \in S$ to any node $l \in L$ is respecified as:

$$U_{sl} = \hat{F}_l - \gamma \cdot (\hat{h}_l + h_{sl} + w_l^v), \quad \forall s \in S, l \in L$$

where $\hat{F}_l$ and $\hat{h}_l$ are the average fare and service time of RVs who get matched at node l. Note that the service time now consists of two parts, respectively corresponding to deadheading to pick up the matched passenger and then delivering him or her. We can expand $\bar{F}_l$ and $\bar{h}_l$ respectively as follows,

$$\hat{F}_l = \frac{\sum\limits_{r \in M^v(l)} (T_{lr}^m + \epsilon^m) \bar{F}_r}{\sum\limits_{r \in M^v(l)} (T_{lr}^m + \epsilon^m)}, \quad \hat{h}_l = \frac{\sum\limits_{r \in M^v(l)} (T_{lr}^m + \epsilon^m)(h_{lr} + \bar{h}_r)}{\sum\limits_{r \in M^v(l)} (T_{lr}^m + \epsilon^m)}, \quad \forall l \in L$$

[1] Intra-node matching flows in this framework are essentially indicated by $\{T_{rr}^m\}$, $\forall r \in R$.

where $\overline{F}_r$ and $\overline{h}_r$ still follow the previous definitions in Eq. (3). By assuming drivers' perceptual errors on utilities follow the Gumbel distribution, we then write the idle RV flow T_{sl}^v under the inter-node matching scenario as

$$T_{sl}^v = \frac{\exp(\theta U_{sl})}{\sum_{k \in L} \exp(\theta U_{sk})} \cdot \sum_{k:(k,s)\in W} T_{ks}^o, \quad \forall s \in S, l \in L$$

2.2.3 Inter-node Matching Functions

Previously, the intra-node matching flow is estimated from the numbers of hailing customers and idle vehicles at the same node, using a single-output matching function. With inter-node matching, the idle vehicles/hailing customers at each node can possibly be matched to customers/RVs at a set of nodes. Consequently, a multi-output matching function needs to be developed to delineate aggregately the matching process. Figure 1a outlines a conceptual instance for inter-node matching, where the blue circles on the left and the green circles on the right respectively represent the pools of idle RVs and hailing customers at different nodes. Let N_l^v and N_r^c respectively denote the average number of entities respectively at node $l \in L$ and $r \in R$ under the stationary state. Then, as shown by Figure 1a, the RV accumulations N_1^v at node 1 are "digested" by flows T_{11}^m and T_{12}^m, while the customer accumulations N_1^c are matched to flows of T_{11}^m and T_{12}^m etc. Therefore, when determining T_{11}^m, the knowledge on N_1^v and N_1^c is not sufficient. We also need to know flows such as T_{12}^m and T_{21}^m, which may further depend on $\{T_{2r}^m\}$ and $\{T_{l2}^m\}$ etc. Propagating by nodes and links, the inter-dependencies are likely to integrate over the whole set of matching flows $\{T_{lr}^m\}_{l \in L, r \in R}$ throughout a well-connected network.

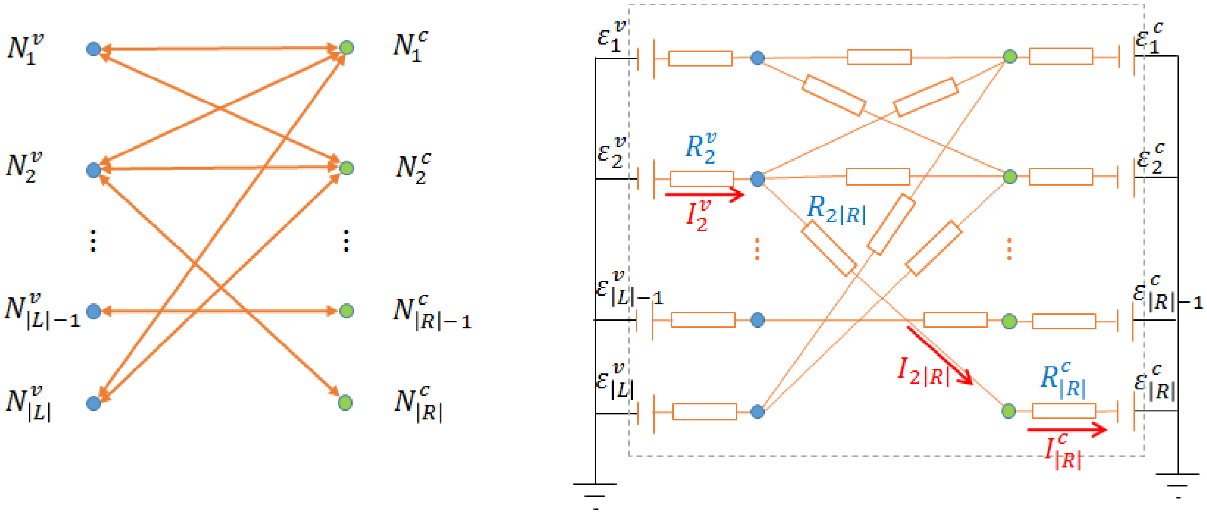
Another challenge arises from the matching priority issue. Taking Figure 1a as an example, there are two possible matching outcomes for RVs in N_1^v, either node 1 or 2. Suppose customer requests from node 1 can be served by RVs from the same node with less pick-up time, compared to the counterparts at node 2. Then, idle RVs at node 1 will be prioritized to match with closer customers, i.e. the customers at node 1 in this case. The modeling of such a priority is nontrivial, because the comparative nature of priorities essentially gives rise to asymmetries for the paired nodes as well as the interdependences between pairs.

Our inter-node matching function is built upon an analogy to electric circuits. For each inter-node matching graph (e.g. Figure 1(a)), an electric circuit can be constructed correspondingly by duplicating the connections with resistances (see Figure 1(b)). Additionally, each node outstretches a separate resistance as well as an external power supply, and connects in parallel to constitute a circuit. Denote ε_i^t, R_i^t and I_i^t respectively as the power voltage, the resistance and the current on each node's individual branch, where $\{(t,i) \mid t \in \{v,c\}; i \in L, \text{if} t = v; i \in R, \text{if} t = c\}$. Denote R_{lr} and I_{lr} respectively as the resistance and the current on connections between the paired nodes l and r, where $l \in L$ and $r \in R$. Then, the currents on the graph are subject to the following relationships as per Kirchhoff's circuit laws:

$$(\varepsilon_l^v - R_l^v \cdot I_l^v) - (R_r^c \cdot I_r^c - \varepsilon_r^c) = R_{lr} \cdot I_{lr} \forall (l,r) \in \{(l,r) \mid r \in M^v(l), l \in L\} \tag{27}$$

$$I_l^v = \sum_{r \in M^v(l)} I_{lr} \forall l \in L \tag{28}$$

$$I_r^c = \sum_{l \in M^c(r)} I_{lr} \forall r \in R \tag{29}$$



(a) (b)

Figure 1 The analogy of (a) inter-node matching flows to (b) currents in an electric circuit

The two terms on the left-hand side of Eq. (1) respectively represent the potentials at the blue and green nodes, while the right-hand side calculates the potential difference in terms of Ohm's law. The latter two equations Eq. (2) and (3) essentially represent current conservation at nodes. We note that such an electric-circuit framework mimics or approximates the inter-node matching process and its outcomes. The current on each branch of the circuit is analogous to the matching flow between the corresponding paired nodes, and the voltage of each power supply measures the accumulation of entities at the corresponding node. For each individual branch, higher voltage/accumulation will yield larger current/flow. Further, the resistances in the electric circuit characterize the time that RVs/customers spend on each process. Specifically, the separate-node resistance represents the waiting time of RVs/customers on each node, while the connecting-node resistance quantifies RVs' deadheading or pick-up time for customers in between. The interdependencies of currents/flows thus transmit rotationally through the two types of resistances/residence time and propagate systematically to the whole graph.

In light by the above analogy, we construct the inter-node matching function following a similar principle. Denote ϕ_l^v and ϕ_r^c respectively as the RV and customer potentials at different nodes, and define them respectively as the following functions[2],

$$\phi_l^v = \log\left(\Phi^v\left(T_l^v, N_l^v\right)\right) \forall l \in L$$

$$\phi_r^c = -\log\left(\Phi^c\left(T_r^c, N_r^c\right)\right) \forall r \in R$$

where the potential function Φ (including Φ^v and Φ^c) ranges in $(0,+\infty)$, decreasing on the cumulative flow T and increasing on the accumulation N, i.e. $\frac{\partial \Phi}{\partial T} < 0$ and $\frac{\partial \Phi}{\partial N} > 0$. Note that the waiting time w does not appear in Φ's variable argument list, because in this case w can be directly written as a function of N over T. Besides, we define the potential difference on each pair of connected nodes as the flow and travel time in between, i.e.

$$\delta\phi_{lr} = \log\left(\Delta(T_{lr}^m, h_{lr})\right) \forall (l,r) \in \{(l,r) \mid r \in M^v(l), l \in L\}$$

[2] The logarithm associated with the potentials (also, the potential differences defined later) seems unnecessary, but it facilitates making a connection with a Cobb-Douglas-type matching function.

where the function Δ also ranges from $(0,+\infty)$, and $\frac{\partial\Delta}{\partial T^m} > 0$, $\frac{\partial\Delta}{\partial h} > 0$.

Then, by setting the accumulations $\{N_l^v\}$, $\{N_r^c\}$ and node-transfer time $\{h_{lr}\}$ as given parameters, the resultant matching flow pattern $\{T_{lr}\}$ in line with Eq. (1) solves the following equation system[3],

$$\Phi^v\left(T_l^v, N_l^v\right)\cdot\Phi^c\left(T_r^c, N_r^c\right) = \Delta(T_{lr}^m, h_{lr})\forall(l,r)\in\{(l,r)\,|\,r\in M^v(l), l\in L\} \tag{30}$$

$$T_l^v = \sum_{r\in M^v(l)} T_{lr}^m \forall l\in L \tag{31}$$

$$T_r^c = \sum_{l\in M^c(r)} T_{lr}^m \forall r\in R \tag{32}$$

We assume the functions $\Phi(T,N)$, including Φ^v and Φ^c, as well as $\Delta(T,h)$ additionally satisfy the following properties (with implications clarified below), given any N and h with positive and finite values:

- Domain of definition - both $\Phi(T,N)$ and $\Delta(T,h)$ are continuous functions defined on $T\in(0,+\infty)$.

  For each pair of nodes with one locating in the matching range of the other, there will have positive flows matched in between.

- Boundary conditions - $\lim_{T\to 0^+}\Phi(T,N) = +\infty$ and $\lim_{T\to+\infty}\Phi(T,N) = 0$;

  These two conditions can be obtained conceptually, as the accumulation N is stuck in the matching process when $T\to 0^+$ and dissipates instantaneously for $T\to+\infty$.

  $\lim_{T\to 0^+}\Delta(T,h) = 0$ and $\lim_{T\to+\infty}\Delta(T,h) = +\infty$.

  As per the Ohm's law, less potential differences are associated with with less matching flows through the time impedance.

- Limiting behavior - there exist $p > 0$ and $q > 0$ such that[4]

$$\Phi(T,N) = \{[l]\Theta(T^{-p}), \text{as} T\to 0^+$$

$$\Theta(T^{-q}), \text{as} T\to+\infty.$$

  This implies a diminishing marginal rate of substitution in the matching process. We let Φ possibly converge/grow on T with different speed when approaching $0^+ / +\infty$, respectively.

Based on these general properties on Φ and Δ, the existence and uniqueness of solutions $\{T_{lr}^m\}$ to system (30)-(32) are proved in Appendix A.

2.2.4 Network Equilibrium Under Inter-node Matching condition

Based on the modifications for inter-node matching derived in this section, we rewrite the network equilibrium conditions (19)-(25) to reflect the inter-node matching flows. The complete set of OD pairs W^c now includes those for idle RV trips, deadheading RV trips, occupied RV trips as well as trips made by regular traffic, i.e.

$$W^c = \{(s,l)\,|\,s\in S, l\in L\}\cup\{(l,r)\,|\,l\in L, r\in M^v(l)\}\cup W\cup W^b$$

Then, the equilibrium link flow distribution $\{x_{ij}^{rs}\}$ satisfies the following conditions:

[3] Eq. (1a) results from the internal relations among potentials that $\phi_l^v - \phi_r^c = \delta\phi_{lr}$.

[4] Big Θ pertains to one of the Bachmann-Landau notations. By writing $f(n) = \Theta(g(n))$, it means f is asymptotically bounded both above and below by g.

Path equilibrium between OD pairs

$$\left[t_{ij}(v_{ij}) - \rho_i^{rs} + \rho_j^{rs}\right] x_{ij}^{rs} = 0 \forall (i,j) \in A, (r,s) \in W^c \tag{33}$$

$$t_{ij}(v_{ij}) - \rho_i^{rs} + \rho_j^{rs} \geqslant 0 \forall (i,j) \in A, (r,s) \in W^c \tag{34}$$

$$v_{ij} = \sum_{(r,s)\in W^c} x_{ij}^{rs} \forall (i,j) \in A \tag{35}$$

$$x_{ij}^{rs} \geqslant 0 \forall (i,j) \in A, (r,s) \in W^c \tag{36}$$

$$T^{rs} = T_{rs}^{v} + T_{rs}^{m} + T_{rs}^{o} + T_{rs}^{n} \forall (r,s) \in W^c \tag{37}$$

$$\sum_{i:(i,k)\in A} x_{ik}^{rs} - \sum_{j:(k,j)\in A} x_{kj}^{rs} = \begin{cases} -T^{rs}, & \text{if} k = r \\ T^{rs}, & \text{if} k = s \quad \forall (r,s) \in W^c \\ 0, & \text{otherwise} \end{cases} \tag{38}$$

$$h_{rs} = \rho_r^{rs} - \rho_s^{rs} \forall (r,s) \in W^c \tag{39}$$

Customer demand

$$Q_{rs} = f_{rs}\left(\beta^o w_r^c + \beta^m \breve{h}_r + (\beta^i + \beta^f) h_{rs}\right) \forall (r,s) \in W \tag{40}$$

$$\breve{h}_r = \frac{\sum_{l\in M^c(r)} (T_{lr}^m + \epsilon^m) h_{lr}}{\sum_{l\in M^c(r)} (T_{lr}^m + \epsilon^m)} \forall r \in R \tag{41}$$

$$F_{rs} = F_{rs}^0 + \beta^f h_{rs} \forall (r,s) \in W \tag{42}$$

Idle RV movements

$$U_{sl} = \hat{F}_l - \gamma \cdot (\hat{h}_l + h_{sl} + w_l^v) \forall s \in S, l \in L \tag{43}$$

$$\hat{F}_l = \frac{\sum_{r\in M^v(l)} (T_{lr}^m + \epsilon^m) \overline{F}_r}{\sum_{r\in M^v(l)} (T_{lr}^m + \epsilon^m)} \forall l \in L \tag{44}$$

$$\hat{h}_l = \frac{\sum_{r\in M^v(l)} (T_{lr}^m + \epsilon^m)(h_{lr} + \overline{h}_r)}{\sum_{r\in M^v(l)} (T_{lr}^m + \epsilon^m)} \forall l \in L \tag{45}$$

$$\overline{F}_r = \frac{\sum_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o) F_{rs}}{\sum_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o)} \forall r \in R \tag{46}$$

$$\overline{h}_r = \frac{\sum_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o) h_{rs}}{\sum_{s:(r,s)\in W} (T_{rs}^o + \epsilon^o)} \forall r \in R \tag{47}$$

$$T_{sl}^{v}=\begin{cases}\dfrac{\exp(\theta U_{sl})}{\sum\limits_{k\in L}\exp(\theta U_{sk})}\cdot\sum\limits_{k:(k,s)\in W}T_{ks}^{o},\ \forall s\in S,l\in L\\ 0,\ \forall(s,l)\in W^{c}\setminus\{(s,l)\mid s\in S,l\in L\}\end{cases}\tag{48}$$

$$T_{rs}^{o}=\begin{cases}Q_{rs} & \forall(r,s)\in W\\ 0 & \forall(r,s)\in W^{c}\setminus W\end{cases}\tag{49}$$

Inter-node matching

$$\Phi^{v}\left(T_{l}^{v},w_{l}^{v}\cdot T_{l}^{v}\right)\cdot\Phi^{c}\left(T_{r}^{c},w_{r}^{c}\cdot T_{r}^{c}\right)=\Delta\left(T_{lr}^{m},h_{lr}\right)\forall l\in L,r\in M^{t}(l)\tag{50}$$

$$T_{l}^{v}=\sum_{r\in M^{v}(l)}T_{lr}^{m}\forall l\in L\tag{51}$$

$$T_{r}^{c}=\sum_{l\in M^{c}(r)}T_{lr}^{m}\forall r\in R\tag{52}$$

$$\sum_{r\in M^{v}(l)}T_{lr}^{m}=\sum_{s\in S}T_{sl}^{v}\forall l\in L\tag{53}$$

$$\sum_{l\in M^{c}(r)}T_{lr}^{m}=\sum_{s:(r,s)\in W}T_{rs}^{o}\forall r\in R\tag{54}$$

$$T_{lr}^{m}=0,\quad\forall(l,r)\in W^{c}\setminus\{(l,r)\mid l\in L,r\in M^{v}(l)\}\tag{55}$$

RV fleet conservations

$$\sum_{(r,s)\in W^{c}}(T_{rs}^{o}+T_{rs}^{v}+T_{rs}^{m})\cdot h_{rs}+\sum_{(s,l):s\in S,l\in L}T_{sl}^{v}\cdot w_{l}^{v}=N\tag{56}$$

Again, $\{\rho_{k}^{rs}\}$ are auxiliary variables. The existence of an equilibrium solution to the above system (33)-(56) is proved in Appendix B[5]. We also prove there that the zero-flow condition will not arise in equilibrium solution. As system (33)-(56) degenerates into (9)-(25) when the matching range of each node shrinks to only cover itself, the proof readily guarantees the intra-node matching system with an equilibrium solution.

2.3 Solution Procedure

In this section, we develop an iterative algorithm to solve the network equilibrium conditions (33)-(55). Before presenting the solution procedure, we first reformulate some conditions into mathematical problems, which can be easily solved by using commercial solvers.

Specifically, given $(\mathbf{T^{v}},\mathbf{T^{m}},\mathbf{T^{n}})$, Eq. (33)-(38) can be reformulated as a convex problem below:

$$\begin{aligned}&\text{PE:}\\&\min_{\mathbf{x},\mathbf{v}}\sum_{(i,j)\in A}\int_{0}^{v_{ij}}t_{ij}(\varpi)d\varpi\\&\text{s.t. }(33)-(38)\end{aligned}$$

Furthermore, treating $\mathbf{T^{m}},\mathbf{T^{o}}$, and $\mathbf{h}$ as exogenous variables, the idle RV movements, i.e., Eq. (42)-(47), can be captured by the following mathematical program:

[5] We suggest readers to go over Section 2.3 before detouring to the appendix, as some mapping systems defined in the former are referred in the proof.

IRVM:

$$\min_{\mathbf{T^v}} \sum_{s\in S}\sum_{l\in L}\left\{T_{sl}^{v}\left[-\hat{F}_l+\gamma\left(\hat{h}_l+h_{sl}\right)\right]+\frac{1}{\theta}T_{sl}^{v}\left(\ln T_{sl}^{v}-1\right)\right\}$$

s.t. (51)

$$\sum_{l\in L}T_{sl}^{v}=\sum_{r:(r,s)\in W}T_{rs}^{o}\forall s\in S \tag{57}$$

The KKT conditions of IRVM can be stated as follows:

(51),(57)

$$-\hat{F}_l+\gamma\left(\hat{h}_l+h_{sl}\right)+\frac{1}{\theta}\ln T_{sl}^{v}+\beta_l+\tau_s=0\forall l\in L,s\in S \tag{58}$$

where $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$ are Lagrangian multipliers associated with (51) and (57). From (58), we have

$$-\hat{F}_l+\gamma\left(\hat{h}_l+h_{sl}\right)+\frac{1}{\theta}\ln T_{sl}^{v}+\beta_l=-\hat{F}_k+\gamma\left(\hat{h}_k+h_{sk}\right)+\frac{1}{\theta}\ln T_{sk}^{v}+\beta_k \quad \forall l,k\in L,s\in S$$

which is equivalent to

$$\frac{T_{sl}^{v}}{T_{sk}^{v}}=\frac{\exp\left\{\theta\left[\hat{F}_l-\gamma\left(\hat{h}_l+h_{sl}+\dfrac{\beta_l}{\gamma}\right)\right]\right\}}{\exp\left\{\theta\left[\hat{F}_k-\gamma\left(\hat{h}_k+h_{sk}+\dfrac{\beta_k}{\gamma}\right)\right]\right\}},\ \forall l,k\in L,s\in S$$

and further yields

$$\frac{T_{sl}^{v}}{\sum_{k\in L}T_{sk}^{v}}=\frac{T_{sl}^{v}}{\sum_{r:(r,s)\in W}T_{rs}^{o}}=\frac{\exp\left\{\theta\left[\hat{F}_l-\gamma\left(\hat{h}_l+h_{sl}+\dfrac{\beta_l}{\gamma}\right)\right]\right\}}{\sum_{r\in R}\exp\left\{\theta\left[\hat{F}_k-\gamma\left(\hat{h}_k+h_{sk}+\dfrac{\beta_k}{\gamma}\right)\right]\right\}},\ \forall l,k\in L,s\in S$$

From the above formula, we can interpret $\frac{\beta_l}{\gamma}$ as the RV waiting time at node l, i.e., $w_l^{v}=\frac{\beta_l}{\gamma},l\in L$. As $\frac{\beta_l}{\gamma}$ is not unique, we can always add a constant η to $\frac{\beta_l}{\gamma}$, such that the above equation still holds.

As per (55), we have

$$\sum_{(r,s)\in W^{c}}\left(T_{rs}^{v}+T_{rs}^{m}+T_{rs}^{o}\right)h_{rs}+\sum_{(s,l):s\in S,l\in L}\left(\frac{\beta_l}{\gamma}+\eta\right)T_{sl}^{v}=N$$

which gives rise to

$$\eta=\frac{N-\sum_{(r,s)\in W^{c}}(T_{rs}^{v}+T_{rs}^{m}+T_{rs}^{o})h_{rs}-\sum_{(s,l):s\in S,l\in L}T_{sl}^{v}\cdot\dfrac{\beta_l}{\gamma}}{\sum_{(s,l):s\in S,l\in L}T_{sl}^{v}}$$

and accordingly,

$$w_l^v = \frac{\beta_l}{\gamma} + \eta$$

Given the above deduction, we know that given $\mathbf{T^m}, \mathbf{T^o}$, and $\mathbf{h}$, the idle RV movements, i.e., Eq. (42)-(47), can be delineated by IRVM. The following solution procedure is developed to solve the system (33)-(56) (see Figure 2 for the corresponding flow chart):

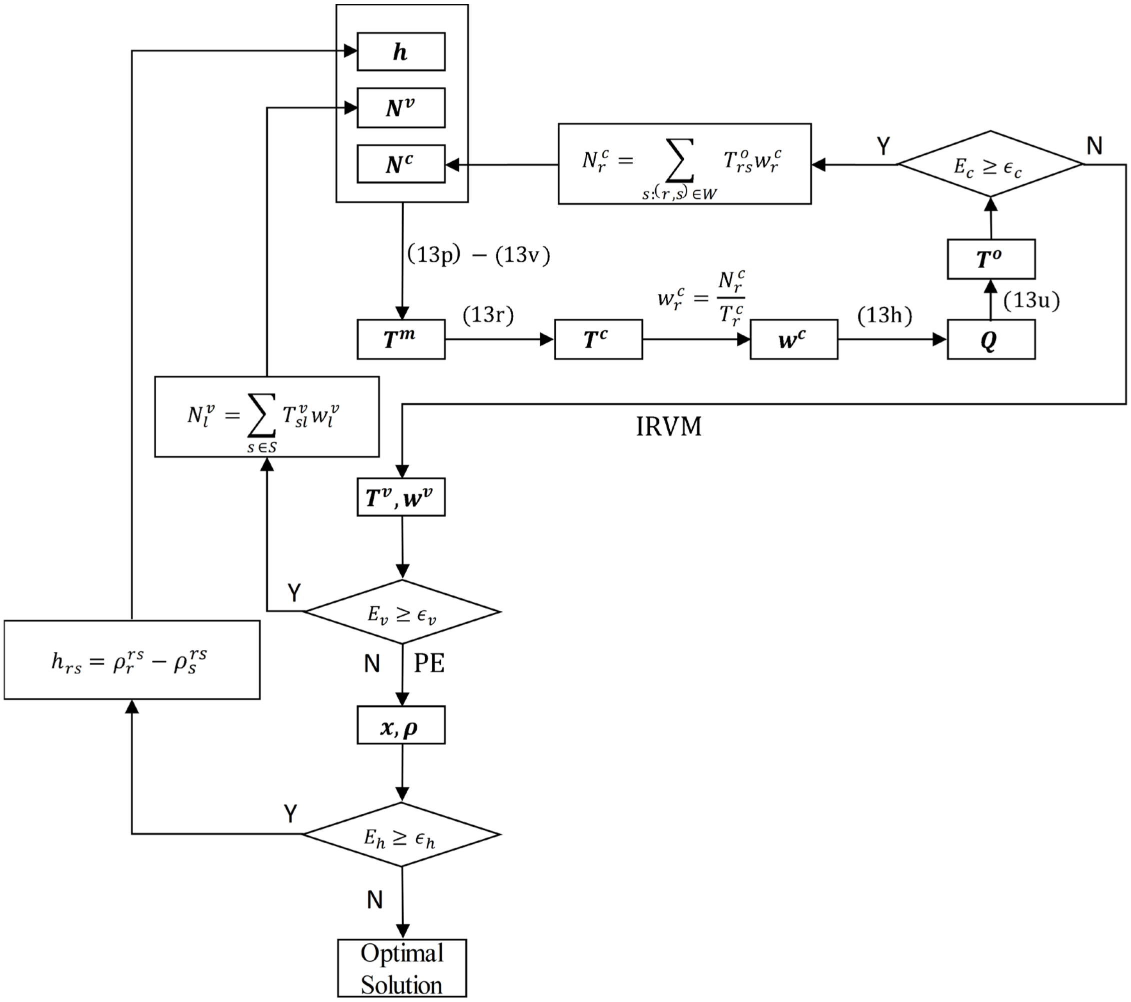


Figure 2 Solution procedures

1. Initialize $\mathbf{h}$, $\mathbf{N^v}$, and $\mathbf{N^c}$.
2. Obtain $\mathbf{T^m}$ by solving (48)-(50), and then deduce $\{\mathbf{T^c}, \mathbf{w^c}, \mathbf{Q}, \mathbf{T^o}\}$. If $E_c = \sum_{r\in R} | N_r^c - \sum_{s:(r,s)\in W} T_{rs}^o w_r^c | \geq \varepsilon_c$, then update $N_r^c = \sum_{s:(r,s)\in W} T_{rs}^o w_r^c, \forall r \in R$, and repeat this step; otherwise, go to Step 3.
3. Obtain $\{\mathbf{T^v}, \mathbf{w^v}\}$ by solving IRVM. If $E_v = \sum_{l\in L} | N_l^v - \sum_{s\in S} T_{sl}^v w_l^v | \geq \varepsilon_v$, then update $N_l^v = \sum_{s\in S} T_{sl}^v w_l^v, \forall l \in L$, and go to Step 2; otherwise, go to Step 4.
4. Obtain $\{\mathbf{x}, \boldsymbol{\rho}\}$ by solving PE. If $E_h = \sum_{(r,s)\in W^c} | h_{rs} - \rho_r^{rs} + \rho_s^{rs} | \geq \varepsilon_h$, then update $h_{rs} = \rho_r^{rs} - \rho_s^{rs}, \ \forall (r,s) \in W^c$, and go to Step 2; otherwise, stop and the obtained

$\{\mathbf{h}, \mathbf{T}^{\mathrm{m}}, \mathbf{T}^{\mathrm{c}}, \mathbf{T}^{\mathrm{v}}, \mathbf{T}^{\mathrm{o}}, \mathbf{Q}, \mathbf{w}^{\mathrm{v}}, \mathbf{w}^{\mathrm{c}}, \mathbf{x}, \boldsymbol{\rho}\}$ is the optimal solution.

where ε_c, ε_v, and ε_h are given tolerances.

3 Trip-based Graph Partitioning for Parallel Computing in Ridesharing

We start this section by forming a bipartite graph on which the ride-matching problem can be formulated. As the number of system participants increases, finding the max cardinality matching on this graph becomes more computationally expensive. Therefore, we propose a graph portioning scheme that approximates finding the max cardinality matching on the original graph by finding max cardinality matching on a number of mutually disjoint sub-graphs.

3.1 Max Cardinality Bipartite Matching

The one-to-one matching problem can be represented by a bipartite graph $G = (P, L)$, where P is the set of participants and L is the link set. Let us denote the set of riders by $R = \{r\}$ and the set of drivers by $D = \{d\}$. Rider and driver sets are mutually exclusive, and form the set of participants, $P = \{p\} = R \cup D$, which constitute the set of nodes in graph G.

We assume that the study region consists of m stations, in which participants start or finish their trips, denoted by $s_1, s_2, \cdots, s_m$. Let us characterize the trip of participant $p \in P$ by its origin and destination station, denoted by s_p^O and s_p^D, respectively, and the earliest departure time from the trip origin, denoted by t_p^{ETD}. The latest arrival time of the participant at the trip's destination station, denoted by t_p^{LTA}, can be computed as the sum of earliest departure time, travel time on the shortest path from the origin station to the destination station, and a detour time. The maximum detour time for participant p, denoted by t_p^{det}, can either be specified by the participant or decided by the system operator, and correlates inversely with level of service. A link $\ell = (r, d) \in L$ between rider r and driver d exists in graph G if the driver is capable of serving the rider before heading to his/her own destination. This condition can be mathematically expressed by having equations (59) and (60) hold concurrently, where $t_{i,j}$ is the shortest path travel time between stations i and j. Equation (59) ensures that a driver who arrives at a rider's origin station as early as possible would have enough time left to carry the rider to his/her destination. Equation (60) states that after dropping off the rider, the driver should have enough time left to accomplish his/her own trip.

$$\max\{t_d^{ETD} + t_{s_d^O, s_r^O}, t_r^{ETD}\} + t_{s_r^O, s_r^D} \le t_r^{LTA} \tag{59}$$

$$\max\{t_d^{ETD} + t_{s_d^O, s_r^O}, t_r^{ETD}\} + t_{s_r^O, s_r^D} + t_{s_r^D, s_d^D} \le t_d^{LTA} \tag{60}$$

The goal of the ridesharing system is to find the max cardinality solution in this graph, i.e., the solution that yields the highest matching rate. The matching problem can be mathematically formulated as a binary program in model (61)-(64). The decision variable u_{rd} is a binary variable that holds the value 1 if rider r is matched with driver d, and the value 0 otherwise. The objective function in equation (61) maximizes the total number of served riders, which equates to maximizing the matching rate. Constraint (62) ensures that each driver is assigned to at most one rider, and constraint (63) ensures that each rider gets served at most once. Notice that although the decision variable u is by definition a binary variable, the total unimodularity structure of the constraint set allows for relaxing the binary constraint and replacing it with equation (64).

$$\mathit{Maximize} \quad \sum_{r \in R} \sum_{d \in D} u_{rd} \tag{61}$$

$$s.t. \quad \sum_{r\in R} u_{rd} \leq 1, \forall\ d \in D : (r,d) \in L \tag{62}$$

$$\sum_{d\in D} u_{rd} \leq 1, \forall\ r \in R : (r,d) \in L \tag{63}$$

$$0 \leq u_{rd} \leq 1 \tag{64}$$

Although this matching problem can be solved reasonably fast using commercial optimization engines or custom-designed algorithms (see e.g. (Hopcroft, 1973), (Ford, 1956)) for small- to medium-sized problems, in ridesharing systems that operate over a large region, the large scale of the problem can prohibit it from being solved in real-time, or at all. In this paper, we propose a graph partitioning method to bisect the bipartite graph G into disjoint sub-graphs, creating smaller-size problems that are computationally less expensive to solve.

3.2 Connectivity Matrix

Let us form the binary matrix F based on graph G, such that each element $(r,d) \in R \times D$ of this matrix holds the value 1 if r and d are connected in the bipartite matching graph, and the value 0 otherwise. We call matrix F the "connectivity matrix" of graph G, denoted by F_G (hereafter referred to as F for simplicity).

Decomposing the connectivity matrix F into sub-matrices corresponds to partitioning graph G into sub-graphs. Figure 3 displays an example of such partitioning. In this figure, the matrix has been divided into four sub-matrices at splitting points b_v and b_h. Level 1 partitioning as depicted in this figure will provide sub-matrices $F_i, i = \{1,2,3,4\}$, while level 2 partitioning corresponds to decomposing level 1 sub-matrices, and will result in sub-matrices $F_{i,j}, (i,j) \in \{1,2,3,4\} \times \{1,2,3,4\}$. In general, level i of partitioning is equivalent to dividing all partitions in level $(i-1)$ into four partitions.

In Figure 3, the sub-graph corresponding to F_1 includes drivers in set D_1 and riders in set R_1. Similarly, the sub-graph of F_4 includes driver in D_2 and riders in R_2. Hence, since $R_1 \cup R_2 = R$, $R_1 \cap R_2 = \varnothing$, $D_1 \cup D_2 = D$, and $D_1 \cap D_2 = \varnothing$, the sub-graphs associated with F_1 and F_4 are exhaustive and disjoint in set P. Similarly, F_2 and F_3 correspond to exhaustive disjoint sub-graphs.

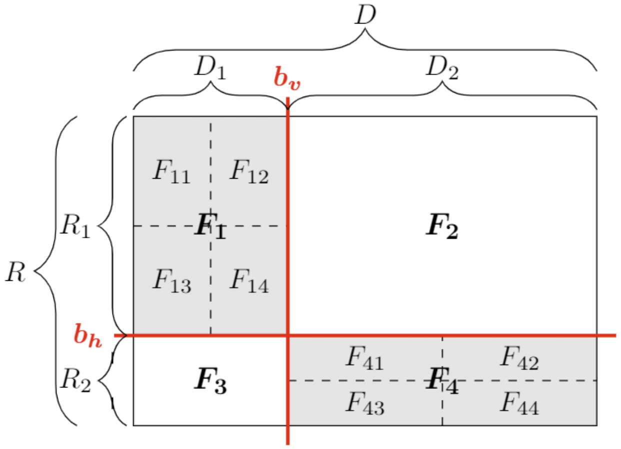


Figure 3 Partitioning the connectivity matrix F at splitting points b_v and b_h. Level 1 partitioning will provide sub-matrices $F_i, i = \{1,2,3,4\}$, while level 2 partitioning will provide sub-matrices $F_{i,j}, (i,j) \in \{1,2,3,4\} \times \{1,2,3,4\}$.

Hence, in order to approximate the matching problem associated with G with two smaller matching problems, we can solve the matching problems associated with either F_1 and F_4, or F_2 and F_3. Without loss of generality, we assume that F_1 and F_4 are always selected. Note that this procedure provides solutions to two sub-problems the union of which provides only an approximation of the solution to the original problem. The reason is that the unit elements of sub-matrices F_2 and F_3 correspond to links in G (i.e., potential matches) that have been removed in order to partition G into two disjoint sub-graphs.

3.3 Properties of a Desirable Partitioning

In the previous section, we noted that the splitting points b_v and b_h are required to decompose matrix F. There are, however, two sets of decisions that affect the quality of the sub-graphs resulting from this partitioning, namely the location of the splitting points, and the row and column orderings in matrix F.

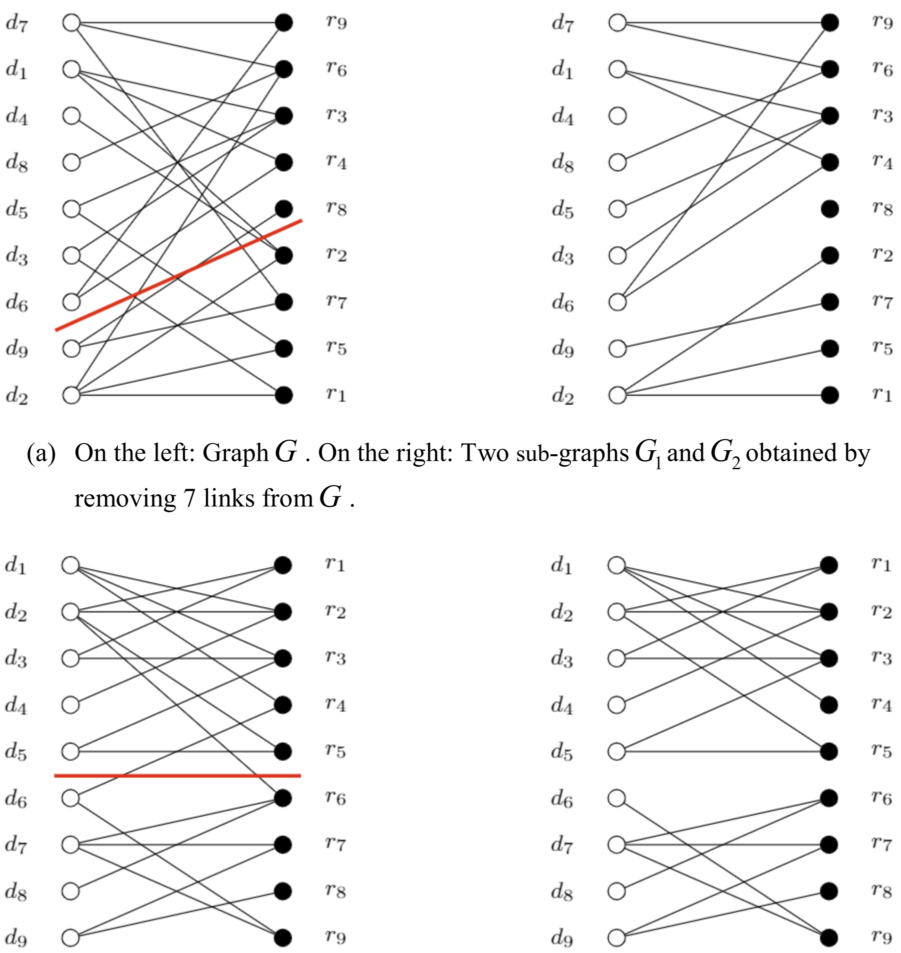
Choice of the splitting points, b_v and b_h, affects the quality of the solutions in two ways. Firstly, by moving the location of the splitting points, we are in fact changing the riders and drivers in the resultant sub-graphs. The proper selection of b_v and b_h would result in a partitioning that (i) removes as few links as possible, and (ii), results in a near uniform split of the graph. The reason for the former condition is that the number of links removed as a result of partitioning of G provides a bound on the performance of the partitioning procedure, since each removed link is a potential matching opportunity that is being ignored. The latter condition stems from the objective of partitioning, which is reducing the computational complexity of the problem. The number of computations required in each iteration of the Simplex algorithm is $\mathcal{O}(PL)$. Hence, a partitioning of the graph that is uniform in the number of nodes and links would provide the highest computational benefit.

Secondly, for a given set of splitting points, the ordering of riders and drivers in F leads to different sub-graphs. It is easy to see that we can re-order the identification numbers of riders and drivers in F before decomposing it, without making any changes to the original problem. Figure 4 showcases an example of how re-ordering riders and drivers in F can lead to more desirable partitions. In Figure 4(a), a good split, subject to uniformity in the size of the sub-graphs, requires omitting 7 links. Re-ordering the rider and driver sets, as presented in Figure 4(b), could lead to a much more desirable partitioning that removes only 2 links. Note that choice of the splitting points and row and column re-orderings need to take place concurrently. In the next section, we present the mathematical formulation of a partitioning that satisfies both conditions presented in this section.

3.4 The ε-uniform Graph Partitioning Problem

The goal of the ε-uniform graph partitioning problem is to partition graph G into K disjoint components by eliminating the minimum number of links, while guaranteeing that the number of nodes and links in partitions are within $\varepsilon_p\,|P|$ and $\varepsilon_\ell\,|L|$ of each other, respectively. This problem is mathematically formulated in model (65)-(70). Let us define the binary decision variable f_{pk} to take the value 1 if node $p \in P$ is assigned to sub-graph k, and the value 0 otherwise. Furthermore, we define the binary variable $h_{\ell k}$ to take the value 1 is link ℓ is assigned to sub-graph k, and the value 0 otherwise.

The objective function in equation (65) maximizes the total number of links within the partitions, hence minimizing the total number of omitted links with end points in different partitions. Constraint (66) ensures that each node is assigned to a single sub-graph. Constraint (67) ensures that only links connecting two nodes within the same sub-graph are selected; if both end nodes of link ℓ, p and p', belong to sub-graph



(a) On the left: Graph G . On the right: Two sub-graphs G_1 and G_2 obtained by removing 7 links from G .

(b) On the left: Graph G after row and column re-ordering. On the right: Two sub-graphs G_1 and G_2 obtained by removing 2 links from G .

Figure 4 Bipartite matching graph: Two re-orderings of same set of participants

k , the left-hand side of the constraint becomes 1, and the max objective function forces $h_{\ell,k}$ to take its upper bound of 1. Any other arrangement of 0 and 1 values assigned to variables f_{pk} and $f_{p'k}$ would make the left hand side of equation (66) smaller than 1, forcing the binary variable $h_{\ell k}$ to take the value 0.

Although equations (66) and (67) partition G into K sub-graphs, they do not impose any restrictions on the size of the resulting sub-graphs. Constraint (68) ensures that links are, within a threshold $\varepsilon_\ell\,|L|$, distributed uniformly among sub-graphs. It is easy to see that setting ε_ℓ closer to zero would distribute the links more uniformly among sub-graphs, reducing the upper bound on the computational complexity of the otherwise larger sub-problem. However, this reduced computational complexity may come at the cost of a higher number of eliminated links, and hence lower accuracy. Analogous to constraint (68), constraint (69) ensures a uniform distribution of nodes among sub-graphs, within the threshold $\epsilon_p\,|P|$. Constraint (70) enforces decision variables to take only binary values.

$$Maximize \quad \sum_{k=1}^{K}\sum_{\ell\in L} h_{\ell k} \tag{65}$$

$$s.t. \quad \sum_{k=1}^{K} f_{pk} = 1, \forall p \in P \tag{66}$$

$$\frac{f_{pk} + f_{p'k}}{2} \geq h_{\ell k}, \forall k \in \{1, \cdots, K\}, \ell \in (p, p') \in L \tag{67}$$

$$\sum_{\ell \in L} h_{\ell k} \leq (1 + \varepsilon_{\ell}) \frac{|L|}{K}, \forall k \in \{1, \cdots, K\} \tag{68}$$

$$\sum_{p \in P} f_{pk} \leq (1 + \varepsilon_{p}) \frac{|P|}{K}, \forall k \in \{1, \cdots, K\} \tag{69}$$

$$f, h \in \{0,1\} \tag{70}$$

It can be easily shown that the mathematical problem in model (65)-(70) is NP-hard. Additionally, creating graph G, which is an input to this problem, can create a computational bottleneck in real-time applications, since it requires $|R| \times |D|$ trip comparisons to generate set L, each trip comparison requiring computations in (59) and (60). In the next section, we present an algorithm to solve this problem fast and with a high level of accuracy, without the necessity of generating graph G a priori.

3.5 The Trip-based ε-uniform Partitioning Algorithm

As an alternative for the problem in (65)-(70), we introduce a "proxy" problem in which generating the entire graph G is not required a priori; rather, we strategically decide the parts of graph G that need to be generated. This problem aims at assigning N objects to K clusters such that the total distance between objects and their assigned cluster centers is minimized, and the clusters are of about the same size. Before presenting the formulation, we need to define the points/objects to be clustered, and elaborate on how the distance between two objects can be measured.

This problem considers each object to be a trip, characterized by latitude and longitude of both trip ends, and the average of its earliest departure time and latest arrival time. Let us define the trip vector $tr_n = (lat_{s_n^O}, lon_{s_n^O}, lat_{s_n^D}, lon_{s_n^D}, \theta_n)^\top$, where the first two terms are the latitude and longitude of the origin station of trip n, respectively. Similarly, the third and forth terms are the latitude and longitude of the destination station of this trip, respectively. The last term is the average of start time and end time of the trip in minutes, with 12 pm set as the reference point (i.e., time zero).

Let us define the distance matrix $C = [c_{nk}], \forall n \in \{1, \cdots, N\}, k \in \{1, \cdots, K\}$, to hold the distance between the center of cluster k (which is the representative trip of the cluster) and trip n. In order to compute the distance between two trips tr_n and $tr_{n'}$, the elements of the two vectors should be expressed in the same unit. Towards this end, we compute the travel time between two points with the same longitude/latitude, but a difference of 1000 meters in latitude/longitude in the region where the ridesharing system is operating. This can be done by first replacing the coordinates of the trips, which are originally in the latitude/longitude coordinate system, with the Universal Transverse Mercator (UTM) coordinate system, and then converting the distances to travel time (in minutes), assuming an average speed for the entire region. Let us denote the travel time between these two points as α. We define $\omega_{lat} / \omega_{lon}$ to be set as $\frac{\alpha}{1000}$, and interpret it as the trip length (in minutes) between two points with the same latitude/longitude and a difference of 1000 meters in longitude/latitude in the region under study. Consequently, let us define the weight vector $\boldsymbol{\omega} = (\omega_{lat}, \omega_{lon}, \omega_{lat}, \omega_{lon}, 1)^\top$, and compute distances as:

$$c_{nk} = (\boldsymbol{\omega} \circ (tr^k - tr_n))^\top (\boldsymbol{\omega} \circ (tr^k - tr_n)), \tag{71}$$

where tr^k is the center of cluster k which can be found by taking the average of trip vectors in cluster k.

As a result, the proxy problem can be formulated as a mixed integer, non-linear program presented in (72)-(77). Let us define the binary decision variable q_{nk} to take the value 1 if point $n \in \{1, \cdots, N\}$ is assigned to cluster $k \in \{1, \cdots, K\}$, and the value 0 otherwise. We further define a non-negative vector tr^k to represent the center of cluster k. The non-linear objective function in equation (72) minimizes the total distance between trips and their associated cluster centers. Note that c_{nk} is not constant in this problem, since it involves the tr^k variables. Constraint (73) ensures that each trip is assigned to a single cluster.

Let us define the indicator parameter I_n to assume the value 1 if point n is a rider trip, and the value 0 if point n is a driver trip. Constraint (74) ensures that the difference between the number of riders between any two clusters is at most $2 \times \varepsilon |R|$. Similarly, constraint (75) ensures that the difference between the number of drivers between any two clusters is at most $2 \times \varepsilon |D|$.

$$Minimize \quad \sum_{k=1}^{K}\sum_{n=1}^{N} c_{nk} q_{nk} \tag{72}$$

$$s.t. \quad \sum_{k=1}^{K} q_{nk} = 1, \forall n \in \{1, \cdots, N\} \tag{73}$$

$$\sum_{n=1}^{N} q_{nk} I_n \le (1+\varepsilon)\frac{|R|}{K}, \forall\, k \in \{1, \cdots, K\} \tag{74}$$

$$\sum_{n=1}^{N} q_{nk}(1-I_n) \le (1+\varepsilon)\frac{|D|}{K}, \forall\, k \in \{1, \cdots, K\} \tag{75}$$

$$tr^k \ge 0 \tag{76}$$

$$q_{nk} \in \{0,1\} \tag{77}$$

The problem in model (72)-(77) is a generalization of the well-known clustering problem, with the addition of the uniformity constraints, and is therefore NP-hard. As such, we devise a polynomial time solution methodology based on Lloyd's algorithm. Lloyd's algorithm is an iterative method that has been implemented in several well-known clustering algorithms, such as k-means and expectation-maximization (EM). It is a 2-step heuristic that iteratively assigns points to their nearest cluster centers, and adjusts cluster centers by averaging the points assigned to them. The proposed algorithm closely resembles the application of Lloyd's algorithm in k-means, but modifies it by adding an optimization problem to step 2 (where points are assigned to given cluster centers) to ensure the uniformity of the clusters within a certain threshold. In the absence of the uniformity constraint, it is easy to show that given cluster centers, the total system-wide within-cluster distance is minimized when each point is assigned to the cluster with the nearest center. In our specific problem of interest, however, the ultimate goal is to reduce the complexity of the matching problems associated with clusters. Consequently, we need to adjust this allocation rule by requiring uniformity in cluster sizes.

Algorithm 1 provides a high level description of the proposed method. The algorithm is initialized by randomly selecting K points (i.e., trips) as cluster centers. In step 1, points are assigned to clusters by solving the optimization problem in model (78)-(82). Constraint (79) ensures that each trip is assigned to a single cluster, and constraints (80) and (81) guarantee an ε-uniform distribution of riders and drivers between the K sub-graphs, respectively. Note that we are taking the integer part of the expressions on the left hand side of constraints (80)-(81) to obtain an integer right-hand side vector, which combined with the totally unimodular structure of the constraint set (see Proposition 9), allows for relaxing the binary constraints on the decision variables. Constraint set (82) relaxes the binary constraint on variable q. Note that model (78)-(82) can be transformed into an unbalanced Hitchcock Transportation problem (see

Proposition 10). For such problems with m source nodes and n sink nodes (where $n \ll m$), (Brenner, 2008) developed an efficient algorithm with a worst-case running time of $\mathcal{O}(nk^2(\log n + k\log k))$. Hence, the computational requirement for solving this problem is very minimal.

In step 2, cluster centers are re-calculated by averaging the points in clusters, and a new distance matrix is generated based on the updated cluster centers. The algorithm terminates when the change in the objective function in equation (78) is bound by a pre-determined threshold, or when a max number of iterations is reached. Note that the structure of the graph G is not an input to Algorithm 1, and the spatio-temporal compatibility of trips will be captured through the distance matrix C.

```
Algorithm 1 The $\varepsilon$-uniform partitioning algorithm
Initialization Randomly select $K$ points (i.e., trips) as cluster centers;
        Create the distance matrix $C = [c_{nk}], \forall\, n \in \{1,\cdots,N\}, k \in \{1,\cdots,K\}$.
Step 1 (Cluster allocation)
    Allocate $N$ nodes to $K$ clusters by solving the optimization problem below:
        Minimize $\sum_{k=1}^{K}\sum_{n=1}^{N} c_{nk} q_{nk}$                                   (78)
        s.t. $\sum_{k=1}^{K} q_{nk} = 1, \forall\, n \in \{1,\cdots,N\}$                       (79)
        $\sum_{n=1}^{N} q_{nk} I_n \le [(1+\varepsilon)\frac{|R|}{K}], \forall\, k \in \{1,\cdots,K\}$       (80)
        $\sum_{n=1}^{N} q_{nk}(1-I_n) \le [(1+\varepsilon)\frac{|D|}{K}], \forall\, k \in \{1,\cdots,K\}$    (81)
        $q_{nk} \ge 0$                                                                      (82)
Step 2 (Update cluster centers)
    Update cluster centers by averaging the points in each cluster;
    Update the distance matrix $C = [c_{nk}], \forall\, n \in \{1,\cdots,N\}, k \in \{1,\cdots,K\}$.
Termination Stop when the reduction in equation (78) in two consecutive iterations is less than a pre-determined threshold, or when a max number of iterations is reached.
```

Using Algorithm 1, each sub-graph can be recursively partitioned into smaller sub-graphs using the same procedure, as needed. After the graph partitioning is done, the optimization problem in model (61)-(64) needs to be solved for each sub-graph.

After partitioning, sub-problems are independent of each other and can be solved in parallel. Therefore, the highest computational complexity after partitioning can be attributed to the largest sub-problem. This complexity has two sources: generating the bipartite graph within the cluster, and solving the max cardinality problem on this graph. In Appendix D, we show how one can benefit from using the proposed partitioning method by presenting upper bounds on the computational requirements of both sources.

It is obvious that as the number of clusters increases, or when a more uniform distribution of riders and drivers in sub-graphs is required (i.e., lower ε values), the upper bound on the number of computations to generate the graph in the largest sub-problem decreases. However, it is very likely that being less flexible in terms of rider and driver uniformity in sub-problems and/or considering more clusters lead to eliminating more links from the original graph, hence reducing the accuracy of the solution obtained from Algorithm 1. Finally, while partitioning the graph would lead to smaller sub-problems that can be solved faster, a more important contribution of partitioning is solving problems that cannot be solved at all due to the large number of variables and constraints.

3.6 Network Equilibrium

In this section, our goal is to devise a procedure, which incorporates the proposed graph partitioning technique, to obtain the equilibrium state of a traffic network with ride-sourcing. This goal, however, requires us to somehow relate the two problems at hand. On the one hand, the trip-based graph partitioning is prescribed for a system with microscopic view in which the trip information of every single participant is available. On the other hand, the user-equilibrium problem requires a macroscopic view of system with hourly OD tables and traffic flows. In order to address this issue, we consider a rolling horizon framework with length $L^{horizon}$. In this framework, we first generate participant profiles from hourly demand, and apply the partitioning algorithm. Then, we solve the ridematching problems for each partition and convert the result of matching back to the hourly flows to solve the equilibrium problem. The length of rolling horizon must be chosen carefully such that it does not violate the time window of any ride-sourcing demand. Next, we elaborate on the assumptions and different steps of the proposed algorithm.

The algorithm starts by assuming no ride-sourcing service within the network. Thus, the total demand between every two nodes, T_{rs}^{c}, is set to the background traffic flow (T_{rs}^{n}). As a result, we have no waiting time for customers and drivers. Having the demand, we can solve for the user-equilibrium to obtain the equilibrated flow and shortest path travel times denoted by x_{rs} and h_{rs}, respectively. Next, we use equation (1) to find the number of ride-sourcing customers between every two nodes, denoted by Q_{rs}.

Now, let's consider one rolling horizon. We generate N driver profiles and $\sum_{(r,s)\in W} Q_{rs}$ rider profiles from available information. Let's assume that all participants including riders and drivers are available at the beginning of a time horizon. We further set the time window of riders to the length of the time horizon and assume a very large time window for drivers. Moreover, we assume that every driver can serve at most one rider in each time horizon. Finally, we assume that if served any, a driver will stay at the destination station of that rider until the start of next time horizon. Otherwise, they will wait at their own origin station until the next time horizon. Furthermore, we initialize the drivers' origin stations by drawing random stations from a discrete uniform distribution. In order to find a potential destination of every driver at the end of each time horizon, suppose that the driver rationally chooses a destination that maximizes their expected utility for the next time horizon based on the ride-sourcing demand, shortest path times and driver waiting times in this time horizon. Having the profiles generated, we apply Algorithm 1 to distribute riders and driver between K clusters. For each cluster, we solve a maximum weighted bipartite matching problem formulated as in equation (83)-(86).

$$Maximize \quad \sum_{r\in R_k}\sum_{d\in D_k} \delta_{rd} u_{rd} \tag{83}$$

$$s.t. \quad \sum_{r\in R_k} u_{rd} \le 1, \forall\ d \in D_k : (r,d) \in L_k \tag{84}$$

$$\sum_{d\in D_k} u_{rd} \le 1, \forall\ r \in R_k : (r,d) \in L_k \tag{85}$$

$$0 \le u_{rd} \le 1 \tag{86}$$

In this equation, δ_{rd} represents the expected benefit of driver d serving rider r, and can be computed as follows:

$$\delta_{rd} = F_{s_r^O, s_r^D} - \gamma \cdot (h_{s_d^O, s_r^O} + w_{s_r^D}^{v}), \quad \forall (r,d) \in L_k\ . \tag{87}$$

After solving the optimization problems, we update the total traffic demand by adding the traffic flow of occupied and idle ride-sourcing vehicles to the background traffic flow. We further, update the waiting time of riders and drivers in each station accordingly. Solving for the user-equilibrium using the updated network information provides new values for the flows, shortest path travel times and ride-sourcing demand. The convergence test will be run between the current (x_{rs}) and previous flows (x''_{rs}). If the criterion is not met, we update the number of riders in each cluster and repeat the process for the same horizon. Upon the convergence of the flows in a horizon, we run a convergence test for the flows from this horizon (x_{rs}) and the previous one (x'_{rs}). If the criterion is met, the algorithm terminates, otherwise, the origin station of drivers will be updated and the process will be repeated for another rolling horizon. A summary of the proposed procedure is presented in Algorithm 2.

Algorithm 2 Equilibrium Algorithm For Traffic Network with Ride-sourcing And Graph Partitioning

```
Initialization
Set $T^c_{rs} = T^n_{rs}, \quad \forall\,(r,s) \in W^c$;
Set $w^c_r = 0, \quad \forall\, r \in S$;
Set $w^v_r = 0, \quad \forall\, r \in S$;
Set $s^O_d \sim unif\{r \mid \sum_s Q_{rs} > 0\}, \quad \forall\, d \in D$;
Set $\Delta^h = \infty$;
Solve the user-equilibrium and find $x_{rs}$, $h_{rs}$ and $Q_{rs} \quad \forall\,(r,s) \in W$;
While ($\Delta^h > \epsilon_h$) do:
    Set $s^D_d = \underset{s\in S}{argmax}\{\underset{r\in S}{argmax}\{\frac{Q_{rs}}{\sum_s Q_{rs}}(F_{rs} - \gamma \cdot (h_{s^O_d,r} + w^v_s))\}\}, \quad \forall d \in D$;
    Generate riders and drivers;
    Solve the trip-based partitioning algorithm to cluster participants into $K$ regions;
    Set $\Delta^m = \infty$
    While ($\Delta^m > \epsilon_m$) do:
        Update number of riders within clusters;
        For ($k \in \{1,...,K\}$) do:
            Solve the max wighted bipartite matching for cluster $k$;
        end
        Update $T^v_{rs}$, $T^o_{rs}$, $\forall\,(r,s) \in W$;
        Update $w^c_r$ and $w^v_r$, $\forall\, r \in S$;
        Set $T^c_{rs} = T^v_{rs} + T^o_{rs} + T^n_{rs}, \quad \forall\,(r,s) \in W^c$;
        Solve the user-equilibrium and find $x_{rs}$, $h_{rs}$ and $Q_{rs} \quad \forall\,(r,s) \in W$;
    end
    Set $\Delta^m = \frac{\sqrt{\sum_{(r,s)\in W}(x_{rs} - x''_{rs})^2}}{\sum_{(r,s)\in W} x'_{rs}}$; Set $\Delta^h = \frac{\sqrt{\sum_{(r,s)\in W}(x_{rs} - x'_{rs})^2}}{\sum_{(r,s)\in W} x'_{rs}}$;
    Set $s^O_d = \begin{cases} s^D_r, & \text{if} u_{rd} = 1; \\ s^O_d, & \text{if} u_{rd} = 0; \end{cases} \quad \forall$
end
```

4 Numerical Experiments

Numerical experiments are carried out on the Nguyen-Dupius network, which consists of 13 nodes, 38 links, and 5 OD pairs as shown in Figure 5. The performance on each link is assumed to follow a linear relationship with the corresponding link flow, i.e.

$$t_{ij}(v_{ij}) = a_{ij} + b_{ij} \cdot v_{ij} \forall (i,j) \in A$$

The values of constants $\{a_{ij}\}$ and $\{b_{ij}\}$ as well as the regular traffic demand $\{T_{rs}^{n}\}$ for the network are enclosed respectively in Table 2(a) and 2(b), Appendix E.

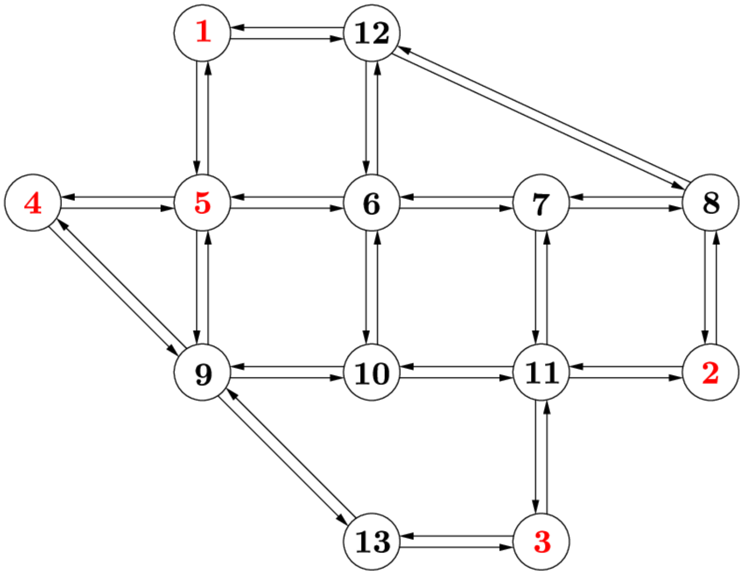


Figure 5 The Nguyen-Dupuis network

Let there be ride-sourcing service demands between the two node sets $\{1,4,5\}$ and $\{2,3\}$ located in the corners of the network (see the red nodes in Figure 5), and the fare structure of the service be

$$F_{rs} = F_0 + \tau \cdot h_{rs} \forall (r,s) \in W$$

where F_0 denotes the flag-drop fee, and $\tau \cdot h_{rs}$ represents the time/distance-based portions. Define the customer demand function as:

$$f_{rs}(C_{rs}, \hat{C}_{rs}) = Q_{rs}^{0} \cdot \frac{\exp(-\hat{\theta} C_{rs})}{\exp(-\hat{\theta} C_{rs}) + \exp(-\hat{\theta} \hat{C}_{rs})} \forall (r,s) \in W$$

where Q_{rs}^{0} indicates the potential customer demand from node r to s (see Table 2(c)); $\hat{\theta}$ features a constant representing the degree of travelers' perceptual dispersion over the costs of using different modes; the costs $\hat{C}_{rs}$ of alternative modes is assumed to also take the form $\hat{F}_{rs} + \hat{\beta}^{o} \hat{w}^{c} + \hat{\beta}^{i} h_{rs}$, in which parameters and variables hold the same meanings as their hatless counterparts in C_{rs}.

For the inter-node matching, we further specify the potential functions Φ^{v}, Φ^{c} and the potential difference function Δ in Equation (48) to respectively adopt the following forms,

$$\Phi^{v}\left(T_{l}^{v}, N_{l}^{v}\right) = (T_{l}^{v})^{-q_{T}^{v}} \cdot (N_{l}^{v})^{q_{N}^{v}}$$

$$\Phi^c\left(T_r^c, N_r^c\right) = (T_r^c)^{-q_T^c} \cdot (N_r^c)^{q_N^c}$$

$$\Delta(T_{lr}^m, h_{lr}) = \eta \cdot T_{lr}^m \cdot (h_{lr})^{q_h}$$

where η, q_h and $\{q_j^i\}_{i\in\{v,c\}, j\in\{T,N\}}$ are constants. Note that all the parametric values implemented in the experiment are summarized in Table 2(d), Appendix E for reference.

4.1 Numerical Results For Inter-node And Intra-node Matching

Based on the nominal system defined above, the scenarios of intra-node and inter-node matching are produced by using two different specifications of matching sets below:

Matching range	$M^c(1)$	$M^c(4)$	$M^c(5)$	$M^c(2)$	$M^c(3)$
Intra-node	$\{1\}$	$\{4\}$	$\{5\}$	$\{2\}$	$\{3\}$
Inter-node	$\{1,5,6,12\}$	$\{4,5,9\}$	$\{1,4,5,6,9\}$	$\{2,8,11\}$	$\{3,11,13\}$

Equilibrium solutions are then retrieved by applying the procedure developed in Section 2.3.

The equilibrium solution of inter-node matching is firstly presented. As shown in Figure 6, beside each source node, a pair of red and green columns are attached to indicate the matching time that RVs and customers undertake therein, respectively. The exact time of waiting is numbered directly on the top of each column.

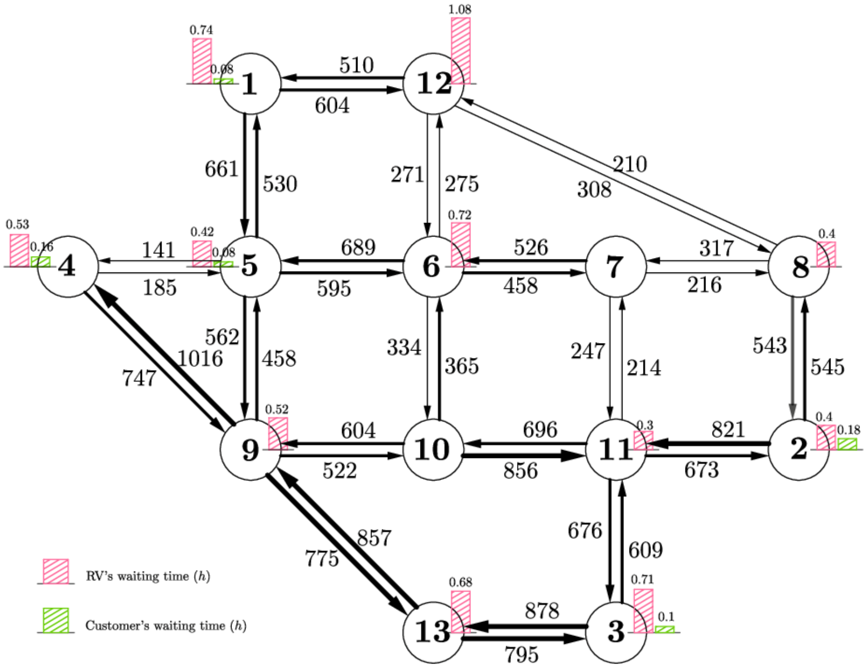


Figure 6 Equilibrated system states under inter-node matching

In general, it is clear that the system under investigation experiences an excessive supply of RVs, i.e., customers enjoy relatively quick matches while RVs suffer from long time of waiting. The degree of excess peaks at node 1 and 3 but appears milder at other source nodes. Additionally, we observe in Figure 6 that there are idle vehicles accumulating on nodes that have no customer demand inside, e.g. node 12 and 6. This is perhaps due to the fact that idle drivers in e-hailing systems are potentially less incentivized for chasing hotspots, as such a strategy may not necessarily reduce their deadheading time and render higher profitability. Even on hotspots, idle drivers may receive matches with distant pickups that make their long-distance cruise futile.

By taking the above equilibrium as the baseline, we then compare the differences between the intra-node and inter-node matching scenarios. In contrast to inter-node matching, results show that RVs under intra-node matching on average experience longer time of idleness but less time in pickups. Also, both the rate of customers being served and the profitability of drivers are slightly lower under intra-node matching, intimating the potential for ride-sourcing platforms to match RVs and customers staying apart.

Figure 7 displays the relative changes on the link flows as well as RVs and customers' waiting time by shrinking down the matching range. Specifically, for each variable x illustrated on the map, its relative change Δ_x is defined as

$$\Delta_x = \frac{x_{\text{intra}} - x_{\text{inter}}}{x_{\text{inter}}}$$

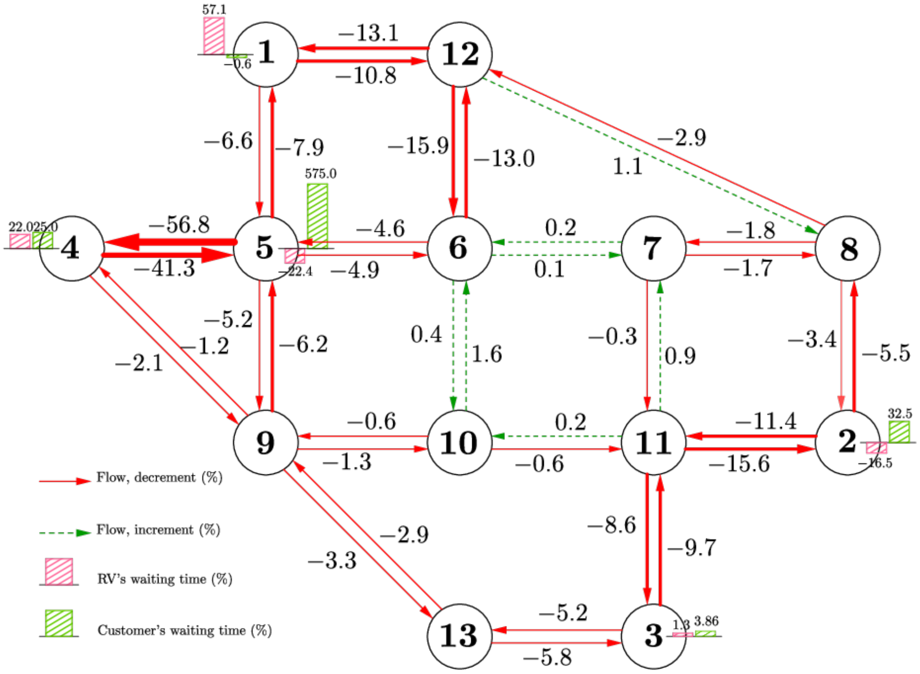


Figure 7 Comparisons of the inter-node and intra-node matching scenarios

The red and green arrows in Figure 7 indicate percentage decrements and increments of flows on different links, respectively. The exact percent difference is numbered next to each arrow and shown graphically by its thickness. As can be observed, there are significantly greater amount of flows on links around the two sets of source nodes under inter-node matching (e.g. link $5 \to 4$ and $12 \to 1$). Such differences suggest that for e-hailing systems with unignorably long distance of pickups, an intra-node matching model may bias the flow estimates due to the lack of strength in modeling vehicles' deadheading.

Further, the red and green columns in Figure 7 present the percentage changes on RVs' and customers' waiting time, respectively. Since inter-node matching enlarges the matching range and thereby reducing the meeting frictions between drivers and customers, it is likely that both parties' waiting time in matching decreases compared to the intra-node matching counterparts (see, e.g., node 4 and 3 in Figure 7). Counterexamples, however, may arise when the matching sets of two source nodes grow large enough to overlap with each other. Taking the pair of nodes 1 and 5 as an example, node 1 exhibits severer excess of supply compared to node 5 (see Figure 6). Then, as we enlarge the matching range and have node 1 and 5 own a large portion of matching sets in common, the supply excess condition on the two nodes neutralizes, which yields longer wait for customers on node 1 and RVs on node 5 (see Figure 7).

4.2 Numerical Results For Graph Partitioning

In this study, we assume that all trips will be completed on the shortest paths. To use the trip-based ε-uniform partitioning, we need to assume some xy-coordinates for each station. Therefore, we assume that the network is located on a grid of 4×3 units. In order to calibrate the wight vector ω for the region, we assume an average travel time distance of 10 minutes between every two points with the same x/y but a difference of 1 unit in y/x. As a result, we have $\omega=(10,10,10,10,1)^\top$. Further, let the number of clusters (K), the value of ε, and the maximum number of iterations in algorithm 1 be 2, 0.1, and 20, respectively.

For solving the user-equilibrium problem, we adopt the well-known method of "Convex Combination" with a threshold of 10^{-5}. Also, we let the flow convergence threshold of ϵ_m and ϵ_h be equal to zero. Since we start Algorithm 2 with a random choice of origin stations, we run the algorithm for 10 samples. Hence, the results are reported as the average of 10 runs. Finally, the length of the rolling horizon is set to be 1 hour.

The equilibrium result of Algorithm 2 is presented in Figure 8. Similar to the result of previous section, we observe high rates of RVs' waiting times that suggests an excess of RV supply. However, unlike the previous cases, we observe zero waiting times for customers in all stations. This is duo the fact that there are enough RVs at each station to serve all the customers in that station. Therefore, no customer has to wait for RVs from other station to pick them up. Low rate of flow between node 4 and node 5 also confirms that RVs are mostly incentivized to serve the customers at their origin station.

Figure 9 and Figure 10 display the relative changes on the link flows as well as RVs and customers' waiting time compared to inter-node matching and intra-node matching, respectively. Roughly speaking, we observe that Algorithm 3.5 results in lower flow estimates. This is duo to the fact that the matching rate in this method is slightly lower than the 2 other methods. Since we assumed that unmatched RVs will stay in their origin station, we can expect this result. We speculate that increasing the length of rolling time horizon would increase the matching rate and hence the flows on network links.

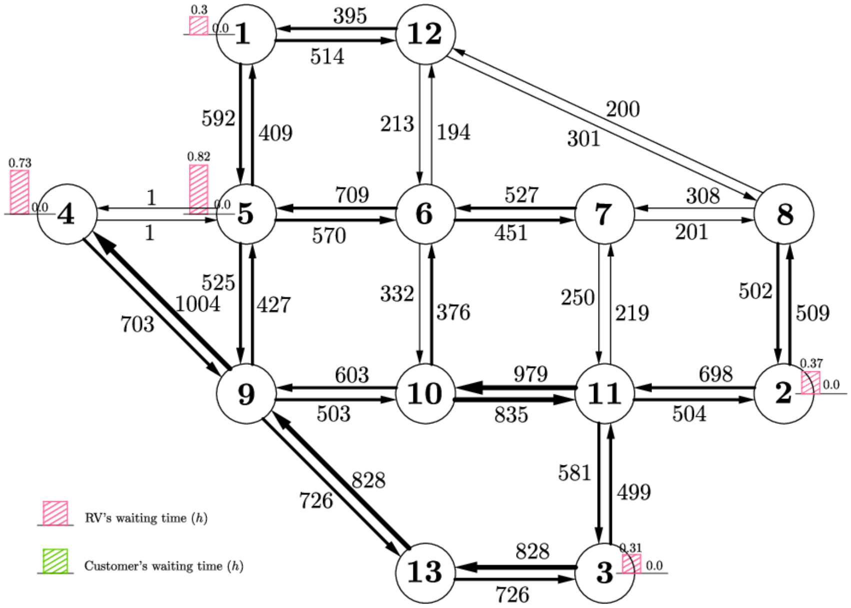


Figure 8 Equilibrated system states under ride matching with graph partitioning

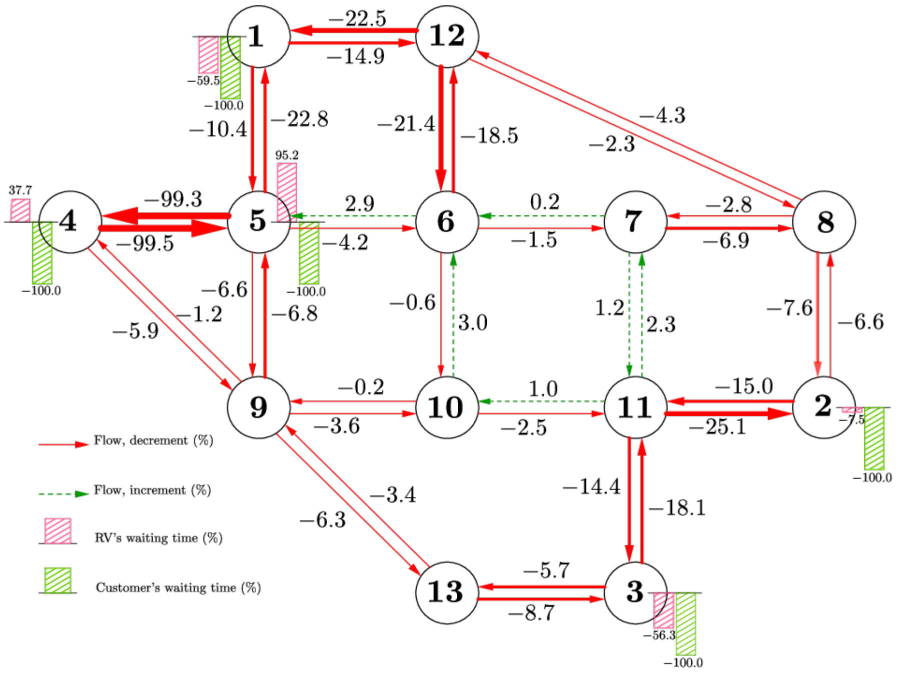


Figure 9 Comparisons of the inter-node matching and graph partitioning scenarios

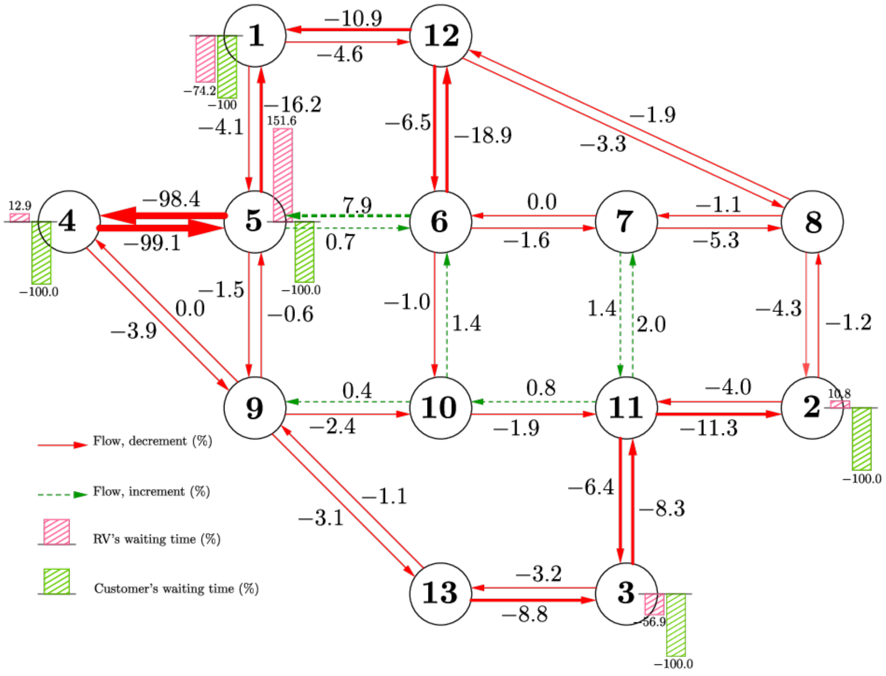


Figure 10 Comparisons of the intra-node matching and graph partitioning scenarios

5 Findings

This project proposes a network equilibrium model to deal with the traffic assignment problem in systems where a large portion of travel demands get served by ride-sourcing services. In such systems, besides the occupied vehicular trips transporting travelers from their origins to destinations, massive traffic is contributed by vacant RV trips that originate from the end of one customer trip to the start of the next. Different from the existing models, our aggregate framework takes into account the two major types of vacant trips generated by RVs, i.e. cruising and deadheading. We firstly present a basic model by considering intra-node matching between customers and idle RVs. The model is only capable of capturing the congestion effects of cruising trips. To cope with the distant matching, a common practice adopted by ride-sourcing systems nowadays, we base on the basic model and develop an enhanced modeling framework for inter-node matching. By using an analogy to electricity circuits, the proposed inter-node matching function handles the spatial interactions between neighboring zones in the matching process. Such a specification endows us with higher flexibility in evaluating the matching strategies of platforms and depicting the movements of vacant RVs. An iterative solution procedure is proposed to solve the equilibrium of an inter-node matching system.

Although the presented methodologies provide an effective strategy to obtain network equilibrium in the existence of ride-sourcing, their operational efficacy may be far from reality duo to utilizing an approximate function of matching. In order to mitigate this issue, we further present a multi-layer framework that combines the microscopic and macroscopic view of the system, as it solves a ride-matching problem in the lower level whose results are fed to a user-equilibrium problem in the upper level. To achieve a high performance level in finding the matches, the ridesharing operator needs to make the matching decision based on a global view of the system that includes all active riders and drivers when proposing ride-matches. Consequently, the ride-matching problem that needs to be solved can become computationally expensive, especially when the system is operating over a large region, or when it faces high demand levels during certain hours of the day. In this project, we proposes a methodology to decompose the matching problem into multiple sub-problems with the goal of reducing the overall computational complexity of the problem as well as providing high quality solutions.

6 Recommendations

For the future research, we intend to feed the real-world data into the proposed framework to calibrate the inter-node matching function and empirically examine its practicality. We are interested in further understanding the physics behind inter-node matching, and studying how parameters in the matching function associate with the macroscopic performance of a system. Also, we intend to enhance our operational methodology by exploiting more complicated, yet more realistic matching problems (e.g. one-to-many matching problem) to enhance the efficiency of our ride-sourcing vehicles. We expect that our enhanced assignment model as well as the derived knowledge will help government agencies/platforms with their critical policy/operational decision-makings in managing a ride-sourcing system.

7 Acknowledgment

The work described in this paper was partly supported by research grants from the Midwest USDOT Center for Connected and Automated Transportation (CCAT).

Nomenclature

Table 1 Notation list of sets, variables, parameters and functions

Notation	Description
Sets	
V	Set of nodes
A	Set of links
R	Set of the original nodes of customer trips; $R \subseteq V$
S	Set of the destination nodes of customer trips; $S \subseteq V$
W	Set of OD pairs of ride-sourcing customer demands
W^b	Set of OD pairs of background regular traffic
W^c	Complete set of OD pairs, including those of RVs and regular vehicles
$M^c(r)$	Set of nodes hailing customers at node $r \in R$ can potentially be matched to
L	Set of nodes with positive accumulations of idle RVs
$M^v(l)$	Set of nodes idle RVs at node $l \in L$ can potentially be matched to
Variables	
C_{rs}	Monetary travel cost between OD pair $(r,s) \in W$
F_{rs}	Fare of trips from node r to s
w_r^c	Customer's average waiting time at node r, $r \in R$
h_{rs}	The equilibrium or shortest vehicular travel time between node r and s
Q_{rs}	Customer demand on OD pair $(r,s) \in W$
$U_{sr}(U_{sl})$	Driver's utility of cruising his/her idle RV from node $s \in S$ to $r \in R(l \in L)$
$w_r^v(w_l^v)$	Idle RV's average additional waiting time at node $r \in R(l \in L)$
$\bar{F}_r$	Average fare of the customer trips originating from node $r \in R$
$\hat{F}_l$	Average fare earnings of RVs who get matched at node $l \in L$
$\bar{h}_r$	Average service time of the customer trips originating from node $r \in R$
$\hat{h}_l$	Average service time of RVs who get matched at node $l \in L$
T_{rs}^o	Occupied RV flow that serves customer demand from node r to s, $(r,s) \in W$
$T_{sr}^v(T_{sl}^v)$	Idle RV flow from node $s \in S$ to $r \in R(l \in L)$
T_{sr}^n	Regular traffic flow from node r to s, $(r,s) \in W^b$
$N_r^v(N_l^v)$	Number of idle RVs at node $r \in R(l \in L)$
N_r^c	Number of hailing customers at node $r \in R$
T_{lr}^m	Rate of RVs matched from node $l \in L$ to $r \in M^v(l)$
Parameters	
β^o	Customer's out-of-vehicle value of time (\$/h)
β^i	Customer's in-vehicle value of time (\$/h)
γ	RV driver's value of time $\$ / h$
θ	Degree of drivers' perceptual dispersion

N	Total number of RVs in the network
Functions	
f_{rs}	Function of customer demand versus travel costs for OD pair $(r,s) \in W$
g_{rs}	Function of trip fare versus travel time for OD pair $(r,s) \in W$
m_r	Aggregate matching function for node $r \in R$
$\Phi^v (\Phi^c)$	Node potential function on accumulations of idle RVs(hailing customers)
Δ	Potential gap function for paired nodes with positive matching flows

References

Ban, X. J.-S. (2019). A general equilibrium model for transportation systems with e-hailing services and flow congestion. *Transportation Research Part B: Methodological, 129, 273-304. Brenner, U. (2008). A faster polynomial algorith*m for the unbalanced hitchcock transportationproblem. *Operation Research Letters, 36*(4), 408-413.

De la Fuente, A. (2000). Mathematical methods and models for economists. *Cambridge University Press*.

Douglas, G. W. (1972). Price regulation and optimal service standards: The taxicab industry. *Journal of Transport Economics and Policy*, 116-127.

Ford, L. R. (1956). Maximal flow through a network. *Canadian journal ofMathematics, 8*(3), 399–404.

Hall, M. A. (1978). Properties of the equilibrium state in transportation networks. *Transportation Science, 12*(3), 208–216.

Hopcroft, J. E. (1973). An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing, 2*(4), 225–231.

Lu, S. a. (2010). Lu, S. and Nie, Y. M.Stability of user-equilibrium route flow solutions for the trafficassignment problem. *Transportation Research Part B: Methodological, 44*(4), 609–617.

Yang, H. a. (2011). Equilibrium properties of taxi markets with search frictions. *Transportation Research Part B: Methodological, 45*(4), 696–713.

Appendix A Existence And Uniqueness of Solutions For the Inter-node Matching Equations

This appendix proves that given $\{N_l^v\}$, $\{N_r^c\}$ and $\{h_{lr}\}$ to be fixed and positively valued, there exists a unique flow pattern $\{T_{lr}^m\}$ that satisfies the following aggregate matching equation set ((88)),

$$\Phi^v\left(\sum_{k\in M^V(l)} T_{lk}^m, N_l^v\right)\cdot\Phi^c\left(\sum_{k\in M^C(r)} T_{kr}^m, N_r^c\right) = \Delta(T_{lr}^m, h_{lr})\forall l\in L, r\in M^v(l) \tag{88}$$

For mathematical convenience, we use $\{\Phi_l^v\}_{l\in L}$, $\{\Phi_r^c\}_{r\in R}$ and $\{\Delta_{lr}\}_{l\in L, r\in M^V(l)}$ to indicate the functions of Φ^v, Φ^c and Δ conditional on $\{N_l^v\}$, $\{N_r^c\}$ and $\{h_{lr}\}$, respectively. Generic properties that pertain to $\{\Phi_l^v(T_l^v), \{\Phi_r^c(T_r^c)\}$ (abbrev. $\Phi(T)$) and $\{\Delta_{lr}(T_{lr}^m)\}$ (abbrev. $\Delta(T)$) are summarized as the following conditions C1-C3:

C1: $\Phi(T)$ and $\Delta(T)$ are continuous functions defined on $T\in(0,+\infty)$.

C2: For any $T\in(0,+\infty)$, we have $\Phi(T)>0$, $\Phi'(T)<0$ and $\Delta(T)>0$, $\Delta'(T)>0$.

C3: There exist $p>0$ and $q>0$ such that:

$$\Phi(T)=\begin{cases}\Theta(T^{-p}), & \text{as } T\to 0^+\\ \Theta(T^{-q}), & \text{as } T\to +\infty.\end{cases}$$

These properties serve as the foundation for proofs below.

On the first stage, we prove the existence of a feasible solution $\{T_{lr}^m\}$ to Eq. (88). The proof begins with the Proposition 1 stated below.

Proposition 1 *There exist positive constants* $(\omega_o, \omega_u, \tau_o, \tau_u)$ *such that*

$$(\omega_o T^p + \omega_u T^q)^{-1} \leqslant \Phi(T) \leqslant \tau_o T^{-p} + \tau_u T^{-q}$$

Proof. Condition C3 suggests there exist $T_o>0, \omega_o'>0, \tau_o'>0$ and $T_u>0, \omega_u'>0, \tau_u'>0$ such that

$$\omega_o' T^{-p} \leqslant \Phi(T) \leqslant \tau_o' T^{-p} T\in(0,T_o]$$

$$\omega_u' T^{-q} \leqslant \Phi(T) \leqslant \tau_u' T^{-q} T\in[T_u,+\infty)$$

Collectively, the above two inequalities yield

$$(\omega_o'^{-1} T^p + \omega_u'^{-1} T^q)^{-1} \leqslant \Phi(T) \leqslant \tau_o' T^{-p} + \tau_u' T^{-q} T\in(0,T_o]\cup[T_u,+\infty) \tag{89}$$

If $T_o \geqslant T_u$, the proposition is proven. Otherwise, as Φ is monotonically decreasing, we have that for any $T\in[T_o,T_u]$,

$$\Phi(T)\geqslant\Phi(T_o)\cdot\frac{\Phi(T_u)}{\Phi(T_o)}\geqslant(\omega_o'^{-1}T^p+\omega_u'^{-1}T^q)^{-1}\cdot\frac{\Phi(T_u)}{\Phi(T_o)} \tag{90}$$

$$\Phi(T)\leqslant\Phi(T_u)\cdot\frac{\Phi(T_o)}{\Phi(T_u)}\leqslant(\tau_o'T^{-p}+\tau_u'T^{-q})\cdot\frac{\Phi(T_o)}{\Phi(T_u)} \tag{91}$$

where the first inequality results from the monotonicity of Φ and the second one is due to the above relation (89). Since $\Phi(T_o)$ and $\Phi(T_u)$ characterize two constants with $\Phi(T_o)\geqslant\Phi(T_u)$, the inequalities (8) are extensible for the entire domain of $T\in(0,+\infty)$, which essentially indicates Proposition (1).

Define $Z_{lr}(T)$ as $T^{2p+2q}\Delta_{lr}(T)$. Then, each Z_{lr} characterizes a monotonically increasing function on T

. By multiplying both side of Eq. (88) with T^{2p+2q} , we can base on that equation to construct the following mapping Γ that maps $\mathbf{T^i}(=\{T_{lr}^i\})$ to $\mathbf{T^d}(=\{T_{lr}^d\})$, i.e.

$$\Gamma:\left\{T_{lr}^d = Z_{lr}^{-1}\left(T_{lr}^{i\,2p+2q}\cdot\Phi_l^v\left(\sum_{k\in M^v(l)} T_{lk}^i\right)\cdot\Phi_r^c\left(\sum_{k\in M^c(r)} T_{kr}^i\right)\right), \forall l\in L, r\in M^v(l)\right\} \quad (92)$$

Then, the existence of a solution to Eq. (88) is equivalent to prove that the mapping ((92)) has a fixed point. As per Brouwer's fixed point theorem (de2000mathematical), this requires the existence of a compact convex set Ω , such that $\mathbf{T^d}=\Gamma(\mathbf{T^i})\in\Omega$ for any $\mathbf{T^i}\in\Omega$. We thus prove the second proposition as follows,

Proposition 2 *A pair of positive bounds B_o and $B_u(\geqslant B_o)$ can always be found, with $\Omega=\{\mathbf{T}\,|\,T_{lr}\in[B_o,B_u],\forall l\in L, r\in M^v(l)\}$ satisfying the above condition required by the existence of fixed points.*

Proof. According to Proposition (1), there will be a set of constants $(\omega_{k,o}^t,\omega_{k,u}^t,\tau_{k,o}^t,\tau_{k,u}^t)$ for each Φ_k^t such that

$$(\omega_{k,o}^t T^p+\omega_{k,u}^t T^q)^{-1}\leqslant\Phi_k^t(T)\leqslant\tau_{k,o}^t T^{-p}+\tau_{k,u}^t T^{-q}$$

where $(t,k)\in S=\{k\in L \text{ if } t=v \text{ and } k\in R \text{ if } t=c\}$. By defining $\omega_o=\max\{\omega_{k,o}^t\},\omega_u=\max\{\omega_{k,u}^t\}$ and $\tau_o=\max\{\tau_{k,o}^t\},\tau_u=\max\{\tau_{k,u}^t\}$ over all $(t,k)\in S$, we thus have

$$(\omega_o T^p+\omega_u T^q)^{-1}\leqslant\Phi_k^t(T)\leqslant\tau_o T^{-p}+\tau_u T^{-q}\forall(t,k)\in S$$

Next, we define two univariate functions $\Delta_o(T)$ and $\Delta_u(T)$ based on $\{\Delta_{lr}(T)\}$, i.e.

$$\Delta_o(T)=\min\{\Delta_{lr}(T),\forall\in L,r\in M^v(l)\}$$
$$\Delta_u(T)=\max\{\Delta_{lr}(T),\forall\in L,r\in M^v(l)\}$$

Also, define $Z_i(T)=T^{2p+2q}\Delta_i(T)$ for either $i\in\{o,u\}$. As all functions in $\{\Delta_{lr}(T)\}$ increase monotonically on T , the resultant Δ_o,Δ_u and Z_o,Z_u are all monotonically increasing functions. Suppose T_{lr} will be valued from a given set $[B_o,B_u]$, $\forall l\in L,r\in M^v(l)$. Then,

$$\begin{aligned}\Gamma_{lr}(\mathbf{T})&\leqslant Z_o^{-1}\left(T_{lr}^{2p+2q}\cdot\Phi_l^v\left(\sum_{k\in M^v(l)} T_{lk}\right)\cdot\Phi_r^c\left(\sum_{k\in M^c(r)} T_{kr}\right)\right)\\&\leqslant Z_o^{-1}\left(T_{lr}^{2p+2q}\cdot\Phi_l^v\left(T_{lr}\right)\cdot\Phi_r^c\left(T_{lr}\right)\right)\\&\leqslant Z_o^{-1}\left(T_{lr}^{2p+2q}\cdot\left(\tau_o\cdot T_{lr}^{-p}+\tau_u\cdot T_{lr}^{-q}\right)^2\right)\\&\leqslant Z_o^{-1}\left((\tau_o B_u^q+\tau_u B_u^p)^2\right)\end{aligned}$$

Let $\xi(B)$ denote $(\tau_o B^q+\tau_u B^p)^2$. The upper bound B_u can thus be defined as solving $Z_o(B_u)=\xi(B_u)$. Note that by solving the equation, we can always receive a feasible B_u because

$$Z_o(B_u)=\Theta\left(\Delta_o(B_u)\cdot B_u^{2p+2q}\right)\underset{(<)}{>}\Theta(B_u^{2p+2q})\underset{(<)}{>}\Theta\left((\tau_o B_u^q+\tau_u B_u^p)^2\right)\mathrm{as B_u}\rightarrow\underset{(0^+)}{+\infty}$$

given $p>0,q>0$ and $\Delta_o(\cdot)$ increases monotonically ranging from 0^+ to $+\infty$.

On the other hand, by defining constants $n_v=\max_{l\in L}\{|M^v(l)|\}$ and $n_c=\max_{r\in R}\{|M^c(r)|\}$,

$$\Gamma_{lr}(\mathbf{T}) \geqslant Z_u^{-1}\left(T_{lr}^{2p+2q}\cdot\Phi_l^v\left(\sum_{k\in M^v(l)}T_{lk}\right)\cdot\Phi_r^c\left(\sum_{k\in M^c(r)}T_{kr}\right)\right)$$

$$\geqslant Z_u^{-1}\left(T_{lr}^{2p+2q}\cdot\Phi_l^v\left(T_{lr}+(n_v-1)B_u\right)\cdot\Phi_r^c\left(T_{lr}+(n_c-1)B_u\right)\right)$$

Definition: Let $B_u^v=(n_v-1)B_u$ and $B_u^c=(n_c-1)B_u$

$$\geqslant Z_u^{-1}\left(\frac{T_{lr}^{p+q}}{\omega_o\cdot(T_{lr}+B_u^v)^p+\omega_u\cdot(T_{lr}+B_u^v)^q}\cdot\frac{T_{lr}^{p+q}}{\omega_o\cdot(T_{lr}+B_u^c)^p+\omega_u\cdot(T_{lr}+B_u^c)^q}\right)$$

$$\geqslant Z_u^{-1}\left(\frac{B_o^{p+q}}{\omega_o\cdot(B_o+B_u^v)^p+\omega_u\cdot(B_o+B_u^v)^q}\cdot\frac{B_o^{p+q}}{\omega_o\cdot(B_o+B_u^c)^p+\omega_u\cdot(B_o+B_u^c)^q}\right)$$

Define the above term on B_o within the parenthesis behind Z_u^{-1} as $\zeta(B_o)$. Then, the lower bound B_o can be defined as solving $Z_u(B_o)=\zeta(B_o)$. Again, such a B_o always exists because

$$Z_u(B_o)=\Theta\left(\Delta_u(B_o)\cdot B_o^{2p+2q}\right)\underset{(<)}{>}\Theta\left(B_o^{2p+2q}\right)\underset{(\leqslant)}{>}\Theta\left(\zeta(B_o)\right)\mathrm{asB_u}\rightarrow\underset{(0^+)}{+\infty}$$

Further, since $Z_u(B_u)\geqslant Z_o(B_u)=\xi(B_u)\geqslant\zeta(B_u)$, there must exist a B_o in $(0,B_u]$. It is then straightforward that with any pair of B_o and B_u defined above and $\mathbf{T}=\{T_{lr}\mid T_{lr}\in[B_o,B_u],\forall l\in L,r\in M^v(l)\}$, all the mappings $\{\Gamma_{lr}(\mathbf{T})\}$ will compliantly fall in $[B_o,B_u]$.

With Proposition (2) held, the existence of a feasible solution $\{T_{lr}^m\}$ to the equation set (88) is always guaranteed. On the second stage, we prove that the ensured solution $\{T_{lr}^m\}$ is unique.

Proposition 3 *The matching flow pattern* $\{T_{lr}^m\}$ *that solves the following Eq. (3) is unique.*

$$\Phi_l^v\left(T_l^v\right)\cdot\Phi_r^c\left(T_r^c\right)=\Delta_{lr}(T_{lr}^m)\forall l\in L,r\in M^v(l) \tag{93}$$

$$T_l^v=\sum_{k\in M^v(l)}T_{lk}^m l\in L \tag{94}$$

$$T_r^c=\sum_{k\in M^c(r)}T_{kr}^m r\in R \tag{95}$$

Proof. We prove the proposition by contradiction. Assume there are two distinct flow patterns $\{T_{lr}^{m,1}\}$ and $\{T_{lr}^{m,2}\}$ solving Eq. (3). Firstly, if $T_l^{v,1}=T_l^{v,2}$ and $T_r^{c,1}=T_r^{c,2}$ for all $l\in L$ and $r\in R$, then Eq. (3a) suggests the equivalence of $\{T_{lr}^1\}$ and $\{T_{lr}^2\}$. Suppose $T_{l_1}^{v,1}>T_{l_1}^{v,2}$ for $l_1\in L$, and then $T_{l_1r_1}^{m,1}>T_{l_1r_1}^{m,2}$ for $r_1\in M^v(l_1)$. These two inequalities yields

$$\Phi_{r_1}^c(T_{r_1}^{c,1})=\frac{\Delta_{l_1r_1}(T_{l_1r_1}^{m,1})}{\Phi_{l_1}^v(T_{l_1}^{v,1})}>\frac{\Delta_{l_1r_1}(T_{l_1r_1}^{m,2})}{\Phi_{l_1}^v(T_{l_1}^{v,2})}=\Phi_{r_1}^c(T_{r_1}^{c,2})$$

which further gives rise to $T_{r_1}^{c,1}<T_{r_1}^{c,2}$. Thus, there exists and $l_2\in M^c(r_1)$ such that $T_{l_2r_1}^{m,1}<T_{l_2r_1}^{m,2}$. Through a simple conduction, we obtain the following connection between $\Phi_{l_2}^v$ and $\Phi_{l_1}^v$,

$$\frac{\Phi_{l_2}^v(T_{l_2}^{v,1})}{\Phi_{l_2}^v(T_{l_2}^{v,2})}=\frac{\Delta_{l_2r_1}(T_{l_2r_1}^{m,1})/\Delta_{l_1r_1}(T_{l_1r_1}^{m,1})}{\Delta_{l_2r_1}(T_{l_2r_1}^{m,2})/\Delta_{l_1r_1}(T_{l_1r_1}^{m,2})}\cdot\frac{\Phi_{l_1}^v(T_{l_1}^{v,1})}{\Phi_{l_1}^v(T_{l_1}^{v,2})}<\frac{\Phi_{l_1}^v(T_{l_1}^{v,1})}{\Phi_{l_1}^v(T_{l_1}^{v,2})}<1 \tag{96}$$

The inequality ((96)) implies two folds of relationships: $1.\,l_2\neq l_1$; $2.\,T_{l_2}^{v,1}>T_{l_2}^{v,2}$, which together evidences the existence of an $r_2 \in R$ with $T_{l_2 r_2}^{m,1} > T_{l_2 r_2}^{m,2}$ and

$$\frac{\Phi_{r_2}^{c}(T_{r_2}^{c,1})}{\Phi_{r_2}^{c}(T_{r_2}^{c,2})} = \frac{\Delta_{l_2 r_2}(T_{l_2 r_2}^{m,1}) / \Delta_{l_2 r_1}(T_{l_2 r_1}^{m,1})}{\Delta_{l_2 r_2}(T_{l_2 r_2}^{m,2}) / \Delta_{l_2 r_1}(T_{l_2 r_1}^{m,2})} \cdot \frac{\Phi_{r_1}^{c}(T_{r_1}^{c,1})}{\Phi_{r_1}^{c}(T_{r_1}^{c,2})} > \frac{\Phi_{r_1}^{c}(T_{r_1}^{c,1})}{\Phi_{r_1}^{c}(T_{r_1}^{c,2})} > 1$$

Therefore, by repeating the above process iteratively, we can retrieve two ceaselessly nonrepetitive sequences $\{l_3, l_4, l_5, \cdots\} \subseteq L$ and $\{r_3, r_4, r_5, \cdots\} \subseteq R$. This yields an obvious contradiction because both L and R are finite sets.

Integrating Proposition (2) and (3) proves the existence and uniqueness of a solution for Eq. (88). We present this conclusion formally as the following Theorem (1):

Theorem 1 *By fixing* $\{N_l^v\}$, $\{N_r^c\}$ *and* $\{h_{lr}\}$ *with positive values, the aggregate matching equation set (88) solves a unique flow pattern* $\{T_{lr}^m\}_{l \in L, r \in M^v(l)}$.

Appendix B Existence of an Equilibrium For the Inter-node Matching system

This appendix proves that for any practical parametric setting, there always exists an equilibrium solution to the inter-node matching system (33)-(56). Before starting the proof, we firstly enumerate two general properties that our system should comply with:

- All the link performance functions $\{t_{ij}\}_{(i,j)\in A}$ are monotonically increasing on the corresponding link flows, with the free-flow travel time $\{t_{ij}^0\}_{(i,j)\in A}$ being positive constants.
- All the customer demand functions $\{f_{rs}\}_{(r,s)\in W}$ are convex and monotonically decreasing on the corresponding trip costs, with the limit $\lim\limits_{w_r^c \to +\infty} w_r^c \cdot f_{rs}(w_r^c \mid \mathbf{h})$ being positively finite. The potential demands (under the zero-cost condition) $\{Q_{rs}^0\}_{(r,s)\in W}$ are positive constants.

These properties then lead to the following two propositions:

Proposition 4 *For any pair of nodes* $(i,j)\in V^2$ *and* $i \neq j$*, there exists a pair of positive bounds* h_{ij}^o *and* $h_{ij}^u (> h_{ij}^o)$ *such that the equilibrated travel time* h_{ij}^* *is always in* $[h_{ij}^o, h_{ij}^u]$.

The proposition holds as the free-flow travel time and the bounded travel demand respectively block h_{ij}^* from approaching 0 and $+\infty$.

Proposition 5 *Both the accumulations of idle vehicles* N_l^v *and waiting customers* N_r^c *are bounded under equilibrium for any node* $l \in L$ *and* $r \in R$.

The boundedness of N_l^v and N_r^c are respectively ensured by the finite number of RV fleets N as well as the boundedness of the term $w_r^c \cdot f_{rs}(w_r^c \mid \mathbf{h})$. We then specify $\hat{N}^c$ as a constant greater than $\max\limits_{r\in R}\{\sup\limits_{w_r^c} w_r^c \cdot f_{rs}(w_r^c \mid \mathbf{h^o})\}$, where $\mathbf{h^o} = \{h_{ij}^o\}_{(i,j)\in V^2}$.

Next, we bring back the mapping Γ defined in Appendix A,

$$\Gamma : \left\{ T_{lr}^d = Z_{lr}^{-1}\left(T_{lr}^{i2p+2q} \cdot \Phi_l^v\left(\sum_{k\in M^v(l)} T_{lk}^i, N_l^v \right) \cdot \Phi_r^c\left(\sum_{k\in M^c(r)} T_{kr}^i, N_r^c \right), h_{lr} \right), \forall l \in L, r \in M^v(l) \right\} \tag{97}$$

where the variables $\{N_l^v\}, \{N_r^c\}$ and $\{h_{lr}\}$ previously specified as fixed parameters are now explicitly incorporated. We then have the following proposition,

Proposition 6 *An upper bound* B_u *can always be found with* $\Omega_m = \{\mathbf{T} \mid T_{lr} \in [0, B_u], \forall l \in L, r \in M^v(l)\}$ *, such that* $\Gamma(\mathbf{T}, \mathbf{N^v}, \mathbf{N^c}, \mathbf{h}) \in \Omega_m$ *for all* $T \in \Omega_m$ *when* $\mathbf{N^v} \in [0, N]^L$ *and* $\mathbf{N^c} \in [0, \hat{N}^c]^R$.

Since $Z_{lr}^{-1}(T,h)$ is monotonically increasing on T and decreasing on h, we have

$$\Gamma_{lr}(\mathbf{T}, N_l^v, N_l^c, h_{lr}) \leqslant \Gamma_{lr}(\mathbf{T}, N, \hat{N}^c, 0), \quad \forall l \in L, r \in M^v(l)$$

Then, by following the same procedure as to find the upper bound in Proposition 2, we can retrieve a bound B_u. Also, we bring back the mapping from the solution procedure section,

Proposition 7 *the complex mapping is continuous. and* T^v*,* T^c *shrink down. and* w^c *goes up.*

$\{T_r^c, T_l^v, w_r^c, w_l^v, h_{rs}\} \in \Omega$ should be determined firstly.

Steps for the proof:

- Also, we bring back the mapping from the solution procedure section,

Proposition 8 *The complex mapping is continuous. And* T^v *,* T^c *shrink down. And* w^c *goes up.*

- Necessary step: Define the state of mapping when w^c is infinitely large.
- Define the mapping and prove the the compact and convexity, continuity.
- Prove that the zero flow condition won't happen.

$$\begin{array}{ll} & (w_l^v, T_{lr}^v) \to N^v \\ (N^v, N^c, \mathbf{h}) \to \{T_{lr}^m\} \to & (w_l^c, Q_{rs}) \to N^c \\ & \{T^{rs}\} \to \mathbf{h} \end{array}$$

(see (Lu, 2010), (Hall, 1978)).

Appendix C Properties of the ε-uniform Partitioning Algorithm

In what follows, we prove certain properties of the ε-uniform partitioning algorithm.

Proposition 9 *The coefficient matrix in model (78)-(82) is totally unimodular.*
Proof. Let us assume that the N point are ordered such that the first $|R|$ points are rider trips and the remaining $|D|$ points are driver trips, i.e. $n \in \{1, \cdots, |R|, |R|+1, \cdots, |R|+|D|=N\}$. As a result, the coefficient matrix in model (78)-(82) can be presented as:

$$\begin{array}{r|ccccccccccccccc|}
 & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
 & 0 & 1 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 0 & 1 & \cdots & 0 & 0 & \cdots & 0 \\
 & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\text{Cons. 79} & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \\
 & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 \\
 & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
 & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 1 \\
\hline
 & 1 & 1 & \cdots & 1 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\text{Cons. 80} & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 1 & 1 & \cdots & 1 & 0 & \cdots & 0 \\
\hline
 & 0 & 0 & \cdots & 0 & 1 & \cdots & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\
\text{Cons. 81} & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
 & 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 & 1 & \cdots & 1
\end{array}$$

The columns in this matrix correspond to the following variables:

$$q_{11}, q_{21}, \ldots, q_{|R|,1}, q_{|R|+1,1}, \ldots, q_{|R|+|D|,1}, \ldots, q_{1K}, q_{2K}, \ldots, q_{|R|,K}, q_{|R|+1,K}, \ldots, q_{|R|+|D|,K}.$$

Since total unimodularity is invariant to multiplying a row by a constant, we can multiply the sub-matrices associated with constraints (80) and (81) by -1. As a result, the coefficient matrix becomes equivalent to a node-edge incidence matrix of a graph, where every column has exactly a single 1 and a single -1, and all other entries are zeros. As all node-edge incidence matrices are totally unimodular, the result follows.

Proposition 10 *The problem in model (78)-(82) can be transformed to an unbalanced Hitchcock Transportation problem with* $N+1$ *source nodes and* $2 \times K$ *sink nodes.*
Proof. Let us define 2 disjoint sets of $\mathcal{U} = \{1, \cdots, N+1\}$ and $\mathcal{V} = \{1, \cdots, 2K\}$ to represent the set of sources and sinks, respectively. Without loss of generality, we assume that the first $|R|$ elements in $\mathcal{U}$ are rider trips, each with a supply of 1, followed by D elements of driver trips, each with a supply of 1. The last element in $\mathcal{U}$ denotes a dummy node with a supply of $k([(1+\varepsilon)\frac{|R|}{K}]+[(1+\varepsilon)\frac{|D|}{K}])$. We further assume that the first K elements in $\mathcal{V}$ represent the riders in clusters 1 to K, each with a demand of $[(1+\varepsilon)\frac{|R|}{K}]$, followed by another K elements to represent the drivers in cluster 1 to K, each with a demand of $[(1+\varepsilon)\frac{|D|}{K}]$. The supply and demand vectors are presented by $\mathcal{S}$ and $\mathcal{D}$, respectively. The

arcs can be defined by set $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \mathcal{A}_3$, where $\mathcal{A}_1 = \{1, \cdots, |R|\} \times \{1, \cdots, K\}$, $\mathcal{A}_2 = \{|R|+1, \cdots, N\} \times \{K+1, \cdots, 2K\}$, and $\mathcal{A}_3 = \{N+1\} \times \{1, \cdots, 2K\}$. The cost of each arc, denoted by $\mathcal{C}_{ij}$, for the arcs in sets $\mathcal{A}_1$ and $\mathcal{A}_2$ can be determined from distance matrix C. Finally, the arcs in set $\mathcal{A}_3$ have a zero cost. As a result, the problem can be formulated as in model (98)-(101).

$$Minimize \quad \sum_{(i,j)\in\mathcal{A}} \mathcal{C}_{ij} f_{ij} \tag{98}$$

$$s.t. \quad \sum_{\substack{j\in\mathcal{V}: \\ (i,j)\in\mathcal{A}}} f_{ij} = \mathcal{S}_i, \forall\ j \in \mathcal{V} \tag{99}$$

$$\sum_{\substack{i\in\mathcal{U}: \\ (i,j)\in\mathcal{A}}} f_{ij} = \mathcal{D}_j, \forall\ i \in \mathcal{U} \tag{100}$$

$$f_{ij} \geq 0 \tag{101}$$

In this formulation, the decision variable is the flow from source i to sink j denoted by f_{ij}. The model in (98)-(101) is the mathematical formulation of a Hitchcock Transportation problem with unbalanced number of source and sink nodes (i.e. $2k \lll N+1$), and the result follows.

Appendix D Computational Complexity Before and After Partitioning

In the following, we compare the computational complexity of the problem before and after partitioning to quantify the effectiveness of the proposed algorithm.

Proposition 11 *The objective function of the proxy problem in (3.5) decreases with iterations of Algorithm in 3.5.*

Proof. Let us denote the objective function in (3.5) by Δ which is a function of the assignment variable q_{nk} and cluster center vector tr^k. In Step 1, the cluster centers are fixed, and Δ will decrease from the previous iteration by solving the model in (3.5), since it assigns the points to the clusters so as to minimize Δ.

In Step 2, the assignment variables are fixed, and we find the cluster centers that minimize Δ with respect to tr^k. since Δ is a convex function with respect to tr^k, the optimum point is a minimum. Thus, we have for each $k \in \{1, \cdots, K\}$:

$$\frac{\partial \Delta}{\partial tr^k} = 0 \tag{102}$$

$$\sum_{n=1}^{N} q_{nk}(tr^k - tr_n) = 0 \tag{103}$$

$$tr^k = \frac{\sum_{n=1}^{N} q_{nk}\, tr_n}{\sum_{n=1}^{N} q_{nk}} \tag{104}$$

The equation in (104) simply indicates that Δ will be minimized if we choose the center of a cluster to be the average of all trip vectors in that cluster. Hence, Δ will decrease with iterations.

Proposition 12 *Algorithm 3.5 converges in a finite number of iterations.*

Proof. Let us denote the objective function in (3.5) by Δ which is a function of the assignment variable q_{nk} and the cluster center vector tr^k. The proposed algorithm consists of two steps. In the assignment step, it solves an assignment problem by minimizing Δ given the cluster centers, subject to a set of uniformity constraints. In the update step, it minimizes Δ by letting each cluster's center be the centroid of the points in that cluster given the current assignment of points to clusters. Thus, we can infer that:

1. There are possibly many but finite number of ways to assign N points to K clusters.
2. From one iteration to the next, Δ will not increase (see Proposition 11). Since we stop the algorithm when Δ does not change significantly between two consecutive iterations (i.e., when cluster centers do not change), we can further state that Δ will strictly decrease as iterations proceed.

From 1, we know that the solution set is finite. For each solution in this finite set, there is a unique minimum Δ based on the update step. From 2, we know that the solution should improve, and therefore, we will visit any solution at most once. As such, we conclude that the algorithm in (3.5) converges in a finite number of iterations.

Proposition 13 *Algorithm 3.5 has polynomial time complexity.*

Proof. As noted earlier, the cluster allocation step of one iteration has a worst-case running time of $\mathcal{O}(nk^2(\log n + k \log k))$. Also, the update step requires a running time of $\mathcal{O}(nkd)$, where d is the dimension of a trip vector and is set to 5, at every iteration. Given the maximum number of iterations, denoted by i, the running time complexity of the ε-uniform partitioning is $\mathcal{O}(ink^2(\log n + k\log k))$.

Remark 1 *After partitioning the matching problem based on Algorithm 3.5, the worst-case computational complexity of the original problem is reduced as follows: Graph generation by a factor of* $\frac{(1+\varepsilon)^2}{K}$*, and solving the matching problem by a factor of* $\frac{(1+\epsilon)^3}{K^2}$.

Proof. The most complex sub-problem generated by Algorithm 3.5 could have $(1+\varepsilon)\frac{|R|}{K}$ riders and $(1+\varepsilon)\frac{|D|}{K}$ drivers, requiring $(1+\varepsilon)^2\frac{|R||D|}{K^2}$ trip comparisons to generate the bipartite graph, which is by a factor of $\frac{(1+\varepsilon)^2}{K^2}$ smaller than the original number of trip comparisons. Once graph G_k for partition k is generated, we solve a matching optimization problem whose complexity depends on the size of G_k. More specifically, the computational complexity in each iteration of the Simplex algorithm when solving the max cardinality problem in model (3.1) is proportionate to the product of number of nodes and links, i.e., $\frac{(1+\epsilon)^3}{K^3}|R||D|(|R|+|D|)$, which is by a factor of $\frac{(1+\epsilon)^3}{K^3}$ smaller than the complexity of each Simplex iteration when solving the original problem.

Appendix E Parametric settings for the numerical experiment

Table 2 Characteristics of the Nguyen-Dupuis network

(a) Parameters for the link performance functions

Link	a_{ij}	b_{ij}	Link	a_{ij}	b_{ij}	Link	a_{ij}	b_{ij}
1-5	7	0.0125	6-10	13	0.005	10-11	6	0.0025
1-12	9	0.01	6-5	3	0.0075	10-6	13	0.005
2-8	9	0.0125	6-12	7	0.0025	10-9	10	0.005
2-11	9	0.005	7-8	5	0.0125	11-2	9	0.005
3-11	8	0.01	7-11	9	0.0125	11-3	8	0.01
3-13	11	0.01	7-6	5	0.0125	11-7	9	0.0125
4-5	9	0.01	8-2	9	0.0125	11-10	6	0.0025
4-9	12	0.005	8-7	5	0.0125	12-6	7	0.0025
5-6	3	0.0075	8-12	14	0.01	12-8	14	0.01
5-9	9	0.0075	9-10	10	0.005	12-1	9	0.01
5-1	7	0.0125	9-13	9	0.005	13-3	11	0.01
5-4	9	0.01	9-4	12	0.005	13-9	9	0.005
6-7	5	0.0125	9-5	9	0.0075			

*The units of a_{ij} and b_{ij} are minute and minutes per unit flow, respectively.

(b) Regular traffic demand

O\D	1	2	3	4	5
1	0	225	600	0	0
2	150	0	0	450	300
3	450	0	0	300	240
4	0	375	150	0	0
5	0	150	225	0	0

(c) Potential ride-sourcing service demand

O\D	1	2	3	4	5
1	0	150	400	0	0
2	100	0	0	300	200
3	300	0	0	200	160
4	0	250	100	0	0
5	0	100	150	0	0

(d) Parametric values for the system setting

Param	Value	Unit	Param	Value	Unit	Param	Value
$\beta^0, \hat{\beta}^0$	20	$/h	F_0	2	$	q_N^v, q_N^c	1, 1
$\beta^i, \hat{\beta}^i$	6, 12	$/h	τ	60	$/h	q_T^v, q_T^c	0.1, 0.1
γ	10	$/h	$\hat{\theta}$	0.01	1/$	q_h	0.1
θ	0.5		$\hat{w}^c$	0.5	h	η	10
N	2200		$\hat{F}_{rs}$	$0.8 F_{rs}$	$		

Appendix F Impact

1. Presentations
   (a) Peer-to-Peer Ridesharing: Using the Existing Passenger-Movement Capacity to Serve the Transportation Demand, Apr. 2018, Neda Masoud, ASCE speaker series, University of Michigan
   (b) An Optimization Framework for Shared Mobility in Dynamic Transportation Networks, 2017, Neda Masoud, Michigan Innovative Mobility Symposium, Ann Arbor, MI
   (c) Modeling Spatial Effects of Surge Pricing in Ride-Sourcing Markets, July 2017, Yafeng Yin, 2017 Conference for Computational Transportation Science, Lanzhou, China
   (d) Research Needs for Achieving Connected and Automated Mobility, August 2017, Yafeng Yin, Workshop for Future Mobility Systems, University of Illinois Urbana-Champaign
   (e) Modeling and Analysis of Ride-Sourcing Services, March 2018, Yafeng Yin, Invited Seminar, Tongji University, Shanghai, China

2. Journal Papers/Reports (full citation)
   (a) Zhengtian Xu, Zhibin Chen and Yafeng Yin (2019) Equilibrium Analysis of Urban Traffic Networks with Ride-Sourcing Services, *Transportation Research Part B: Methodological* (submitted)
   (b) Tafreshian Amirmahdi and Neda Masoud (2020) Trip-based Graph Partitioning for Parallel Computing in Ridesharing, *Transportation Reserach Part C: Emerging Technologies. 114, 532-553*

3. New Courses (Title, Undergraduate/Graduate, Date)
   (a) Course modules on equilibrium analysis of urban traffic networks with ride-sourcing services, CEE 557 Transportation Network Modeling, Graduate, Winter 2019.