# Trading flexibility for adoption: Dynamic versus static walking in ride-hailing

S. Martin[a], S. J. Taylor[b] and J. Yan[c,*]

[a] Kellogg School of Business, Northwestern University, Evanston, IL, USA
sebastien.martin@kellogg.northwestern.edu
[b] Lyft, San Francisco, CA, USA
sjt@lyft.com
[c] Sauder School of Business, University of British Columbia, Vancouver, BC, Canada
julia.yan@sauder.ubc.ca
* Corresponding author

*Extended abstract submitted for presentation at the 11th Triennial Symposium on Transportation Analysis conference (TRISTAN XI)*
*June 19-25, 2022, Mauritius Island*

April 1, 2022

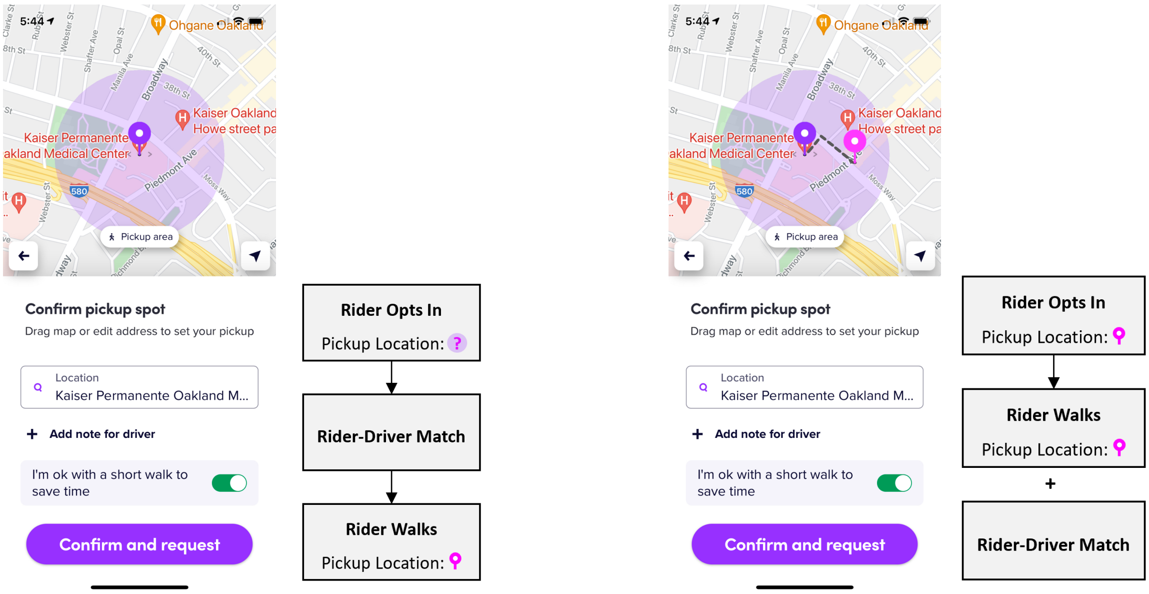
Keywords: walking, ride-hailing, flexibility

## 1 INTRODUCTION

Ride-hailing is an on-demand, individual transportation service that can be appealing for its convenience and comfort. By contrast, mass transit and microtransit services ask riders to be *flexible* in time (waiting) and space (walking), enabling the services to operate more efficiently than purely on-demand systems. A key question, then, is whether on-demand ride-hailing can incorporate a degree of rider flexibility into its operations. Cost savings from improved operations in ride-sharing could be passed onto riders while also mitigating congestion and emissions.

Riders' flexibility in time has been studied under the Vehicle Routing Problem with Time Windows (Kolen *et al.*, 1987), and their flexibility in space has been studied under the Vehicle Routing Problem with Floating Targets (Gambella *et al.*, 2018, Zhang *et al.*, 2020). In both problems, riders are asked to accept a window of time or space for their pickup to occur, expanding the feasible region of the optimization problem relative to problems with fixed pickup times and locations. Driver assignments and routes are then optimized jointly within these windows to find higher-quality solutions. Riders' flexibility in space has attracted relatively less attention than their flexibility in time. However, walking can shorten trips, bring riders closer to drivers for faster pickup times, or coalesce demand for shared rides, especially when traffic and one-way streets restrict drivers' mobility (Stiglic *et al.*, 2015).

In 2021, Lyft introduced a product that encouraged time-sensitive riders to accept a short walk to pickup locations that would enable faster travel times. A screenshot of Lyft's current walking user interface is shown in Figure 1a. As a rider selects the origin location with a pin, a bubble with a short walking radius is shown around the pin, representing the "floating target" in the Vehicle Routing Problem with Floating Targets (VRPFT). Riders are then asked via a toggle switch whether they are willing to walk to a pickup location that could lie anywhere within the bubble.




(a) *Floating/Dynamic Walk* (b) *Fixed/Static Walk*

Figure 1 – *Two proposed user interfaces for walking.*

The VRPFT is attractive for the flexibility gained from the expansion of the feasible region, but it takes as given that riders are willing to accept the *uncertainty* that comes with this flexibility. The uncertainty arises from the joint optimization of riders' pickup locations and drivers' assignments, which means that riders' pickup locations remain unknown until the solution is computed. Rider opt-in is critical to realizing the benefits of walking, motivating our interest in lessening the burden on riders while still leveraging their flexibility.

We make the following contributions. First, we formulate **a *dynamic walking* model.**, which jointly determines riders' pickup locations and driver assignments as in Figure 1a. Second, we propose the new ***static walking* paradigm**, which alleviates uncertainty for riders by showing a static, predetermined pickup location to the rider before asking for the opt-in and assigning a driver. A user interface is shown in Figure 1b; here, the rider is asked to walk along the gray dashed line from the purple origin pin to the pink pickup pin. We propose algorithms to determine static pickup locations and discuss network characteristics that make static walking viable. Third, we provide **detailed simulations quantifying the *value of walking***, defined as the travel time savings for walking relative to no walking, in minutes per ride. The simulations are built using real Lyft data on drivers' locations and riders' origins and destinations in Manhattan, and a detailed representation of the underlying road network. We show that despite being more constrained, static walking performs competitively, achieving as much as 94% of the value of dynamic walking. Consequently, static walking can outperform dynamic walking if it achieves as little as a 6% relative increase in adoption rate. Finally, we provide **empirical evidence of the value of static walking.** Using a fixed-effects model on hundreds of thousands of Lyft rides, we show that riders who are very close to our static pickup locations see substantial improvements in travel times.

## 2 METHODOLOGY

**Basic matching** A service region is represented on a directed graph $G = (\mathcal{V}, \mathcal{E})$. The function $c(u, v)$ returns the driving time from node $u$ to node $v$. In each period, sets of riders and available drivers arrive; we use the random variables $N$ and $M$ to denote the number of riders and drivers arriving in a period, respectively.[1] Each rider $i$ is associated with origin-destination

[1] We will consider driver availability high enough that we can assume $M \geq N$, which is usually true in practice.

$(O_i, D_i) \in \mathcal{V}^2$, and each driver $j$ with location $L_j \in \mathcal{V}$.

We use the binary variable $x_{i,j}$ to denote whether driver $j$ is matched to rider $i$, and seek to minimize the total travel time (pickup/deadheading plus in-vehicle). The formulation is then:

$$\min_{\mathbf{x}\in\mathcal{X}} \quad \sum_{i=1}^{N}\sum_{j=1}^{M} \left(c(L_j, O_i) + c(O_i, D_i)\right) x_{i,j}, \tag{1}$$

where $\mathcal{X} = \{\mathbf{x} \in \{0,1\}^{N\times M} | \sum_{i=1}^{N} x_{i,j} \leq 1 \ \forall j \in [M],\ \sum_{j=1}^{M} x_{i,j} = 1 \ \forall i \in [N]\}$. Problem (1) is a bipartite matching problem, standard in the ride-hailing literature.

**Walking** Now, consider that riders can walk from their origin to a nearby pickup location. To this end, we use the function $\delta(u, v)$ to denote the walking time from node $u$ to node $v$, and define the set $\mathcal{W}_\Gamma(u)$ to be the set of nodes within a walking radius $\Gamma$ of node $u$.

We introduce the idea of a *walking function* $f : \mathcal{V}^3 \to \mathcal{V}$, which maps an origin-destination pair and driver location to a pickup location for the rider to walk to. For a given walking function $f$, we can express the matching problem as:

$$J(S \,|\, f) = \qquad \min_{\mathbf{x}\in\mathcal{X}} \quad \sum_{i=1}^{N}\sum_{j=1}^{M} \bar{c}\left(O_i, D_i, L_j, f(O_i, D_i, L_j)\right) x_{i,j}, \tag{2}$$

where we use $\bar{c}(O, D, L, w) = \max\{c(L, w), \delta(O, w)\} + c(w, D)$ for the total travel time when a rider for origin-destination pair $(O, D)$ is picked up at $w$ by a driver coming from location $L$. We use $S = (\{(O_i, D_i)\}_{i=1}^{N}, \{L_j\}_{j=1}^{M})$ as shorthand for the state of all riders and drivers in a period.

For *no walking* and *dynamic walking*, we apply the following functions to Problem (2):

$$f^{\mathsf{NO}}(O, D, L) = O, \tag{3a}$$

$$f^{\mathsf{DYN}}(O, D, L) = \operatorname*{argmin}_{w\in\mathcal{W}_\Gamma(O)} \ \bar{c}(O, D, L, w). \tag{3b}$$

A *static walking function* will assign a riders' pickup node based only on their origin and destination. It therefore must satisfy the following conditions: (i) $f(O, D, L) \in \mathcal{W}_\Gamma(O)$, and (ii) $f(O, D, L) = f(O, D, L')\ \forall L, L'$ ($L$ could of course be removed from the arguments). We use $\mathcal{F}^{\mathsf{STAT}}$ to denote the set of walking functions that satisfy these conditions. Among the feasible static walking functions, we are interested in achieving the best expected performance, namely:

$$f^{\mathsf{STAT}} \in \operatorname*{argmin}_{f\in\mathcal{F}^{\mathsf{STAT}}} \mathbb{E}_S[J(S \,|\, f)]. \tag{4}$$

In the full paper, we will propose algorithms to solve (4). We can quantify their performance by computing the following values of walking:

$$V^{\mathsf{DYN}} = \frac{1}{\mathbb{E}[N]} \left(\mathbb{E}_S[J(S \,|\, f^{\mathsf{NO}})] - \mathbb{E}_S[J(S \,|\, f^{\mathsf{DYN}})]\right), \tag{5a}$$

$$V^{\mathsf{STAT}} = \frac{1}{\mathbb{E}[N]} \left(\mathbb{E}_S[J(S \,|\, f^{\mathsf{NO}})] - \mathbb{E}_S[J(S \,|\, f^{\mathsf{STAT}})]\right). \tag{5b}$$

As $V^{\mathsf{DYN}} \geq V^{\mathsf{STAT}}$, we have a convenient measure of the optimality gap when we solve (4).

## 3 RESULTS

Here, we present a selection of simulated results and empirical analysis on historical Lyft data in Manhattan. The road network was represented as a graph structure using the `OSMnx` package in Python, with travel times from Uber Movement. On the Manhattan network, we simulated arrivals of riders and drivers using internal Lyft data on riders and idle drivers. Our analysis included both a *short walk* setting that would ask riders originating on a road segment to simply walk to one of the two intersections at either end of the origin road segment, as well as a *long walk* setting that would ask riders to walk to any intersection within about a five minute walk.

Table 1 shows results for dynamic and static walking. Naturally, dynamic walking has the highest value, saving 1.69 minutes per ride for short walks and 3.47 minutes per ride for long walks. But surprisingly, static walking performs competitively with dynamic walking, achieving 94% of the value for a short walk radius, and 85% of the value even for the long walk radius.

Table 1 – *Performance metrics on Manhattan simulation*

| **Walk Radius** | **Design** | **Walk Time** | **Value of Walking** (min./ride) | | |
|---|---|---|---|---|---|
| | | (min./ride) | Pickup | In-Vehicle | Total (%Dyn) |
| Short Walk | Dynamic | 1.04 | 0.81 | 0.88 | 1.69 (100%) |
| | Static | 1.05 | 0.64 | 0.95 | 1.59 ( 94%) |
| Long Walk | Dynamic | 2.98 | 1.27 | 2.20 | 3.47 (100%) |
| | Static | 3.24 | 0.46 | 2.48 | 2.94 ( 85%) |

We also fit a fixed-effects model to data on hundreds of thousands of historical Lyft rides in Manhattan to estimate the value of walking at different distances from our static pickup locations. The estimates are plotted in Figure 2 and show that the value of walking can be substantial, with especially high value for shorter walks. These estimates give us confidence that our static pickup locations are indeed valuable, and that the value in our simulations would translate well in practice.

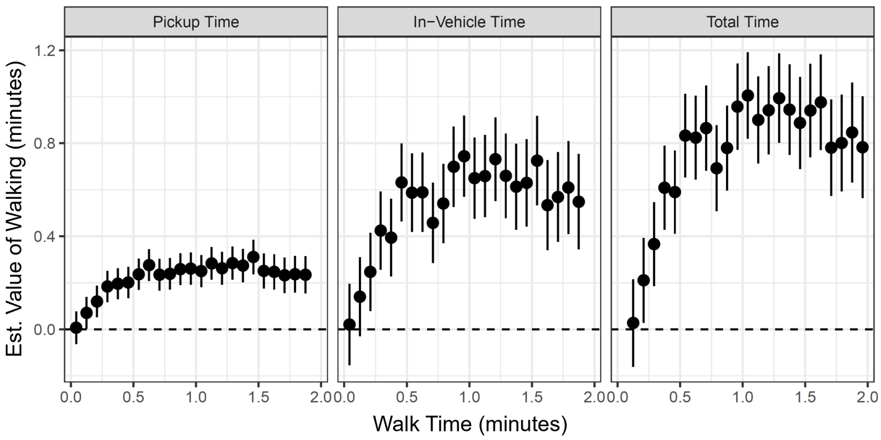



Figure 2 – *Empirical estimates of value of walking for different bins of walking time.*

## References

Gambella, Claudio, Naoum-Sawaya, Joe, & Ghaddar, Bissan. 2018. The vehicle routing problem with floating targets: Formulation and solution approaches. *INFORMS Journal on Computing*, **30**(3), 554–569.

Kolen, Antoon WJ, Rinnooy Kan, AHG, & Trienekens, Harry WJM. 1987. Vehicle routing with time windows. *Operations Research*, **35**(2), 266–273.

Stiglic, Mitja, Agatz, Niels, Savelsbergh, Martin, & Gradisar, Mirko. 2015. The benefits of meeting points in ride-sharing systems. *Transportation Research Part B: Methodological*, **82**, 36–53.

Zhang, Wei, Jacquillat, Alexandre, Wang, Kai, & Wang, Shuaian. 2020. Geometric Optimization for Routing with Floating Targets. *In: TSL Second Triennial Conference*.