Mobility21

A USDOT NATIONAL UNIVERSITY TRANSPORTATION CENTER

Carnegie Mellon University

Penn University of Pennsylvania

The Ohio State University

CCAC

Smart Curbspace: Optimized Parking Reservations for Diverse Stakeholders

Aaron Burns (https://orcid.org/0000-0001-9616-8104)
Connor Forsythe, Ph.D. (https://orcid.org/0000-0002-3734-1840)
Jeremy Michalek, Ph.D. (https://orcid.org/0000-0001-7678-8197)
Kate Whitefoot, Ph.D. (https://orcid.org/0000-0001-7678-8197)

FINAL RESEARCH REPORT – 31 July 2023

Contract # 69A3551747111

ESTIMATING THE POTENTIAL TO REDUCE DOUBLE PARKING AND CRUISING WITH A DYNAMIC CURB SPACE RESERVATION SYSTEM FOR DIVERSE USERS

Aaron Burns
Department of Engineering and Public Policy
Department of Mechanical Engineering
Carnegie Mellon University, Pittsburgh, PA, 15213
aaronbur@andrew.cmu.edu
https://orcid.org/0000-0001-9616-8104

Connor Forsythe, Ph.D.
Department of Mechanical Engineering
Department of Engineering and Public Policy
Carnegie Mellon University, Pittsburgh, PA, 15213
cforsyth@andrew.cmu.edu
https://orcid.org/0000-0002-3734-1840

Jeremy J. Michalek, Ph.D.
Department of Engineering and Public Policy
Department of Mechanical Engineering
Department of Civil and Environmental Engineering
Carnegie Mellon University, Pittsburgh, PA, 15213
jmichalek@cmu.edu[1]
https://orcid.org/0000-0001-7678-8197

Kate Whitefoot, Ph.D.
Department of Mechanical Engineering
Department of Engineering and Public Policy
Carnegie Mellon University, Pittsburgh, PA, 15213
kwhitefoot@cmu.edu
https://orcid.org/0000-0001-8961-5367

[1]Corresponding author

DISCLAIMER

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

ABSTRACT

As the demand for curb parking increases and new types of curb space users compete for space, the need to more efficiently manage how vehicles interact with the curb becomes more apparent. One solution is to allow curb space users to submit a reservation ahead of their arrival that can be centrally managed and scheduled if the resources are available. To study the potential benefits and drawbacks of an intelligently managed curb, we develop a dynamic parking reservation system that continually collects parking requests from delivery and private vehicles and re-optimizes the schedule of accepted parking requests periodically throughout the day. This process employs model predictive control (MPC) to iteratively apply a mixed integer linear programming parking slot assignment optimization formulation adapted from prior literature. In our preliminary results we observe that our MPC algorithm can reduce computation cost and provide tractable parking schedules, which is not often possible with complex day-ahead optimal schedule generation. Additionally, we compare the reduction in total minutes of double parking and cruising between a first-come first-serve (FCFS) paradigm and our MPC algorithm. We preliminarily observe that the MPC algorithm can reduce double parking and cruising under some conditions, but correlations between request time and dwell time lead to cases where dynamically optimized reservations can increase double parking and cruising relative to FCFS. We intend to further explore and characterize conditions under which reservation systems improve parking metrics in future research.

Keywords: Smart Curbspace, Model Predictive Control, Dynamic Parking Reservations, Mixed Integer Linear Program

INTRODUCTION

Competition for curb space parking is a growing concern for cities due to increasing demand from delivery companies, private vehicles, and transportation network companies (TNCs), among others (*1–5*). Unfortunately, ineffective curb management can result in additional congestion, emissions, and safety issues when vehicles unable to find a parking space either "cruise" in search of another location or double park adjacent to an occupied space and block a lane of traffic as noted by Jaller (*6*). Several researchers have further explored and studied vehicle "cruising" (*7–11*) and the negative impacts of double parking (*12–16*).

A possible solution to more effectively use curb space is a technology concept known as "Smart Curbspace" which broadly envisions an intelligently managed curb parking space system. One version of a "Smart Curbspace" system which consists of an optimized reservation schedule for delivery vehicles based on day-ahead parking requests was characterized by Burns et al. (*12*) and a representation of the system borrowed from the authors is shown in Figure 1. The authors uncovered a range of impacts from zero benefit to over $300,000 in energy and congestion related savings per year per parking space due to a reduction in minutes of double parked vehicles.

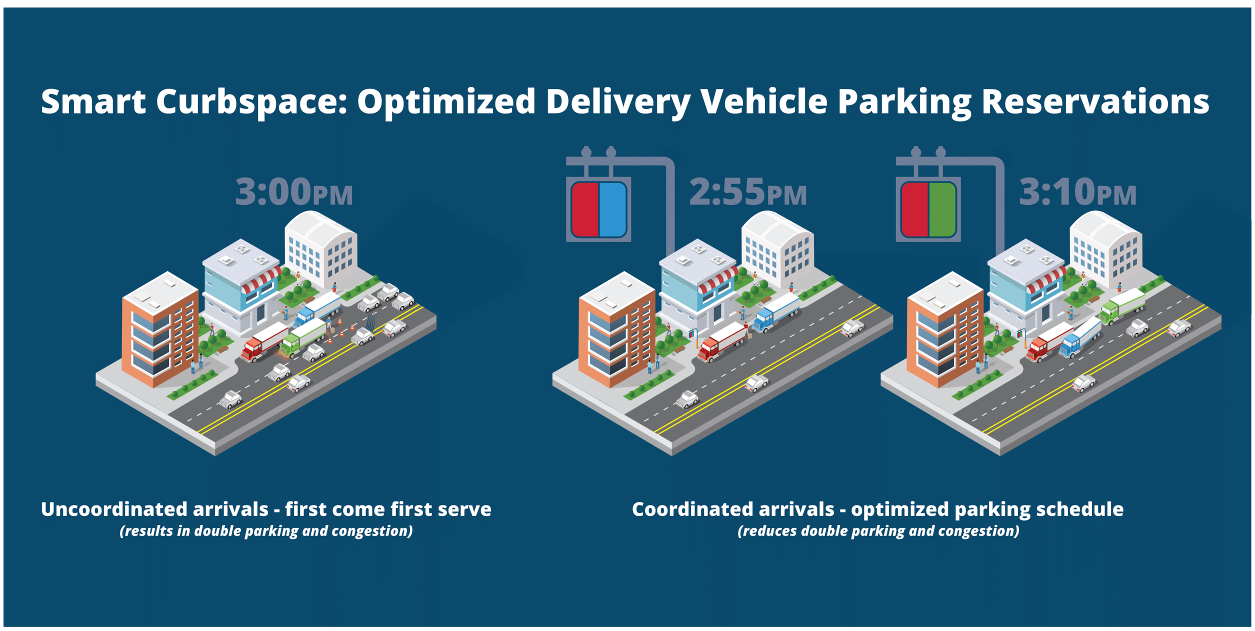


FIGURE 1: A Smart Curbspace system allows for optimized scheduling of delivery vehicle arrivals to parking spaces which has the potential to reduce the traffic congestion caused by uncoordinated, first-come first-serve delivery vehicles and double parking. Figure taken from Burns et al. (*12*).

In this research we explore the performance of a dynamic reservation system when compared with a first-come first-serve (FCFS) paradigm while considering uniquely characterized parking requests for different types of parking space customers. To create a dynamic real-time optimal curbspace reservation management system, we expand on the formulation in Burns et al. (*12*) and apply the principles of model predictive control (MPC) to iteratively solve many, but smaller optimal schedules. Periodically solving parking schedules also allows for the incorporation of new parking requests instead of assuming all requests will be placed a day-ahead as in Burns et al. (*12*).

Dynamic requests may also be more relevant for mixed-use spaces. To explore this idea we include vehicle parking requests typical of both private passenger vehicles and delivery vehicles.

After exploring 9,000 cases, we preliminarily observe that the MPC algorithm can have both beneficial and negative impacts on the total minutes of double parking and cruising when compared with a first-come first-serve paradigm. The cases with negative impacts are primarily present for double parking and are likely due to correlations between request time and dwell time present in the data used to generated test scenarios. We plan to investigate the impact of parking assignment lead time in our future research. We also preliminarily observe a mostly homogeneous impact of the multi-objective function weights on the reduction of double parking and cruising i.e. a similar impact to double parking regardless of how we prioritize vehicles assignments based on their expected double parking or cruising behavior, suggesting our weighting values have insufficient fidelity. An updated assessment will be provided in future research after studying both smaller values of parking assignment lead time as well as a large set of objective weights.

Finally, we also preliminarily observe that the MPC algorithm is more computationally tractable than the full day optimal formulation across a normalized measure of parking space demand. This behavior is expected as the MPC conducts many less complex and faster optimization formulations which can be iteratively completed, whereas a single full day formulation may be very complex and require a substantially longer compute duration.

LITERATURE REVIEW

Model Predictive Control (MPC) Methods for Incorporating Dynamic Parking Requests

Our research seeks to study performance improvements from a dynamic optimal curbspace reservation management system when applied to a diverse set of parking space customers. We define a dynamic optimal system as a process which incorporates an iterative decision making behavior, or more specifically is based on model predictive control (MPC) principles. The MPC process generally contains the following steps, 1) understand the status of the system at the current time step, 2) have a model to predict future performance of the system over an upcoming time horizon, 3) update key variables such that the predicted performance approaches the intended behavior, 4) repeat this process at the next time interval with a new time horizon (*17*). An early survey of MPC techniques is available in Garda et al. (*17*).

In reviewing the literature relevant, several methods, including optimization, simulation, and machine learning, address portions of our research question, however none capture all of our elements. Common across all identified research however is an iterative mechanism which approximates MPC principles. A summary of the relevant methods and characteristics is shown in Table 1.

The optimization studies from Table 1 generally employ a repetitive process where the optimal algorithm is resolved periodically and can incorporate updated vehicle and parking information. Also common across most studies is an objective function which minimizes total travel distance which can include vehicle driving/cruising and walking time.

The studies which most closely apply MPC principles include Zhao et al. (*18*), Mladenović et al. (*19*), and Hakeem et al. (*20*) primarily due to their models' ability continually update parking assignment prior to arrival. Zhao et al. (*18*) develops the D2Park: Diversified demand-Aware on-street parking guidance process which combines machine learning-based parking demand prediction methods with an ILP optimization formulation for aligning parking requests with available parking spaces. The authors follow MPC principles by periodically updating parking demand and

Dynamic Parking Reservation Studies
(based on MPC Principles)

Author (Year)	Method	Types of Vehicles	Metrics
Our Paper	O (MILP)*	Delivery, Private	reduction in surrounding vehicle travel time due to double parking and cruising
Zhao (2020)	ML* + O (ILP)*	Not Specified	extra driving time (cruising), walking time, the sum of which is delay time
Mladenovic (2021)	O (ILP)*	Not Specified	driving time, walking distance, parking request fulfillment; proxy for cruising duration, utilization of parking lots
Chen (2015)	O (ILP)	Not Specified	average social cost (weighted combo of cruising time and walking time), average revenue
Mejri (2016)	O (ILP) + SA	Not Specified (SUMO-based)	parking space occupancy rate; request fulfillment; walking distance
Hakeem (2016)	O (ILP)*	Not Specified (SUMO-based)	total travel time (driving and walking)
Errousso (2021)	O (ILP) + GA	Delivery, Private	total walking distance; number of fulfilled requests
Letnik (2018)	DES	Delivery	total travel; CO2 emissions; fuel consumption
Carvalho (2017)	DES	Private	total system travel time
Comi (2018)	DES	Delivery	time in queue, number of vehicles in the queue
Zhang (2022)	ML	Working, Shopping, Visiting	total travel time (cruising and walking), parking lot occupancy

*closely aligns with MPC principles
O = Optimization
MILP = Mixed Integer Linear Program
ML = Machine Learning
ILP = Integer Linear Program
SA = Simulated Annealing
GA = Genetic Algorithm
DES = Discrete Event Simulation

TABLE 1: A summary of studies which explored dynamic parking reservation systems across optimization, simulation, and machine learning methods.

subsequently reassessing guidance for vehicles throughout the scenario. Zhao et al. (*18*) most notably observe that the average total delay time (cruising + walking time) with D2Park decreases by 33% when compared with a "No Guidance" greedy parking space search algorithm. Mladenović et al. (*19*) also closely adheres to MPC principles with their "Dynamic Parking Allocation Problem (DPAP)" formulation which sequentially executes an integer programming model (*21*) to achieve near real-time updates for parking space assignment. The integer program seeks to minimize total travel time (driving and walking time) given a set of parking requests with initial positions and desired destinations, but the authors do not consider this to be a reservation based formulation. Instead, the authors maintain the ability to update vehicle to parking space assignment up to vehicle arrival which may change based on new parking requests and updated vehicle positions. The vehicles are then formally assigned a parking space when they arrive to their allocated location. Hakeem et al. (*20*) developed the Free Parking System (FPS) which implements an integer programming optimization model to assign vehicles to geographically disperse parking spaces while minimizing the total travel time for all of the vehicles including driving and walking time. Similar to Mladenović et al. (*19*) this model also incorporates a minimum distance between the vehicle an its destination before a parking space is assigned. This allows for relevant parking requests to be considered in the optimization algorithm along with the most up to date parking availability information for the vehicle approaching its destination.

Several additional optimization studies also iteratively solve vehicle parking assignments based on updated vehicle and parking information, but tend to lock in reservations early and are less flexible at future time steps. Mejri et al. (*22*) develop the Reservation-based multi-Objective SmArt Parking (ROSAP) process which solves an integer linear program and is aided by a simulated annealing heuristic. Their multi-objective model considers walking distance, travel distance to end destination at the time of the request and a parking congestion impact variable based on parking demand of the destination. They find that ROSAP results in higher parking space occupancy and a larger percentage of requests fulfilled than a greedy method at the expense of slightly longer walking distances. Errousso et al. (*23*) and Errousso et al. (*24*) describe different iterative approaches to assigning vehicles to parking spaces. Errousso et al. (*23*) applies a fuzzy logic method composed of two phases which determine vehicle assignment and a third deconfliction phase. Errousso et al. (*24*) leverage the parking space allocation based on demand phase from Errousso et al. (*23*) and develops two integer linear programming models for specific vehicle assignment. Errousso et al. (*24*) also has many unique considerations including, the size the vehicle and parking space, the amount of cargo being transported, a method to differentiate private and commercial vehicle parking requests, and the probability that an assigned parking space will be available upon arrival. Both studies show a reduction in total walking distance and an increase in the quantity of fulfilled parking requests based on 10 or 15 minutes between executing the assignment algorithm. Additionally, Chen et al. (*25*) iteratively resolve their optimization formulation with an objective of minimizing total travel time which includes driving and walking distance. Instead of specifying a specific time between each optimization run however, the authors focus on the total number of re-optimizations over the scenario and conduct numerical experiments between 1 and 100 intervals. They find the total travel time is generally less than a first-come first-serve scenario with their model and that the extent of the improvement is dependent on the number of time intervals considered.

Simulation methods are often structured around discrete events, such as a vehicle submitting a parking request, which enables an iterative, dynamic and "MPC-like" process of assigning parking spaces given the opportunity to incorporate updated parking availability information. However, if the parking assignment is made as soon as the request is received, then the simulation cannot chose between multiple new requests to select the best option. Optimization models however can collect new incoming requests over a period of time before making assignments. For example, Comi et al. (*26*) and Comi et al. (*27*) discusses an intelligently managed parking system for delivery vehicles (DynaLOAD) which includes a reservation process. The authors implement their system as a discrete event simulation where each new parking request is assessed and assigned a parking space based on the current scenario assignment paradigm, e.g. drive to the nearest parking space and if the space is occupied, go to the next closest space, etc. The authors find that their system can reduce the number of vehicles waiting for parking and their time in the queue. The authors also consider the queues that form while waiting for parking and double parked vehicles, but they do not address the impact of the surrounding resulting traffic congestion. Letnik et al. (*28*) also creates a simulation which first optimizes the location of loading bays based on delivery destinations with fuzzy k-means clustering and then assigns delivery vehicles to either the most convenient bay or from a set of alternative bays if the best bay is occupied. This can be considered a dynamic reservation-based model because parking space availability is updated when the parking requests are processed and under certain paradigms, the system can choose from set of possible parking locations. The output metrics of the model focus on delivery vehicle parameters such as total travel time and distance and resulting CO2 emissions and fuel consumption along with total waiting time outside of the city until a parking space is available. The authors do not model for the congestion impact of double parked vehicles or vehicles cruising in search of a parking space. Carvalho e Ferreira and de Abreu e Silva (*29*) develops a different discrete event simulation to assess the impact of a parking reservation system on total system travel time while incorporating cruising vehicle behavior. Based on empirical observations, they model 10% of the traffic population as cruising for parking, where vehicles may find a space quickly or cruise for a randomly sampled duration. Combining the cruising behavior with vehicles passing through and those with reservations creates a queue of vehicles moving through the area which contributes to total system travel time. The authors find that in their best case scenario, they are able to reduce total travel time by 3%.

A third method to consider includes machine learning, specifically a deep reinforcement learning algorithm as described by Zhang et al. (*30*). This algorithm rewards accurate parking assignment decisions for connected vehicles which use the reservation system while also mitigating for non-connected vehicles which may unexpectedly occupy an assigned parking space. The algorithm is compared to several machine learning structure variants along with traditional first-come first-serve, day ahead optimal, and periodic optimal parking assignment paradigms. The authors see improvements in reducing the total travel time, which includes cruising and walking distance as the number of connected vehicles increases. Additionally, the authors study parking lot occupancy as another metrics to assess algorithm performance.

METHODS

Our research develops a model predictive control (MPC) algorithm which implements a mixed integer linear program (MILP) schedule optimization formulation to assign vehicles to curb parking spaces. The MPC algorithm processes parking requests periodically throughout the scenario such that only a subset of the scenario parking requests within the current time horizon are considered for schedule optimization. The subset of parking requests are then input into an MILP formulation known as the Parking Slot Assignment Problem (PAP) which is originally derived in Roca-Riu et al. (*31*) adapted and modified in Burns et al. (*12*) and further modified in this research. This process is repeated at predetermined time steps throughout the scenario and in aggregate creates a parking schedule which contains the vehicles which were assigned parking spaces and their start time, parking duration, and departure time. The MPC algorithm and optimization formulation are discussed in more detail in the subsequent sections.

Baseline Models

Two baseline models of parking space assignment, first-come first-serve (FCFS) and full day optimal, are considered in this research for comparison with our MPC algorithm. Central to all models is a set of vehicle parking requests where vehicles are indexed by i, and each vehicle has a requested arrival time window from a_i to b_i, a parking duration $s_i > 0$ and a latest departure time $d_i = b_i + s_i$. Our first-come first-serve model represents a traditional parking space paradigm where vehicles arrive to parking spaces in an uncoordinated manner and may quickly saturate available parking spaces in some cases. In the FCFS model only, the actual arrival time, $t_i = a_i \ \forall i$.

The second baseline model, full day optimal, assumes vehicle parking request for the entire scenario are submitted prior to the start of the scenario, e.g. delivery vehicles submit parking request a day ahead of time, and can then be optimized and shifted to accommodate the best combination of vehicles. In this model, t_i may not necessarily be equal to $a_i \ \forall i$. The traffic congestion benefits of the full day optimal model as compared with first-come first-serve is explored in Burns et al. (*12*).

Model Predictive Control (MPC) Algorithm

The model predictive control (MPC) algorithm is generally defined by two parameters, τ and ζ, where τ is length of the future time horizon under consideration and ζ is the time step between the start of each time horizon. The MPC algorithm iterates through the scenario in ζ step increments of time which in one example could be every 5 minutes or some other amount of time. The time horizon defines which vehicle parking requests should be considered for schedule optimization where example lengths may be 30 minutes or some other time period. In our specific implementation, we also create a lead time variable, ρ_i, which defines the earliest time a vehicle can be considered in the optimization and is a proxy for the maximum length of time horizon. ρ_i can be heterogeneous for every vehicle, but we keep the variable homogeneous across vehicle types, e.g. a 30 minutes lead time for all vehicle requests. Also important to the MPC algorithm is the parking request received time, r_i. In order for a request to be considered in the optimization subset both, r_i and ρ_i must be $\leq$ the current time. An example of the MPC algorithm with vehicles assigned to different time horizons in shown in Figure 2.

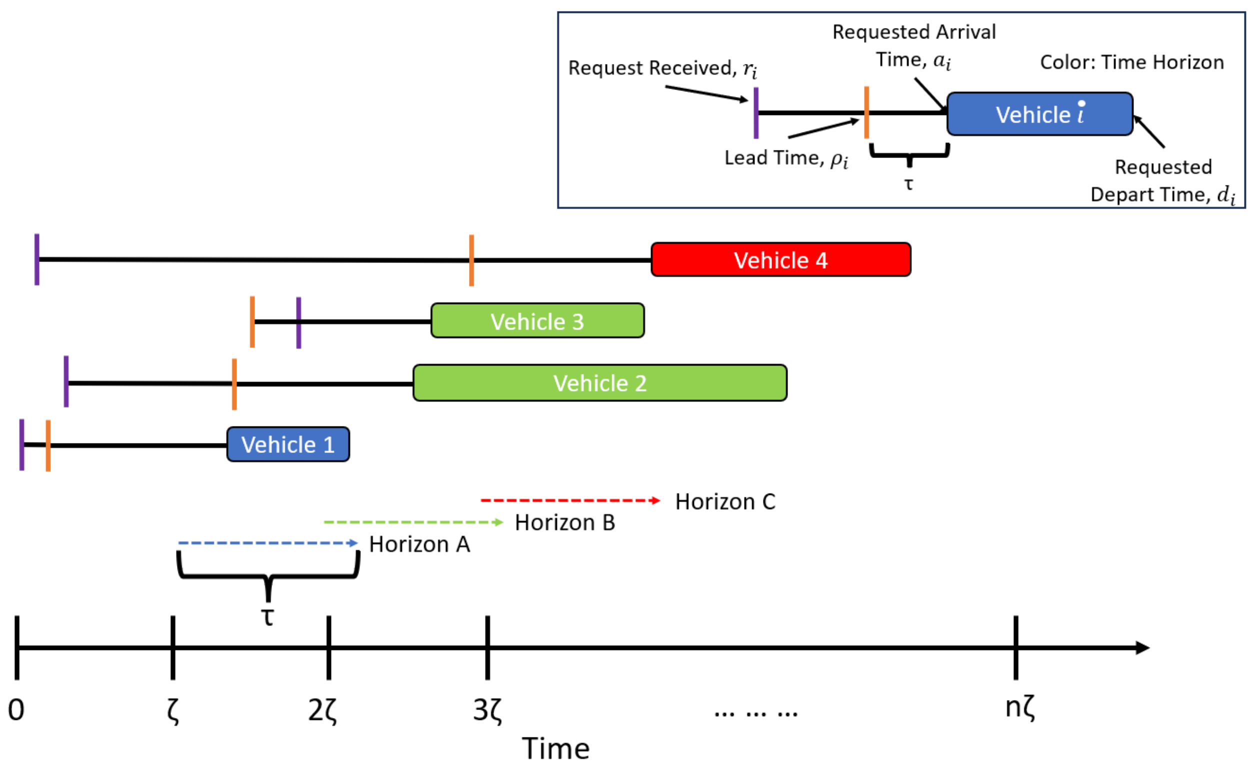


FIGURE 2: A visual representation of the ζ and τ parameters in the Model Predictive Control algorithm where ζ is the time step between the start of each time horizon and τ is the length of the horizon.

Parking Slot Assignment Optimization Formulation

After the subset of parking requests has been determined in the MPC algorithm, the parking request are input into our mixed integer linear program (MILP) Parking Slot Assignment Optimization formulation. Our formulation was initially developed by Roca-Riu et al. (*31*), adapted and modified in Burns et al. (*12*) and further updated in our current research as shown below.

$$\underset{\mathbf{x}_w}{\text{minimize}} \sum_{k \in \mathscr{K}} \omega_k f_k(\mathbf{x}_w) \qquad \text{Minimize double parking, cruising in planning horizon } w \tag{1}$$

subject to

$$\sum_{j \in \mathscr{V}} x_{0jw} \leq n, \forall w \in \mathscr{W} \qquad \text{\# parallel vehicle parking sequences} \leq \text{parking capacity} \tag{2}$$

$$\begin{cases} x_{ijw} = 0 \quad \forall (i,j) \in \mathscr{A}, \forall w \in \mathscr{W} & \max r_i, a_i - \rho_i \leq (t_{w\text{START}}), \\ \sum_{j \in \mathscr{A}} x_{ijw} \leq 1 \quad \forall i \in \mathscr{V}, \forall w \in \mathscr{W} & \text{Otherwise,} \end{cases} \qquad \text{Vehicle may be considered if in current horizon} \tag{3}$$

$$\sum_{(i,j) \in \mathscr{A}} x_{ijw} - \sum_{(j,i) \in \mathscr{A}} x_{jiw} = 0, \quad \forall i \in \mathscr{V}, \forall w \in \mathscr{W} \qquad \text{Assigned vehicles are part of a sequence} \tag{4}$$

$$t_j \geq t_i + s_i + \psi - (1 - x_{ijw})M, \;\; i, j \in \mathscr{A} \qquad \text{Prevent overlap in sequential vehicle arrivals} \tag{5}$$

$$t_i \geq t_{0i}, \quad \forall i \in \mathscr{V} \qquad \text{Each arrival is not before its requested start time} \tag{6}$$

$$t_i \leq t_{0i} + b_i, \quad \forall i \in \mathscr{V} \qquad \text{Each arrival is not after its latest permitted start time} \tag{7}$$

$$t_i + s_i \leq t_{\text{END}}, \quad \forall i \in \mathscr{V} \qquad \text{Vehicle must complete service before end of scenario} \tag{8}$$

$$\sum_{j \in \mathscr{V}} x_{ijw} = \sum_{j \in \mathscr{V}} x_{ij,w-1}, \quad (i,j) \in \mathscr{A} \qquad \text{Decisions persist across time windows considered} \tag{9}$$

where

$$\mathbf{x}_w = [t_i \; \forall i \in \mathscr{V}, x_{ijw} \; \forall (i,j) \in \mathscr{A}]^\top \qquad \text{Decide vehicles to schedule and arrival times} \tag{10}$$

$$t_i \in \mathbb{R}, \quad \forall i \in \mathscr{V} \qquad \text{Scheduled arrival time for each vehicle} \tag{11}$$

$$x_{ijw} \in \{0,1\}, \quad (i,j) \in \mathscr{A}, \quad w \in \mathscr{W} \qquad \text{1 iff vehicle } j \text{ follows } i \text{ in parking sequence} \tag{12}$$

$$f_k(x_w) = \sum_i c_{ik} \beta_i \qquad \text{Dbl. parking or cruising for each unscheduled vehicle} \tag{13}$$

$$\beta_i = \left(1 - \min(1, \sum_{w \in \mathscr{W}} \sum_{(i,j) \in \mathscr{A}} x_{ijw})\right) \quad \forall i \in \mathscr{V} \quad \text{0 if vehicle } i \text{ is assigned a space, 1 otherwise} \tag{14}$$

$$t_{0i} = a_i, \quad \forall i \in \mathscr{V} \qquad \text{Arrival time for unscheduled vehicles (for efficiency)} \tag{15}$$

$$M = \text{Arbitrarily large value} \qquad \text{Implement a “Big M” disjunctive constraint} \tag{16}$$

$$t_{w\text{END}} = t_{w\text{START}} + \tau \qquad \text{Definition of the time horizon of length } \tau \tag{17}$$

$$t_{w\text{START}} = t_{w-1,\text{START}} + \zeta \qquad \text{Shift time frames across optimization windows} \tag{18}$$

where $i \in \mathscr{V}$ indexes the set of vehicles (and a "depot" node $i = 0$ that serves as the start and end node of each parking sequence), $(i, j) \in \mathscr{A}$ is the set of arcs in a directed network, $w \in \mathscr{W}$ is the current time horizon in the set of horizons for the scenario, x_{ijw} is a binary decision variable that is equal to 1 if vehicle i is scheduled immediately before vehicle j in the same parking spot in the current time horizon (0 otherwise), ω_k is the weight of objective function $k \in \mathscr{K}$, f_k is the kth function given vehicle assignments β, C_{ik} is the cost associated with either double parking or cruising for the ith vehicle, parking durations s_i and the expected value that vehicle i cruises E[R], or double parks E[D] if not assigned a parking space,[2] t_i is a continuous decision variable that represents the scheduled start time of vehicle i, t_{0i} is the arrival time value used for vehicles that are not assigned (imposed to improve efficiency), t_{wSTART} and t_{wEND} is the start and end of the current time horizon, ζ is the time step between time horizons, τ is the length of the time horizon, t_{START} and t_{END} are the start and end of the scenario, n is the number of parking spaces, a_i and b_i are the bounds of the requested arrival time window for vehicle i, ψ is the buffer enforced between parking reservations, r_i and ρ_i are defined as the request received time and the lead time before arrival at which a request can be assigned a parking space, and M is a "Big M" representation of a disjunctive constraint using an arbitrarily large number.[3]

Datasets

Our optimization formulation requires a set of vehicle parking requests which populate $\mathscr{V}$, where each request contains the vehicle's requested arrival time, a_i, service duration, s_i, departure time, d_i, and the time the request was received r_i. In this research we model private passenger vehicles and freight vehicles, both of which have different types of parking requests. Our private passenger vehicle data is sampled from the National Household Transportation Survey (NHTS) which contains a diverse set of vehicle trips collected in 2017 US Department of Transportation and Federal Highway Administration (*32*). The freight delivery vehicle data is derived from commercial company Coord's 2021 pilot program in Aspen, CO Coord (*33*) where delivery drivers were able to reserve specific loading zones for their loading and unloading operations. Arrival time and service duration distributions for NHTS and Aspen, CO data are available in the appendix.

Additionally, in our modelling, the objective function penalty for not assigning a parking space to a vehicle is either double parking for the vehicle's service duration, s_i or cruising for an expected duration of 3 minutes. The expected cruising duration was determine based on our preliminary assessment of observed cruising behaviors from Chiara et al. (*34*), Dalla Chiara and Goodchild (*11*), and Shoup (*7*).

Parameter Settings

We explored a range of parameters in our research as described in Table 2 which resulted in 9,000 total cases. Many of the parameters contained a range of values, while other parameters were fixed.

For each of these cases we generated a parking schedule based on a first-come first-serve paradigm, a full day optimal formulation, and our MPC algorithm. Of note, the full day optimal formulation achieved optimality in 3,487 cases and the MPC algorithm achieved optimality across

[2]Further description of how E[R] and E[D] is estimated is available in the appendix.

[3]It should be noted that the implementation of the proposed formulation is slightly different but mathematically consistent. The implementation removes certain trip requests from the formulation after they are no longer relevant to the algorithm's decision-making process. This involves a transformation of certain x_ijw values, but reduces overhead when constructing the optimization problem.

Study Parameters*

Parameters	Values
Vehicles per Parking Space per Hour	[1, 2, 3]
Number of Parking Spaces, n	[1, 10, 20, 30, 40, 50]
Weight Double Parking**	[0, 0.25, 0.5, 0.75, 1]
Proportion Delivery Vehicles***	[0, 0.25, 0.5, 0.75, 1]
Time Step, ζ	[2, 5] minutes
Lead Time, ρ	30 minutes
Buffer, ψ	[1, 6] minutes

*Full factorial exploration replicated 5x, total of 9,000 cases
**Weight Cruising = (1 - Weight Double Parking)
***At least one delivery and one passenger vehicle is present in each test case

TABLE 2: Key parameters and range of values explored

all iterations in 8,934 cases.

PRELIMINARY RESULTS

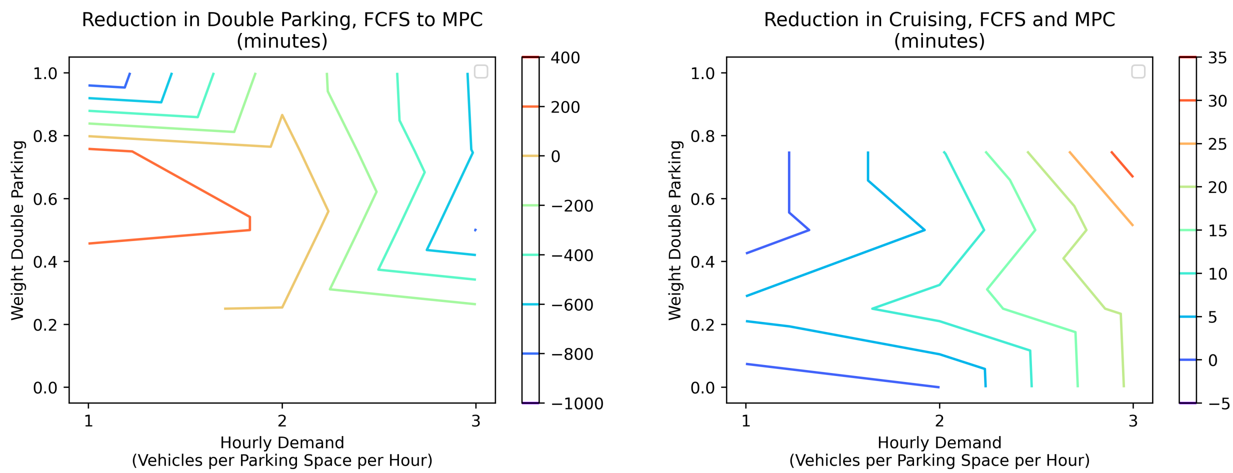
The below preliminary results were generated using a Mac Studio with a 24-core CPU M2 Ultra and 64GB of RAM running Python 3.10.11 and Gurobi 10.0.2.

Efficiency Improvements from a Dynamic Optimal Curbspace Reservation Management System

To assess the effectiveness of the MPC algorithm, we collected the total minutes of double parking and vehicle cruising from each parking management system paradigm. From these data, we explored the reduction in both metrics when switching from a first-come first-serve (FCFS) to MPC model across our set of parameters. The preliminary results shown in Figure 3 describe the average[4] reduction metrics with respect to the weight of double parking in the objective function and the hourly demand for parking in terms of the average number of vehicles requesting parking per parking space per hour. Of note, the optimization objective function assigns weights to both double parking and cruising penalties where the weights must sum to one. Also of note, Figure 3 does not include data where the objective function receives a zero weight for either double parking or cruising, e.g. the double parking graphic only contains optimization schedules which were generated from an objective function with a weight for double parking greater than zero, i.e. weight equal to [0.25, 0.5, 0.75, 1].

We preliminarily observe different regions and magnitudes of performance for reducing double parking and cruising. Double parking appears to reduced the most with an equally weight objective function and lower parking demand. There are also regions on Figure 3a with a negative reduction in double parking which indicates that first-come first-serve has less double parking than the MPC system. This is likely due to the correlations present between request lead and service time among passenger vehicles. In appendix, we see that shorter lead times tend to be

[4]results are averaged across all parameters described in Table 2 which are not included in the primary axes, e.g. average over all parking space cases.



(a) (b)

FIGURE 3: Reduction in double parking and cruising with respect to the objective function weight assigned to double parking penalties and the average hourly demand of vehicles for parking space. Of note, the data shown only contains simulation schedules which provides a weight greater than zero for either double parking or cruising, e.g. the reduction in double parking graphic only contains preliminary results with weights [0.25, 0.5, 0.75, 1].

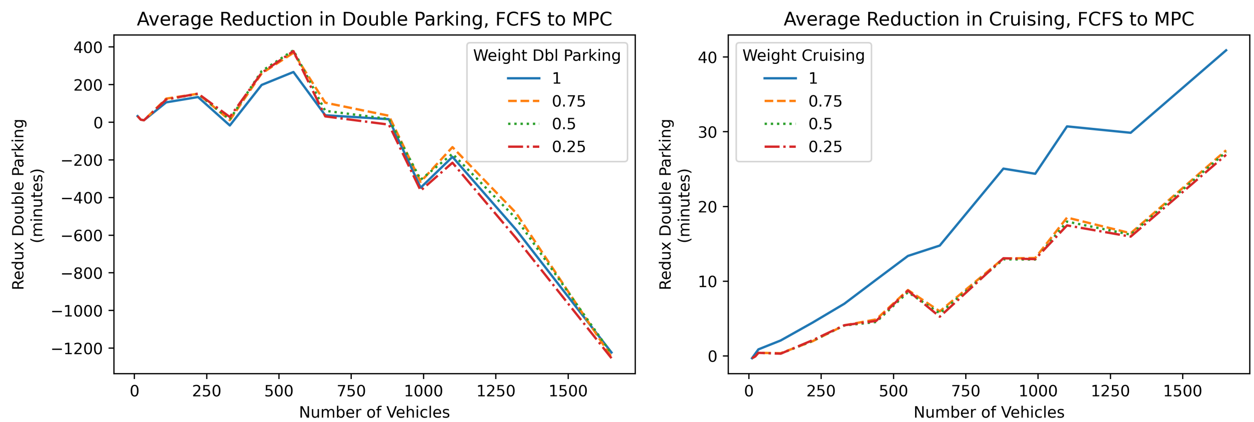
correlated with longer dwell times. In turn, as the algorithm iterates through the scenario it is more likely to receive shorter dwell time requests before longer dwell time requests. As parking decisions are required to be consistent over time, this could lead to reduced performance for the MPC algorithm. We plan to explore smaller lead time values, which should alter this behavior, in our future work. For cruising, we also preliminarily observe variations in performance with larger reductions given higher parking demand. Unlike double parking however, we do not observe relatively large negative reduction values.

Additionally, we explored the reduction in double parking and cruising when differentiated by objective function weights as shown in Figure 4. We preliminarily observe similar behavior to Figure 3, where the reduction in double parking can be positive, but does have regions of large negative performance and the reduction in cruising appears to be primarily positive. Also of note, in Figure 4a, there appears to be a homogeneous impact on double parking across vehicle cases. In Figure 4b, however, assigning a 100% weight to cruising appears to result in heterogeneous performance. It is unclear what is causing this difference between double parking and cruising at this time, but this will be part of our future research.

Runtime Comparison of Full Day Optimization Formulation and the MPC Algorithm

From each of the cases explored, we recorded the runtime of the parking schedule generated from the full day optimization formulation and the MPC algorithm. Figure 5 displays a preliminary set of boxplots of the runtime for each algorithm in seconds by the average number of vehicles requesting a parking space per hour with an optimization solver time limit of 10 minutes. Two cases of the MPC were explored with different values for ζ, the time step between each time horizon and re-optimization.

Preliminary results based on the median and interquartile range indicate that the MPC al-



(a) (b)

FIGURE 4: Preliminary observations of reduction in double parking and cruising minutes when comparing a first-come first-serve paradigm to an MPC system across the number of vehicles in the scenario and differentiated by objective function weight. The reduction in double parking appears to have regions of positive and negative reduction whereas the reduction in cruising appears to be positive.

gorithm is faster across the cases we examined. While there are instances of the full day schedule performing quickly and on par or better with MPC algorithm, the full day optimization runtime tends to increase rapidly with increasing hourly demand. Additionally, many of the full day optimization cases hit the runtime limit of 10 minutes without converging to a solution, indicating even longer actual runtimes than shown in Figure 5.

Of note, some runtimes exceed 10 minutes (600 seconds) which is due to two factors. First, as implemented in our simulation, the 10 minute runtime limit is only applied to the optimization solver and does not include any time required for additional computational overhead, but this additional time is included in the runtime calculator. Additionally, the MPC algorithm executes the optimization formulation multiple times, but the limit is only applied to a single implementation which allows the sum of the runtime across MPC iterations to exceed 10 minutes in some cases.

CONCLUSIONS

This work proposed a model-predictive control influenced optimization formulation that allows for the determination of optimal parking assignments across a given scenario. We implement and test this formulation based on real-world data that characterizes the behavior of both delivery and private passenger vehicles. Generally, we preliminarily find that the MPC algorithm often underperforms in reducing double-parking relative to an intuitive base case that is first-come first-serve parking assignment. However, this same algorithm is able to reduce the amount of cruising that would otherwise occur in in a first-come first-serve scenario. Additionally, we see that the MPC algorithm enables both faster solving across a wide variety of scenarios and makes many problems computationally tractable. These preliminary findings provide potential insight into the contexts in which the proposed algorithm could be of most value to stakeholders such as city governments, local citizens, and private business.

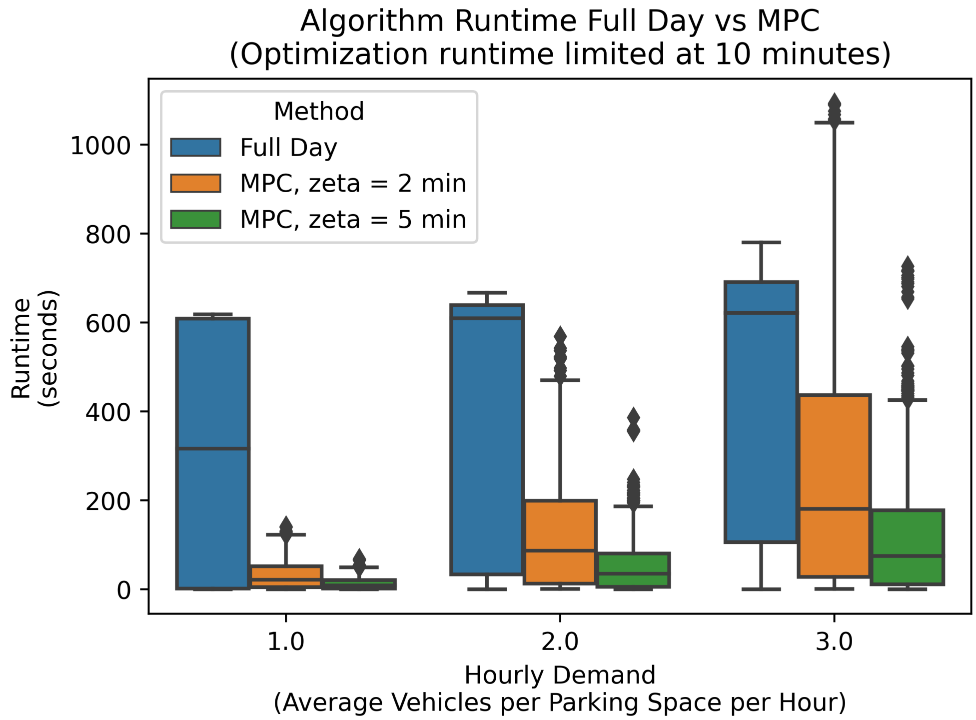


FIGURE 5: Runtime comparison between the full day optimization formulation and MPC algorithm. Preliminary results show that the full day schedule can resolve quickly, however the MPC algorithm generally tends to have faster runtime across the average number of vehicles requesting parking per parking space per hour. Of note, the optimization solver runtime was limited in each case to 10 minutes (600 seconds), however this did not include additional time required to store variables in memory or the sum of the successive optimization iterations in the MPC algorithm. As a result, runtime observations can exceed the 10 minute threshold.

FUTURE WORK

Our future research intends to further characterize the performance of the MPC algorithm relative to the full day optimal and first-come first-serve parking schedules. This includes additional parameter exploration of the reservation lead time, ρ, objective weighting, and an increased resolution and range of the hourly parking demand per parking space. We also hope to explore larger parking space cases which we believe will be uniquely tractable with the MPC algorithm. Further advancements in our modeling may also include parking requests from ride-hailing vehicles along with different requests for freight and parcel vehicles.

ACKNOWLEDGEMENTS

This project is funded in part by Carnegie Mellon University's Mobility21 National University Transportation Center, which is sponsored by the US Department of Transportation. The funding is associated with agreement #397. Thank you to Professor Sean Qian for his guidance on parking behaviors and impacts on the surrounding environment.

APPENDIX

NHTS Distributions

One of the primary datasets used in the construction of parking requests is the 2017 National Household Travel Survey (NHTS) US Department of Transportation and Federal Highway Administration (*32*). NHTS contains travel diaries for a nationally representative set of Americans. Although the dataset is nationally representative, we use a particular subset of data relevant to our context of interest. We leverage data for trips that are made in urban areas, to non-home or primary work locations, by light-duty passenger vehicles or rental vehicles, and the entirety of the trip is made during the hours of 7AM to 6PM. This dataset contains information on several aspects of any given trip such as travel time, dwell time, reason for travel, and many others. In this context, we pay particular attention to both travel times and dwell times. Specifically, we use travel times as a proxy for the time before arrival that a parking request is made, and dwell time serves as a proxy for parking request length. A two-way histogram showing the relationship between travel and dwell time is shown in Figure 6.

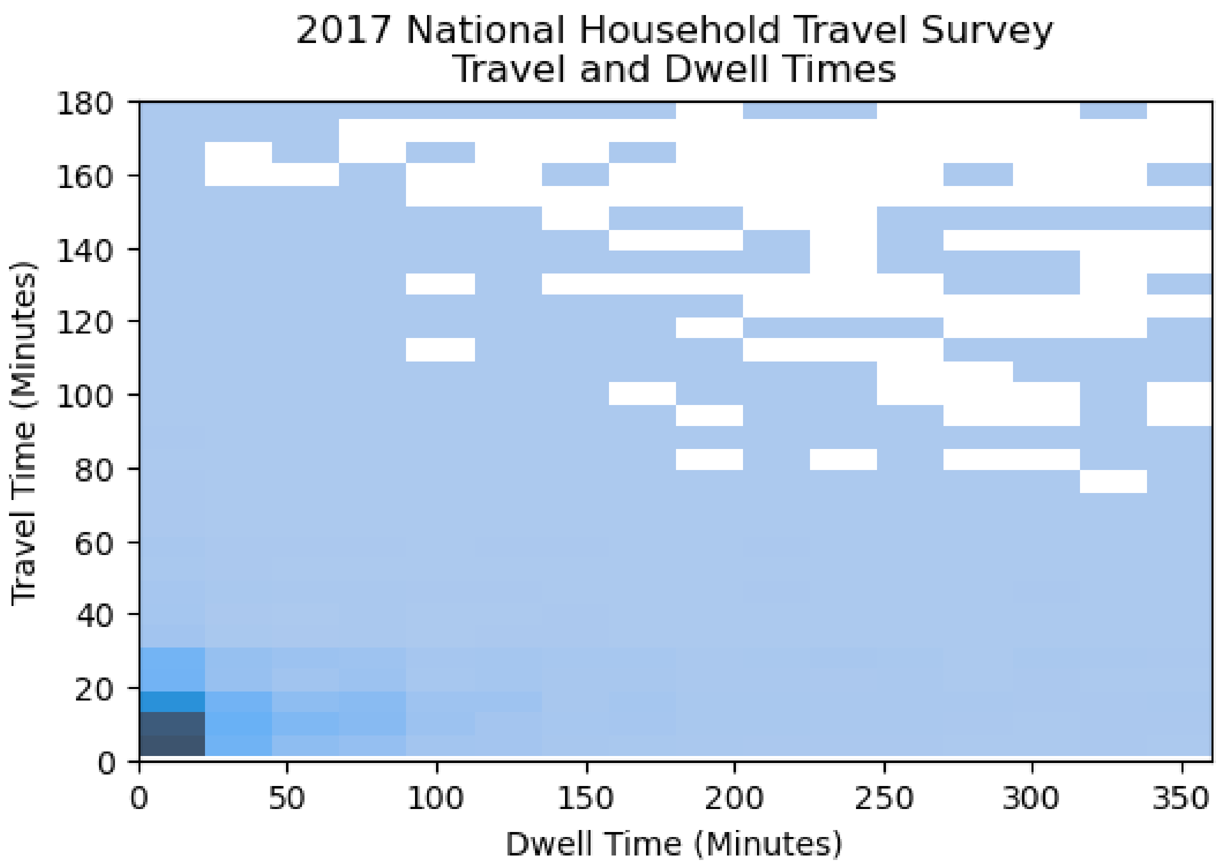


FIGURE 6: Two-way histogram showing the bivariate distribution of travel and dwell times US Department of Transportation and Federal Highway Administration (*32*). Note, axes do not cover full range of data to improve interpretability.

The NHTS dataset is leverage as an empirical distribution used to construct instantiations of a given parking day. Specifically, each scenario (or day) that the algorithm optimizes is constructed of a given number of parking requests. Those parking requests are constructed by random sampling with replacement of the relevant NHTS trip dataset.

Aspen Distributions

The Aspen, CO dataset is derived from empirical data collected by commercial company Coord over the first 100 days of their Smart Zone pilot program in Aspen, CO, from November 2020 through January 2021 (*33*). The pilot program implemented a loading zone reservation system for a set of curbside and alley commercial loading zones and supported 1520 parking sessions. Using the Aspen dataset, which provides data at an hourly resolution, we determined the set of requested vehicle arrival times a by sampling the empirical frequency distribution for vehicle arrivals by hour and adding a random value uniformly drawn from between 0 and 60 minutes. The empirical distribution is shown in Figure 7.

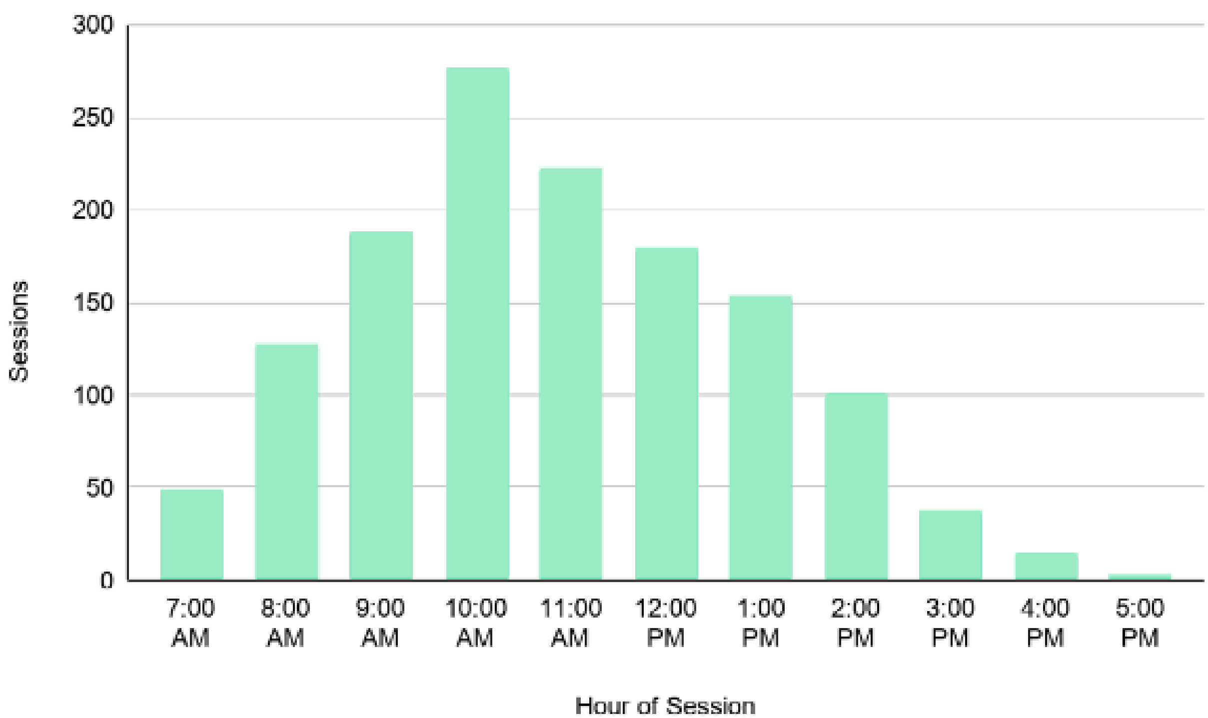


FIGURE 7: The distribution of freight vehicle arrivals from Coord's reservable loading zone pilot program in Aspen, CO (*33*).

For service duration s, we sampled from a normal distribution with a heterogeneous mean parameter dependent on the hour of the vehicle arrival as provided by Coord (*33*) and shown in Figure 8.

We are, however, missing data on the variance associated with each of the average service duration data points, so we follow (*31*) in assuming a standard deviation of five minutes for each hour (based on empirical observations of Barcelona, Spain in 1997). Even though this is an older data source, we believe that five minutes is a reasonable estimate that can be explored through future sensitivity analysis. Of note, a_i, s_i are each rounded to the nearest integer for easier interpretation of the optimization results.

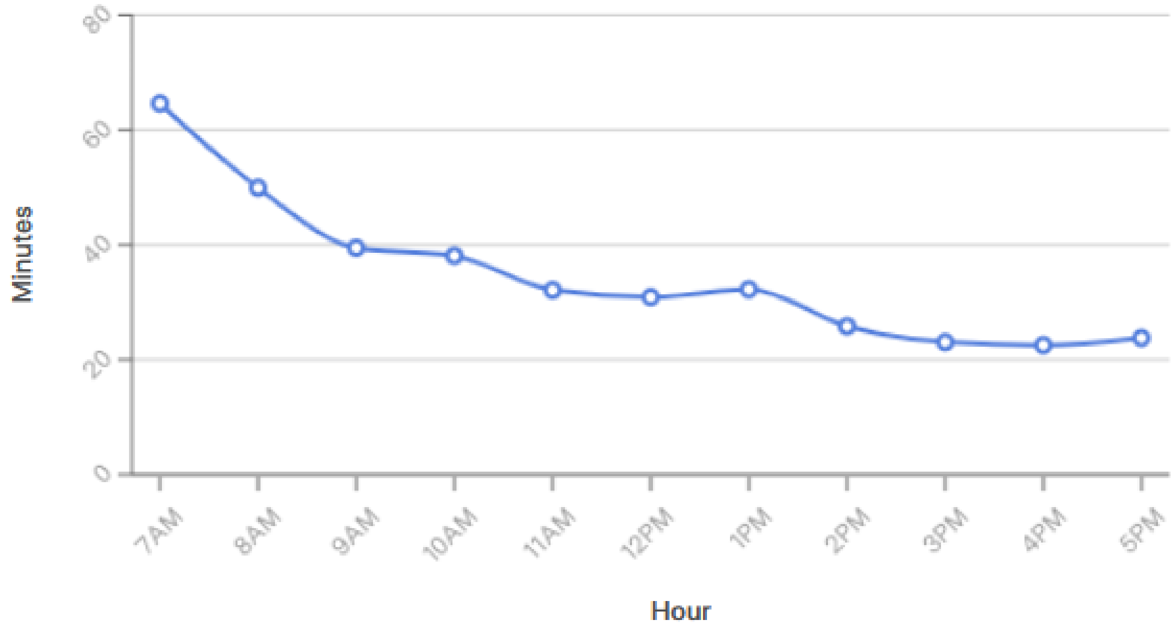


FIGURE 8: The distribution of freight vehicle parking duration from Coord's reservable loading zone pilot program in Aspen, CO (*33*).

Street-Sense Data

For our research we also leverage data collected by Street Sense Inc. and their camera monitoring system on Liberty Ave in downtown Pittsburgh from September to November of 2021. Over this period, Street Sense Inc. used a machine learning image classification system to characterize more than 6500 cars and 500 trucks both parked legally in parking spaces and illegally double parked in a lane of traffic. Each parking event is described by the vehicle arrival and departure time and whether or not the vehicle was double parked in a lane of traffic or legally parked in a parking space.

Using these data, we are able to construct a simple method of estimating the probability that an individual vehicle will double-park or cruise when a parking spot is not available to them upon arrival. We construct our estimate of double-parking probability using the number of vehicles who have double parked for a given amount of time and in total. Equation (19) shows the specific formulation. Intuitively, the probability that a vehicle with a parking request t seconds long will double park is equal to the proportion of vehicles double parked at least as long as t seconds among all double-parked vehicles. We then assume that the only other alternative is cruising, so the probability of cruising is equal to one minus the probability to double park.

$$\text{P(Double parking with service duration } t) = \frac{\text{\# of Vehicles Double Parked for Duration} \geq t}{\text{Total \# of Double Parked Vehicles}} \tag{19}$$

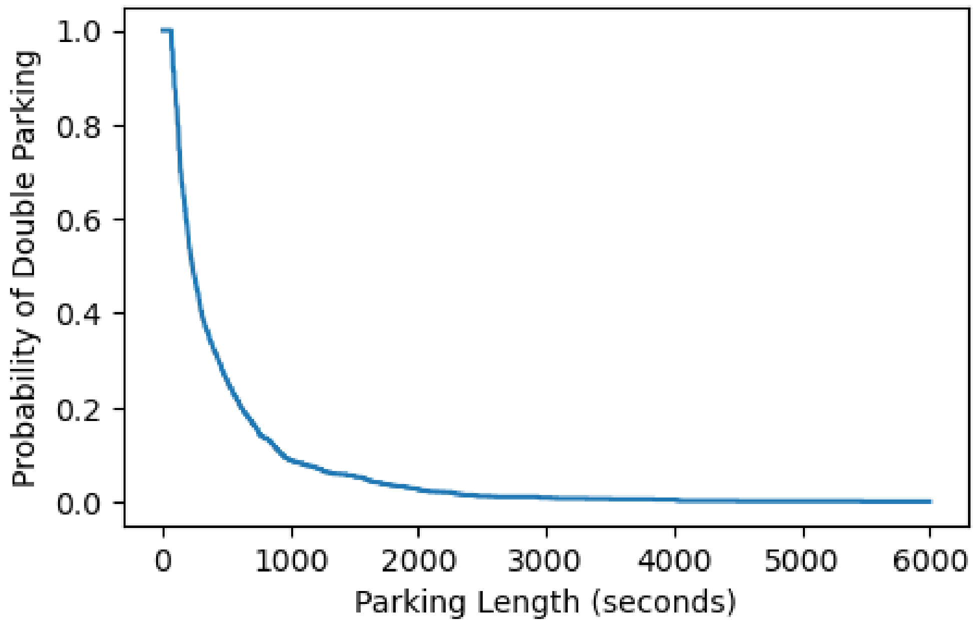


FIGURE 9: Double parking probability values with respect to dwell time.

REFERENCES

1. Surico, J., *Can Technology Help Cities Manage Curbs Better?* New York, 2022.
2. The City of Pittsburgh, *Curbside Management Toolkit*, 2023.
3. Chang, K., A. Goodchild, A. Ranjbari, and E. Mccormack, *MANAGING INCREASING DEMAND FOR CURB SPACE IN THE CITY OF THE FUTURE FINAL*, 2022.
4. Jaller, M., C. Rodier, and M. Zhang, *Fighting for Curb Space: Parking, Ride-Hailing, Urban Freight Deliveries, and Other Users*, 2021.
5. Nichols, C. M. and J. Dorsett, *Poor Curb Management Is Costing Cities Billions*, 2021.
6. Jaller, M., *Curbside Management Is Critical for Minimizing Emissions and Congestion*. UC David, 2021.
7. Shoup, D., Pricing curb parking. *Transportation Research Part A: Policy and Practice*, Vol. 154, 2021, pp. 399–412.
8. Dalla Chiara, G., K. F. Krutein, A. Ranjbari, and A. Goodchild, Providing curb availability information to delivery drivers reduces cruising for parking. *Scientific Reports*, Vol. 12, No. 1, 2022.
9. Hampshire, R. and D. Shoup, How much traffic is cruising for parking? *Transfers Magazine*, 2019.
10. Van Ommeren, J. N., D. Wentink, and P. Rietveld, Empirical evidence on cruising for parking. *Transportation Research Part A: Policy and Practice*, Vol. 46, No. 1, 2012, pp. 123–130.
11. Dalla Chiara, G. and A. Goodchild, Do commercial vehicles cruise for parking? Empirical evidence from Seattle. *Transport Policy*, Vol. 97, 2020, pp. 26–36.
12. Burns, A. J., J. J. Michalek, and C. Samaras, *RESERVING THE CURB: OPTIMIZED*

CURB MANAGEMENT CAN REDUCE DELIVERY VEHICLE DOUBLE PARKING, TRAFFIC CONGESTION AND ENERGY CONSUMPTION, 2023.

13. Kladeftiras, M. and C. Antoniou, Simulation-based assessment of double-parking impacts on traffic and environmental conditions. *Transportation Research Record*, Vol. 2390, No. 1, 2013, pp. 121–130.
14. Chrysostomou, K., A. Petrou, G. Aifadopoulou, and M. Morfoulaki, Microsimulation Modelling of the Impacts of Double-Parking Along an Urban Axis. In *Conference on Sustainable Urban Mobility* (Data Analytics: Paving the Way to Sustainable Urban Mobility, ed.), Springer, Cham, 2019, pp. 164–171.
15. Delaitre, L., A new approach to diagnose urban delivery plans. In *International Conference on Computers & Industrial Engineering*, IEEE, 2009.
16. Gao, J. and K. Ozbay, Modeling Double Parking Impacts on Urban Street. *Transportation Research Record*, 2016.
17. Garda, C. E., D. M. Preti't, and M. Morari˜, *Model Predictive Control: Theory and Practice a Survey**, 1989.
18. Zhao, D., Z. Cao, C. Ju, D. Zhang, and H. Ma, D2park: Diversified demand-Aware on-street parking guidance. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, Vol. 4, No. 4, 2020.
19. Mladenović, M., T. Delot, G. Laporte, and C. Wilbaut, A scalable dynamic parking allocation framework. *Computers and Operations Research*, Vol. 125, 2021.
20. Hakeem, A., N. Gehani, X. Ding, R. Curtmola, and C. Borcea, *On-The-Fly Curbside Parking Assignment*. New Jersey Institute of Technology, 2016.
21. Mladenović, M., T. Delot, G. Laporte, and C. Wilbaut, The parking allocation problem for connected vehicles. *Journal of Heuristics*, Vol. 26, No. 3, 2018, pp. 377–399.
22. Mejri, N., M. Ayari, R. Langar, and L. Saidane, Reservation-based multi-objective smart parking approach for smart cities. In *IEEE 2nd International Smart Cities Conference: Improving the Citizens Quality of Life, ISC2 2016 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2016.
23. Errousso, H., J. E. Ouadi, E. A. A. Alaoui, S. Benhadou, and H. Medromi, A hybrid modeling approach for parking assignment in urban areas. *Journal of King Saud University - Computer and Information Sciences*, 2020.
24. Errousso, H., J. El Ouadi, E. A. Abdellaoui Alaoui, and S. Benhadou, Dynamic parking space allocation at urban scale: Problem formulation and resolution. *Journal of King Saud University - Computer and Information Sciences*, 2021.
25. Chen, Z., Y. Yin, F. He, and J. L. Lin, Parking reservation for managing downtown curbside parking. *Transportation Research Record*, Vol. 2498, 2015, pp. 12–18.
26. Comi, A., M. M. Schiraldi, and B. Buttarazzi, Smart urban freight transport: tools for planning and optimising delivery operations. *Simulation Modelling Practice and Theory*, Vol. 88, 2018, pp. 48–61.
27. Comi, A., B. Buttarazzi, M. M. Schiraldi, R. Innarella, M. Varisco, and L. Rosati, DynaLOAD: A simulation framework for planning, managing and controlling urban delivery bays. In *Transportation Research Procedia*, Elsevier B.V., 2017, Vol. 22, pp. 335–344.
28. Letnik, T., A. Farina, M. Mencinger, M. Lupi, and S. Božičnik, Dynamic management of loading bays for energy efficient urban freight deliveries. *Energy*, Vol. 159, 2018, pp. 916–928.

29. Carvalho e Ferreira, D. and J. de Abreu e Silva, Tackling cruising for parking with an online system of curb parking space reservations. *Case Studies on Transport Policy*, Vol. 5, No. 2, 2017, pp. 179–187.
30. Zhang, X., C. Zhao, F. Liao, X. Li, and Y. Du, Online parking assignment in an environment of partially connected vehicles: A multi-agent deep reinforcement learning approach. *Transportation Research Part C: Emerging Technologies*, Vol. 138, 2022.
31. Roca-Riu, M., E. Fernández, and M. Estrada, Parking slot assignment for urban distribution: Models and formulations. *Omega (United Kingdom)*, Vol. 57, 2015, pp. 157–175.
32. US Department of Transportation and Federal Highway Administration, *2017 National Household Travel Survey v1.2*, 2019.
33. Coord, *Smart Zones Deliver Aspen Smart Zone Pilot Program*, 2021.
34. Chiara, G. D., K. F. Krutein, A. Ranjbari, and A. Goodchild, Understanding Urban Commercial Vehicle Driver Behaviors and Decision Making. *Transportation Research Record: Journal of the Transportation Research Board*, 2021.