

Model-Based Vehicle-Miles Traveled and Emission Evaluation of On-Demand Food Delivery Considering the Impact of COVID-19 Pandemic

Haishan Liu¹, Peng Hao¹, Yejia Liao¹ , Kanok Boriboonsomsin¹ , and Matthew Barth¹ 

Abstract

The market for on-demand food delivery (ODFD) has increased considerably, especially during the COVID-19 pandemic. It is crucial for transportation and environmental agencies to understand how ODFD has reshaped the travel patterns of people, affecting vehicle-miles traveled (VMT) as well as pollutant emissions in the transportation system. However, the lack of public data from food delivery companies makes it challenging to quantify the impact of on-demand delivery on the real-world transportation network. In this research, we propose a comprehensive framework to quantify the VMT and emissions incurred by ODFD with three main components: (i) a daily activity generation tool, Comprehensive Econometric Micro-simulator for Daily Activity-travel Patterns, to create a simulation scenario of ODFD behaviors based on a real-world roadway network and population demographics in the City of Riverside, California; (ii) an efficient order dispatching and routing algorithm, adaptive large neighborhood search, to obtain a high quality order dispatching and routing plan; (iii) an emission evaluation model, emission factor (EMFAC), to evaluate pollutant emissions from all dining-related trips. Both short-term and long-term impacts of the COVID-19 pandemic are evaluated. Experimental results show that ODFD has great potential to reduce the dining-related VMT and emissions. The total dining-related VMT in the during-pandemic case decreased by 38% and in the after-pandemic case reduced by 6% to 9%, and the corresponding environmental impacts were reduced accordingly. Meanwhile, emissions reduced significantly with more electric vehicles involved in food delivery. With 100% electric delivery fleet, the ODFD service can save 14% to 22% of emissions after the COVID-19 pandemic.

Keywords

on-demand food delivery, COVID-19 Pandemic, pollutant emission, adaptive large Neighborhood Search(ALNS), impact evaluation

The COVID-19 pandemic has reshaped people's activity patterns, not only in the way they work, but also the way they shop, eat, and play. According to surveys, most customers switched to online shopping and delivery services because of government requirements of social distancing and personal fears of transmission of the virus (1, 2). With regard to the online ordering of cooked food, on-demand food delivery (ODFD) platforms have recently gained prevalence around the world because they benefit both consumers and restaurants by providing contactless, efficient, and convenient online food ordering and offline food delivery services (3). Especially when self-quarantine and social distancing were required during the COVID-19

pandemic (4), this service met both the exact needs of restaurant owners for survival of their business and consumer demand for convenient food supplies without generating any personal hygiene concerns. According to Statista, by 2021 ODFD comprised 14% of the total restaurant market in the U.S.A. and it is expected to keep growing steadily in the next few years (5).

¹Center for Environmental Research & Technology, University of California, Riverside, CA

Corresponding Author:
Haishan Liu, hliu240@ucr.edu

Transportation Research Record

1–14

© National Academy of Sciences:
Transportation Research Board 2023



Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/03611981231169276
journals.sagepub.com/home/trr



Most recent work on ODFD focuses on optimizing the service quality and developing faster and more efficient algorithms. Liu et al. (6) presented a Food Delivery Network (FooDNet) in which taxi drivers play both opportunistic and dedicated shared-rider roles in ODFD. Tu et al. (7) built a dynamic optimization framework to process order collection, solution generation, and sequential delivery decisions in each time window. Zheng et al. (8) proposed a two-stage fast heuristic to solve the food delivery routing problem, leveraging geographic information to speed up the insertion process. From the algorithm perspective, there are three main approaches: (i) using an exact algorithm, (ii) using local search heuristics, and (iii) using a large neighborhood search. Exact algorithms aim to obtain the global optimal solution but are often limited by the size of the problem, and in some cases even fail to find a feasible solution (9, 10). Local search heuristics only make small modifications in each iteration, which may get stuck in a local minima though some of them are integrated with a Tabu search (11) and simulated annealing framework (12). Large neighborhood search (LNS) is designed to solve large-scale scenarios by making a large amount of change in each iteration to diversify the solution space (13). Ropke and Pisinger (14) combined simulated annealing with LNS and introduced a roulette wheel selection mechanism to select various destroy and repair operators. Their approach, named adaptive large neighborhood search (ALNS), shows great capability in solving the “pick-up and delivery problem with time window” (PDPTW) problem, having been applied in Liu et al. (6), Tu et al. (7), and Emeç et al. (15).

With the unexpected and fast outbreak of the COVID-19 pandemic, most research did not quantify the impact of the pandemic on ODFD demand. There is great uncertainty about the extent to which this fast-growing food delivery service will affect the transportation system and urban environment. On the other hand, little attention has been paid to investigating the environmental challenge and opportunity from the delivery operation perspective. In the literature, the objectives and evaluation metrics of most research to date are minimization of travel distance (15), order delay (7), and number of vehicles (6). To the authors' best knowledge, no existing research evaluates the environmental impact of ODFD on a large-scale traffic network. In this paper, to fill the gaps, we thus aim to study the effect on vehicle-miles traveled (VMT) and pollutant emissions of ODFD in the context of the COVID-19 pandemic. (In the rest of this paper, we refer to the COVID-19 pandemic which was declared by the World Health Organization early in 2020 as “COVID” for short.)

A comprehensive framework is developed to evaluate the ODFD service from both operational and sustainable perspectives considering the short-term and long-

term impact of COVID and it is validated by numerical simulation. As shown in Figure 1, a CEMDAP (Comprehensive Econometric Micro-simulator for Daily Activity-travel Patterns) model, integrating inputs from multiple sources, is utilized to generate high-resolution behaviors and activities. CEMDAP is a micro-simulation implementation of a continuous-time activity-travel modeling system. Given as inputs various socio-demographic data from SynthPoP (16), land use, and transportation level-of-service attributes from Southern California Association of Governments (17), the system provides as output the complete daily activity-travel patterns for all individuals of a population (18). For details about the CEMDAP model, interested readers should refer to Bhat et al. (18) and Hao et al. (19). In this research, we mainly focus on dining-out activities. We take reference from existing research and survey results on the impact of COVID on dine-out choices and extract the food delivery demand based on which ODFD orders can be obtained (5). Driver information is generated according to the order amount and traffic information is extracted from BEAM (20) to execute driver routing with the real traffic network. The ODFD order dispatching and routing system is based on the ALNS algorithm to solve a large-scale optimization problem efficiently. With the driver routing plan, we can gain the total VMT and each order visiting plan. Finally, the driver planned routes will be used as input of the emission factor (EMFAC) model to exactly quantify the pollutant emissions of delivery drivers. The rest of this paper is organized as follows. In the next section, we will first introduce the ODFD system and formulate a mathematical model to present the food delivery problem. The third section presents the ALNS algorithm in detail. The fourth and fifth sections will show the scenario setup considering COVID and the numerical results analysis with both operational and environmental evaluation metrics, followed by the concluding remarks.

Problem Description and Model Formulation

In this section, we present the ODFD problem description and the mathematical modeling. There are four main stakeholders in the ODFD system: customers, restaurants, delivery drivers, and online platform. Each customer places a food order individually. Delivery drivers are a group of local and non-professional drivers who are willing to work temporarily to deliver food orders using their private vehicles. The drivers will log-on to the ODFD platform and provide their availability for work. The ODFD platform will gather information of new orders and active drivers. The goal of ODFD is to dispatch orders to currently available drivers and provide a

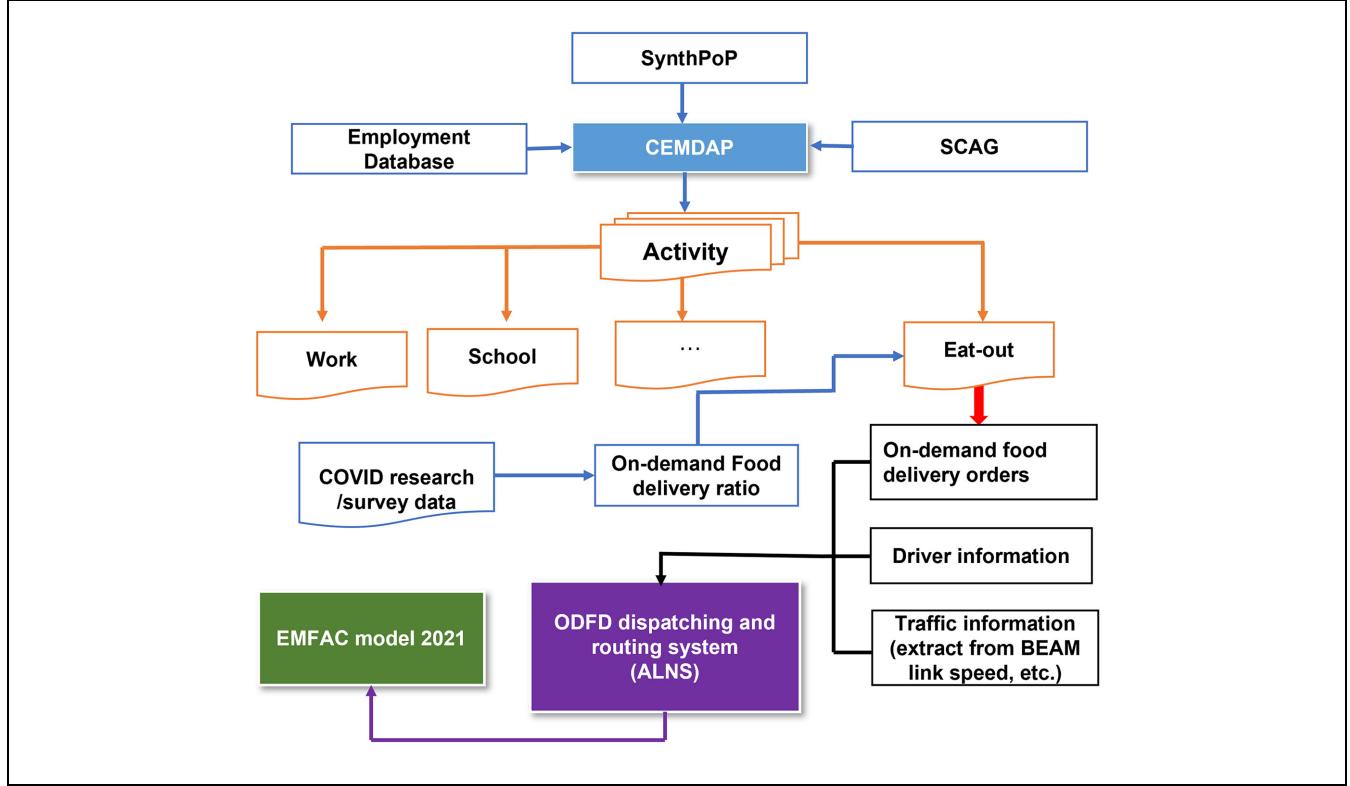


Figure 1. Framework for evaluating vehicle-miles traveled (VMT) and emission impact of on-demand food delivery (ODFD).

routing plan simultaneously considering some specific constraints and objectives.

The general ODFD process is illustrated as follows, and the key time stamps associated with one order are shown in Figure 2. In theory, a customer first places one order i at time t_p^i . After receiving the order request, the delivery platform confirms the order information with the corresponding food provider and receives a time t_r^i indicating the time the food will be ready. Meanwhile, the system will provide an expected drop-off time t_{do}^{ie} to the customer considering the food preparation time, customer experience, peak hour demand, and traffic condition. Based on a collection of food orders and available drivers, the platform needs to assign each order to the best driver with regard to driver's location, availability, capacity, and so forth. After finding the dispatching solution, the assigned driver will receive the delivery tasks and will pick up order i from the restaurant at time t_{pu}^i and drop off the food at time t_{do}^i . Note that the driver can arrive at the restaurant earlier than the order ready time but should wait until t_r^i to pick up the order. With all the time stamp information, we can calculate the following information for each order (in Figure 2):

1. Food preparation time T_{pre} : the time difference between order place time and order ready time.

2. Order click-to-door time T_{ctd} : the time difference between order place time and drop-off time.
3. Order expected click-to-door time T_{ctd}^e : the time difference between order place time and the system-offered expected drop-off time, which represents a delivery time commitment.
4. Order ready-to-door time T_{rtd} : the time difference between order ready time and drop-off time, which can indicate the food freshness.
5. Order delivery delay: the time difference between the expected drop-off time and the actual drop-off time, calculated as $\max(0, t_{do}^i - t_{do}^{ie})$.

Figure 3 illustrates an example of ODFD service with six orders. Each order contains the following information: restaurant location (R_i), customer location (C_i), order place time t_p^i , ready time t_r^i , and expected drop-off time t_{do}^{ie} . Without the ODFD service, in this paper we assume that each customer would have a roundtrip to visit the restaurant individually and then return to their origin location (shown in the left part of Figure 3). With ODFD, three routes are generated for drivers $k1$, $k2$, and $k3$ to finish the assigned pick-up and drop-off tasks sequentially (shown in the right part of Figure 3). In this paper, we use restaurant/customer and pick-up/drop-off tasks interchangeably. Note that delivery drivers are

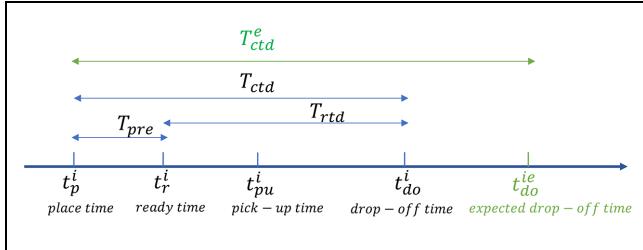


Figure 2. Time record for each order in the on-demand food delivery (ODFD) system.

“shared” among multiple restaurants and this flexibility can offer extra opportunities to minimize delivery cost and improve system efficiency.

With the above description, in general, the ODFD problem can be formulated as a pick-up and delivery problem with time window (PDPTW) model. Assuming that there are n orders and m drivers in the system, then an undirected graph $G = (V, E)$ can be defined. Node set V can represent the locations of a customer, a restaurant, or a driver. Each arc ($E = V \times V$) represents the movement from one node to another. All parameters and variables are listed in Table 1. Specifically, order i can be defined as $\langle i, i + n, q_i, q_{i+n}, t_p^i, t_r^i, t_{do}^i, t_{doe}^i \rangle$, where i represents the pick-up node, $i + n$ represents the drop-off node, q_i and q_{i+n} are the quantity of orders at the corresponding node. The last

three components are the key time stamps defined above. Then we can formulate the ODFD problem as follows.

The objective of this problem is to minimize the total travel distance of delivery drivers and total order delay cost. The order delay is defined as the difference between actual drop-off time t_{do}^i and predefined expected drop-off time t_{doe}^i . α and β are weight factors designed to balance the distance and delay. Constraint 1 ensures that any customer or restaurant will be visited exactly once and all orders in the system will be serviced. Constraints 2 and 3 define the driver’s first stop and last stop of the trip. Constraint 4 guarantees the flow conservation of the route. Constraint 5 ensures that each order should be picked up and delivered by the same driver. The driver capacity change along the path and its limit is defined in Constraints 6 and 7. Constraint 8 states that driver’s arrival time at node j is no less than arrival time of the previous point i plus travel time from node i to j and service time at node i . Constraints 9 and 10 allow each driver to pick-up the order no earlier than ready time, and then deliver the order to the corresponding customer’s location. Constraint 11 states the driver’s order number is no less than zero. Decision variables are defined in Constraints 12 and 13.

$$\min F = \alpha \sum_{k \in K} \sum_{(i,j) \in E} d_{ij} x_{ij}^k + \beta \sum_{i \in D} \max(0, t_{do}^i - t_{doe}^i)$$

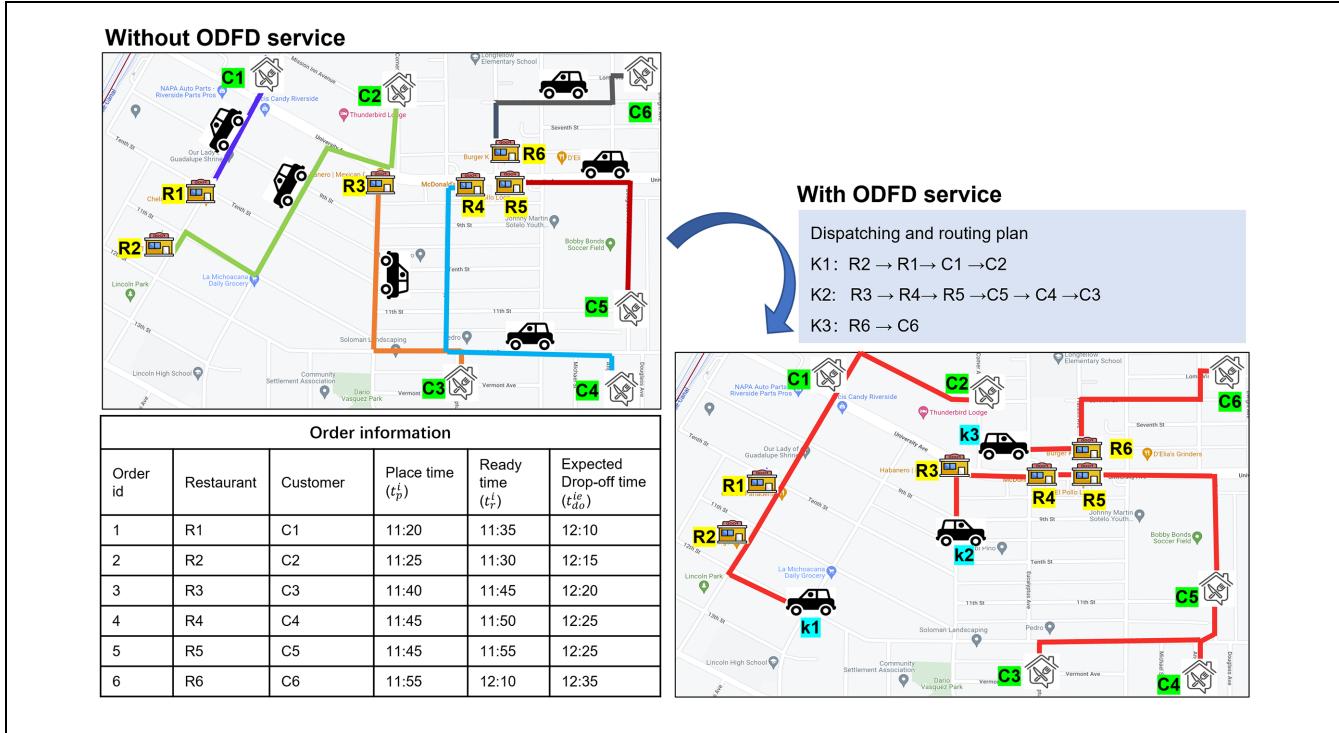


Figure 3. Illustration of on-demand food delivery with six orders, compared with customers traveling to restaurants (left side).

Table I. Definition of Variables

Type	Variable	Description
System input	α, β n m P D K q_i	Weight factor to balance distance and delay Total number of orders Total number of delivery drivers Set of restaurants. Pick-up point $\{1, \dots, n\}$ Set of customers. Delivery point $\{n+1, \dots, 2n\}$ Set of drivers. Initial location $\{2n+1, \dots, 2n+m\}$ Orders needed to be transported at node i . Positive when i is a pick-up node, negative when i is a drop-off node.
	Q^k s_i t_p^i t_r^i t_{de}^i t_{do}^i t_{ij}^k d_{ij} Q_i^k t_{pu}^i t_{do}^i x_{ij}^k T_i^k	The maximum number of orders that can be accepted by driver k , $\forall k \in K$ Service time at node i (pick-up/drop-off) The food order place time, $i \in D$ The food order ready time, $i \in D$ The expected drop-off time, $i \in D$ Travel time of link ij with driver k Link length of link ij Number of orders when driver k leaves node i Order pick-up time, same as T_i^k , when $i \in P$ Order drop-off time, same as T_i^k , when $i \in D$ 1 if driver k use link(i, j), otherwise 0 Time when driver k arrives at node i
Intermediate variable		
Decision variable		

subject to

$$\sum_{k \in K} \sum_{j \in V} x_{ij}^k = 1 \quad \forall i \in P \cup D \quad (1)$$

$$\sum_{j \in P} x_{2n+k, j}^k = 1 \quad \forall k \in K \quad (2)$$

$$\sum_{i \in D} x_{i, 2n+k}^k = 1 \quad \forall k \in K \quad (3)$$

$$\sum_{i \in V} x_{ij}^k - \sum_{i \in V} x_{ji}^k = 0 \quad \forall j \in P \cup D, \forall k \in K \quad (4)$$

$$\sum_{j \in V} x_{ij}^k - \sum_{j \in V} x_{j, n+i}^k = 0 \quad \forall i \in P, \forall k \in K \quad (5)$$

$$x_{ij}^k = 1 \Rightarrow Q_i^k \geq Q_i^k + q_j \quad \forall (i, j) \in E, \forall k \in K \quad (6)$$

$$Q_i^k \leq Q^k \quad \forall i \in V, \forall k \in K \quad (7)$$

$$x_{ij}^k = 1 \Rightarrow T_j^k \geq T_i^k + t_{ij} + s_i \quad \forall (i, j) \in E, \forall k \in K \quad (8)$$

$$T_i^k \geq t_r^i \quad \forall i \in P \quad (9)$$

$$T_i^k \leq T_{n+i}^k \quad \forall i \in P, \forall k \in K \quad (10)$$

$$Q_i^k \geq 0 \quad \forall i \in V, \forall k \in K \quad (11)$$

$$x_{ij}^k \in \{0, 1\} \quad \forall (i, j) \in E, \forall k \in K \quad (12)$$

$$T_i^k \geq 0 \quad \forall i \in V, \forall k \in K \quad (13)$$

The solution of the ODFD problem is a set of task sequences assigned to multiple drivers. In this problem, the expected drop-off time is defined as the regular expected time of arrival of each order assuming regular traffic and delivery demand level, representing the platform's commitment as well. Under peak hour conditions, the congested traffic and high demand may cause

unavoidable additional delay to the orders. To accommodate this condition, we set time window as a soft constraint in this problem rather than a hard constraint in the general PDPTW model (14). If the actual drop-off time is later than the expected one, this route is still feasible but a penalty will be recorded in the objective function.

Order Dispatching and Routing Algorithm

The ODFD problem is a variant of vehicle routing problem (VRP) which is a famous combinatorial optimization problem that possesses non-deterministic polynomial-time hardness. This is because the real-world order routing/dispatching operation involves many continuous variables such as order pick-up time and drop-off time. To solve this problem in a computationally efficient manner, a widely used meta-heuristic is presented in this paper. It consists of two steps. First, we will use a construct algorithm to obtain an initial solution. Thereafter, an ALNS meta-heuristic is applied to further improve the initial solution by minimizing the total cost.

Construction Algorithm

A simple heuristic is proposed to generate the initial solution (shown in algorithm 1). It consists of two steps. (i) Construct the sorted order sequence with regard to expected drop-off time t_{do}^{ie} and store in a priority queue (line 3). Then assign orders to nearest active driver (line 4). After this step, each active driver will be assigned one order. This strategy ensures that the most urgent orders

are paired with the nearest drivers. (ii) Exploit greedy insertion (described in the “Repair Process” section below) to plan all remaining orders according to the minimum increase of objective. The best insertion position for order i is calculated with $\min_k c_i^k$, where c_i^k represents the change in objective value after inserting order i into driver k ’s delivery sequence incurring least objective change (line 5–9). An insertion will be rejected if it will cause a driver to exceed their capacity or work schedule.

Algorithm I Construction Algorithm

```

1: Input:  $N$  : the order set,  $K$  : the driver set
2: Output: The initial route for each driver  $k$ 
3: Priority queue  $L \leftarrow$  Sort orders by  $t_{do}^{ie}$ 
4: Pop out order sequentially and assign to a nearest active
   driver
5: for order  $i$  remains in  $N$  do
6:   for driver  $k$  in  $K$  do
7:     greedy insertion( $i, k$ )
8:   end for
9: end for

```

Adaptive Large Neighborhood Search (ALNS)

Although the initial solution is feasible, it might involve unacceptable delivery delay and inappropriate task sequences. Therefore, we need to further improve the initial solution. In this paper, we utilize the ALNS framework, in which multiple removal and repair operators are selected based on an adaptive selecting mechanism, to diversify and intensify the initial solution then obtain the optimized solution. ALNS can explore a large neighborhood in a structured way, thus it has the potential to escape the local minimum and provide high quality solutions. The details of ALNS are described in the following sub-sections.

Removal Process. First, the current solution is destroyed with one of the following operators. An important parameter in the removal process is the degree of destruction. If only too small a part is destroyed, then it can be difficult to escape a local minimum. If an enormous part is destroyed, then it may require extensive time to construct a new solution. In this paper, a random number b is chosen from $\{4, \theta n\}$ to decide the number of removal orders, where n is the total number of orders and θ is the destroyed ratio. The removed tasks will be placed in a task pool.

- (a) Random removal: This operator randomly selects b tasks to remove to diversify the solution space.

- (b) Worst removal: This operator ranks the insertion cost of every order in an increasing order, while introducing a random number $y \in (0, 1)$ and a parameter p , then removes the order located at $y^p n$. This randomization is implemented to avoid removing the same task repeatedly.
- (c) Shaw removal: This operator was first proposed to solve VRP problems based on evaluating the similarity of two locations (21). In the ODFD problem, slight modifications are needed since the smallest unit in our problem is an order which consists of two locations. Specially, we calculate the relatedness of two orders with the following equation. The first part is the distance between order i and j including the restaurant’s distance $d_{p(i), p(j)}$ and the customer’s distance $d_{d(i), d(j)}$. The second part is the pick-up and drop-off time difference between two orders in the current assignment. Smaller $R_{i,j}$ indicates higher similarity between two orders. The Shaw removal operator removes most related orders each time.

$$R_{i,j} = p_1 (d_{p(i), p(j)} + d_{d(i), d(j)}) + p_2 \left(|t_p^i - t_p^j| + |t_{do}^{ie} - t_{do}^{je}| \right) \quad (14)$$

- (d) Distance-based path removal: This operator is designed to remove a route with longest distance. All tasks on this route will be placed into the removed task pool directly.
- (e) Delay-based path removal: This operator picks the route with longest total delay, then removes all tasks on that route.

Repair Process. A repair operator is employed to re-insert the removed tasks. In this paper, parallel insertion heuristics is chosen so that multiple routes are built simultaneously. Further, to reduce the computational complexity, we sort orders using the expected drop-off time t_{do}^{ie} then pop out each one to be re-inserted.

- (a) Random repair operator: Randomly select a feasible position, then insert the task. Similar to the random removal operator, this operator also perturbs the solution space.
- (b) Greedy insertion operator: Greedily insert every task into the best position such that the change of objective function is minimized.
- (c) Regret-q insertion operator: The main drawback of greedy insertion is that it might leave the most “expensive” task to the last iteration where we lack flexibility. Regret-q insertion could avoid

this situation by incorporating look-ahead information. Let Δc_i^j indicate the objective change when task i is inserted into j th cheapest position. Then we need to find the order i that maximizes the regret value in Equation 15. In this paper, Regret-2 and Regret-3 insertion operators are constructed.

$$\max_i \left\{ \sum_{j=2}^q (\Delta c_i^j - \Delta c_i^1) \right\} \quad (15)$$

Adaptive Weighting and Selection Mechanism. Instead of selecting only one removal and one insertion operator in the entire searching process, ALNS uses all the operators proposed above. In each iteration, one removal and one repair operator are selected independently based on the roulette wheel selection principle. At the beginning, each operator is equally weighted. Weights will be updated after a segment of iterations. Assume that we have n operators and operator i with weight w_i^k at segment k . Then the probability of choosing operator i at segment k is defined as $P_i^k = \frac{w_i^k}{\sum_{j=1}^n w_j^k}$. An adaptive weight adjustment method is introduced (Equation 16) to update the weight according to the operator performance.

$$w_i^k = (1 - \rho)w_i^{k-1} + \rho \frac{\mu_i}{\pi_i} \quad (16)$$

where the weight of operator i at segment k (w_i^k) is derived based on the weight at segment $k-1$ (w_i^{k-1}). ρ is a reaction factor that controls the speed of this algorithm reacting to the effectiveness of operators. π_i indicates the number of times that operator i is chosen in this segment. μ_i is the accumulated score of operator i (Equation 17). At the beginning of each segment, every accumulated score will be set to zero. In each iteration, a score γ^k will be assigned to the operator, which represents the performance of the operator (Equation 18).

$$\mu_i = \sum_{j=1}^{\pi_i} \gamma_j^k \quad (17)$$

$$\gamma_j^k = \begin{cases} \gamma^1 & \text{if a new best solution is obtained} \\ \gamma^2 & \text{if the solution is better than the current solution} \\ \gamma^3 & \text{if the solution is worse than the current one but still accepted.} \end{cases} \quad (18)$$

Acceptance and Termination Criteria. To avoid becoming trapped in a local minimum, we use simulated annealing strategy to accept a worse solution s' with probability of $e^{-\frac{f(s')-f(s)}{T}}$, where f is the objective function and T is the

temperature. T will decrease with a cooling rate δ : $T = \delta T (0 < \delta < 1)$. Considering practicality, we prefer to obtain good results in a short time rather than the optimal solution that takes a long computational time. Thus, we propose the following termination criteria to stop the ALNS algorithm. The algorithm will terminate if one of the rules is met: the maximum number of iterations φ_{max} is reached; φ iterations have been executed without any improvements.

EMFAC Model

The EMFAC model is calibrated from the average value of repeated measurements of total emissions per driving cycle. Developed by the California Air Resources Board, the EMFAC model integrates a mobile sources emission model where local-specific emission rate and vehicle activity are combined to generate seasonal or annual emissions for various geographic areas in California (22). We construct an EMFAC model for the City of Riverside, California, and specify the delivery vehicle to be a gasoline-fueled private vehicle. The following emission rates are obtained (Table 2).

The EMFAC model provides a convenient way to model area-wide vehicle emission levels which only requires the vehicle travel speed as input. We obtain the link-level travel speed of the City of Riverside and assume that each driver will follow the link speed in the network, thus determining the emission rate of the delivery vehicle per link. With the delivery routing result, we can sum up all emissions link-by-link to evaluate the emission impact. In this research, we evaluate the impact of greenhouse gases (GHG), NOx, CO, and PM2.5.

Scenario Design and Experiment Setup

In this section, we first set up ODFD service scenarios with consideration of the COVID-19 pandemic. A well-calibrated CEMDAP model is first applied to generate eat-out demand in the City of Riverside during the lunch hour, that is, 11:30 a.m. to 12:30 p.m. CEMDAP generates a total of 1,328 eat-out trips. This includes all eat-out options: dine-in, pick-up, or home delivery. We need to decide the home delivery ratio based on the survey data and current research related to COVID.

Before the COVID-19 pandemic, in the U.S. home delivery sales was reported to be 8% of the eat-out market (23). COVID brought major changes to eat-out behaviors. Some new requirements and strategies, such as social distancing and remote working, resulted in an increasing demand for ODFD services. With regard to the intention to eat out, Tahlyan et al. (24) studied behavior change in relation to multiple eat-out modes, including dine-in, pick-up, and on-demand delivery during

Table 2. Pollutant Emission Rates From Private Delivery Vehicle Using EMFAC Model (22)

Speed (mph)	NOx_RUNEX	PM2.5_RUNEX	CO_RUNEX	GHG_RUNEX
5	0.1047548	0.0095916	1.9610494	730.490871661
10	0.0920467	0.0060364	1.7142107	587.889228881
15	0.0807634	0.0040165	1.5665320	481.750639020
20	0.0721445	0.0028265	1.4482002	401.929636697
25	0.0654610	0.0021013	1.3648953	346.068099773
30	0.0607778	0.0016363	1.2398813	306.232655164
35	0.0570647	0.0013466	1.1472837	284.141599805
40	0.0546770	0.0011704	1.0674726	275.301472327
45	0.0532620	0.0010726	0.9946617	276.339908564
50	0.0526385	0.0010371	0.9344414	284.848677089
55	0.0534785	0.0010568	0.8774290	296.359939367
60	0.0548256	0.0011285	0.8142389	307.219363871
65	0.0565380	0.0012766	0.7742411	317.223470809
70	0.0610274	0.0013955	0.7685599	319.747283997

Note: GHG = greenhouse gases; RUNEX = running exhaust emissions from vehicle tailpipe while traveling on the road. It is assumed that the fuel used is gasoline. The unit for all emission rates is grams per mile.

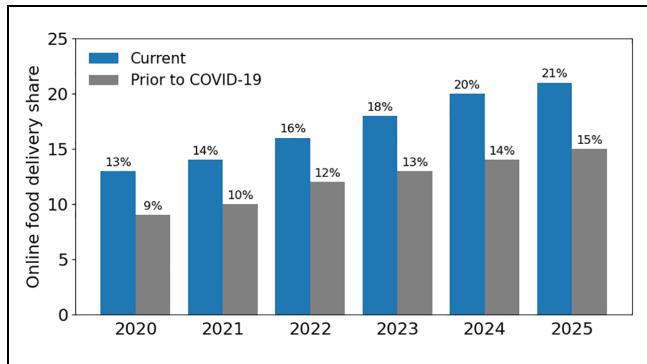


Figure 4. Prediction of online food delivery penetration share in the U.S. (24).

COVID. According to the data analysis of this study, we found that the total eat-out demand reduced by 34% because of regulations and concerns on COVID and more people opted to order food online.

The effect of COVID has evolved with time, with changes in the pandemic condition, public attitudes, and shopping preferences. Many studies have focused on shopping behavior during and after COVID. For example, Wang et al. (25) showed that the long-term effect of COVID on less essential goods (i.e., takeout food) would be approximately half that of the near-term effects. The long-term impact of COVID on eat-out choice is still under debate. Multiple surveys have explored the change in online shopping behaviors during COVID. One survey by Morgan Stanley exactly focuses on the change in online food delivery penetration share (5). The main survey result is shown in Figure 4. The black bar shows the prediction result before COVID, and blue bar is the

adjusted prediction result considering the short-term and long-term impacts of COVID. With this information, we averaged the ratio over the years 2020 to 2023 to obtain the delivery ratio during COVID. Since the COVID-19 virus has many variants and its effects are still not completely terminated, in this research, we set the expected time for “after COVID” at 2025. Depending on the long-term effects of COVID, we set up two scenarios to discuss the post-COVID food delivery situation. With all the research and survey results, we set up the following experiment scenarios.

1. Scenario B: Before COVID, the total eat-out demand is unaffected by COVID. The ODFD ratio is 8%.
2. Scenario D: During COVID, the total eat-out demand shrinks by 34% because of restaurant closures and restrictions. The ODFD ratio increases to 15%.
3. Scenario A1: After COVID, the total eat-out demand recovers to the before-COVID level. The ODFD ratio is 15% (conservative).
4. Scenario A2: After COVID, the total eat-out demand recovers to the before-COVID level. The ODFD ratio is 21% (optimistic).

We sampled from the total eat-out trips according to the scenario settings. As an example, Figure 5 shows the ODFD customer location and restaurant distribution in scenario A2, with a total of 278 orders, scattered in the City of Riverside.

To simulate the ODFD operation, we set up the ODFD scenario as follows. After the customer places one order, the food preparation time is a random



Figure 5. On-demand food delivery orders in case A2. Red point = restaurant, green point = customer location, arc = meal order.

number within interval [5, 20], with which we can define the order's ready time. The expected click-to-door time is 40 min (26), indicating the system aims to deliver every food order within 40 min. Each driver needs an extra 1 min to pick up or drop off when they arrive at the restaurant/customer location. Moreover, link travel time is obtained with real-world link length divided by link speed, both of which were extracted from the Riverside BEAM model (20). Then we do the routing using the Networkx module (27) to obtain the fastest route between any two locations. We first generate drivers' initial locations around the residential area, assuming that drivers will depart from home to pick up a food order. For drivers with multiple delivery rounds, we assume they will be idle around the last visit location to wait for new orders. The generated driver number is according to the order/driver ratio and is set to five. The driver capacity is 10 orders per trip. In general, food orders will arrive according to time. We set the system to update in every 10 min to dispatch the new arriving orders. Finally, based on multiple experiments, we set the weight α and β both equal to one in the objective function.

For parameter settings in the ALNS algorithm, we only list the key parameters. Other detailed parameter settings can be found at (14). The destruction degree θ of removal operator is 0.2. The reaction factor ρ in the operator weight adjustment is set at 0.6. The operator scores based on the performance is set at (30,18,12). Cooling rate in the simulated annealing process is set at 0.99. In the ALNS iteration, we run a total of 100 segments, where each segment contains 50 iterations, thus φ_{max} is 5,000 iterations. If 500 continuous iterations execute without any improvement, we terminate the algorithm.

Experiment Result and Analysis

Performance of ALNS Algorithm

In this part, we first illustrate the performance of the proposed algorithm. We compare with two sets of baselines: (i) iterated greedy algorithm (IG) and (ii) 12 LNS algorithms that combine the previous proposed removal and repair operators. The IG method, proposed in Wang et al. (28), is based on the local search mechanism to select long delay orders or earlier drop-off orders and then rearrange the delivery routes to minimize total cost. In LNS, a fixed pair of operators will be used repeatedly to improve the solution. The order numbers in scenario B (106 orders) and D (131 orders) are similar and computational complexity is also similar, thus we only choose scenarios D, A1, and A2 to evaluate the algorithm performance. To ensure fair comparison, we apply the same construction algorithm to obtain an initial solution and the same termination criteria. The algorithm performance is summarized in Table 3.

In all three cases, we can note that ALNS outperforms both IG and LNS algorithms in improving the delivery performance. The IG algorithm is mainly based on a local search mechanism which will limit the algorithm to search for a wide range of neighborhoods and can be easily trapped in local minima. With the same termination criterion, the IG algorithm is around 4% worse than the ALNS algorithm. Among the 12 LNS algorithms, we can see that the SR-G operator pair performs best in case D and case A1, which is only 0.7% worse than ALNS, though more computational time is required to achieve such a high quality solution. But in case A2, the SR-G algorithm is 6% worse. This shows the dependency of the LNS algorithm of different cases and the lack of robustness. We then average 12 LNS algorithm performances to obtain the LNS-Avg result and compare it with ALNS performance. This comparison again shows the superiority of the ALNS algorithm. ALNS can perform 9%, 10%, and 13% better in D, A1, and A2 case respectively. Further, we can note that the average running time of ALNS is longer than other algorithms, because ALNS iteratively searches for more potential neighborhoods to escape the local minima. Another reason is that our algorithm is implemented on an ordinary computer and written in the Python script language. In practice, one can consider using parallel or distributed computing and coding in a much faster programming language, such as C++ or Java to further accelerate the computational speed.

Operational Performance Analysis of ODFD Service

We ran our proposed algorithms over all four cases: B, D, A1 and A2. With the obtained order dispatching and

Table 3. Performance Comparison of Different Algorithms for On-Demand Food Delivery with Three Scenarios

Algorithm	Scenario								
	D			A1			A2		
	Obj	Avg time (s)	Obj gap	Obj	Avg time (s)	Obj gap	Obj	Avg time (s)	Obj gap
IG	1232.44	24.91	-5.08%	2026.00	89.56	-7.29%	2443.30	130.35	-3.82%
WR-G	1198.53	79.55	-2.19%	1972.89	222.03	-4.29%	2494.82	137.82	-5.77%
WR-R2	1311.72	16.02	-11.84%	2275.94	41.00	-17.03%	2705.60	65.68	-13.11%
WR-R3	1324.60	15.64	-12.94%	2262.77	46.87	-16.55%	2710.53	65.20	-13.26%
SR-G	1179.98	189.17	-0.61%	1901.20	165.16	-0.68%	2507.26	132.69	-6.23%
SR-R2	1270.83	14.78	-8.35%	2170.90	35.24	-13.02%	2635.25	54.50	-10.79%
SR-R3	1294.17	13.31	-10.34%	2168.16	41.51	-12.91%	2452.58	158.54	-4.14%
DisR-G	1263.87	17.69	-7.76%	2117.80	39.00	-10.84%	2512.31	159.47	-6.42%
DisR-R2	1378.91	7.82	-17.57%	2313.73	15.67	-18.39%	2502.13	136.33	-6.04%
DisR-R3	1374.19	7.75	-17.17%	2117.80	38.40	-10.84%	2455.63	169.37	-4.26%
DeR-G	1278.00	17.08	-8.97%	2138.98	48.72	-11.72%	2589.19	54.93	-9.20%
DeR-R2	1366.01	7.53	-16.47%	2314.81	15.23	-18.43%	2714.14	21.26	-13.38%
DeR-R3	1378.61	7.47	-17.54%	2327.35	17.03	-18.87%	2721.21	20.82	-13.61%
LNS-Avg	1301.62	32.82	-10.98%	2173.53	60.49	-13.12%	2583.39	98.05	-9.00%
ALNS	1172.85	158.14	0%	1888.26	447	0%	2350.98	834.12	0%

Note: Avg = average; IG = iterated greedy; WR = worst removal; SR = Shaw removal; DisR = distance-based path removal; DeR = delay-based path removal; G = greedy insertion; R2 = regret-2 insertion; R3 = regret-3 insertion; LNS = large neighborhood search; ALNS = adaptive large neighborhood search; Obj = optimized objective value of delivery problem; Avg time = average running time to obtain an optimized solution when system updates.

"Obj gap" is calculated with $(\text{ALNS_obj} - B)/B$, where ALNS_obj is the objective value obtained with our proposed ALNS algorithm and B denotes the objective value obtained with the corresponding baseline algorithm. A negative gap indicates a better solution with ALNS. ALNS can achieve the minimal cost compared to baseline algorithms.

Table 4. Operational Performance of the Solution

Scenario	Number of orders	Vehicle-miles traveled (km)	Total delivery delay (min)	Unsatisfied orders	Average T_{ctd} (min)	Average T_{rtd} (min)
B	106	914	159	4	33.99	20.48
D	131	1052	120	3	33.37	20.75
A1	200	1658	230	4	33.86	20.68
A2	278	2073	277	7	33.45	20.66

routing result, we further look into the operational performance of the ODFD service. In Table 4, we show the general operational performance. Among four cases, the total VMT cost increases because of the rising demand for the ODFD service. From the order delivery delay, we can note that the average delay per order is within 2 min. The average click-to-door time is around 33 min, indicating that on average customers can receive food orders within 33 min after placing the order online, which is acceptable in the real world. Meanwhile, the food freshness can be guaranteed because the average order ready-to-door time is around 20 min.

Figure 6 shows the box plot of the click-to-door time and ready-to-door time distribution in detail. The x-axis represents multiple scenarios results and the y-axis is the time unit (minutes). In the experiment setup, we set the expected click-to-door time for all orders at 40 min. Thus, the first threshold indicates whether any delay

occurs. T_{ctd} of less than 40 min means no delay occurs and the order has been delivered earlier or on time. It is important to note that up to 75% of orders are delivered within T_{ctd}^e . The second threshold indicates whether the food order is delivered more than 10 min later than the expected time, and we tagged these orders as unsatisfied order. We can see that only a few orders are delivered later than 10 min from the expected time, that is, there were only three unsatisfied orders in case D (listed in Table 4). All orders are delivered within 60 min, as we can identify the maximum T_{ctd} in the box plot as well. This attractive operational performance is achieved because the ODFD system can gather several orders and determine order dispatching and routing decisions in coordination using the proposed ALNS algorithm. Thus, the delivery drivers are able to pick up multiple food orders in the restaurant areas then deliver to the customer zones, or pick up new orders along the delivery

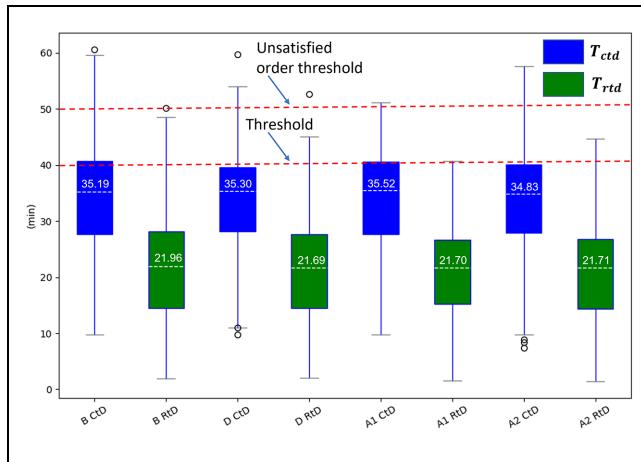


Figure 6. Distribution of click-to-door time and ready-to-door time.

route, delivering food orders more efficiently. To sum up, we can achieve satisfactory delivery performance measured by delivery speed and food freshness.

VMT and Emissions Impacts of ODFD

In this section, we quantify the VMT and emissions impacts of all eat-out trips. For ODFD trips, the routing result for a delivery driver is utilized as the input of the EMFAC model to calculate the emission amount. For other eat-out demands, we assume people visit the restaurant either to dine in or take out; these trips are in-person roundtrips between customer and restaurant locations. We use the Networkx module to obtain the fastest route for in-person trips and quantify the corresponding VMT. We then use the EMFAC model to obtain the emissions impact. We combine the impacts of ODFD trips and in-person trips together to evaluate the VMT and emissions impacts under different COVID periods (B, D, A1, and A2). The detailed results are summarized in Table 5. For all four scenarios, the average distance cost for an ODFD order is 7.5 to 8.5 km/order. While for the in-person trips to pick up an order, the distance cost is around 18 km/order. This difference illustrates the superiority of ODFD service to reduce VMT with the optimized solution.

Take case B as the baseline scenario; during COVID the total VMT and emissions all reduced by 38%, including vehicles on the road for the purpose of eating out. This might be explained by the following two reasons. First, the government-imposed lock downs, social distancing requirements, and fear of infection, meant that the eat-out demand reduced drastically, which can undoubtedly reduce the VMT and emissions. Second, more people tended to use the ODFD service as it is a desirable choice for convenient and contactless food delivery under

the wide concern of personal health. With the ODFD service, the system can bundle more orders together to pick up most food orders in the commercial zone first then deliver them to the customers sequentially, so one would expect more saving of VMT and emissions.

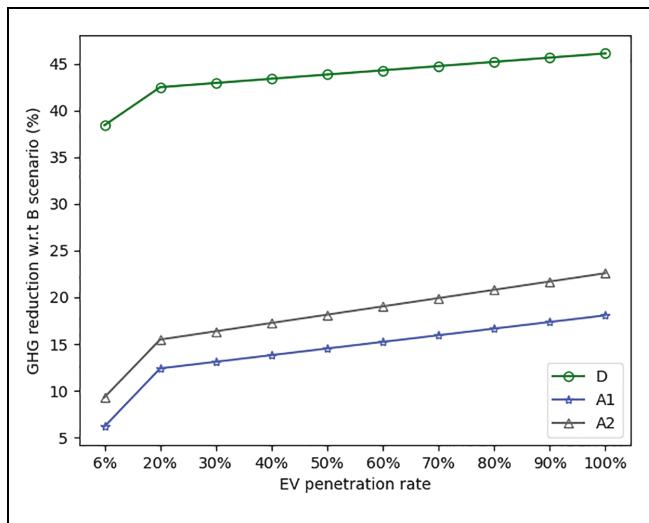
Meanwhile, the ODFD service may change people's eat-out behavior in the long term considering the potential change in dining style. When we compare case A1 and case A2 with case B, we can note that although the eat-out demand recovers to the before-COVID level, with higher ODFD ratio, the VMT and emissions can still be decreased by around 6% to 9% depending on the future ODFD penetration rate. The dining-related vehicles on road are reduced by 6% to 11% as well. Thus, high penetration rate of ODFD can help reduce VMT in the traffic network and to build an eco-friendly transportation system.

It is also noticed that a similar reduction rate can be found in VMT and pollutant emissions for the same scenario. The main reason is that with EMFAC model, VMT and speed pattern are two key inputs for emission estimation. As the objective function only aims to minimize the VMT, the speed pattern is hard to optimize or influence. Thus, the percentage change in emissions is the same as the percentage change in VMT during different periods. We will introduce the fuel-consumption/emissions into the objective function to avoid this limitation in future research. Another possible reason is that we assume all vehicles to be gasoline fueled. With the homogenous vehicle fleet, VMT saving is proportional to emission savings. If some vehicles are replaced with electric vehicles, then we can expect more emission reduction in the urban context.

To validate this assumption, we introduced electric vehicles (EVs) into the delivery vehicle fleet under multiple penetration rates and calculated the emissions change. EVs are reported to occupy 6% of the private vehicle market share in California (29). Our baseline scenario is set up as 6% dining-related vehicles being replaced with EVs. Both dine-in and food delivery emissions are reduced proportionally with the EV penetration rate in these two sectors. If dine-in vehicles and delivery vehicles were electrified by the same proportion, then environmental impacts are reduced by the same amount as in the all gasoline vehicle case. To specifically study the benefits of delivery fleet electrification, we conducted a sensitivity analysis by steadily increasing the EV penetration rate only in the delivery fleet and assumed that 94% of in-person trips still use gasoline vehicle. In the real world, in practice, delivery platforms can subsidize EV drivers to increase the EV penetration rate, which can be potentially higher than the total EV penetration rate in the private car market. For example, Uber has already launched a pilot program to subsidize EV drivers

Table 5. Vehicle-Miles Traveled (VMT) and Emission Impacts of On-Demand Food Delivery (ODFD) in Different Scenarios

Scenario	Eat-out trips	No. orders	No. vehicles	VMT (km)	Greenhouse gases (GHG) (kg)	CO (g)	PM2_5 (g)	NOx (g)
B	ODFD trips	106	23	914.22	175.69	530.23	0.75	32.21
	In-person trips	1,222	1,222	22,428.84	4,317.86	12,138.02	17.79	784.69
	All eat-out trips	1,328	1,245	23,343.06	4,493.55	12,668.25	18.54	816.90
D	ODFD trips	131	28	1,052.00	202.64	612.31	0.87	37.14
	In-person trips	746	746	13,351.50	2,570.75	7,238.60	10.56	467.37
	All eat-out trips	877	774	14,403.50	2,773.39	7,850.91	11.43	504.51
A1	ODFD trips	200	45	1,658.12	318.24	958.78	1.35	58.36
	In-person trips	1,128	1,116	20,298.74	3,907.52	11,000.86	16.03	710.28
	All eat-out trips	1,328	1,161	21,956.86	4,225.76	11,959.64	17.38	768.64
A2	ODFD trips	278	57	2,073.08	397.80	1,203.50	1.69	72.99
	In-person trips	1,050	1,050	19,189.46	3,692.47	10,404.08	15.14	671.27
	All eat-out trips	1,328	1,107	21,262.54	4,090.27	11,607.59	16.83	744.26
Comparison		D versus B		A1 versus B		A2 versus B		
Metric	No. vehicles	-37.83%		-6.75%		-11.08%		
	VMT (km)	-38.30%		-5.94%		-8.91%		
	GHG (kg)	-38.28%		-5.96%		-8.97%		
	CO (g)	-38.03%		-5.59%		-8.37%		
	PM2_5 (g)	-38.36%		-6.23%		-9.20%		
	NOx (g)	-38.24%		-5.91%		-8.89%		

**Figure 7.** Reduction of greenhouse gas emissions under increasing electric vehicle (EV) penetration rates.

with an extra one dollar per order (30). Taking GHG emissions as an example, we present the results in Figure 7. The result shows that the GHG emissions reduced significantly with higher EV penetration rate. With the same VMT saving as in Table 5, in the during-COVID case, the emissions can be reduced by 45%. In the after-COVID case with 100% EV delivery fleet, the ODFD service can save 14% to 22% of emissions and 6% to 9% of VMT cost. This sensitivity analysis provides evidence to encourage the food delivery platforms to pay

attention to the delivery vehicles' composition and encourage more EV drivers to participate in the delivery process, since delivery fleet electrification can reduce further energy consumption and mitigate negative environmental externalities.

Conclusion

The COVID-19 pandemic has brought major changes to eat-out behaviors and since the start of the pandemic more people tend to order food online. There is great uncertainty about the extent to which this fast-growing food delivery service will affect the transportation system (VMT) and urban environment (pollutant emissions). In this paper, a comprehensive framework is proposed to evaluate the VMT and emissions impact of ODFD services. An effective order dispatching and routing algorithm is implemented to assign order sequences to available drivers. The ALNS outperforms 13 baseline algorithms in obtaining high quality solutions. The operational analysis of the experiment results further shows that ALNS can provide an ODFD solution with fast delivery speed and high standard of food freshness. From the city-wide VMT and emission impact analysis of all dining-related trips through different COVID periods, we can note that the total VMT and pollutant emissions (including GHG, CO, PM2.5, and NOx) all reduced by 38% during COVID and by 6% to 9% in the after-COVID case. With increasing EV ratio of the delivery fleet, the results show that emissions reduced significantly with higher EV penetration rate. With 100% EV

delivery fleet, the ODFD service can save 14% to 22% of emissions in the post-COVID context.

This study has potential limitations. First, our evaluation of the VMT and environmental benefits may be optimistic. (i) The VMT and emissions impact are based on the optimized order dispatching and routing result, whereas in the real world this depends on the delivery platform's operation strategy and optimization objective. (ii) We assume delivery drivers will remain idle at a position to wait for new orders instead of actively moving around to enhance the chance of receiving new delivery tasks, while in practice such deadheading trips might account for a higher amount of total VMT which may offset the environmental benefits of ODFD (31). Second, our food delivery setup is static without considering the dynamism of order demand and driver resources. This might not exactly represent the real-world delivery process since it is still possible that drivers may be assigned new orders during the delivery round. Finally, because of the limitation of existing research on the impact of COVID on ODFD, we have to take reference from other research and survey results. This might introduce a slight gap since the research area and context are different in our research and other references.

In the future, we will work on improving the model, for example, setting fuel consumption, which is also directly correlated with GHG emissions, as the main objective of the optimization model to investigate the potential of eco-friendly ODFD. Moreover, factors in real-world ODFD operation such as profit, incentives, and compensation for drivers will be considered in the future work. Another interesting research direction is the integration of EVs with ODFD, such as considering EV range limits and access to charging, utilizing EVs to reduce emissions, and optimizing the order dispatching decision with a mixed energy fleet of traditional vehicles and EVs. Finally, this proposed framework and algorithms could be readily employed to solve other on-demand delivery problems, such as groceries and parcels, and evaluate the corresponding impact.

Author Contributions

The authors confirm contribution to the paper as follows: study conception and design: H. Liu, P. Hao, Y. Liao, K. Boriboonsomsin, M. Barth; data collection: H. Liu, P. Hao, Y. Liao; analysis and interpretation of results: H. Liu, P. Hao, Y. Liao, K. Boriboonsomsin, M. Barth; draft manuscript preparation: H. Liu, P. Hao. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This study is funded by the National Center for Sustainable Transportation (NCST).

ORCID iDs

Yejia Liao  <https://orcid.org/0000-0003-4997-7528>
 Kanok Boriboonsomsin  <https://orcid.org/0000-0003-2558-5343>
 Matthew Barth  <https://orcid.org/0000-0002-4735-5859>

References

1. Schmidt, S., C. Benke, and C. A. Pané-Farré. Purchasing Under Threat: Changes in Shopping Patterns During the COVID-19 Pandemic. *PLoS One*, Vol. 16, No. 6, 2021, p. e0253231.
2. Moon, J., Y. Choe, and H. Song. Determinants of Consumers' Online/Offline Shopping Behaviours During the COVID-19 Pandemic. *International Journal of Environmental Research and Public Health*, Vol. 18, No. 4, 2021. <https://doi.org/10.3390/ijerph18041593>.
3. Zhao, Y., and F. Bacao. What Factors Determining Customer Continuously Using Food Delivery Apps During 2019 Novel Coronavirus Pandemic Period? *International Journal of Hospitality Management*, Vol. 91, 2020, p. 102683.
4. Centers for Disease Control and Prevention. Interim Infection Prevention and Control Recommendations for Healthcare Personnel During the Coronavirus Disease 2019 (COVID-19) Pandemic. 2020. <https://pesquisa.bvsalud.org/global-literature-on-novel-coronavirus-2019-ncov/resource/en/grc-747106>
5. Statista. Impact of COVID-19 on Online Restaurant Delivery Market Share in the U.S. 2020–2025. <https://www.statista.com/statistics/1170614/online-food-delivery-share-us-coronavirus/>. Accessed June 17, 2022.
6. Liu, Y., B. Guo, C. Chen, H. Du, Z. Yu, D. Zhang, and H. Ma. FooDNet: Toward an Optimized Food Delivery Network Based on Spatial Crowdsourcing. *IEEE Transactions on Mobile Computing*, Vol. 18, No. 6, 2019, pp. 1288–1301.
7. Tu, W., T. Zhao, B. Zhou, J. Jiang, J. Xia, and Q. Li. OCD: Online Crowdsourced Delivery for On-Demand Food. *IEEE Internet of Things Journal*, Vol. 7, No. 8, 2020, pp. 6842–6854.
8. Zheng, H., S. Wang, Y. Cha, F. Guo, J. Hao, R. He, and Z. Sun. A Two-Stage Fast Heuristic for Food Delivery Route Planning Problem. *Proc., INFORMS Annual Meeting*, Seattle, WA, October 20–23, 2019. <https://doi.org/10.1287/infp.2019.XXXX>.
9. Ropke, S., J.-F. Cordeau, and G. Laporte. Models and Branch-and-Cut Algorithms for Pickup and Delivery Problems with Time Windows. *Networks: An International Journal*, Vol. 49, No. 4, 2007, pp. 258–272.
10. Ropke, S., and J.-F. Cordeau. Branch and Cut and Price for the Pickup and Delivery Problem with Time

Windows. *Transportation Science*, Vol. 43, No. 3, 2009, pp. 267–286.

11. Li, H., and A. Lim. A Metaheuristic for the Pickup and Delivery Problem with Time Windows. *International Journal on Artificial Intelligence Tools*, Vol. 12, No. 02, 2003, pp. 173–186.
12. Lim, H., A. Lim, and B. Rodrigues. Solving the Pickup and Delivery Problem with Time Windows using “Squeaky Wheel” Optimization with Local Search. *Proc., American Conference on Information Systems AMCIS 2002*, Dallas, TX, 2002. https://ink.library.smu.edu.sg/lkcsb_research/1963AIS, 2002.
13. Bent, R., and P. Van Hentenryck. A Two-Stage Hybrid Algorithm for Pickup and Delivery Vehicle Routing Problems with Time Windows. *Computers & Operations Research*, Vol. 33, No. 4, 2006, pp. 875–893.
14. Ropke, S., and D. Pisinger. An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows. *Transportation Science*, Vol. 40, No. 4, 2006, pp. 455–472.
15. Emeç, U., B. Çatay, and B. Bozkaya. An Adaptive Large Neighborhood Search for an E-Grocery Delivery Routing Problem. *Computers & Operations Research*, Vol. 69, 2016, pp. 109–125.
16. *Synthpop: Synthetic Populations from Census Data*. Github. <https://github.com/UDST/synthpop>
17. Southern California Association of Governments - SCAG. <https://scag.ca.gov/>. Accessed March 14, 2023.
18. Bhat, C. R., J. Y. Guo, S. Srinivasan, and A. Sivakumar. Comprehensive Econometric Microsimulator for Daily Activity-Travel Patterns. *Transportation Research Record: Journal of the Transportation Research Board*, 2004. 1894: 57–66.
19. Hao, P., C. Wang, G. Wu, S. Tanvir, B. Sun, J. Holden, A. Duvall, J. Gonder, and M. Barth. Evaluate the System-Level Impact of Connected and Automated Vehicles Coupled with Shared Mobility: An Agent-Based Simulation Approach. 2021. <https://trid.trb.org/view/1759510>
20. BEAM. The Modeling Framework for Behavior, Energy, Autonomy, and Mobility. Lawrence Berkeley National Laboratory Sustainable Transportation Initiative. <https://transportation.lbl.gov/beam>. Accessed July 1, 2022.
21. Shaw, P. Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems. In *Principles and Practice of Constraint Programming — CP98* (M. Maher, and J. F. Puget, eds.), Springer, Berlin and Heidelberg, Germany, 1998, pp. 417–431.
22. EMFAC. Emission Inventory. <https://arb.ca.gov/emfac/emissions-inventory/>. Accessed June 7, 2022.
23. Statista. Share of Delivery Sales in the Food Service Market in the U.S. 2013–2022. *Statista*. <https://www.statista.com/statistics/1091419/food-service-sales-delivery-share-us/>. Accessed July 30, 2022.
24. Tahlyan, D., M. Said, H. Mahmassani, A. Stathopoulos, J. Walker, and S. Shaheen. *Latent Transition Analysis of Consumer Spending Behavior and Adaptation Across Online and In-Person Channels Through the Pandemic*. Presented at 101st Annual Meeting of the Transportation Research Board, Washington, D.C., 2021.
25. Wang, X. C., W. Kim, J. Holguín-Veras, and J. Schmid. Adoption of Delivery Services in Light of the COVID Pandemic: Who and How Long? *Transportation Research Part A: Policy and Practice*, Vol. 154, 2021, pp. 270–286.
26. New Food Delivery Study Shows What Consumers Crave In a Service. <https://www.usfoods.com/our-services/business-trends/2019-food-delivery-statistics.html>. Accessed January 27, 2022.
27. Hagberg, A., P. Swart, and D. S. Chult. Exploring Network Structure, Dynamics, and Function Using Networkx. 2008. <https://www.osti.gov/biblio/960616>
28. Wang, X., S. Wang, L. Wang, H. Zheng, J. Hao, R. He, and Z. Sun. An Effective Iterated Greedy Algorithm for Online Route Planning Problem. *Proc., IEEE Congress on Evolutionary Computation (CEC)*, Glasgow, UK, IEEE, New York, 2020, pp. 1–8.
29. Kane, M. California: Plug-Ins Capture Almost 11% of the Market in Q1 2021. *InsideEVs*. <https://insideevs.com/news/516906/california-plugin-car-sales-2021q1/>. Accessed July 12, 2022.
30. Uber Together on the road to zero emissions. <https://www.uber.com/us/en/drive/services/electric/>.
31. Wenzel, T., C. Rames, E. Kontou, and A. Henao. Travel and Energy Implications of Ridesourcing Service in Austin, Texas. *Transportation Research Part D: Transport and Environment*, Vol. 70, 2019, pp. 18–34.

The contents of this paper reflect only the views of the authors, who are responsible for the facts and the accuracy of the data presented in the paper.