

Agent-Based Modeling for Transportation Planning: A Method for Estimating Parking Search Time Based on Demand and Supply

Nir Fulman, Itzhak Benenson

Department of Geography and Human Environment,
Porter School of the Environment and Earth Sciences, Tel Aviv University
nirfulma@post.tau.ac.il; bennya@post.tau.ac.il

Abstract

We estimate parking cruising time curves - the probability $P_i(\tau)$ of longer than τ parking search for destination N_i located within an area with heterogeneous demand and supply. To do that, we estimate cruising time curves for an area of homogeneous demand and supply and then average these curves based on (1) a model of parking search behavior established in a serious parking game; and (2) a “Maximally Dense” parking pattern obtained for the case where drivers possess full knowledge of the available parking spots and are able to park at the spot closest to their destination that is vacant at the moment they start searching for parking. We verify the proposed methods by comparing their outcomes to the cruising time curves obtained in an agent-based model of parking search in a city. As a practical example, we construct a map of cruising time for the Israeli city of Bat Yam. We demonstrate that despite low (0.65) overall demand-to-supply ratio in Bat Yam, high demand-to-supply ratio in the center of the city may result in longer than 10 minutes parking search there. We discuss the application of the proposed approach for urban planning.

1 Introduction: Demand-to-Supply Ratio as a Major Determinant of Parking Search Time

Long parking search time is a perpetual problem of every big city, and quantitative estimation of parking cruising time is a long-standing challenge for transportation research. Given only a moment's thought, the inherent reason for this problem is that demand D exceeds supply S and the demand to supply ratio $R = D/S > 1$. A greater level of detail is necessary to estimate parking search time for a designated area, and should include vehicles arrival and departure rates in the area, parking occupation rate, spatial distributions of the departing drivers, and of destinations of the arriving drivers.

The analytical study of cruising for parking can be performed with stochastic or deterministic models [Arnott and Rowse 1999; 2013; Anderson and de Palma, 2004; 2013;

Levy *et al.*, 2013], while simulation modeling makes it possible to estimate parking search time and the distance between a driver's place of overnight parking and destination [Levy *et al.*, 2013; Levy and Benenson, 2015]. Note that simulation models of cruising for parking include car following effects [Levy and Benenson, 2015; Arnott and Williams, 2017], but we are not aware of analytical models that account for this phenomenon.

Importantly, the analytical and simulation approaches result in qualitatively different estimates of cruising time, as dependent on the occupation rate. According to [Levy *et al.*, 2013] the average search time in a homogeneous grid-like city area remains low in analytical models, even when the occupation rate is very high, ca. 98%, whereas simulation studies of cruising time for the same area result in essentially higher estimates starting from ca. 90% occupation (Figure 1).

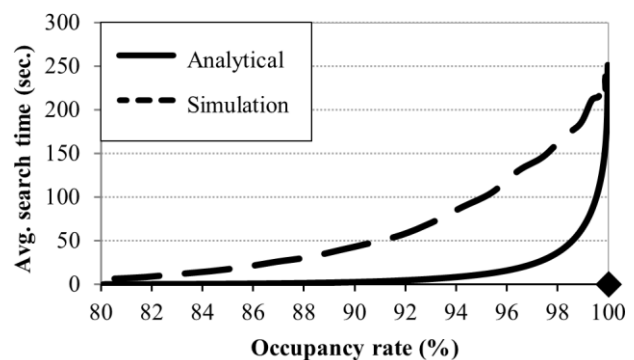


Figure 1: Cruising time as dependent on occupancy rate in analytical and simulation models of [Levy *et al.*, 2013].

According to [Levy *et al.*, 2013], the reason for the gap in Figure 1 is the primarily clustered distribution of vacant parking places that inevitably emerges in a parking model with stochastic arrivals and departures.

High occupation rate and above 100% demand-to-supply ratio are characteristic of the central part of every large city. At the same time, the spatial patterns of demand and supply there are always heterogeneous and the level of heterogeneity is dictated by the city: the demand for parking is defined

by the size and use of the buildings, while the supply is defined by the parking capacity of street links and off-street lots, as well as parking permissions and prices. In this paper we demonstrate that this heterogeneity has far-reaching consequences and local mismatch results in the emergence of essentially larger areas where drivers have to cruise for longer. We investigate this idea in depth with an agent-based model of parking search, and present a fast and efficient algorithm for estimating parking search time based on the patterns of parking demand and supply. The output of the algorithm is a map of cruising time that is validated with the help of the simulation model. As a practical example we construct the map of cruising time for the Israeli city of Bat Yam, with a population of 120,000, and discuss the application of the proposed approach for urban management and planning.

2 The PARKGRID Agent-Based Model of Parking Search

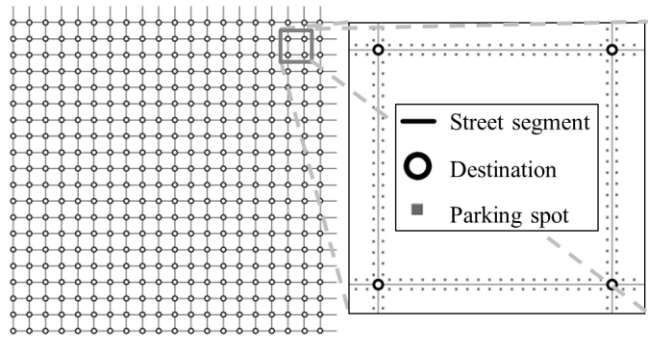
Cruising is the collective outcome of individual drivers' parking search. In what follows we investigate the problem of parking search with the spatially explicit agent-based PARKGRID model that is based on the knowledge of parking search behavior obtained in a serious parking search game [Benenson *et al.*, 2015]. PARKGRID is a stand-alone C# application and can be freely downloaded from https://www.researchgame.net/profile/Nir_Fulman.

PARKGRID continues the tradition of PARKAGENT [Levy *et al.*, 2013; Levy and Benenson, 2015] and is a GIS-based application that is based on the layers of streets, destinations, and parking places. In this paper, we consider an abstract grid city for estimating basic dependencies, and then apply our results to a real city.

2.1 Urban Street Network in PARKGRID

PARKGRID simulates on-street parking in an abstract grid city where the street network is represented by two-way links L_i and junctions N_j (Figure 2). The length of a street link is 100 m. To avoid boundary effects, the grid is folded into a torus - the right ends of its rightmost links in Figure 2a are connected to the leftmost junctions and the top ends of the top links are connected to the junctions at the bottom. In this way each junction has exactly four incident links. For further simplicity, we set drivers' destinations at the junctions.

In the current version of the model, we assume that drivers' destinations are located at the junctions and each destination junction N_i is characterized by its demand D_i that can vary between buildings. Each link contains 20 parking places of 5m length on each of its sides, 40 parking spots altogether. This entails the ratio of the total number of destinations to the total number of curb parking spots equal to $R_{city} = 80$.



Street links and junctions are stored as GIS layers, with the demand being an attribute of a junction, and the number of parking spots an attribute of a link. Model experiments are performed on a 20x20 grid with $N = 400$ destinations (junctions), $L = 800$ links and $P = L \cdot 40 = N \cdot 80 = 32,000$ parking places. We do not consider off-street parking lots in the current version of this model.

2.2 PARKGRID Basic Assumptions

PARKGRID agents - *drivers* are explicitly considered from the moment they reach their destination and start cruising for parking; whereas drivers en route to their destinations are ignored. While cruising, a model driver either finds a vacant parking spot and parks, or leaves the system after a long unsuccessful cruise. We assume that a driver cruises at a constant speed of 12 km/h [Carrese *et al.*, 2004] that is, it takes a driver 30 seconds to traverse a 100m link. We thus consider 30 sec as a model time unit - tick. At each model tick, the list of cruising and due-to-depart drivers is constructed, randomly re-ordered, and each driver acts in its turn.

Driver Types, Arrivals, Departures

Each model driver c is assigned a destination N_i ; c appears at N_i and starts its parking search driving along a randomly chosen link that is incident to N_i . Each driver is also assigned a parking time, the distribution of which is uniform on the $[TP_{min}, TP_{max}]$ time interval. Drivers that aim at N_i are generated according to a Poisson process with a per-hour average λ_i that depends on whether a driver is an employee or a visitor to the destination, and is proportional to the destination's N_i 's demand D_i . The car vacates the spot after the parking time is over.

We consider two types of drivers: *employees* who park in the city and do not leave until the end of the simulation ($TP_{min} = 8$ hours); and *visitors* with $TP_{min} = 1$ hour, $TP_{max} = 2$ hours. Employees arrive to the city in the morning, and their arrival time is uniformly distributed on the time interval [9:00, 10:00]. Visitors arrive to the city and leave it between 9:00 and 16:00. The simulation starts at 9:00 with an empty city and stops at 16:00.

Decision at a previous junction	$d < 100$		$100 \leq d < 200$		$200 \leq d < 300$		$d \leq 300 < 400$		$d \geq 400$	
	Closer	Further	Closer	Further	Closer	Further	Closer	Further	Closer	Further
Closer	0.00	1.00	0.65	0.35	0.85	0.15	0.90	0.10	Irrelevant	
Further	Irrelevant		0.00	1.00	0.80	0.20	0.85	0.15	1.00	0.00

Table 1: Probability to choose a link that takes a driver closer to/further away from a destination, as depending on a distance d between a junction and a destination and the decision made at the previous junction [Benenson *et al.*, 2015].

Drivers' Cruising Behavior

The parking search behavior that we implement in the model is based on the results of the PARKGAME serious game [Benenson *et al.*, 2015] and is formalized as a biased, towards destination, random walk [O'Sullivan and Perry, 2013]. Visually, a driver cruises around the destination until finding a free, on street parking spot, repeatedly approaching the destination and driving further away from it (Figure 3a). Drivers' turn decisions at junctions depend on two parameters: the distance between the junction and destination, and the decision taken at the previous junction – to approach the destination or drive further away from it. The probabilities to turn closer to/further away from the destination, as dependent on the distance to destination and the decision made at the previous junction, were based on more than 200 PARKGAME game sessions with 35 participants (Table 1). Given a driver's destination N_i , the biased towards destination random walk model of parking search determines the driver's search neighborhood $U(N_i)$ and, for each link $l \in U(N_i)$, the probability w_l of traversing this link during a period of search (Figure 3b). PARKGAME experiments demonstrate that these probabilities do not depend on a driver's characteristics (risk-taker or risk avoider) and parking occupation rate around the destination.

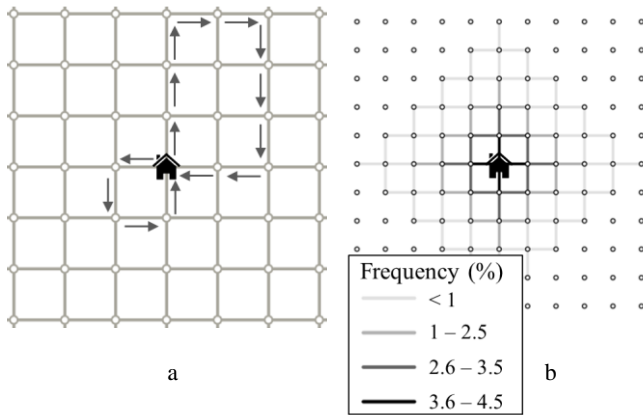


Figure 3: Driver's parking search as a biased random walk. Typical search path (a); $U(N_i)$ and probabilities to visit links in it (b).

In the model, a driver parks on the first street link that is not fully occupied. If, during a 30 sec iteration, a driver parks on a traversed link that had f free parking places at the

previous time step, then its search time on this link is estimated as $30/(f + 1)$ sec.

Maximum cruising time M in all investigated scenarios is set to $M = 20$ min; during this time a model driver traverses 40 street links and 1,600 parking places. We assume that drivers that fail to find curb parking during the maximum search time, park at a "distant off-street parking lot" that always has vacant spots. We ignore them when estimating average cruising time.

3 Model Study

3.1 Homogeneous Demand and Supply Patterns

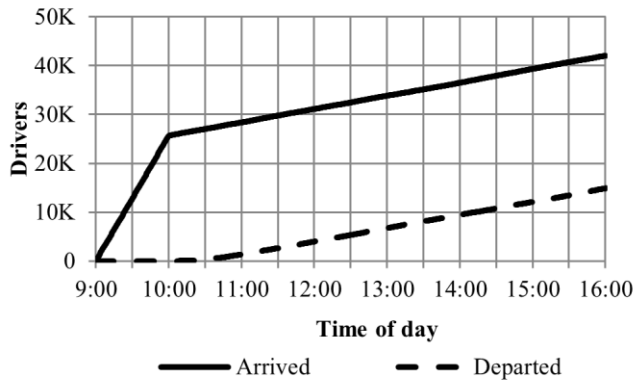
In the basic scenario we consider a homogeneous city in which the average number of drivers who aim at destination N_i is $D_i = q * R_{city}$, $q < 1$. Note that q is an average over the city occupation rate in this case.

Let a fraction e of drivers who arrive to the city in the morning be employees who stay there until the end of the day. For a city with N destinations this means that $e * q * R_{city} * N$ drivers arrive, uniformly, to the city between 9:00 and 10:00, search for parking, park (if successful) and the car stays at the parking spot until 16:00, the end of the model day. The rest of the drivers that arrive throughout the day and depart the same day, are visitors whose parking time is uniformly distributed on the $[1, 2]$ (hours) time interval. The average parking time of a visitor is thus 1.5 hours and to compensate visitors' departures, we assume that visitors' arrival rate λ is $(1 - e) * q * R_{city} * N$ per 1.5 hour that is, $\lambda = ((1 - e)/1.5) * q * R_{city} * N$ per hour.

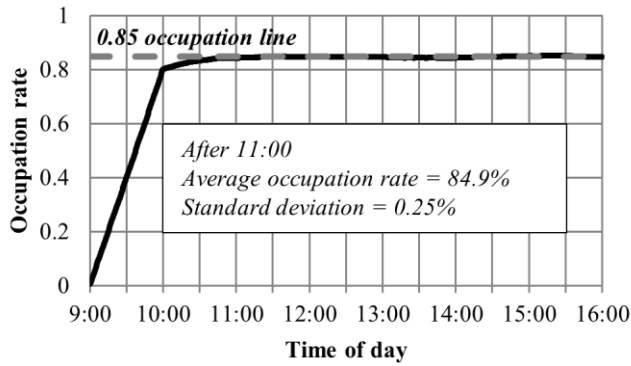
All drivers in the basic scenario employ the biased random walk search tactic, with the parameters presented in Table 1. We start with investigating the dependency of parking search time in a city with a homogeneous distribution of demand and supply, that is, D_i , λ_i and $[T_{i,min}, T_{i,max}]$ are identical for every destination, and estimate the probability $p(q, \tau)$ to find parking in time less than τ ("cruising time curve"), as dependent on q . We then extend these results to the case of heterogeneous demand.

3.2 Homogeneous Scenario Outcomes

We start with the case of relatively low demand, $q = 0.85$ and $e = 0.85$. That is, the average number of employees that aim at each destination equals to $e * q * R_{city} = 0.85 * 0.85 * 80 = 57.8$ cars, while the visitors arrive during the whole day at an average rate $((1 - e)/1.5) * q * R_{city} = 0.1 * 0.85 * 80 = 6.8$ cars/hour/destination.

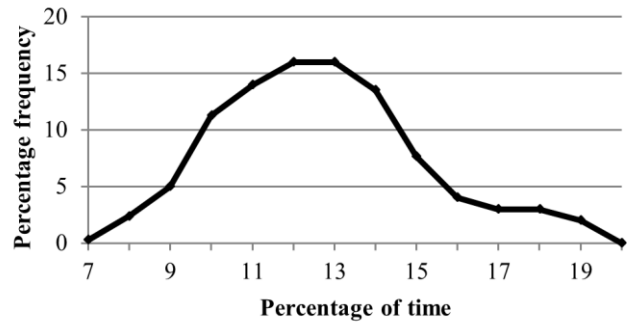


a

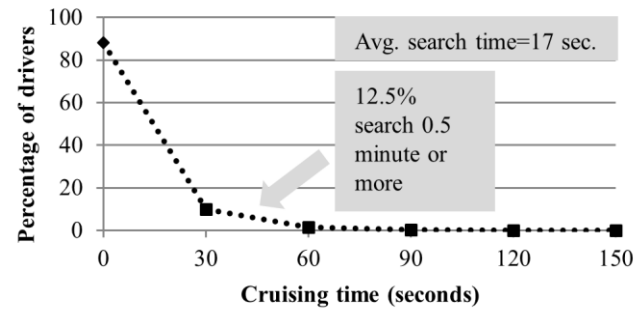


b

Figure 4: PARKGRID basic scenario, $q = 0.85$, $e = 0.85$. Dynamics of the total arrivals and departures (a), and the fraction of occupied spots (b).



a



b

Figure 5: Model output for the basic scenario of homogeneous demand for $q = 0.85$, $e = 0.85$. Percentage of time the link is fully occupied (a), cruising time curve (b).

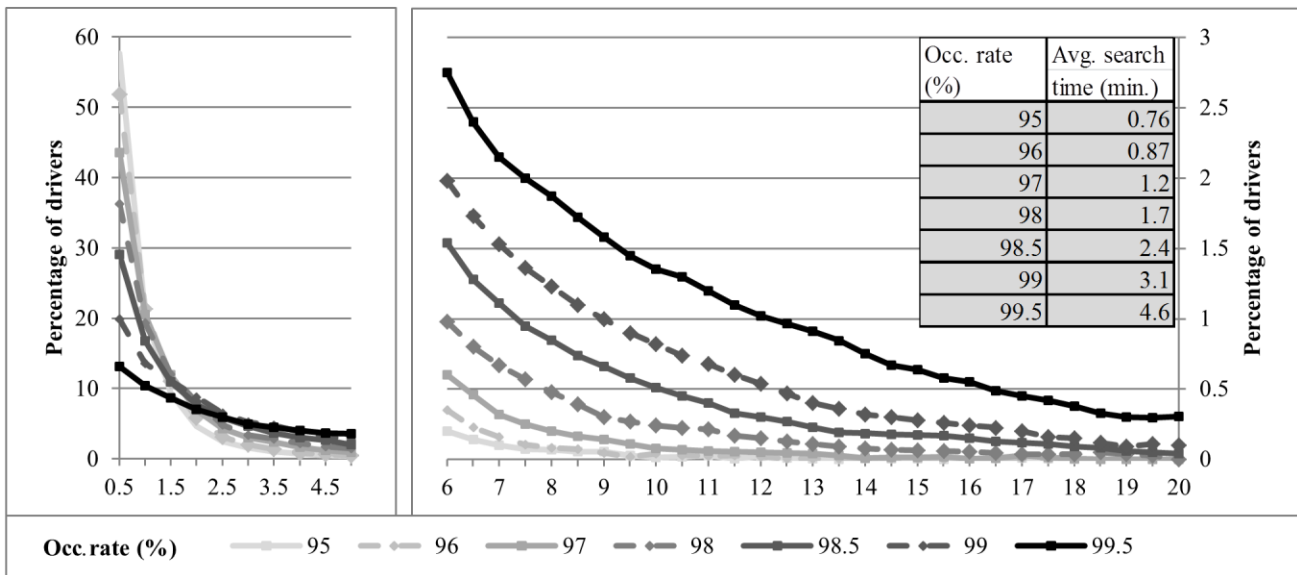


Figure 6: Cruising time curves (fraction of driving cruising for a time t before finding a vacant parking) for q varying between 0.950 – 0.995 and $e = 0.85$

As presented in Figure 4, the average occupation rate in the city stabilizes, as expected, at $q = 0.85$ towards 11:00 and from then on remains steady, fluctuating around 0.85 with the STD of 0.0025. In what follows, we consider the steady period 11:00 - 16:00 only.

As should be expected, a street link's occupation rates are symmetrically distributed around 85% average, with an $STD = 1\%$. On average, a link is fully occupied during ca. 7 minutes per hour that is, 13% of the time (Figure 5a). High parking availability results in an average cruising time of 17 seconds, with only 12% of drivers cruising for longer than 30 seconds, a consequence of not finding a vacant spot along the first link after the destination (Figure 5b). With an increase in q , the expected search time becomes longer and longer (Figure 6).

The cruising time curves in Figure 6 enable estimating the average search time as dependent on the occupation rate and Figure 7 merges between Levy *et al* [2013] outcomes presented in Figure 1 and the PARKGRID estimates of the average cruising time as dependent on occupation.

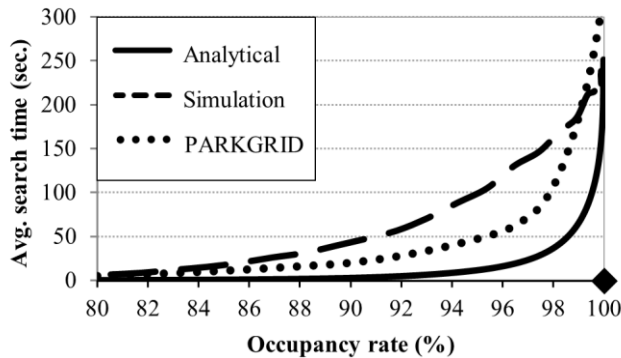


Figure 7: PARKGRID search time vs Levy *et al* [2013] results.

As can be seen, the PARKGRID average search time is higher than obtained in Levy *et al* [2013] analytical model, while lower than the estimates obtained in simulations for the occupation rates below ~99.5% and higher for higher average occupation rates. Several explanations can be proposed: Levy *et al* [2013] (1) artificially preserved a constant number of drivers in the system, substituting one driver that leaves the system by one driver that enters it; (2) accounted for the parking search on the way to the destination; and (3) accounted for the car-following and the time that it takes a driver to occupy a vacant spot. In PARKGRID, the arrival and departure processes are independent, parking search starts after a destination is reached, and car-following and the time that it takes a driver to occupy a spot are ignored.

3.3 The Case of Heterogeneous Demand

To investigate the consequences of heterogeneous demand, we consider a city with two neighborhoods H and L, where the demand differs from the average over the city. We assume that in the neighborhood H the demand is higher than $q \cdot R_{\text{city}}$, while the demand in L is lower and adjusted to the demand in H, to preserve the overall $q \cdot R_{\text{city}}$. Formally, for

each destination $N_i \in H$ the demand is set equal to $D_i = (q + \alpha) \cdot R_{\text{city}}$, while for destinations in L, $D_i = (q - \alpha) \cdot R_{\text{city}}$.

Figure 8 presents the case of $\alpha = 0.5$ and H and L as 5x5 neighborhoods. The demand D_i of every destination in H is equal to $D_i = (q + \alpha) \cdot R_{\text{city}} = (0.85 + 0.5) \cdot 80 = 108$ and, to compensate, the destination's demand in L is equal to $D_i = (q - \alpha) \cdot R_{\text{city}} = (0.85 - 0.5) \cdot 80 = 28$. For the rest of the destinations we preserve the demand $D_i = q \cdot R_{\text{city}} = 0.85 \cdot 80 = 68$.

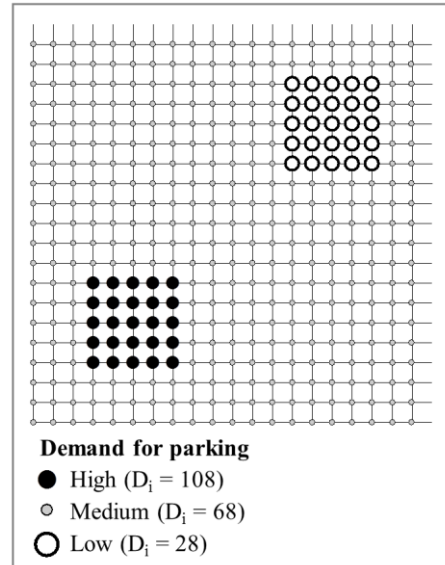


Figure 8: Demand patterns for the heterogeneous scenario, $q = 0.85$, $e = 0.85$, $R = 80$, $\alpha = 0.5$.

The effects of the H and L neighborhoods on the city parking pattern are different. The capacity of the links inside H is insufficient for absorbing all drivers who aim to park there and some of them eventually park beyond H, increasing parking occupation in H's surroundings. The L neighborhood hardly influences the parking pattern, because the demand there is far below parking capacity.

To reflect the effect of spillovers generated by drivers who aim to park at H, we apply the PARKFIT algorithm (Levy and Benenson [2015]) that, based on the PARKGRID demand and supply patterns, generates a Maximally Dense (MD) parking occupation pattern.

3.4 PARKFIT Algorithm and Maximally Dense Parking Pattern

The PARKFIT algorithm aims at estimating the parking pattern in a "city of autonomous vehicles". Its major assumption is that each car "knows" the vacant parking place that is closest to its destination when starting its parking search, "books" it when entering the system, and drives there directly, meanwhile the spot cannot be occupied by other drivers.

Let $k, k = 1, 2, 3, \dots, K$ be destinations of the D_k demand, and drivers (i.e., autonomous cars) know distances between each of the parking spots in the area and their destinations.

The steps of the PARKFIT algorithm are as follows:

(1) Build a list L of all <driver, destination> pairs (the length of this list is $D_1 + D_2 + D_3 + \dots + D_K$) and randomly reorder it. This list defines the order of drivers' arrival to the area

(2) Loop by drivers in L . For each driver consider the parking spot closest to its destination that is vacant at a moment of the driver's arrival to the system and assign it to the driver.

(3) Randomly release spots in respect to the departure rate per time period that, on average, is necessary to traverse the link.

In the areas where destinations' demand is lower than the parking supply around, as in the neighborhood L , PARKFIT algorithm generates patches of 100% occupation around each destination with intervals of vacant spots between patches. In cases where destinations' demand is higher than the supply nearby, as in the neighborhood H , PARKFIT spreads the excess demand beyond the area of the high-demand destinations (Figure 9). In both cases, the occupation rate of the link within highly occupied patches is equal to 100% minus departure rate per the time unit necessary for traversing the link (30 seconds for the grid city that we consider).

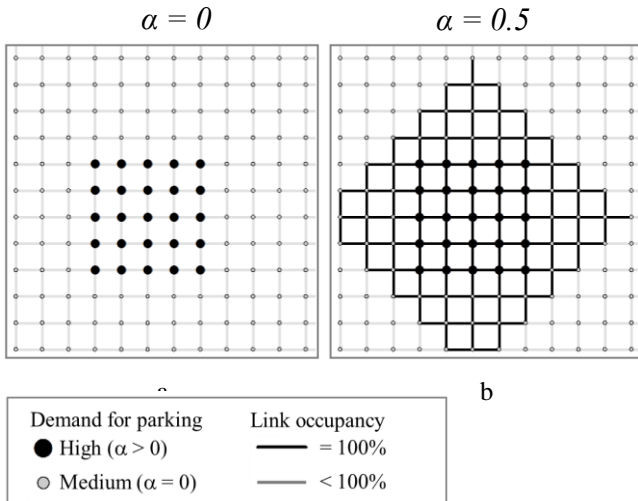


Figure 9. MD patterns generated by PARKFIT for $q = 0.85$, $R_{city} = 80$, $\alpha = 0$ (a) and $\alpha = 0.5$ (b).

3.5 Cruising Time as Dependent on Local Demand-to-Supply Ratio

A driver's search conditions are very different depending on whether its destination N_i is located within H , L or over the rest of the area. The success of a driver's parking search is defined by the overlap of the search neighborhood $U(N_i)$ and the MD pattern. For drivers with destinations N_i that are close to the center of H , the only chance to park is to occupy a spot that is freed by a departing driver. Drivers whose destination is close to the boundary of the MD-extension of H have a significantly higher chance to find a free spot beyond

the border of this extension, where the links' occupation rate is lower than 100%. Drivers whose destinations are not within H and its MD extension will cruise over a neighborhood with an average or even lower than average occupation rate.

The average occupation rate $r_{i,ave}$ over the driver's search neighborhood $U(N_i)$ can be estimated as

$$r_{i,ave} = \sum_{l \in U(N_i)} \{w_l * r_l\} \quad (1)$$

Where $U(N_i)$ is the driver's random walk search neighborhood, w_l is the probability of traversing each link $l \in U(N_i)$ during its search as presented in Figure 3b, and r_l is the average occupation rate of link l in the maximally dense pattern.

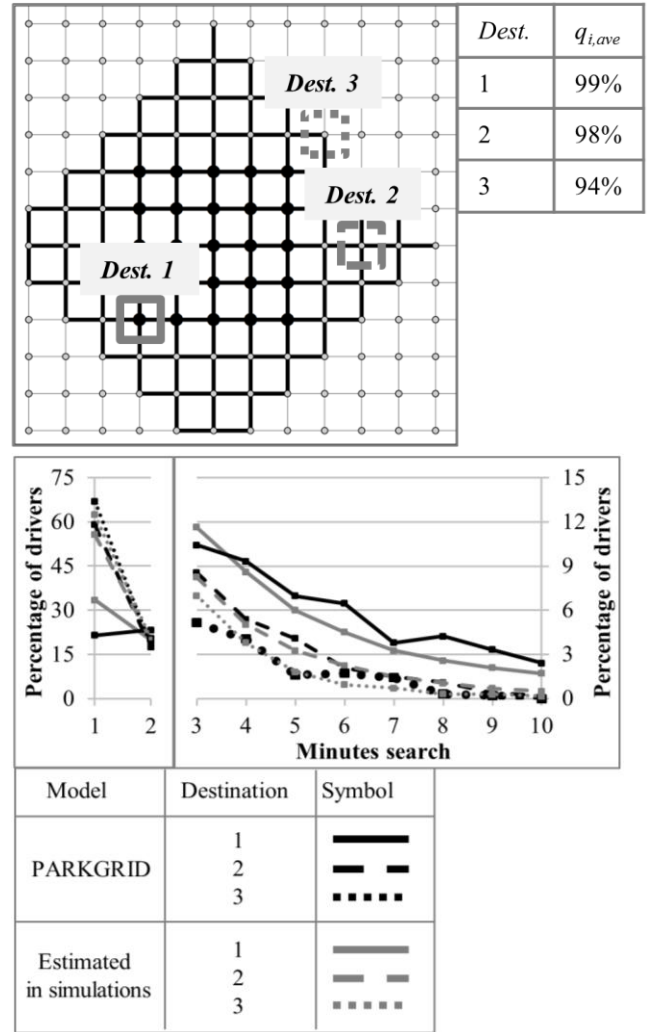


Figure 10: Directly simulated vs obtained according to (2) cruising time curves for three selected destinations within and outside H .

Consequently, we can roughly estimate the cruising time curve $P_i(\tau)$ for the destination N_i located within the heterogeneous neighborhood (according to the demand and supply) $U(N_i)$ based on $r_{i,ave}$. The simplest approximation is as follows:

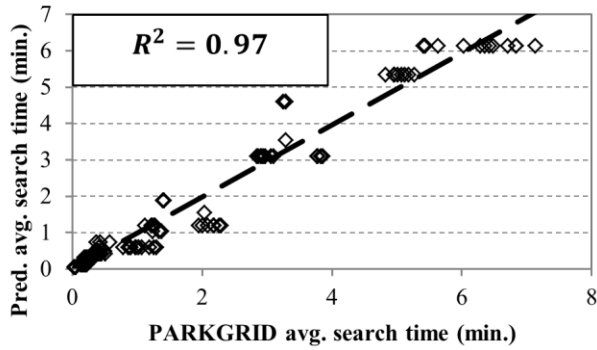
$$P_{i,1}(\tau) = p(r_{i,ave}, \tau) \quad (2)$$

Instead of (2) that is based on the average occupation rate (1), we can directly average cruising time curves $p(r_i, \tau)$ that are characteristic of the links of the MD pattern:

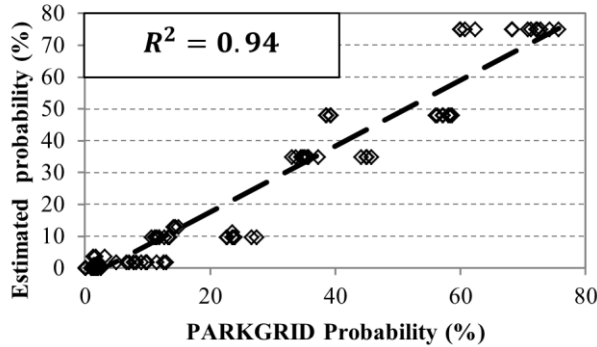
$$P_{i,2}(\tau) = \sum_{i \in U(N_i)} \{w_i * p(r_i, \tau)\} \quad (3)$$

We have verified approximation (2) by comparing cruising time curves $P_i(\tau)$ that are estimated directly in simulations and $P_{i,1}(\tau)$ for locations within and outside (yet close to) H , and for which $U(N_i)$ neighborhoods are heterogeneous. We employed the weights w_i as presented in Figure 3b and cruising time curves for the homogeneous case as presented in Figure 6. Figure 10 presents $P_{i,1}(\tau)$ and directly estimated $P_i(\tau)$ curves, and the fit is very good.

Figure 11 presents the results of systematic comparison between aggregate properties of the $P_i(\tau)$ and its $P_{i,1}(\tau)$ approximation for all 400 destinations N_i of the demand pattern presented in Figure 8. As can be seen, direct and approximate estimates of the average search times as well as the probability to cruise for over 3 minutes strongly correlate with $R^2 \sim 0.95$.



a



b

Figure 11: $P_{i,1}(\tau)$ estimates according to (2) versus direct estimates with the PARKGRID. Average search times (a); Probability to cruise longer than 3 minutes (b).

4 Predicting Cruising Time in Bat Yam

As a practical example, we estimate the time of residents' search for overnight parking in the Israeli city of Bat Yam.

4.1 Parking Demand and Supply in Bat Yam

Our estimates are based on the demand and supply data of 2010, when Bat Yam's population was ca. 130,000, total car ownership 35,000, and the total number of residential buildings 3,300 with 51,000 apartments. These data as well as layers of streets, off-street parking facilities and buildings were supplied to us by the Bat Yam municipality. Residential buildings in Bat Yam provide their tenants a total of 17,500 dedicated parking places that should be excluded from the demand and supply data. We associate destinations of the overnight parking with residential buildings and estimate the demand for parking in each as $(35,000 - 17,500) / 51,000 = 0.34$ times the number of apartments (i.e. households) in the building.

Parking supply data is based on two GIS layers - a layer of streets and a layer of off-street parking facilities. Based on the layer of streets, 27,000 spots for curb parking were constructed automatically, 5 meters apart on both sides of two-way street links, and on the right side of one-way links, with a necessary gap from the junction. In addition, 1,500 spots are available for the city's residents in its parking lots and in the evening Bat Yam residents can park at these spots free of charge. The average overnight demand/supply ratio is thus very low $(35,000 - 17,500) / (27,000 + 1,500) \approx 0.61$ car/parking spot.

However, the distribution of demand and the road network that characterizes the supply in Bat Yam are both highly heterogeneous, and the demand in the center of Bat Yam is high and significantly exceeds the supply there (Figure 12).

4.2 Maps of Parking Search Time in Bat Yam

We estimate cruising time in Bat-Yam assuming, as above, an 85:15 ratio between the numbers of parking residents and visitors that come to visit Bat-Yam residents in the evening. Starting from the building-based estimates of demand, and the link and lot-based estimates of supply, we have (1) transferred buildings' demand to the nearest junction; (2) established Bat Yam MD pattern for 85:15 ratio of residents and visitors (Figure 13a); and then (3) estimated the cruising time curve for every destination applying formula (3), assuming that a driver's area of search and cruising behavior is the same as revealed in the experiments presented in Figure 3b and Table 1. Figure 13b presents estimates of the average cruising time in Bat Yam, while Figure 14 presents the probability to cruise for parking for more than a certain time as dependent on driver's destination.

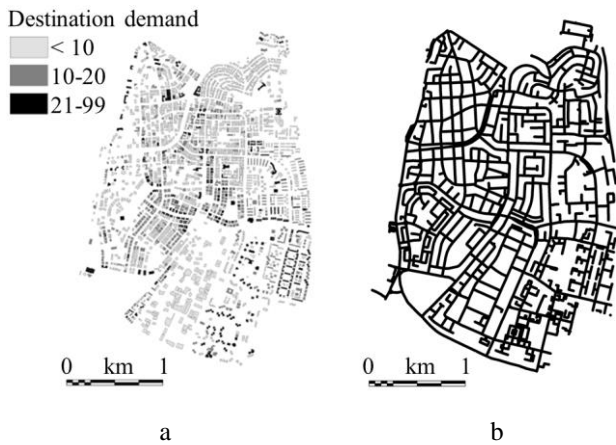


Figure 12: Bat Yam: Parking demand by buildings (a), road network (b) Demand/Supply ratio by Transport Analysis Zones (c).

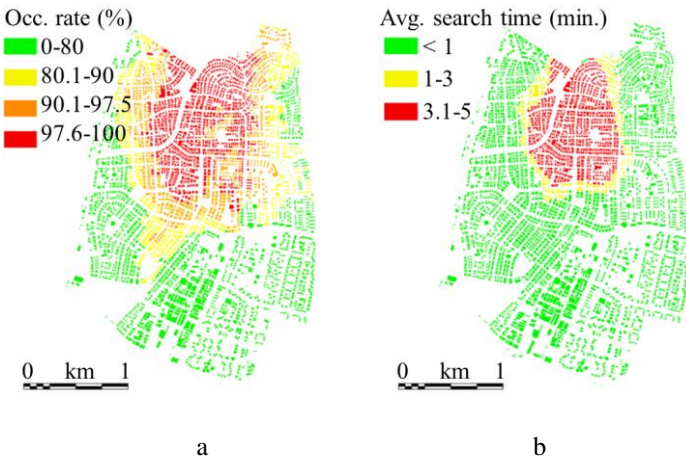


Figure 13. Bat-Yam MD (a) and average cruising time (b) maps.

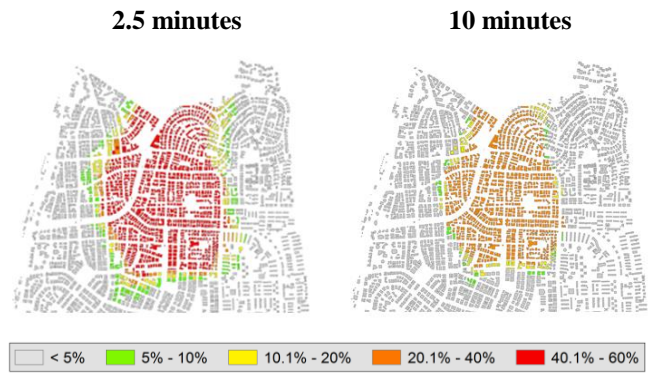


Figure 14. Bat-Yam maps of the probability to cruise longer than 2.5 and 10 minutes.

5 Conclusions

The surplus of demand over supply is critical for parking search in the city. We investigate the dependence of the parking search time on the local demand-to-supply ratio and propose an algorithm for estimating search time based on static demand and supply patterns.

We apply our model to the Israeli city of Bat Yam and show that despite a low, ca. 61%, average demand to supply ratio, spatial heterogeneity of the demand and supply patterns results in lengthy parking searches for a significant fraction of drivers.

The proposed method can be applied to every city where the patterns of parking demand and supply are known at a resolution of buildings, roads and parking lots. We consider our approach as a fast and efficient approximation for direct estimates of the cruising time, which can also be obtained in a dynamic agent-based model simulation, such as PARKAGENT [Levy *et al.*, 2013]. This approach can be applied to any city of arbitrary size.

It should be stressed that the perspectives of agent-based modeling of human-driven systems such as parking, critically depend on our knowledge of agents' behavior. In this respect, we consider serious games as a method to account for the dynamic nature of the system that is missed in the stated and revealed preferences surveys. In the same time, the conditions of serious games are fully controlled by the researcher and can be used to create situations that cannot be observed. To the best of our knowledge, the biased random walk search tactic that is revealed in the game and employed in the PARKGRID model is the first example of a successful merge between a serious game and a parking agent-based model.

From a practical point of view, the estimates of parking search times presented in this paper should serve as initial information for an urban planner who aims at assessing the consequences of construction of, for example, a new office, commercial or residential building. If parking supply in the area is insufficient for the planned demand, a planner can choose to increase supply by adding parking lots. Our method can be applied for predicting the decrease in the search time and its spatio-temporal extent.

The policy maker can also attempt to decrease demand by rigid limitations on vehicular entrance to a designated area to certain groups of drivers, or by increasing parking prices, or even introducing flexible prices that are adapted to the expected demand for parking [SFMTA, 2016]. Incorporation of drivers' reactions to prices when cruising for parking is thus the extension of our approach, and the first step in this direction is presented in [Fulman and Benenson, 2018].

6 References

- [Anderson and de Palma, 2004] Simon P. Anderson and André De Palma. Parking in the city. *Papers in Regional Science*, 86(4): 621–632, 2004.
- [Arnott and Rowse, 1999] Richard Arnott and John Rowse. Modeling parking. *Journal of urban economics*, 45(1): 97-124, 1999.
- [Arnott and Rowse, 2013] Richard Arnott and John Rowse. Curbside parking time limits. *Transportation Research Part A: Policy and Practice*, 55: 89-110, 2013.
- [Arnott and Williams, 2017] Richard Arnott and Parker Williams. Cruising for parking around a circle. *Transportation Research Part B: Methodological*, 104: 357-375, 2017.
- [Carrese et al., 2004] Stefano Carrese, Borja Beltrán Bellés, and Emanuele Negrenti. Simulation of the parking phase for urban traffic emission models. In *TRISTAN V – Triennial Symposium on Transportation Analysis*, Guadeloupe, 2004.
- [Benenson et al., 2015] Itzhak Benenson, Eran Ben Elia, Evgeny Medvedev, Shay Ashkenazi, and Nadav Levy. Serious game-based study of urban parking dynamics. In *Presented at the XIII NECTAR International Conference*, 2015.
- [Fulman and Benenson, in press] Nir Fulman and Itzhak Benenson, Establishing Heterogeneous Parking Prices for Uniform Parking Availability for Autonomous and Human-Driven Vehicles, *IEEE Intelligent Transportation Systems Magazine*, in press.
- [Levy and Benenson, 2015] Nadav Levy and Itzhak Benenson, GIS-based method for assessing city parking patterns, *Journal of Transport Geography*, 46: 220-231, 2015.
- [Levy et al., 2013] Nadav Levy, Karel Martens, and Itzhak Benenson. Exploring cruising using agent-based and analytical models of parking. *Transportmetrica A: Transport Science*, 9(9): 220-231, 2013.
- [O'Sullivan and Perry, 2013] David O'Sullivan and George LW Perry. *Spatial simulation: exploring pattern and process*. John Wiley & Sons, 2013.
- [SFMTA, 2016] SFMTA (San Francisco Municipal Transportation Authority). SFpark: The Basics. [Online], 2016. Available: <http://sfpark.org/about-the-project/faq/the-basics/>