Effect of Routing Constraints on Learning Efficiency of Destination Recommender Systems in Mobility-on-Demand Services

Gyugeun Yoon, Joseph Y. J. Chow, Assel Dmitriyeva, and Daniel Fay

***Abstract*—With Mobility-as-a-Service platforms moving toward vertical service expansion, we propose a destination recommender system for Mobility-on-Demand (MOD) services that explicitly considers dynamic vehicle routing constraints as a form of a "physical internet search engine". It incorporates a routing algorithm to build vehicle routes and an upper confidence bound based algorithm for a generalized linear contextual bandit algorithm to identify alternatives which are acceptable to passengers. As a contextual bandit algorithm, the added context from the routing subproblem makes it unclear how effective learning is under such circumstances. We propose a new simulation experimental framework to evaluate the impact of adding the routing constraints to the destination recommender algorithm. The proposed algorithm is first tested on a 7 by 7 grid network and performs better than benchmarks that include random alternatives, selecting the highest rating, or selecting the destination with the smallest vehicle routing cost increase. The RecoMOD algorithm also reduces average increases in vehicle travel costs compared to using random or highest rating recommendation. Its application to Manhattan dataset with ratings for 1,012 destinations reveals that a higher customer arrival rate and faster vehicle speeds lead to better acceptance rates. While these two results sound contradictory, they provide important managerial insights for MOD operators.**

***Index Terms*—Mobility-on-Demand, destination recommendation, contextual bandit algorithm, insertion heuristics, physical internet.**

I. Introduction

MOBILITY -on-Demand (MOD) systems, which include a wide range of services including rideshare, car- and bike-share, e-hail taxis, and microtransit, provide broader options to travelers [1]. There is a shift from simply operating transportation services to becoming a comprehensive service platform that addresses all of a traveler's needs: helping them plan a journey, book the trips with operators, transport the passenger, pay for the trips, etc. This platform-oriented perspective is called Mobility-as-a-Service (MaaS), increasing accessibility toward door-to-door (DTD) service and improving energy efficiency [2]. It is becoming more prevalent (see [3], [4]), following advances in information and communications technologies (ICT) that deal with real-time interactions between service operators and travelers. In the Manhattan central business district (CBD), the average number of passengers that used app-based transportation services was 202,262 in 2017 [5] while conventional taxis served 249,767 passengers which decreased from 378,166 in 2013.

MOD systems can benefit stakeholders of transportation area in various ways, not to mention travelers and operators. Several studies considered such benefits as improving social welfare by non-myopic dynamic pricing [6], better profit and consumer surplus by altering service types [7], increasing capacity utilization, trip throughput, and welfare with dynamic waiting [8], or improving livability and environment of an urban area [9].

Despite having tangible benefits, high operating costs impede the sustainability of MOD operation even if planned in advance. For example, companies like Uber continue to operate at a loss [10]. Furthermore, Access-A-Ride (AAR) paratransit service in NYC costs as much as $71 per trip to operate [11] (given the accessibility requirements and inflexibility of offline scheduling, it serves as an upper bound cost on MOD services). Moreover, MOD services are vulnerable to unplanned disruptions and cancellations, as such incidents can impact service quality for users sharing the service [12].

Nevertheless, MOD systems can provide substantial support to disadvantaged and senior populations that have limited mobility options. AAR served 6,170,876 trips in NYC throughout 2017 [13], and 60% of users could be older than 65 according to their service satisfactory surveys. Also, 80% of them needed the assistance of devices like canes, walkers, or wheelchairs. These populations tend to have limited access to helpful information to navigate to their destinations, especially for secondary trip purposes including social recreation, dining, shopping, and others. According to surveys conducted with seniors in NYC and El Paso, the percent of seniors using smartphones to navigate trips were between 49% to 62% [14], which is lower than other age groups with ratios around 90% [15]. Thus, seniors may end up choosing only familiar destinations that are outdated and unsatisfactory [16].

Access to information for destinations can be addressed with "recommender systems." For a set of users C and a set of items $\mathbb{A}$, recommender systems are designed to recommend an item $s'_c \in \mathbb{A}$ such the user c's utility $u(c, s)$ is maximized

This research was supported by the C2SMART, a USDOT Tier 1 University Transportation Center, and a grant from NSF, CMMI-1652735.

G. Yoon is with the Department of Civil and Urban Engineering, New York University, Brooklyn, NY, 11201 USA. (e-mail: ggyoon@nyu.edu)

J.Y.J. Chow is an Assistant Professor at the Department of Civil and Urban Engineering, New York University, Brooklyn, NY, 11201 USA. He is also the Deputy Director of C2SMART. (e-mail: joseph.chow@nyu.edu).

A. Dmitriyeva was with Interactive Telecommunications Program, Tisch School of Art, New York University, Manhattan, NY, 10003 USA. (e-mail: assel@nyu.edu).

D. Fay was with Center for Urban Science and Progress, New York University, Brooklyn, NY, 11201 USA. He is now with Microsoft (e-mail: df1383@nyu.edu).

Manuscript received July XX, 2019; .

as shown in Eq. (1) [17]. The use of a recommender system for destinations is a natural next step for MOD services, especially those moving toward a MaaS paradigm. For example, emerging MaaS platforms like moovel and Whim from MaaS Global seek to provide integrated mobility services to match travelers with trips among various operators. It is not a far leap to consider destination recommendation as part of an itinerary planning process for them, particularly in light of the presence of network disruptions that can severely increase the cost of accessing an initial passenger destination, similar to how Waze provides recommendations on alternative routes [18].

$$s'_c = \arg\max_{s \in \mathbb{A}} u(c, s), \quad \forall c \in C \tag{1}$$

There are many examples in practice, where there have been attempts to combine destination recommendation and mobility service to improve user experiences. In Helsinki, Finland, the city government's marketing company collaborated with Whim and WeChat, a Chinese multi-purpose mobile messenger, to provide Chinese tourists with recommendations to local attractions and multimodal mobility options. In Spain, mobility startup Iomob partnered with Smartvel, a destination content provider, to share destination information and offer them to mobility users to help their destination decision-making.

These real-world examples demonstrate the demand that exists for such integrative services, particularly for users looking for shopping and sociorecreational activities. By tying the destination to mobility service like in Figure 1, the mobility service's routing cost, which affect the service fare, can be better customized to a person's destination preferences. For instance, a system may recommend a place that differs from a passenger's top choice but still within acceptability while achieving a cheaper operating cost and resulting fare price. However, such integrations have only considered proximity until now, not directly considering routing as this study does.

We can further infer a rough estimation of expected service demand using public data. According to the dashboard of taxi and ridehailing usage in New York City [19], 749,117 trips were made per day in February 2020. 25% of those trips are for either social (14.4%) or shopping (10.6%) trip purpose, per the 2010/2011 Regional Household Travel Survey conducted by New York Metropolitan Transportation Council [20]. If even 1% of those users want destination recommendation built in, there would be about 1,800 users per day using the feature.

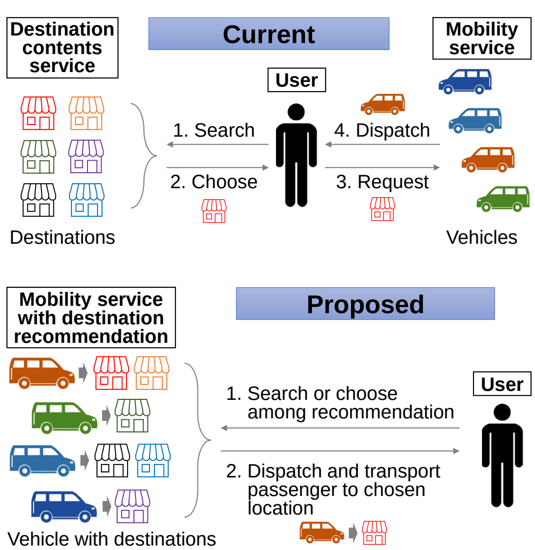


Fig. 1. User interaction with current and proposed system.

Learning in recommender systems can be achieved with multi-armed bandit (MAB) algorithms [21], [22]. These algorithms seek to balance information acquisition with maximizing user satisfaction which corresponds to exploration and exploitation in reinforcement learning [23]. Moreover, the integration of MOD and recommendation service does not need to be done using only smartphone apps. Existing infrastructure can also accommodate a broader range of users including seniors who may not have access to smartphones, such as through phone calls, kiosks at taxi-stands or shared taxi or microtransit virtual stops, or through an interface in each seat of a MOD vehicle. Furthermore, such systems can help connect new local businesses to users as a type of "physical internet search engine" (see [24], on physical internet). Such a mobility-based search engine can do for physical businesses what internet search engines have done for e-commerce.

Despite these advantages, reinforcement learning for destination recommender systems on MOD services are almost non-existent in practice.

Like with the dynamic content of the internet [25], physical destinations are also highly dynamic and large-dimensional: in NYC, the number of restaurants alone was nearly 27,000 in 2017 with a net increase of 587 from 2016 [26]. Unlike internet content, however, destination recommendation regarding MOD service is even more complex because the contextual environment is path dependent. A person being picked up in lower Manhattan may prefer a restaurant in Midtown but another person in Downtown Brooklyn may not. This means a user's preference for different items will vary in each observation. The cost of delivering the person to the destination also depends on the pickup and drop-off locations of other passengers sharing the service and on the time-varying traffic conditions. To the best of our knowledge, location-based recommendation systems have ignored routing constraints of MOD operations in the destination recommendations [27], much less considered these heterogeneities between users and contexts. These conditions suggest that MOD-based destination recommendation can be much harder to efficiently learn from than even internet content.

Such a system is meant to operate in a MaaS environment (see [4], [28]) in which travelers simply access a single platform/gateway (e.g. moovel, Whim, etc.). As we see more *vertical expansion* (see [29]) of services in MaaS, it will go beyond simply providing trips to actively engaging with users in planning/anticipating travel needs, including recommending destinations to them. As an analogy, Blockbuster, a former giant in a movie video rental service market, used to be a hotspot for cinephiles looking for movies they want to watch. On the other hand, today Netflix provides lists of recommended content, and subscribers enter the app to be recommended a movie to watch. Due to innovations in transportation market, we will be able to observe mobility services intertwining with a certain extent of recommendation.

There is no study that quantifies features within a MOD service environment (routing constraints like variable travel times due to traffic, passenger densities, vehicle capacities, etc.) on the efficiency of contextual bandit algorithms. To pave the way for the operation of destination recommender systems as a new type of physical internet search engine, we develop a routing-constrained contextual bandit algorithm for destination recommendation for MOD services. Our primary contribution is the study of the effects of various routing parameters on its learning efficiency using realistic data from NYC.

II. Prior studies

Studies in ride-hailing have worked on various subjects, and some significant ones are analyzing impact on transportation systems (e.g. [30]–[32]), fare pricing and revenue (e.g. [6], [33]–[38]), and vehicle routing and relocation (e.g. [39]–[41]). They solved problems or found insights by manipulating endogenous factors such as system parameters or configurations. One of the representative exogenous factors is the trip demand but studies that considered the demand as variable mainly focused on temporal fluctuation without changing origin or destination. Although there are studies suggesting some walk to nearby locations to increase the chance of matching, they did not mean the alteration of actual origins or destinations.

In our work, we give users an option of changing the actual destination by implementing a destination recommendation in MOD services. Recommender systems are widely used in such online content services as providing articles, products, advertisements, or videos. Systems offer several alternatives to users and observe their choices of acceptance, which usually can be coded in binary outputs of 0 and 1. Based on accumulated data of users' choices, systems decide which alternatives indicate higher probabilities to be chosen and display them in priority as shown in Eq. (1). If effective, those options successfully induce more clicks or visits from users, potentially increasing hits or time staying on their website.

MAB algorithms involve a finite sequence of decisions made to select an alternative s_t from a set of alternatives $\mathbb{A}$ in each trial $t \leq T$ to maximize a cumulative reward $\sum_{t=1}^{T} r_{s_t,t}$. The rewards of trials are randomly drawn from a distribution which is unknown to the decision-maker (the system or service provider), and not dependent on the decision-maker's choice. Because of the uncertainty in each trial, a measure based on regret minimization (Eq. (2a)) is typically used if the true reward can be observed and the other one (Eq. (2b)) if only the user's choice of the recommended alternative is observed.

$$\rho_n = n r_{s^*} - \sum_{t=1}^{n} r_{s_t,t} \quad \text{(2a)} \qquad\qquad \rho_n = \sum_{t=1}^{n} y_{s_t,t} \quad \text{(2b)}$$

ρ_n is the measure of reward, r_{s^*} is the maximum mean reward obtained from alternative s^*, $r_{s_t,t}$ is the reward at t from chosen alternative s_t, and $y_{s_t,t}$ is a binary variable indicating whether the user accepts (1) or rejects (0) the recommended alternative s_t at t.

Earlier work clarified the concept of the MAB problem and linked it to other existing methodologies. For example, Bellman and Brock [42] summarized the concept of the two-armed bandit problem including objective function, assumptions, and possible approaches. Whittle [43] mentioned several different types of sequential decision processes with the MAB problem as one example. Gittins [44] introduced the problem as a Markov decision process and proposed a dynamic allocation index to figure out how the complexity can be reduced. Berry and Fristedt [45] addressed discounting, which puts more weight on current rewards than future rewards, to derive optimal strategies.

Due to the uncertainty in outcomes, the effectiveness of MAB algorithms as selection policies is evaluated in terms of the rate at which the worst-case bounds change over time. Algorithms have been proposed to minimize this rate, such as the Upper Confidence Bound (UCB) algorithm. Auer *et al.* [46] proved that the regret in the UCB algorithm grows at a logarithmic rate and compared it to an ε-greedy algorithm from earlier studies. Vermorel and Mohri [47] conducted empirical evaluations of multi-armed bandit algorithms. Results showed that complicated strategies did not always beat simpler strategies but outperformed them when tested with real-world data instead of artificial datasets. Bubeck and Cesa-Bianchi [48] provided a review of variants and extensions of MABs including contextual bandit problems where the decision-maker can observe contextual feature vectors $\boldsymbol{x_{s,t}}$ for each alternative item $s \in \mathbb{A}$ in each trial $t \leq T$ prior to making a recommendation.

Researchers have made improvements to algorithms for the contextual bandit problem [49]–[51]. Li *et al.* [52] developed a generalized linear contextual bandit algorithm based on the upper confidence bound and achieved a lower regret bound. As the generalized linear function covers various types of functions including random utility functions according to a binary logit model, we focus on their algorithm.

None of the literature on location recommendation have considered routing constraints. Numerous studies focused on applying the algorithm to spatiotemporal data and providing points of interests satisfying filters set by users including distance, reputation, or type [53]–[56]. These don't consider the added context of a MOD routing service. Chow and Liu [39] proposed recommendation for points of interest based on routing costs and activity benefits but did not consider an online system that sequentially learns from user input as a conventional recommender system would. Gutowski *et al.* [57] proposed a conceptual framework that builds context for individuals from each user's profile, mobile device, activity, and environment to recommend general services and information.

Römer *et al.* [58] implemented a contextual bandit process to control charging demands of electric vehicles by adjusting the price and recommending stations to users. Considering station load, charging price, or income as features which affect driver behavior, they analyzed the effect of bandit algorithms on maximum loads at stations and average rewards of drivers. Song [59] proposed optimizing personalized menus for flexible MOD services where items on the menu represent route and mode combinations. An MAB algorithm was used for selecting the alternative to expose and the user would choose to accept or reject the recommendation. The algorithms showed better performance than content-based ones, but the difference significantly decreased when heterogeneity increased. Zhou *et al.* [22] applied an MAB algorithm to sequential departure time and path selection considering on-time arrival reliability. Mean rewards of alternatives incorporated early and late arrival times and corresponding penalties. Zhu and Modiano [60] dealt with travel time delays on the network where delays were collected only in total along the path and those of individual links would be revealed if selected.

Different from prior studies, the proposed study seeks to better understand the behavior of MAB recommender systems where online routing cost is a feature. The reward function for our proposed recommender system is based on an online routing in which a vehicle updates its route with existing customers having pickup and drop-off locations to serve the new request. The routing function is a dynamic dial-a-ride problem (see [6], [61], [62]), an NP-hard problem typically solved using heuristics for practical size problems, especially in an online setting. For convenience in testing the relationship between the MAB algorithm and routing cost parameters, we implemented an insertion heuristic to compute the change in the route cost. Consequently, our contextual bandit algorithm has the added context which muddles the learning further.

III. Methodology

A. Preliminaries

Consider a sequence of T independent trials in which each trial t a new customer from origin d_t asks for a recommendation to a destination $s \in \mathbb{A}_t \subseteq \mathbb{A}$ from a MOD service. For this study, the MOD service is treated as only a single shuttle (as a worst case scenario; a fleet would be more flexible and would reduce the impact of routing constraints on the learning) that provides dynamic pickups and drop-offs to customers (who may not be using destination recommendation). There are $n_t \geq 0$ customers (of which $n_b \leq n_t$ and $n_b \leq q$ are already on-board the vehicle with passenger capacity q) currently being served by the shuttle according to a route p_{0t} (and route cost w_{0t}) that sequences the set of locations $\{1, 2, \cdots, n_t, n_t+1, n_t+2, \cdots, 2n_t\}$, where $i \leq n_t$ is a pickup location and $i + n_t$ is the corresponding drop-off location. Based on the recommendation, the shuttle would have to add the pickup location d_t and chosen destination s_t to form a new route $p_{s_t,t}$ (and route cost $w_{s_t,t}$). The route has an operating cost based on sum of travel costs c_{ij} (assumed to include an average dwell time at locations for picking up or dropping off passengers) between each location pair (i, j).

Each destination $s \in \mathbb{A}_t$ has a preference ranking $\pi_{s,t}$ that varies per t with an observable mean ranking $\bar{\pi}_s$. A user's preference for the destination depends on a feature vector that includes the mean ranking $\bar{\pi}_s$ and route cost difference $\Delta w_{s,t}$. For the purpose of this study, we focus on having only these two features to study their interactions without additional noise, but more complex vectors can be specified for implementation. The user's preference is quantified with an unobservable utility function $u_{st}(\bar{\pi}_s, \Delta w_{s,t})$. Based on the recommended destination s_t, the user responds by either

TABLE I
Summary of notation used

Notations	Explanations
T, t, τ	Number of total trials, moment of trial where $t \in [1, T]$, number of initial learning trials
$\mathbb{A}$, $\mathbb{A}_t$	Universal set of alternatives, alternative subset at t where $\mathbb{A}_t \subseteq \mathbb{A}$
d_t, s	Origin location of new request at t, available alternative destination where $s \in \mathbb{A}_t$
s_t^*, s_t	Optimal destination at t, recommended destination at t where $s_t \in \mathbb{A}_t$
n_t, n_b	Number of total passengers being served at t, number of passengers on-board
n, k_t	Index of passenger where $n \in [1, n_t]$, number of pickups where $k_t \in [1, 2n_t]$
q, $\bar{v}$, λ	Vehicle capacity, mean vehicle speed, passenger arrival rate
η, γ_t, α	Travel time conversion factor (pace), degree of congestion at t, exploration factor
l_0, o_n, d_n	Initial vehicle location, origin of passenger n, destination of passenger n
p_{0t}, w_{0t}	Initial route before the new request at t, route cost of p_{0t}
p_{st}, w_{st}	Revised route including the new request visiting s at t, route cost of p_{st}
p^n, p_i^n, p_{ij}^n	Shortest route with n OD pairs, p^n with the new origin at $(i+1)$th place, p_i^n with the new destination at $(i+j)$th place
w^n, w_{ij}^n	Route cost of p^n, route cost of p_{ij}^n
$\Delta w_{s,t}$	Route cost increment when the vehicle serves d_t and s additionally
$\lVert \boldsymbol{x} \rVert_A$	Weighted l_2-norm associated with a positive definite matrix A where $\lVert \boldsymbol{x} \rVert_A := \sqrt{\boldsymbol{x}^T A \boldsymbol{x}}$
c_{ij}, c_{ij0}	Travel cost between location i and j, free flow travel cost between i and j
$\pi_{s,t}$, $\bar{\pi}_s$	Preference ranking of destination s at t, observable preference
$\boldsymbol{\theta}$, $\boldsymbol{\theta_t}$, $\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}$, $\boldsymbol{\theta_t^T}$	True coefficient vector, $\boldsymbol{\theta}$ at t, estimated coefficient vector at t, transposed vector of $\boldsymbol{\theta_t}$
$U_{s,t}$, $V_{s,t}$	Utility of s at t, estimated systematic utility of s at t
P_t, $\mathbb{P}$	Probability that the passenger accepts recommended alternative at t, passenger pool
$\boldsymbol{x_{s,t}}$, y_t	Feature vector of s at t, passenger response at t
$\boldsymbol{X_t}$, $\boldsymbol{Y_t}$	Accumulated feature vectors until t, accumulated responses until t
R_t, $\phi(t)$, $\rho(t)$	Regret at t, acceptance rate at t, average regret at t
μ	A strictly increasing function representing cumulative probability of acceptance

accepting s_t ($y_t = 1$) or not ($y_t = 0$). In practice, the system can provide multiple options to choose from instead of the single s_t to enhance customer satisfaction level, as we can see with such systems belonging to examples like Netflix and Amazon. Nevertheless, we keep the number of recommended alternatives to one per trial for the purpose of experimental efficiency in our simulation design. For example, the destination recommender system can be implemented using a multinomial logit choice model instead of a binary choice model, but the evaluation of experiments is clearer using simpler measures like number of clicks/acceptances designed for binary choice. The objective is to maximize the acceptance rate ($\sum_{t=1}^{T} y_t/T$) of s_t over T trials. A notation is summarized in Table I.

The proposed mechanism can work with any routing heuristic provided as an exogenous input. For this study, we assume the operator uses a standard insertion heuristic for constructing routes as shown in Algorithm 1 (see [61]). The proposed learning mechanism will be evaluated against other mechanisms that also use the same routing algorithm to be consistent.

B. Proposed algorithm

The proposed algorithm, RecoMOD, assumes a learning period τ and an exploration factor α. The exploration factor is used to balance between exploration and exploitation; higher values of α result in placing more value on exploration; the algorithm becomes myopic when $\alpha = 0$. RecoMOD is adapted from [52] and modified to include routing-based features, as presented in Algorithm 2.

When the system encounters a new request in a trial, the mobility service may be in the middle of serving a queue of n_t jobs. The request arrives after the system has already picked up $k_t \leq n_t$ passengers. As such, only the remaining portion of the original route, including $2n_t - k_t$ points, is reconsidered for rerouting. If there are no existing passengers, the existing route cost is set to $w_{0t} = 0$. While the origin of the new request is fixed, the destination to be recommended is not yet decided. Therefore, for all candidate destinations $s \in \mathbb{A}_t$, excess travel costs $\Delta w_{s,t} = w_{st} - w_{0t}$ are calculated. The criteria of choosing $\mathbb{A}_t$ from $\mathbb{A}$ can vary; setting them equal can be computationally expensive. The code for the algorithm is available on our lab Github site: https://github.com/BUILTNYU/DestinationRecoMOD.

Once $\Delta w_{s,t}$ is obtained, it is added to $\boldsymbol{x_{s,t}}$ for each $s \in \mathbb{A}_t$, which can include other features. We assume that users' behaviors are explained by a random utility model with a utility function shown in Eq. (4).

$$U_{s,t} = \boldsymbol{\theta}^T \boldsymbol{x_{s,t}} + \varepsilon \quad (4) \qquad V_{s,t} = \boldsymbol{\hat{\theta}_t^T} \boldsymbol{x_{s,t}} \quad (5)$$

ε is a disturbance term. The coefficient vector $\boldsymbol{\theta}$ represents the true relationship between $\boldsymbol{x_{s,t}}$ and perceived utilities of the prevailing population. The algorithm estimates $\boldsymbol{\theta}^T$, the transpose of $\boldsymbol{\theta}$, for each t by collecting recommendation responses from users. The systematic utility, $V_{s,t}$ ($= E[U_{s,t} \mid \boldsymbol{x_{s,t}}]$) in Eq. (5), is calculated by an inner product of $\boldsymbol{x_{s,t}}$ and $\boldsymbol{\hat{\theta}_t}$,

---

Algorithm 1 Routing (Insertion Heuristic)

Input: vehicle capacity q, mean speed $\bar{v}$, and initial location l_0, $|\mathbb{P}| > 0$, passenger OD pairs $\{o_n, d_n | n \in \mathbb{P}\}$

  Initialization: Identify $2|\mathbb{P}|+1$ locations should be visited with passengers randomly labeled. Build an initial route $p^1 = \{l_0, o_1, d_1\}$ and $w^1 = c_{l_0,o_1} + c_{o_1,d_1}$

  If $|\mathbb{P}| \geq 2$ **then**

    For $n = 2$ **to** $|\mathbb{P}|$ **do**

      For $i = 1$ **to** $2n - 1$ **do**

        Insert o_n to $(i+1)$th place of p^n and create p_i^n

        If $n_b \leq q$ for entire p_i^n **then**

          For $j = 1$ **to** $|p_i^n| - i$ **do**

            Insert d_n to $(i+j)$th place of p_i^n, create p_{ij}^n, and calculate route cost w_{ij}^n

          End For

        End If

      End For

      $(i^*, j^*) = \arg\min_{1 \leq i \leq 2n-1, 1 \leq j \leq i} w_{ij}^n$

      $w^n = w_{i^*,j^*}^n$

      $p^n = p_{i^*,j^*}^n$

    End For

  End If

Output: Routing cost $w^{|\mathbb{P}|}$, route sequence $p^{|\mathbb{P}|}$

Algorithm 2 RecoMOD

Input: total trials T, initial learning period τ, exploration factor α, and pool of alternatives $\mathbb{A} \supset \mathbb{A}_t, \forall t \in \{T\}$

  For $t = 1$ to T **do**

    1. Given an existing route visiting $2n_t$ points with a subset of k_t pickups already made, with remaining routing cost w_{0t} obtained from Algorithm 1, a new request for destination recommendation comes in

    For $\forall s \in \mathbb{A}_t$ **do**

      2. Construct shortest routes p_{st} and route cost w_{st} visiting $2n_t - k_t$ points and serving new request from d_t to s using Algorithm 1, and calculate $\Delta w_{s,t}$

      3. Add $\Delta w_{s,t}$ to feature vector $\boldsymbol{x_{s,t}}$

    End For

    If $t \leq \tau$ **then**

      4-1. Randomly recommend $s_t \in \mathbb{A}_t$

    Else If $t > \tau$ **then**

      4-2. $s_t = \arg\max_{s \in \mathbb{A}_t} \left(V_{s,t} + \alpha \left\| \boldsymbol{x_{s,t}} \right\|_{V_t^{-1}} \right)$ is recommended, where estimated systematic utility $V_{s,t} = \boldsymbol{\hat{\theta}_t x_{s,t}^T}$ and $\|\boldsymbol{x}\|_A := \sqrt{\boldsymbol{x}^T A \boldsymbol{x}}$

    End If

    5. Observe the user's response where $y_t = 1$ if the user accepts the recommendation and $y_t = 0$ otherwise.

    6. Add $\boldsymbol{x_{s,t}}$ and y_t to $\boldsymbol{X_t}$ and $\boldsymbol{Y_t}$

    If $t \geq \tau$ **then**

      7. Solve the Eq. (3) to obtain the maximum-likelihood estimator $\boldsymbol{\hat{\theta}_{t+1}}$

$$\sum_{i=1}^{t} \Big(\boldsymbol{Y_i} - \mu(\boldsymbol{\theta X_i}) \Big) \boldsymbol{X_i} = 0 \quad (3)$$

    End If

  End For

---

where $\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}$ is estimated from the previous trial. Given $\mathbb{A}_t$, the algorithm recommends s_t to maximize $V_{s,t}$ over T trials.

The function $\mu(\boldsymbol{X})$ represents a cumulative distribution function relating $\boldsymbol{X_t}$ to $\boldsymbol{X_t}$, and any monotonically increasing, generalized linear model is applicable. For our study, we use a binary logit model in Eq. (6) to be consistent with the random utility theory (see [4]), which assumes ε is Gumbel distributed.

$$\mu\left(\boldsymbol{\theta}^T \boldsymbol{X}\right) = \left(1 + \exp\left(-\boldsymbol{\theta}^T \boldsymbol{X}\right)\right)^{-1} \tag{6}$$

The true coefficient of the population and alternatives, $\boldsymbol{\theta}$, is substituted by the estimator $\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}$. When the algorithm improves the precision of $\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}$, it can predict the choice behavior of passengers better. The estimation is performed using maximum likelihood in Eq. (3), as shown in [52].

During the initial sampling period $t \leq \tau$, the algorithm simply selects s_t randomly from $\mathbb{A}_t$ to obtain initial data from which to estimate $\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}$. After $t = \tau$, the algorithm calculates $V_{s,t}$ and exploration term $\alpha \|\boldsymbol{x_{s,t}}\|_{V_t^{-1}}$ of all s and recommends the best one with the largest sum of both. The exploration term places value on selecting solutions that are most different from prior selected solutions by selecting those with largest L_2 norms. Multiple recommendations may be given; for our study we only investigate providing single recommendations. Once a recommendation is received, the passenger will either accept the recommended destination or not. The term y_t is updated with 1 if it is accepted and 0 if rejected.

Figure 2 illustrates an simple example with an additional context of routing constraints on a 4×4 grid. Suppose there is a shortest route for Passenger 1 initially. When Passenger 2 enters the system in an early trial and requests a destination recommendation, the system may suggest the location with the least route cost increase because the system has not explored sufficiently yet and destination contextual information is limited. Using only routing cost consideration, the system may recommend the destination of Passenger 1 to Passenger 2. If Passenger 2 refuses the suggestion, it means some features pertaining to the recommended one have a negative effect on Passenger 2's utility. The system accumulates this information, so that it can determine how to behave next time. Regarding the choice of Passenger 2, it can either stick to the recommendation of destinations with the least routing cost increase or recommend other appropriate locations.

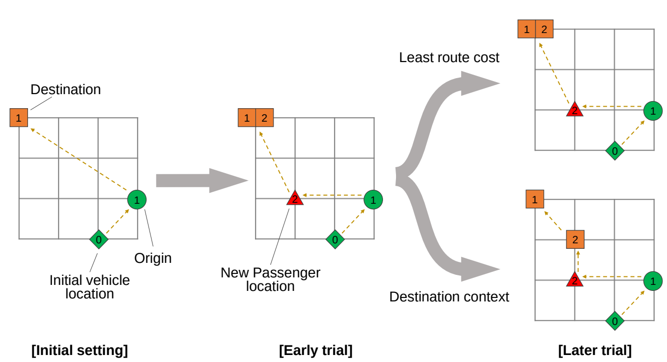


Fig. 2. Illustration of consideration of destination context.

TABLE II
USE CASES AND CORRESPONDING SYSTEM DESIGN VARIANTS

Use case	System design variant
Single MOD shuttle, mixed passengers using mobile apps	Proposed algorithm as is
Single MOD shuttle, mixed passengers reserving online	Algorithm with time windows
Taxi company dispatch, microtransit service, paratransit	Algorithm with fleet of vehicles
MaaS platform	Algorithm with stable matching for multiple operators
Tourism company with travelers reserving online	Algorithm with fleet of vehicles and time windows
Personalized car navigation system	Algorithm without user heterogeneity in parameters and single passenger
Personalized car navigation system with waypoints (e.g. to pick up or drop-off friends, tourism company with hotel pickups)	Algorithm with fixed locations in sequence considering precedence constraints
Incident management for a MOD shuttle to reroute all its passengers due to incident	Simultaneous routing of multiple passengers with recommendation as a generalized traveling salesman problem [63]

C. Variant system design and use cases

The underlying algorithm of this study can be adopted to a variety of systems with minimal modifications. Table II shows a list of use cases and corresponding system design modifications. An illustration using the incident management case recommending destinations to multiple passengers can be an example to describe how these variants may be implemented. Consider a shuttle serving 4 onboard passengers. An incident on the road initiates a query from the shuttle to its passengers for whether they would each accept an alternative destination.

Assuming that 4 passengers each consider different destination types, the system might retrieve a list of 5 alternatives per passenger as shown in Figure 3. The underlying routing

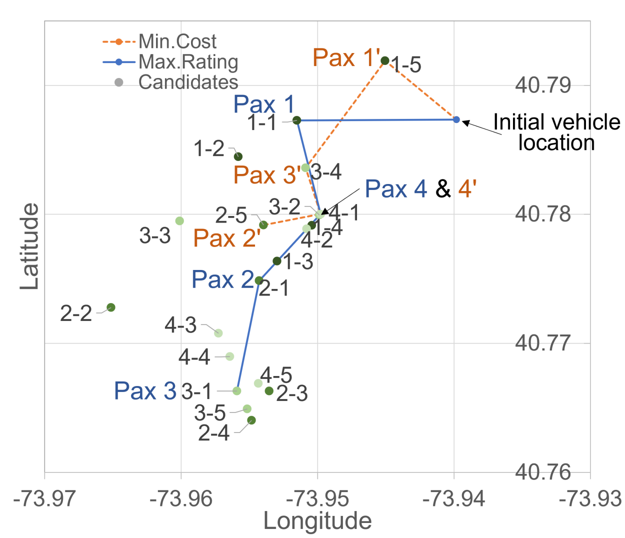


Fig. 3. Two extreme examples of routing solutions.

algorithm can be a modified insertion heuristic to address the “generalized traveling salesman problem” [63], where one candidate from each of n clusters is chosen such that the overall objective (routing cost, prize-collecting) is optimized.

We implement such a heuristic in Figure 3 to demonstrate how the solution might vary. Solutions for two extremes are shown (minimizing routing cost in blue/solid versus maximizing ratings attained in orange/dashed). According to the result, a cost-minimized route takes 14.9 minutes to drop off all passengers, while a rating-maximized route requires 20.7 minutes. Only Passenger 4 gets the same recommendation. When the system recommends the choices to the passengers, it will include prices based on the routing cost. Passengers would then have time to accept or reject, and rejected recommendations would lead to a new trial with a subset of the destinations (from those accepting) to be fixed. A recommendation system for this use case would use the results of this one trial to update the estimate of preferences to be applied to the next time an incident occurs to a shuttle of passengers.

For all the other use cases in Table II, however, the arrivals of passengers are independent from one another. The aim of the study is to understand how routing constraint parameters impact the effectiveness of the bandit algorithms. In designing the simulation experiments to explore this research question, we therefore focus on trials of independent passenger arrivals that are mixed in with other passengers that already have their destinations assigned. Insights from the experiments would be of use to systems designed for most of the cases in Table II.

D. Proposed simulation experiment design for algorithm evaluation

As illustrated above, the complexity of the learning setting is due to the routing operation which we control for by using $\Delta w_{s,t}$ as a feature. Since not all passengers will request recommendations, each new request may be made under very different routing settings. The same destination may be highly recommended for one trial but be undesirable in the next due to the context of the queue of existing passengers and their optimal route. Even the same route may feature different costs from one trial to the next because of changing traffic conditions. Adding the cost increase as a feature controls for the variability somewhat, but the degree of impact this added context puts on the learning efficiency of the system is not clear. Does the impact depend on vehicle capacity? Number of passengers queued up (demand density)? Variability in traffic congestion? We decipher these relationships with the learning mechanism by proposing a simulation experiment.

We need to design a controlled simulation-based experiment to parameterize the key routing and learning variables and estimate their relationships to the algorithm's efficiency. Figure 4 is the flow chart of the simulation using the RecoMOD algorithm with simulated factors and outputs during a single simulation of T trials. To be clear, each trial represents the event of a passenger querying the system in an independent ongoing service of passengers. From one trial to the next, the system encounters a newly simulated job queue of passengers (not all of which are using the recommender system) being served by a vehicle when a new passenger request comes in. The simulation is not of the progression of a fleet of vehicles over time because not every passenger will request a recommendation, so it is more computationally efficient to simulate only the occurrences of recommendation requests.

Initially, several simulation settings are given to the algorithm. The true distribution of $\boldsymbol{\theta}$ needs to be assumed from which true values of $\boldsymbol{\theta}_t$ are simulated for every trial t. The network should provide the spatial information for the calculation of distance, and $\bar{v}$ converts this distance to travel time. T and τ are also preset to determine the length

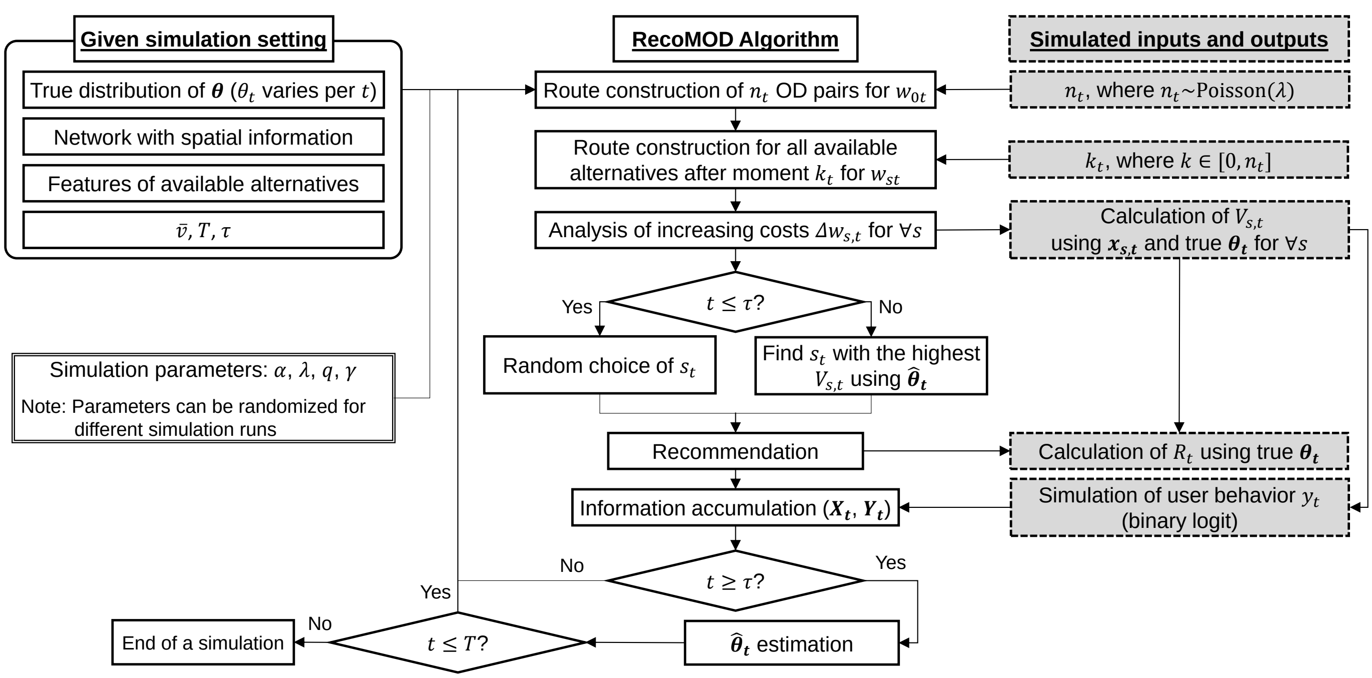


Fig. 4. Flow chart of simulation with underlying RecoMOD for one simulation scenario.

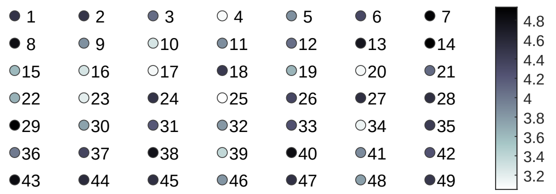


Fig. 5. Distribution of ratings.

of learning and recommendation. Other simulation parameters including exploration factor α, passenger arrival rate λ, and vehicle capacity q, which should be constant during a single simulation. We call these the scenario parameters. Since the purpose of the simulation is analyzing the influences of these parameters on the system learning performance, different simulation scenarios are generated in which these parameters are randomly drawn. The outputs of the multiple scenario simulations are then used as observations to estimate a regression model and measure the influence of the parameters.

α determines the extent of the exploration in which RecoMOD takes risks. q limits the maximum number of onboard passengers. γ_t reflects the degree of congestion when spatial distance is converted to travel time by multiplying $\eta = \gamma_t/\bar{v}$ to the distance, where $\gamma_t \geq 1$ and can vary by trial. η can also be considered as "pace" (see [4], Chapter 2), indicating a travel time per unit distance, because it is inversely proportional to $\bar{v}$. At each t, n_t is randomly drawn from a Poisson distribution with λ. The number of pickups served among n_t before receiving a new request is designated by k, which is uniformly randomly chosen between $0 \leq k \leq n_t$ because it occurs before the first passenger is dropped off.

As a controlled simulation experiment, we know the true $\boldsymbol{\theta_t}$ and can calculate true $V_{s,t}$ for all alternatives s, as well as other measures like the regret R_t and simulation of users' acceptance y_t according to Eqs. (7) and (8),

$$R_t = V_{s^*,t} - V_{s_t,t} \tag{7}$$

$$P_t = \left(1 + \exp\left(-\boldsymbol{\theta_t}^T \boldsymbol{x_{s_t,t}}\right)\right)^{-1} \tag{8}$$

where P_t is the probability that the passenger at t accepts and decides to visit recommended s_t. Users' responses are simulated assuming they behave according to a binary logit model. Consequently, the simulation reproduces how users will react to recommended alternatives, and RecoMOD yields $\hat{\boldsymbol{\theta}}_{\boldsymbol{t}}$ used in subsequent trials and further updated. The simulation code is available at https://github.com/BUILTNYU/DestinationRecoMOD.

IV. Simple network experiment - 7×7 network

The first set of experiments are conducted to verify the proposed Algorithm 2 and illustrate the sensitivities and trade-offs that can be modeled. All test data are available at https://github.com/BUILTNYU/DestinationRecoMOD. The experiment is tested on a simple network with 49 zones, the same configuration as the one for the example of insertion heuristics, using multiple simulation runs of Figure 4 with preset scenario parameters. Horizontal and vertical distances between two adjacent zones are assumed to be 1 mile. The OD locations of existing passengers in each trial are drawn from a uniform distribution. When a new passenger shows up in the trial, it calculates the increment of route travel time considering all 48 zones (minus the origin zone) as candidate destinations. These increased routing costs to each candidate zone represent one of their features.

Moreover, a rating is assigned to zones as a second feature representing the average reputation of them. We randomly generate numbers between 3 and 5 and assign them to zones as Figure 5 shows. This set of ratings is used for all the scenarios in this experiment. The structure of the feature vector is a set of two predefined features (rating, increase in vehicle route cost) plus a constant which is discussed later.

We implement three different types of constants to consider potential heterogeneities; θ_0, θ_s, and θ_{ns}. First, θ_0 is a universal constant that represents averaged and aggregated influence of other features except for ratings and route cost increases. It remains the same regardless of demographic and geographic heterogeneities under the assumption that all users share a common perception of miscellaneous features of zones. Second, θ_s, $s \in \mathbb{A}$, is an alternative specific constant (ASC) reflecting geographic differences among zones, where $E(\theta_s) = \theta_0$ as it follows a certain distribution $\theta_s \sim X_s(\theta_0, \sigma_s)$, where σ_s explains the geographic heterogeneity. Lastly, θ_{ns}, $n \in \mathbb{P}$, $s \in \mathbb{A}$, is a heterogeneous ASC (hASC) varying among not only alternatives but also individuals. The mean of θ_{ns} is θ_s and its distribution can be $\theta_{ns} \sim X_{ns}(\theta_s, \sigma_n)$

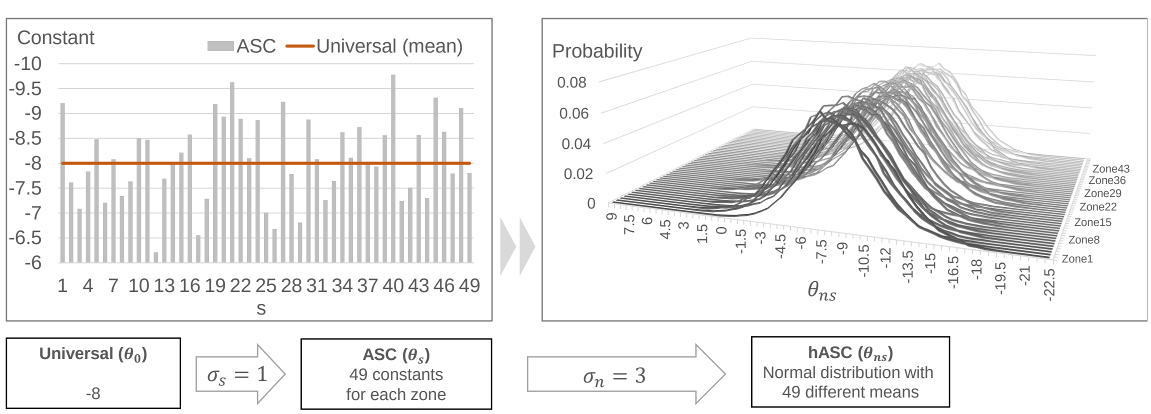
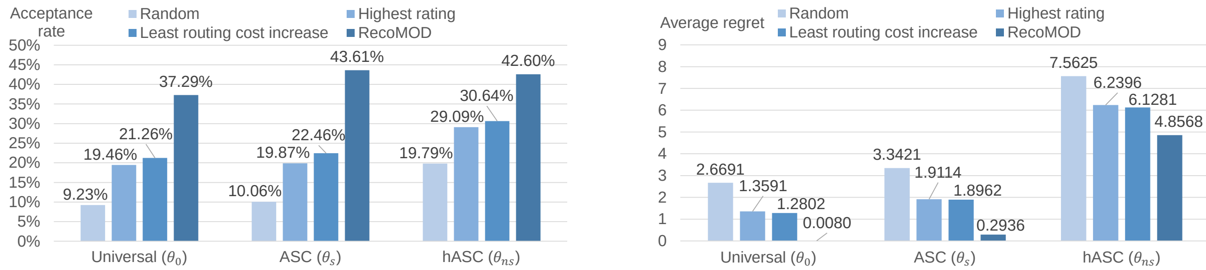
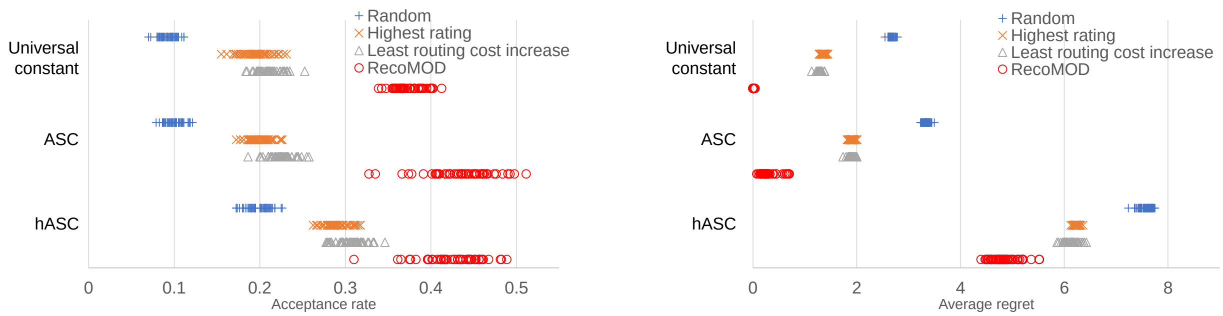


Fig. 6. Assumed distribution of the intercepts.



(a) (b)

Fig. 7. Result from different recommendation schemes. (a) Average acceptance rate and (b) average regret.



(a) (b)

Fig. 8. Distribution of (a) average acceptance rate and (b) average regret from different recommendation schemes.

with σ_n covering the diversity of personal preferences. Figure 6 visualizes modeling ASCs and hASCs. Datasets reflecting these variations are prepared and compared in this experiment.

The ground truth utility function assumed for this simulation is $V_{s,n} = \theta_{ns} - 6\Delta w_{s,n} + 2\pi_s$. For example, if a user was recommended a destination with $\Delta w_{s,n} = 0.05$ and $\pi_s = 4.5$ and they had $\theta_{ns} = -8$ they should have a true probability of 66.82% of accepting the recommendation while if it was $\Delta w_{s,n} = 0.1$ and $\pi_s = 3.5$ it would be 16.80%. This reflects a user who has a strong sensitivity for high ratings and reduced travel costs.

True values of hASCs are not identifiable unless longitudinal data from individuals are tracked. For our experiments we assume users are not tracked, i.e. each trial represents a different user. Therefore, we assume that ASCs follow a normal distribution with specified parameters. For example, with $\theta_0 = -8$ and $\sigma_s = 1$, we can draw 49 ASCs from a normal distribution, $\theta_s \sim N(\theta_0, \sigma_s)$ and assigned to every zone. Furthermore, we consider hASCs with θ_{ns} by conducting a similar approach with $\sigma_n = 3$. As a result, normal distributions with 49 different parameter sets are produced and θ_{ns} values are derived from them for each trial.

For the simulations, the following scenario parameters are set: $\alpha = 1.5$, $q = 4$, $\lambda = 1$, and $\eta = 0.1$ hr/mi. It is derived from the inverse of $\bar{v} = 10$ mph and $\gamma_t \equiv 1$ with a distance of 1 between adjacent zones, meaning it excludes congestion effects. Algorithm 2 is compared against three other benchmark policies (for a total of 4 policies):

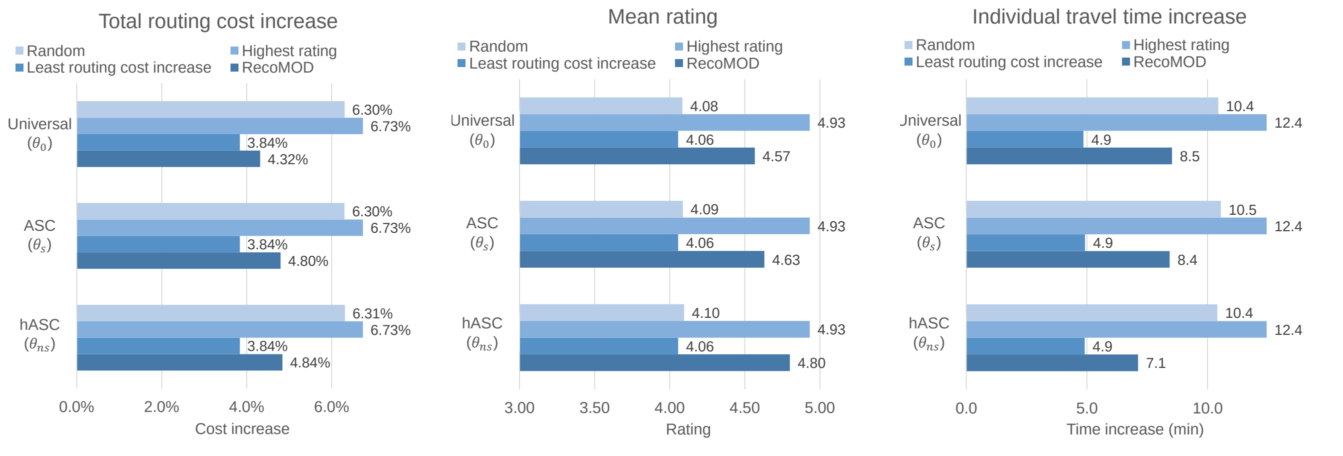
1) Picking a random zone,
2) Picking a zone with the highest rating,
3) Picking a zone with the least route cost increase.

Fifty simulations of the same scenario parameters are run with $T = 1{,}000$ including $\tau = 200$ for each combination of 3 ASC/hASC configurations and 4 recommendation policies, resulting in 12 cases. The performance measures used for comparison are (1) the acceptance rate $\phi(T)$ (Eq. (9)), which indicates the proportion of users who accept recommendations of the system after T trials, and (2) average regret $\rho(T)$ (Eq. (10)) for recommendations made after τ.

$$\phi(t) = \frac{1}{t-\tau}\sum_{i=\tau+1}^{t} y_t \quad (9) \qquad \rho(t) = \frac{1}{t-\tau}\sum_{i=\tau+1}^{t} R_t \quad (10)$$

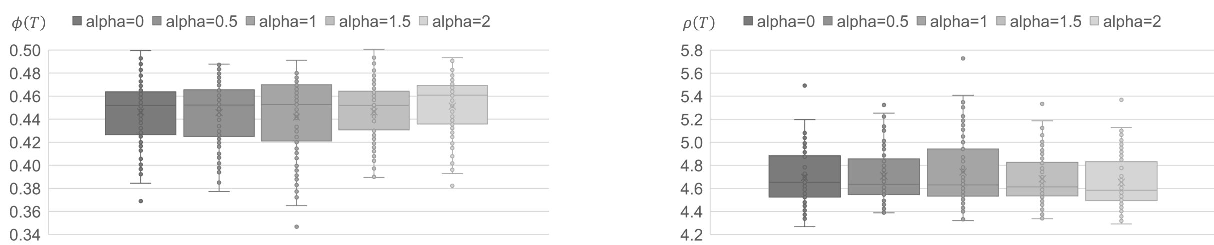
Figure 7 shows that the suggested algorithm achieves higher $\phi(T)$ compared to the other recommendation schemes in the 12 cases. In Figure 8, the $\rho(T)$ of RecoMOD is the lowest and the differences seem to depend highly on the consideration of context. This suggests we should apply the algorithm to different socioeconomic environments to better evaluate its dependence on different scenario parameters.

Among the four recommendation policies, randomly choosing an alternative for a user derives the lowest $\phi(T)$ and the highest $\rho(T)$. It serves as a lower bound threshold of $\phi(T)$ of other schemes. The policies based on rating and routing cost have similar levels of $\phi(T)$ and $\rho(T)$. Their $\phi(T)$ are 9.31∼12.41pp higher while $\rho(T)$ are 1.3100∼1.4459 lower than the random choice scheme. These three policies achieve better performance measures when the population is assumed to behave according to the use of hASCs, θ_{ns}. Their recommendations are accepted with 8.18∼10.56pp higher $\phi(T)$ despite the higher $\rho(T)$ of 4.2204∼4.8934. This implies



(a) (b) (c)

Fig. 9. Performance measure from different recommendation schemes. (a) total routing cost increase, (b) mean rating of destination, and (c) individual travel time increase.



(a) (b)

Fig. 10. Distribution of (a) average acceptance rate, and (b) average regret for different exploration factor.

that they offer acceptable alternatives to users although those recommended options are not necessarily the best ones. The RecoMOD algorithm outperforms other recommendation policies. $\phi(T)$ for this algorithm lay between 37.29% and 43.61%, 1.39∼4.04 times higher than those from the others.

While gaps between the result of the algorithm and others are significant when a case assumes θ_0 or θ_s, they become smaller when constants are disturbed by heterogeneous preferences. We observe an increase of $\phi(T)$ from 9.23% and 10.06% to 19.79% for the policy choosing random alternatives, meaning that recommendations in general work better when there is heterogeneous behavior (which makes intuitive sense).

Nevertheless, generated constants do not lead to better $\phi(T)$ for the RecoMOD, which remains at a similar level of 43.61% and 42.60% even as the base policy improves. This is likely because the algorithm is operating under the assumption that individuals are not tracked, and user-specific features are not included. As it covers the entire population, it estimates means of θ_{ns} of each zone, aggregating them to θ_s. Thus, it may not be easy to respond to users' request by estimating ASCs and customizing recommendations with the proposed algorithm. Distinguishing involved individuals and accumulating their information separately should further improve the $\phi(T)$ of the Algorithm 2 (which is also concluded by [21]).

Various recommendation schemes result in different levels of performance measures that involve total routing cost increase, mean individual travel time increase, and mean rating of recommended destinations as illustrated in Figure 9, using means of each case. In average, total routing cost increases by more than 6% when using random or highest rating recommendation. In contrast, vehicles only travel 4.32∼4.84% more under the RecoMOD scheme compared to 3.84% for the least cost scheme. The increase in individual travel cost follows the trend of total routing cost, but gaps become larger between the minimum cost increase and RecoMOD scheme. New customers are recommended to spend 2.2∼3.6 mins more with the proposed recommendation policy. Nonetheless, they are considered worth doing so due to the compensation in terms of the mean rating of recommended destinations. While random or least routing cost scheme can suggest visiting less preferred alternatives with ratings of 4.06∼4.10, RecoMOD provides options closer to the highest rating, 4.93. Although the mean drops from 4.80 to 4.57 as the level of heterogeneity rises, it significantly outperforms other schemes by 0.51∼0.74. The comparison of individual travel cost increases and mean rating among schemes indicates that customers can experience even better alternatives if they tolerate additional travel time.

Figure 10 graphically summarizes $\phi(T)$ and $\rho(T)$ of 100 simulations, respectively, for different α in box and whisker plots with $T = 2{,}000$. If $\alpha = 0$, the algorithm does not explore other options but focuses on systematic utilities of alternatives using estimated coefficients. While this can be highly efficient, there is also a risk that no exploration results in values trapped in local optima. Table III brings some statistics of performance

TABLE III
STATISTICS OF AVERAGE ACCEPTANCE RATE AND REGRET FOR DIFFERENT EXPLORATION FACTOR

	Average acceptance rate				Average regret			
α	Mean	Standard deviation	Median	p-value from t-test	Mean	Standard deviation	Median	p-value from t-test
0	44.63%	2.77%	45.19%	–	4.6954	0.2310	4.6527	–
0.5	44.57%	2.61%	45.22%	0.433	4.7068	0.2243	4.6355	0.362
1	44.19%	3.23%	45.28%	0.149	4.7421	0.2847	4.6293	0.102
1.5	44.64%	2.50%	45.19%	0.488	4.6826	0.2181	4.6132	0.343
2	45.12%	2.37%	46.08%	0.090	4.6544	0.2169	4.5838	0.099

measures and t-test results of whether differences between results of $\alpha = 0$ and others are statistically significant.

Cases with $\alpha = 2$ produce better measures than having $\alpha = 0$ in Figure 10. The t-tests proves that it is statistically significant at 0.1 level, as shown in Table III. These results imply that α should be carefully selected and that learning opportunities do exist even with untracked individuals. Overall results of this section demonstrate the improved performance of the proposed RecoMOD compared to benchmarks that lack learning. In the next section, another controlled experiment is conducted using more realistic datasets drawn from NYC to evaluate the effects of different routing constraints and other scenario parameters on RecoMOD's performance.

V. NYC SIMULATION EXPERIMENTAL DESIGN

A. Experiment objective

In the previous section, it was proven that the algorithm can improve the acceptance rate of destination recommendations for MOD systems that explicitly consider routing costs. To better understand the dependency of this acceptance rate on different routing constraints and parameters, we construct an experiment using real location ratings data and travel patterns for existing passenger OD locations to evaluate the algorithm performance under different sensitivities.

In this experiment, we set out to parameterize the simulation in Figure 4 to real data to achieve the primary experimental objective: simulate multiple scenarios with random setting parameters and routing constraints to determine the elasticity of $\phi(T)$ and $\rho(T)$ of the proposed algorithm with respect to them. Although real users have their interest in all types of destinations, without loss of generality we conduct our simulation using destination recommendations only for the restaurant category. Preferences for restaurants can vary significantly across a population, especially compared to some other destination types like hospitals and schools. New restaurants open and close permanently on a regular basis, adding to users' need for restaurant recommendation. From the physical internet search engine perspective, restaurant recommendation serves as a good initial market. Easily accessible private information services like Yelp and Google have archived reputations of those places and provide almost-objective ratings.

For the purpose of having a controlled experimental setting, we opt to assume our own ground truth utility function so that true values of $\phi(T)$ and $\rho(T)$ can be computed. The downside is that this prevents us from evaluating the elasticity of the algorithm to the coefficients. We also need to ensure that the specified utility function is within a realistic range.

The simulation consists of multiple runs to accumulate dependent and independent variables to estimate a linear regression model so that average elasticities can be quantified. For each run, pools of passenger locations and recommendable places are randomly drawn, and such indices as α, λ, η, and q are randomly generated.

Furthermore, the algorithm analyzes different combinations of passengers and restaurants for every trial to simulate the fluctuating environment. After a run, the algorithm produces the $\phi(T)$ and $\rho(T)$ throughout the recommendation period. Consequently, linear regression analysis is conducted using each run as an observation.

B. Data and simulation parameters

The borough of Manhattan, NYC, is considered as the study area. NYMTC conducted a Regional Household Travel Survey in 2010/2011 [64] that provides zone-aggregated trip data. We use the OD location distribution of trips in Manhattan departing between 4 and 7 p.m. to simulate existing passenger pickups and drop-offs, as well as pickup locations of new passengers at each trial looking for a destination recommendation. For the zones we use 29 Neighborhood Tabulation Areas (NTAs), which are generally aggregations of the traffic analysis zones with some mismatched boundaries. Accounting for the mismatched boundaries, 35 modified NTAs are created. The locations of 35,000 candidate passengers, 1,000 per NTA, were randomly placed using a "Random Points" function in QGIS (https://qgis.org/en/site/). These points are used to provide a pool of potential pickup locations which are then

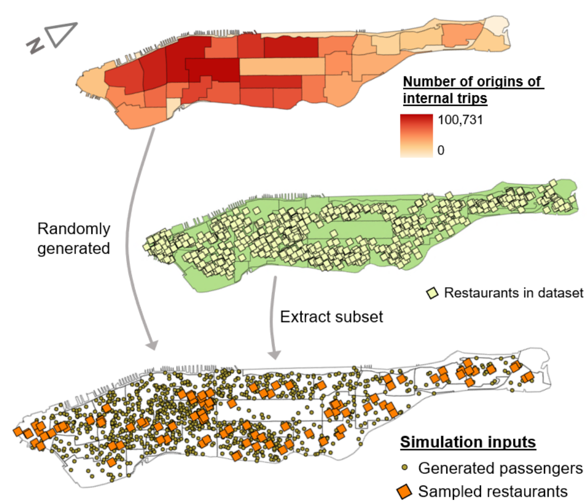


Fig. 11. Distribution of average regret for different exploration factor.

pre-generated to determine the travel times more efficiently for the randomly generated trials in each scenario.

Yelp is a widely used application that collects and distributes reputations of restaurants and other businesses. The company provides an application programming interface (API) called "Fusion API" (https://www.yelp.com/fusion) to allow the public to access their dataset in real-time, limiting the number of queries at the same time. A set of 1,012 restaurants were sampled in November 2018. Although the original dataset includes various types of information including name, coordinates, rating, zip code, number of reviews, and price level, the algorithm brings only rating and coordinates for extracting features of each alternative to prevent privacy and private property violation. Figure 11 illustrates the process of generating both random passengers and restaurant pools.

Travel time is assumed proportional to Euclidean distance. This should be adequate for this experiment because the network is dense and primarily all arterial with fairly homogeneous characteristics. The parameter γ_t is introduced to reflect the overall time of day congestion level occurring during a trial, which will vary by trial and scenario between 1 and 1.5.

The ground truth utility functions assumed for this simulation is $V_{s,n} = \theta_{ns} - 6\Delta w_{s,n} + 2\pi_s$. We generate θ_{ns} for every trial to reflect various preferences, following the methodology introduced previously. The global mean of the constant is set to -5 while the standard deviation for generating location specific constants is set to 1 and the standard deviation of θ_{ns} set to 3. Figure 12 is graphically presenting means of generated hASCs.

While the parameters α, λ, q, and η are constant for one simulation run, we vary them randomly over multiple simulation runs to observe the effects. To prevent confusion, we introduce new notation here to represent these scenario-based, randomly generated values: I_{ex} for α, I_{ar} for λ, I_{vc} for q, and I_{tt} for η. The simulation parameters for each scenario run are generated, assuming they are uniformly distributed between certain upper and lower bounds shown in Table IV.

VI. Results

A. Summary of the 200 scenario runs

We conducted 200 scenario runs of simulations. Each run consists of 100 warmup trials and 900 learning trials. Table IV describes the distributions of the indices and statistics that resulted from the 200 runs.

Table V summarizes the descriptive statistics of the dependent variables simulated from the 200 runs. Under this simulation setting, the mean of average acceptance rate is 81.05% and median is 82.22%. Mean of average regret is 4.7889 while median is 4.8602. Since the magnitudes depend

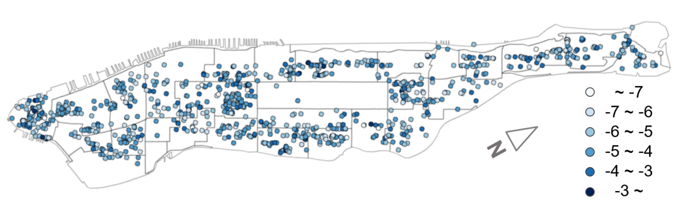


Fig. 12. Distribution of mean of hASCs.

TABLE IV
Distribution of Input Indices

Index	Assumption	Mean	Std. dev.
α (I_{ex})	$0.5k_1, k_1 \in [1, 10]$	2.953	1.326
λ (I_{ar})	$k_2, k_2 \in [1, 5]$	2.980	1.470
q (I_{vc})	$3 + k_3, k_3 \in [1, 5]$	6.015	1.343
η (I_{tt})	$0.1 + 0.05k_4, k_4 \in [0, 1]$	0.125	0.014

TABLE V
Descriptive statistics of average acceptance rate and regret

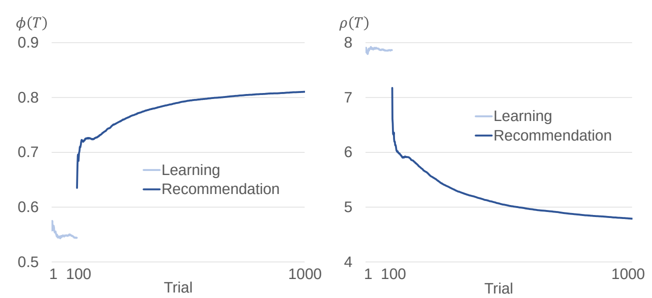
Statistic	Mean	Std. dev.	Median	Minimum	Maximum
$\phi(T)$	0.8105	0.0626	0.8222	0.5889	0.9289
$\rho(T)$	4.7889	0.5335	4.8602	3.4663	6.1317

on the underlying utility functions, the focus should not be on their values (if we specified utility functions with different coefficients it would have changed the magnitudes) but on how they vary with the input parameters for different scenarios.

Trends of $\phi(T)$ and $\rho(T)$ in Figure 13 are shown for both the learning and the recommendation period. The $\phi(T)$ of the learning period starts below 60% and hikes up to over 80% at the end of simulation. $\rho(T)$ decreases by almost 40% when the algorithm starts providing recommendations.

Correlations among the 200 simulated variable observations are provided in Table VI. The three highest correlations are observed between $\phi(T)$ and I_{ar}, $\phi(T)$ and I_{tt}, and $\phi(T)$ and $\rho(T)$ (bolded). Low correlations between independent variables confirm the lack of multicollinearity.

Figure 14 summarizes the distribution of $\bar{p}$, the average accepted recommendations made per scenario run. It indicates 809 alternatives with ratings between 4 and 5 while Table VII covers the complete distribution of locations based on their ratings and average number of choices. $\bar{p}$ is derived by dividing the sum of the number of times that an alternative is



(a) (b)

Fig. 13. Average trend of (a) average acceptance rate and (b) average regret.

TABLE VI
Correlation between variables.

	Independent variable				Dependent variable	
	I_{ex}	I_{ar}	I_{vc}	I_{tt}	$\phi(T)$	$\rho(T)$
I_{ex}	1					
I_{ar}	-0.1823	1				
I_{vc}	-0.1336	-0.0380	1			
I_{tt}	-0.0535	0.0330	0.0274	1		
$\phi(T)$	-0.0307	**0.6136**	-0.0146	$\mathbf{-0.2523}$	1	
$\rho(T)$	-0.0479	0.1468	-0.0402	0.0305	$\mathbf{-0.5478}$	1

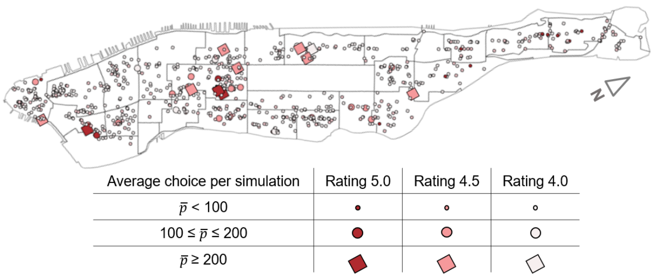


Fig. 14. Distribution of average accepted recommendations per trial.

TABLE VII
CLASSIFICATION OF ALTERNATIVES BY RATINGS AND AVERAGE NUMBER OF CHOICES

Rating	5.0	4.5	4.0	3.5	3.0	≤ 2.5	Sum
$\bar{p} = 0$	-	2	43	26	10	6	87
$0 < \bar{p} < 20$	7	125	489	130	22	8	781
$20 \leq \bar{p} < 40$	-	24	30	-	-	-	54
$40 \leq \bar{p} < 60$	3	22	12	1	-	-	38
$60 \leq \bar{p} < 80$	1	11	2	-	-	-	14
$80 \leq \bar{p} < 100$	-	2	2	-	-	-	4
$100 \leq \bar{p} < 120$	1	3	2	-	-	-	6
$120 \leq \bar{p} < 140$	-	3	-	-	-	-	3
$140 \leq \bar{p} < 160$	-	5	3	-	-	-	8
$160 \leq \bar{p} < 180$	1	3	1	-	-	-	5
$180 \leq \bar{p} < 200$	-	1	-	-	-	-	1
$\bar{p} \geq 200$	3	7	1	-	-	-	11
Sum	16	208	585	157	32	14	1,012

accepted during the simulation by the count of trials in which an alternative is included in an alternative pool. The maximum number is 626, meaning that 626 passengers decide to accept that recommended location among 900 recommendation trials, which is extraordinarily high. In addition, there is a negative correlation between $\bar{p}$ and ratings.

Some locations with the rating close to 5 are not popular to simulated passengers due to being in more secluded locations like Uptown Manhattan. In contrast, if recommended places with similar ratings are located around Midtown, their chances to being accepted become higher as they impact the vehicle routing cost less. Despite influences from randomized θ_{ns}, this result helps to connect the acceptance rate to the features.

B. Estimation of linear regression model

Table VIII summarizes the linear regression model estimated with average acceptance rate as the dependent variable while Eq. (11) specifies the resulting model. With a significance level of 0.05, the I_{ar} and I_{tt} strongly affect the $\phi(T)$. If there are more customers on the route before the new request, the new passenger is more likely to accept the recommendation as the coefficient is positive. It is likely because an existing route covers a wider region when the vehicle is serving more passengers, resulting in more flexibility to adjust the route minimally to accommodate the new passenger. Meanwhile, a negative coefficient for the I_{tt} is derived, meaning that more congested road condition damages the $\phi(T)$.

$$\phi(T) = 0.8585 + 0.0271 \times I_{ar} - 1.1778 \times I_{tt} \tag{11}$$

TABLE VIII
LINEAR REGRESSION RESULT FOR AVERAGE ACCEPTANCE RATE MODEL

Variable	β	t stat	p-value
Intercept	0.8585***	24.5659	1.33×10^{-61}
I_{ex}	0.0035	1.3736	0.1711
I_{ar}	0.0271***	11.8365	1.04×10^{-24}
I_{vc}	0.0013	0.5048	0.6143
I_{tt}	-1.1778***	-5.1038	7.88×10^{-7}

[1] Adjusted $r^2 = 0.4452$, Model F value = 40.9246
[2] *** for $\alpha < 0.01$

It is interesting that the I_{ex} and I_{vc} do not impact the $\phi(T)$ significantly at the 0.05 level. The I_{ex} likely impacts the speed with which the recommender system learns the preferences but by 1,000 trials the $\phi(T)$ is already stabilized. As for the vehicle capacity, at least for the range tested, it does not appear to impact the learning significantly. The implication is that a fleet operator for the single shuttle operation looking to develop the recommender system should focus on investing in marketing rather than focusing on trying different vehicle sizes.

A regression estimated on average regret is specified in Eq. (12) and summarized in Table IX. No index is significant at the 0.05 level and in fact the adjusted r^2 is insignificant. The result suggests the $\rho(T)$ is constant of 4.6441 under the significance level of 0.05. The low adjusted r^2 suggests that this model does not explain the relationship with regret well. This is likely because the range of regret varies significantly over trials but upon reaching 1,000 trials the level is fairly constant regardless of the parameters.

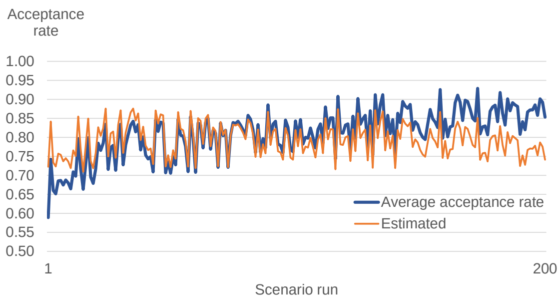
$$\rho(T) = 4.6441 + 0.0507 \times I_{ar} \tag{12}$$

TABLE IX
LINEAR REGRESSION RESULT FOR AVERAGE ACCEPTANCE RATE MODEL

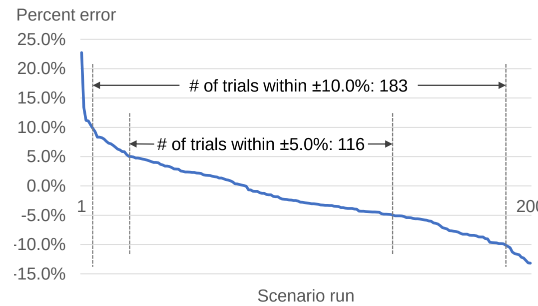
Variable	β	t stat	p-value
Intercept	4.6441***	11.6342	4.22×10^{-24}
I_{ex}	-0.0106	-0.3617	0.7180
I_{ar}	0.0507*	1.9369	0.0542
I_{vc}	-0.0155	-0.5469	0.5850
I_{tt}	0.9515	0.3610	0.7185

[1] Adjusted $r^2 = 0.0041$, Model F value = 1.2051
[2] *** for $\alpha < 0.01$, * for $\alpha < 0.01$

Figure 15 presents how the estimation errors are distributed among the different scenario runs. Figure 15a indicates differences between actual and estimated $\phi(T)$, and patterns of curves resemble each other. Figure 15b is the arrangement of percent errors in order of magnitude, ranging between 22.7% and -13.2%. If we set the tolerance to 10.0%, 183 (91.5%) trials out of 200 were accurately estimated. The number decreases to 116 (58.0%) when a stricter tolerance of 5.0% is applied.



(a)



(b)

Fig. 15. (a) Actual and estimated acceptance rate, and (b) percent error, across the 200 runs.

C. Elasticity analysis

Having estimated a good fitting model for $\phi(T)$, we now consider the elasticity of the $\phi(T)$ to different parameters. Elasticity explains the magnitude of the influence of a 1% change of one variable on the other. For linear models, the value of elasticity varies across values. We report the means for variables over the 200 points in the dataset using Eq. (13).

$$\epsilon_{xy} = \frac{\bar{x}}{\bar{y}}\theta_x \tag{13}$$

where, $\bar{x}$ and $\bar{y}$ are the mean of variable x and y, and θ_x is the coefficient of x in the model.

TABLE X
AVERAGE ELASTICITY OF AVERAGE ACCEPTANCE RATE

Significant index	Average elasticity
Mean passenger arrival rate (I_{ar})	0.1020
Pace (I_{tt})	-0.1852

[1] *** for $\alpha < 0.01$

Table X summarizes the elasticities for the statistically significant parameters. First, if the I_{ar} increases by 1%, the average $\phi(T)$ may increase by 0.1020%. The elasticity for I_{tt} is almost twice as high, which suggests that effective control of the learning efficiency of the MOD recommender system should focus on operating in less congested periods than on operating during peak demand conditions.

A MOD service considering implementing a destination recommender system faces the additional challenge of more contextual setting due to the operations within a routing environment. To mitigate this challenge, the findings of this analysis suggests that operators should consider operating the recommender system under periods of high user demand but low congestion level to provide the most efficient learning setting for the system, which also fits with intuition. For restaurant recommendations, the implication is to implement during weekday lunch periods or evenings, or weekends when there's likely more demand for MOD service and less overall commuter activity impacting the roadway congestion. The implication is also for focusing the service in neighborhoods that are more retail-oriented for restaurant destinations. Having tested this with only NYC data, we cannot generalize this conclusion to all other cities. However, these findings provide a starting point for investigating other cities.

VII. Conclusion

MOD service assists passengers to conduct their trips with better convenience by reducing access cost and interacting with them in real-time. However, service disruptions such as request cancellations or pickup/drop-off location changes and a growing population of users with limited access to information support a need for a "physical internet search engine." We introduce a destination recommender system for MOD service as a solution to reduce the unreliability and increase the efficiency. A contextual bandit algorithm is modified to incorporate routing features. A behavior model of passengers is assumed to follow a binary logit model with choices of recommendation acceptance or rejection.

Due to the uncertainty from added contextual features from routing, understanding the relationship of the recommender system's learning efficiency to different routing constraints and parameters is an important research question. To answer the question, we developed a controlled simulation and experimented with both synthetic and real data.

A test on a synthetic 7×7 grid network proves the effectiveness of the proposed algorithm based on average acceptance rate and regret compared to benchmark policies. A simulation experiment is further conducted on Manhattan, NY. The elasticity of the acceptance rate from 1,000 trials with respect to passenger arrival rate is 0.1020, while with the pace it is -0.1852. This implies that lower demand for the MOD service and slower vehicle speed because of either congestion or operational failure may harm the performance of the system by reducing the average acceptance rate. For restaurant recommendations, the implication from our findings is to implement during weekday lunch periods or evening, or weekends when there's likely more demand for MOD service and less overall commuter activity impacting the roadway congestion. The implication is also for focusing the service in neighborhoods that are more retail-oriented for restaurant destinations.

One of the main purposes of the study is validating a potential of the MOD with destination recommendation and identifying the possible relationship between system performance and route constraints. For that purpose, many aspects of recommender systems were simplified to have a more controlled experimental setting. By implementing destination recommendation in MOD services, operators can expect benefits including routing cost reduction, additional demand information learning, and balancing the vehicle supply since users are more likely to visit places favorable for operators' perspective. In contrast, because the benefit from their journey becomes more uncertain, users may be less motivated to accept the recommendations unless they are risk-taking people. Thus, operators may offer a fare reduction as an incentive since it has obvious appeal for users seeking cost-efficient alternatives. This may imply the importance of cost allocation between users' and operators' side, bargaining the proper fare level that attracts users and improves the revenue at the same time.

For future implementation, we can consider several enhancements. First, user tracking is required to predict users' behavior with higher precision. Moreover, additional features for alternatives should be discovered as decisions to accept or reject recommended options beyond just ratings and vehicle routing costs. For instance, we considered the systematic cost increase as a proxy of fare increase imposed on passengers. Using increasing journey cost for individual passengers may be more straightforward to model their behavior. In addition, the system relies on the static reputation dataset of points of interests which are not updated in real-time, and it can be more realistic to employ a dynamic system. These will be investigated in future research.

References

[1] P. Viechnicki, A. Khuperkar, T. Fishman, and W. Eggers, "Smart mobility: Reducing congestion and fostering faster, greener, and cheaper transportation options," *Deloitte Smart Mobility Research Report*, 2015, last accessed on May 11, 2019.

[2] L. Greer, J. Fraser, D. Hicks, M. Mercer, and K. Thompson, "Intelligent transportation systems benefits, costs, and lessons learned: 2018 update report." United States Department of Transportation, 2018.

[3] G. Smith and D. A. Hensher, "Towards a framework for mobility-as-a-service policies," *Transport Policy*, 2020.

[4] J. Chow, *Informed Urban Transport Systems: Classic and Emerging Mobility Methods toward Smart Cities*. Elsevier, 2018.

[5] FixNYC Advisory Panel, "Advisory panel report," 2018.

[6] H. R. Sayarshad and J. Y. Chow, "A scalable non-myopic dynamic dial-a-ride and pricing problem," *Transportation Research Part B: Methodological*, vol. 81, pp. 539–554, 2015.

[7] B. Atasoy, T. Ikeda, X. Song, and M. E. Ben-Akiva, "The concept and impact analysis of a flexible mobility on demand system," *Transportation Research Part C: Emerging Technologies*, vol. 56, pp. 373–392, 2015.

[8] C. Yan, H. Zhu, N. Korolko, and D. Woodard, "Dynamic pricing and matching in ride-hailing platforms," *Naval Research Logistics (NRL)*, 2019.

[9] L. Fulton, J. Mason, and D. Meroux, "Three revolutions in urban transportation: How to achieve the full potential of vehicle electrification, automation, and shared mobility in urban transportation systems around the world by 2050," Tech. Rep., 2017.

[10] F. Siddiqui, "Uber reports a $1 billion loss in first quarterly earnings after IPO," Washington Post, 2019, May 30, 2019. [Online]. Available: https://www.washingtonpost.com/technology/2019/05/30/uber-reports-billion-loss-first-quarterly-earnings-after-ipo/?utm_term=.2fd44845339f

[11] R. Goldensohn, "Access-a-ride wastes up to $100 million a year, experts say," Crain's New York Business, September 20 2016. [Online]. Available: https://www.crainsnewyork.com/article/20160920/BLOGS04/160919867/access-a-ride-wastes-up-to-100-million-a-year-experts-say

[12] J. A. Goodwill and H. Carapella, "Creative ways to manage paratransit costs: final report, July 2008 [summary]," National Center for Transit Research (US), Tech. Rep., 2008.

[13] New York City Transit, "2017 Paratransit customer satisfaction study: Access-A-Ride," 2018. [Online]. Available: http://web.mta.info/nyct/paratran/pdf/RPT292%20Access-A-Ride%20Report%202017.pdf

[14] M. Vechione, C. Marrufo, R. A. Vargas-Acosta, M. G. Jimenez-Velasco, O. Gurbuz, A. Dmitriyeva, R. L. Cheu, N. Villanueva-Rosales, G. G. Nunez-Mchiri, and J. Y. Chow, "Smart mobility for seniors: challenges and solutions in El Paso, TX, and New York, NY," in *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, 2018, pp. 1–8.

[15] A. Berenguer, J. Goncalves, S. Hosio, D. Ferreira, T. Anagnostopoulos, and V. Kostakos, "Are smartphones ubiquitous?: An in-depth survey of smartphone adoption by seniors," *IEEE Consumer Electronics Magazine*, vol. 6, no. 1, pp. 104–110, 2016.

[16] J. Y. Chow, A. Dmitriyeva, and D. Fay, "City-scalable destination recommender system for on-demand senior mobility," *C2SMART Final Report*, 2018.

[17] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge & Data Engineering*, no. 6, pp. 734–749, 2005.

[18] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Ghost riders: Sybil attacks on crowdsourced mobile mapping services," *IEEE/ACM transactions on networking*, vol. 26, no. 3, pp. 1123–1136, 2018.

[19] T. W. Schneider, "Taxi and ridehailing usage in new york city," 2020, last accessed on May 3, 2020. [Online]. Available: https://toddwschneider.com/dashboards/nyc-taxi-ridehailing-uber-lyft-data/

[20] New York Metropolitan Transportation Council and North Jersey Transportation Planning Authority, "2010/2011 regional household travel survey: Final report," Tech. Rep., 2014.

[21] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 661–670.

[22] J. Zhou, X. Lai, and J. Y. J. Chow, "Multi-armed bandit on-time arrival algorithms for sequential reliable route selection under uncertainty," *Transportation Research Record*, 2019.

[23] W. B. Powell and I. O. Ryzhov, *Optimal learning*. John Wiley & Sons, 2012, vol. 841.

[24] B. Montreuil, "Toward a physical internet: meeting the global logistics sustainability grand challenge," *Logistics Research*, vol. 3, no. 2-3, pp. 71–87, 2011.

[25] W. Chu and S. T. Park, "Personalized recommendation on dynamic content using predictive bilinear models," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 691–700.

[26] G. David, "Burdens abound, but nyc restaurants' numbers are growing," Crain's New York Business, April 12 2018. [Online]. Available: https://www.crainsnewyork.com/article/20180412/BLOGS01/180419940/burdens-abound-but-nyc-restaurants-numbers-are-growing

[27] F. Rehman, O. Khalid, and S. A. Madani, "A comparative study of location-based recommendation systems," *The Knowledge Engineering Review*, vol. 32, 2017.

[28] S. Ebrahimi, F. Sharmeen, and H. Meurs, "Innovative business architectures (bas) for mobility as a service (maas)-exploration, assessment, and categorization using operational maas cases," in *97th annual meeting of transportation research board. Washington DC*, 2018.

[29] K. R. Harrigan, "Vertical integration and corporate strategy," *Academy of Management journal*, vol. 28, no. 2, pp. 397–425, 1985.

[30] R. R. Clewlow and G. S. Mishra, "Disruptive transportation: The adoption, utilization, and impacts of ride-hailing in the united states," 2017.

[31] Y. Babar and G. Burtch, "Examining the heterogeneous impact of ride-hailing services on public transit use," *Information Systems Research*, 2020.

[32] A. Tirachini and A. Gomez-Lobo, "Does ride-hailing increase or decrease vehicle kilometers traveled (vkt)? a simulation approach for santiago de chile," *International journal of sustainable transportation*, vol. 14, no. 3, pp. 187–204, 2020.

[33] X. Wang, F. He, H. Yang, and H. O. Gao, "Pricing strategies for a taxi-hailing platform," *Transportation Research Part E: Logistics and Transportation Review*, vol. 93, pp. 212–231, 2016.

[34] S. M. Zoepf, S. Chen, P. Adu, and G. Pozo, "The economics of ride-hailing: driver revenue, expenses, and taxes," *CEEPR WP*, vol. 5, 2018.

[35] N. Korolko, D. Woodard, C. Yan, and H. Zhu, "Dynamic pricing and matching in ride-hailing platforms," *Available at SSRN*, 2018.

[36] H. A. Chaudhari, J. W. Byers, and E. Terzi, "Putting data in the driver's seat: Optimizing earnings for on-demand ride-hailing," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 2018, pp. 90–98.
[37] L. Zha, Y. Yin, and Z. Xu, "Geometric matching and spatial pricing in ride-sourcing markets," *Transportation Research Part C: Emerging Technologies*, vol. 92, pp. 58–75, 2018.
[38] M. Nourinejad and M. Ramezani, "Ride-sourcing modeling and pricing in non-equilibrium two-sided markets," *Transportation Research Part B: Methodological*, 2019.
[39] J. Y. Chow and H. Liu, "Generalized profitable tour problems for online activity routing system," *Transportation Research Record*, vol. 2284, no. 1, pp. 1–9, 2012.
[40] S. Samaranayake, K. Spieser, H. Guntha, and E. Frazzoli, "Ridepooling with trip-chaining in a shared-vehicle mobility-on-demand system," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–7.
[41] X. Wan, H. Ghazzai, and Y. Massoud, "A generic data-driven recommendation system for large-scale regular and ride-hailing taxi services," *Electronics*, vol. 9, no. 4, p. 648, 2020.
[42] R. Bellman and P. Brock, "On the concepts of a problem and problem-solving," *The American Mathematical Monthly*, vol. 67, no. 2, pp. 119–134, 1960.
[43] P. Whittle, "Sequential decision processes with essential unobservables," *Advances in Applied Probability*, vol. 1, no. 2, pp. 271–287, 1969.
[44] J. C. Gittins, "Bandit processes and dynamic allocation indices," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 41, no. 2, pp. 148–164, 1979.
[45] D. A. Berry and B. Fristedt, *Bandit problems: sequential allocation of experiments (Monographs on statistics and applied probability)*. Springer, 1985.
[46] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.
[47] J. Vermorel and M. Mohri, "Multi-armed bandit algorithms and empirical evaluation," in *European conference on machine learning*. Springer, 2005, pp. 437–448.
[48] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Foundations and Trends® in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.
[49] J. Langford and T. Zhang, "The epoch-greedy algorithm for contextual multi-armed bandits," in *Proceedings of the 20th International Conference on Neural Information Processing Systems*. Citeseer, 2007, pp. 817–824.
[50] T. Lu, D. Pál, and M. Pál, "Contextual multi-armed bandits," in *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, 2010, pp. 485–492.
[51] M. Dimakopoulou, S. Athey, and G. Imbens, "Estimation considerations in contextual bandits," *arXiv preprint arXiv:1711.07077*, 2017.
[52] L. Li, Y. Lu, and D. Zhou, "Provably optimal algorithms for generalized linear contextual bandits," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 2071–2080.
[53] M. Ye, P. Yin, and W. C. Lee, "Location recommendation for location-based social networks," in *Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems*. ACM, 2010, pp. 458–461.
[54] B. Berjani and T. Strufe, "A recommendation system for spots in location-based online social networks," in *Proceedings of the 4th Workshop on Social Network Systems*, ser. SNS '11. New York, NY, USA: ACM, 2011, pp. 4:1–4:6.
[55] J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel, "Lars: A location-aware recommender system," in *2012 IEEE 28th international conference on data engineering*. IEEE, 2012, pp. 450–461.
[56] H. Wang, M. Terrovitis, and N. Mamoulis, "Location recommendation in location-based social networks using user check-in data," in *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2013, pp. 374–383.
[57] N. Gutowski, T. Amghar, O. Camp, and S. Hammoudi, "A framework for context-aware service recommendation for mobile users: A focus on mobility in smart cities," *From Data To Decision*, 2017. [Online]. Available: https://www.openscience.fr/IMG/pdf/iste_fromd2dv1n1_1.pdf
[58] C. Römer, J. Hiry, C. Kittl, T. Liebig, and C. Rehtanz, "Charging control of electric vehicles using contextual bandits considering the electrical distribution grid," *arXiv preprint arXiv:1905.01163*, 2019.
[59] X. Song, "Personalization of future urban mobility," Ph.D. dissertation, Massachusetts Institute of Technology, 2018.
[60] R. Zhu and E. Modiano, "Learning to route efficiently with end-to-end feedback: The value of networked structure," *arXiv preprint arXiv:1810.10637*, 2018.
[61] G. Berbeglia, J.-F. Cordeau, and G. Laporte, "Dynamic pickup and delivery problems," *European journal of operational research*, vol. 202, no. 1, pp. 8–15, 2010.
[62] J. Y. Chow and H. R. Sayarshad, "Reference policies for non-myopic sequential network design and timing problems," *Networks and Spatial Economics*, vol. 16, no. 4, pp. 1183–1209, 2016.
[63] M. Fischetti, J. J. Salazar González, and P. Toth, "A branch-and-cut algorithm for the symmetric generalized traveling salesman problem," *Operations Research*, vol. 45, no. 3, pp. 378–394, 1997.
[64] New York Metropolitan Transportation Council, "2010/2011 regional household travel survey," 2011, last accessed on June 1, 2017. [Online]. Available: https://www.nymtc.org/Portals/0/Pdf/SED/Excel.zip?ver=2016-05-26-130138-000

Gyugeun Yoon is a Ph.D. student in the Department of Civil & Urban Engineering at New York University Tandon School of Engineering. He is also one of current researcher at BUILT@NYU Lab. His research interests involve Mobility-on-Demand, transportation service planning, and others. He obtained his BS and MS at Seoul National University in 2013 and 2015.

Joseph Chow is an Assistant Professor in the Department of Civil & Urban Engineering at New York University Tandon School of Engineering, and Deputy Director of the C2SMART University Transportation Center. His research interests lie in emerging mobility in urban public transportation systems, particularly with Mobility-as-a-Service. He obtained his Ph.D. at UC Irvine in 2010, and a BS and MEng at Cornell University in 2000 and 2001.

Assel Dmitriyeva is a recent graduate of Interactive Telecommunications Program at NYU. As a young researcher in transportation modeling and human-computer interaction, she strives for interdisciplinary collaboration. Assel received a bachelor's degree in Mathematical and Computer Modeling from Kazakh-British Technical University.

Daniel Fay was with Center for Urban Science and Progress, New York University, Brooklyn, NY. He is now a technical solutions professional at Microsoft.