Street Parking Strategy Sensitivity Analysis (EU-SP2132)

Jean-Sébastien Gonsette [1*]

[1] AISIN AW – AWTC Europe SA, Belgium
[*]jean-sebastien.gonsette@awtce.be

Abstract: Parking in European town centres is a challenging problem that can be addressed from multiple lines of attacks. If informed solutions based on metered parking spots are evolving in some big cities, traditional on-street parking generally makes up the majority of the parking capacity. In such case, a probabilistic approach is the only suitable one as there is never any guarantee that a vacant spot will still be available within the time needed to reach it. In this paper, we first define a complete probabilistic model that can be used to simulate realistically the vacancy of parking spots among streets of a city. This model includes the possibility to get short time information about street occupancy, along with the prediction of how this availability evolves on the long run. Based on this model, we present a methodology to evaluate the performances of a route returned by a parking strategy in terms of expected time to reach a target goal. This time includes how long it is expected to park, as well as how long the driver still needs to walk until his destination. Then, we use this methodology to evaluate the gain, in time, of our parking strategy, as well as its sensitivity to the uncertainty of the actual parking difficulty.

1. Introduction

Everybody has already experienced the daunting task of searching for a parking place inside a crowded city centre. In addition to being stressful and time-consuming, it is also a real society problem that may cause up to 30 percent traffic increase in big cities [1], not talking about pollution. This problem is well identified and many solutions have been suggested as countermeasures for this matter of great ecological and social importance. Most of them rely on sensors put around metered parking spots, like [2] [3], but because of their cost are inherently more difficult to scale up. Other approaches to collect statistics or real-time information about parking spot availability consist in using the vehicles themselves as probes. This is a reasonable assumption considering the increase of connected cars today, at least thanks to the driver smartphones. Parking spots could be detected when cars stop along the street, while free spots can be announced as soon as they leave. Many car markers also consider scanning directly parking spot availability while driving with the help of the car ultrasonic sensors or front camera. This leads to approaches consisting in the exploration of the known available free spots in an optimal way considering the driver's destination [4].

If those alternative solutions sound legitimate, they are not flawless. For example, it is especially uneasy to be sure that what appears to be free parking spot is indeed a valid one. It could just as well be a private garage or a not legal parking area. In addition, collecting parking statistics is only valuable if the availability of free spots is scarce, which means that the measures are particularly sensitive to errors made on those free detections. If an attractive neighbourhood contains, on average, only one free spot out of one hundred, evaluating such low density is particularly difficult and put a lot of pressure on the system capability to detect it.

Even putting this problem aside, raw information about parking statistics or real-time availability are not enough to efficiently guide the driver to his destination. Firstly, knowing that a parking spot is vacant right now (short term observation) is not sufficient for a successful search, because it does not mean that it will still be the case upon arrival. This is especially true if there is a lot of competition among drivers. In fact, parking search is probabilistic by nature and working with such probabilities (long term observation) helps addressing this problem by indicating high likelihood parking streets to the driver. However, a good strategy to guide the driver to its destination still needs to be elaborated.

A good parking route should not be deterministic but rather try to accumulate as much probability mass as possible. However, this optimization must be done under the cost constraint of time needed to reach the final destination. In the case of the parking problem, this cost is the expected total time to reach the final destination. This latter can be separated into the expected time to find a free spot, plus the deterministic time to walk to the destination. But because of this indeterminism, the route can potentially be infinite while the cumulated probability to park converges to one. Moreover, testing a street at a given time without success does not mean that testing it later is useless, because there is always a chance that someone leaves a bit later. For example, parking areas with a high churn rate may be more interesting to be tested again.

As we can see, searching for a vacant parking spot requires a stochastic strategy, along with several hypotheses or measurements to rely on. These are mainly the travel time of each road element, the chances of success in each street and the dynamic of the underlying parking events. Of course, all those parameters can strongly vary over time, and sometimes in a very unpredictable way. It is therefore interesting to understand how far this lack of knowledge can be detrimental to the parking strategy.

In this paper, we investigate the performances of our on-street parking algorithm [5] in the context of the partial knowledge of all the street attributes cited above. This strategy, which we called the 'winning strategy', showed promising results provided the different probabilities to park. However, the availability of this information is a strong assumption, especially if they are required with high precision. That is why we developed a test setup enabling to acquire considerable insights about the sensitivity of our probabilistic routing to the different measurements it is based on. As expected, any uncertainty on the underlying parking model deteriorates the driver gain (i.e. the total time to reach his destination). Neththeless, this algorithm is quite robust and tolerates many parameter approximations without losing too much of its advantage. Moreover, we show that the sole knowledge of the parking spot locations is enough to considerably lower the total time to park, provided a reasonable approximation of the overall occupancy rate.

1.1. Background

In the context of parking search, guiding the driver to the closest free spot does not provide a satisfactory solution. Instead, he must visit as much parking spots as possible to maximize the odds of finding a suitable one. Parking is inherently probabilistic and modelling the road network as a graph, each edge having a probability to hold at least one vacant parking spot, is a natural and more powerful extension than the simple recording of free spot locations. Indeed, any free spot detection can be easily modelled with a probability to park equal to one. However, this short-term detection has a decreasing life time: as time progresses, its influence on the probability to find a vacant space decays. Such probability model is introduced in [6]: availability of road network resources are modelled as continuous-time Markov chains. Short time observations ensure to take new fresh information into account, while the model falls back nicely on long-term average information as time passes, allowing depleted resources reappearance. This work does not focus especially on street parking as any kind of resources is considered; however it provides an example for this particular use case. Two algorithms are presented for optimal search of a solution. Both are based on backtracking.

A detailed street parking occupancy model aimed at predicting spot vacancy states is presented in [7]. Built on queueing theory and using a continuous-time homogeneous Markov model, its goal is to estimate how the probability density function to observe a parking spot in a given occupancy state evolves with time. Given any observation of resource availability, the model predicts how it evolves in the future by considering arrival and parking rate as Poisson processes.

An algorithm specifically designed to seek for a probabilistic route in the case of the street parking problem is presented in [8]. This work builds on [7] and proposes two specific algorithms: the first one uses the underlying probability to park to explore the most interesting part of the road network traversal tree, pruning branches that seem costly at a early stage; the second one can be used when such probability information is not available and is based on a dedicated heuristic. It also present a study identifying the impact of inaccurate probabilities on the final cost of their solution.

Finally, we presented our own algorithm to find a solution to the street parking problem in [5]. This one takes a different approach, avoiding the exploration of a search tree whose width is exponential in the length of the route. The methodology is similar to [9] but in the context of on-street parking and not inside a parking lot. Based on a fixed-point iteration method, our solution computes the expected time to reach a given destination from every street in a map, given that each street solution depends on the value of its neighbours. This solution is optimal if the driver manages to park before reaching the street having the minimal expected time. In the other case, an updated solution can be computed based on the new understanding of the probability to park.

1.2. Contribution

The novelty of this work is twofold. The first contribution is a full development and presentation of a statistical model simulating the probabilities to park on-street based on long-term collected statistics and on short-term observations of the street occupancy. Long-term observations drive the steady probability to park in each street and only depends on its capacity and average occupancy rate. On the other hand, short time observations change those probabilities for a while and the model describes the dynamic of the relaxation to the steady state. This model has been implemented in a simulator in order to test the performances of any parking strategy in a realistic configuration. A complete description of the definition of performance is given, along with the step by step procedure to implement it.

In a second time, we use this simulator to evaluate the performances and robustness of the parking strategy presented in [5] and that we call the winning strategy. The goal of our approach is to understand the sensitivity of our methodology against the correct knowledge of its environment. Indeed, a parking route is useless if it finally makes the driver loose time because of the erroneous guess of any variable. Those ones are typically the probabilities to park in streets and are notoriously difficult to estimate or to measure. We propose to evaluate this sensitivity in simulation by running the winning strategy on corrupted maps while analysing the impact on the original one. The idea is to get a better comprehension of the final time a driver could save using such strategy, even in the context of incomplete or erroneous modelling of the environment. Then, it allows to better understand the benefits of the winning strategy and what would be the ROI if any effort is conceded to better measure the variables influencing the probabilities to park on-street.

2. Analytic Street Parking Model

In this section, we define the road network model that we use to make an assessment of our winning strategy, that is, a parking strategy optimizing the total time to destination. The latter is defined as the time needed to find a suitable parking place plus the time to walk to the final destination. It is important to simulate all the quantities required to compute optimal routes, as well as providing a model that mimics as much as possible what we could expect from a ground truth. We begin with the definition of the road network as a collection of edges then, we continue with the behaviour modelling of each edge individually.

2.1. Road Network Graph

The winning strategy operates on a graph $G = (V, E)$ representing the road network to take into consideration to solve the search problem. This network is typically the set of reachable streets from the target destination in less than an arbitrary limit, in time or distance. The vertices $v \in V$

correspond to the crossings (or dead ends) in the network, while the *directed* edges $e \in E \subseteq V \times V$ match the edges joining those intersections. We refer to this graph as a *Road Network Graph*.

A distance function $d_{v_t}(v): V \to \mathbb{R}$ maps every vertex $v \in V$ to its shortest path distance to a given target destination v_t. This distance can be computed for all vertices at once at the beginning of a query with the Dijkstra algorithm.

A travel time function $t(e): E \to \mathbb{R}$ associates each edge $e \in E$ to its travel time. Those values may be time dependent if, for example, they are given by a traffic provider.

A probability function $\mathrm{p(e)}: \mathrm{E} \to [0,1]$ gives how likely it is possible to find a free parking spot for every edge $\mathrm{e} \in \mathrm{E}$ when driving it down once. Those values are time dependent for two reasons. First, the long-term probability to park on a edge may change naturally depending on the time of the day (e.g. probability to be able to park in a residential area is lower during the night). In addition, our model must cope with short-term observations collected when driving along the streets. For example, if someone tries to park in a street with initial probability set to 0.2, without success, then this probability immediately drops to zero after this observation. Afterwards, this value slowly recovers to its initial level as time passes.

2.2. Street Probabilistic Model

In the previous section, we described the minimal model for the winning strategy to operate on. Now, we present further the underlying model that enables to simulate the probability to park of each edge in the graph in a realistic way.

Our approach is similar to [6] and uses a Markov chain to describe the dynamic of the different states in which a street can be. We develop it here for completeness. If the street has a capacity m, that is the maximum number of cars that can park there, then each of the $\mathrm{m}+1$ states represent the respective number of vehicles currently parked. A car is rejected if it tries to park in a fully occupied street.

The flow of incoming vehicles trying to park is assimilated to a Poisson process, which means that it follows an exponentially distributed inter-arrival times of rate λ. In the same way, the parking rate μ of a single car is the inverse of the average time it stays on its parking lot before leaving again. Brought back to the whole edge e, it means that the rate of leaving cars is a function of the number of cars parked n by the relation $\mu_e = \mu.\mathrm{n}$. A street can reach an equilibrium if those two rates cancel each other, that is, if the number $\mathrm{n_e}$ of cars parked on e is given by $n_e = \frac{\lambda}{\mu} \leq m$. In the other case, we say that the street saturates and it impacts directly the adjacent roads as they must be engaged to absorb the incoming flow.

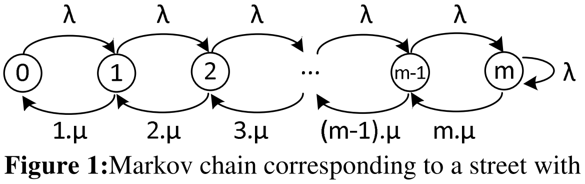

Figure 1: Markov chain corresponding to a street with capacity *m.*

Figure 1 depicts the Markov chain corresponding to our street dynamic model. The instantaneous probabilities of transition from one state to the other can be found by considering the limit of a Poisson random process when time tends towards zero:

$$p_\lambda(k) = \lim_{dt \to 0} \frac{(\lambda.dt)^k}{k!}.e^{-\lambda.dt}$$

with k the number of events happening in an interval of time dt. The result of this limit is given by:

$$\begin{aligned} p_\lambda(0) &= \; 1 - \lambda.\mathrm{dt} + O(dt^2) \\ p_\lambda(1) &= \; \lambda.\mathrm{dt} + O(dt^2) \\ p_\lambda(2) &= \; O(dt^2). \end{aligned}$$

Arrivals and departures are independent processes that can be combined to derive the transition probabilities from a state n to its neighbours. Let's write:

$$\begin{aligned} p_{\mathrm{n}+}(dt) &\coloneqq P(X_{t+dt} = n+1 | X_t = n) \\ p_{\mathrm{n}-}(dt) &\coloneqq P(X_{t+dt} = n-1 | X_t = n) \\ p_{\mathrm{n}0}(dt) &\coloneqq P(X_{t+dt} = n | X_t = n) \end{aligned}$$

Then, by considering the instantaneous case when time tends towards zero and under the reasonable assumption of time homogeneity, we have:

$$\begin{aligned} p_{\mathrm{n}+}(dt) &= \lambda.dt.(1 - n.\mu.dt) \cong \lambda.dt \\ p_{\mathrm{n}-}(dt) &= n.\mu.dt.(1 - \lambda.dt) \cong n.\mu.dt \\ p_{\mathrm{n}0}(dt) &= (1 - n.\mu.dt).(1 - \lambda.dt) \cong 1 - (\lambda + n.\mu).dt \end{aligned}$$

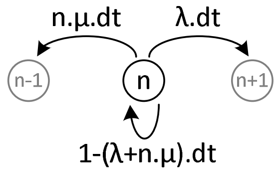

Figure 2: Infinitesimal transition probabilities when street is in state *n.*

The partial view of the corresponding Markov process is given in Figure 2. Let's denote by $\pi_n(t)$ the probability that the street is occupied by n cars at time t, i.e. $\pi_n(t) \coloneqq P(X_t = n)$ and by $\pi(t)$ the probability distribution over all possible states, i.e. $\pi(t) = [\pi_0(t) \cdots \pi_m(t)]$. According to Markov chains theory, the evolution of π is driven by a generator matrix Q whose elements q_{ij} represent the transition rate from state i to state j:

$$\dot{\pi}(t) = \pi(t).Q$$

In our case, $Q = (q_{ij})_{0 \leq i,j \leq m}$ is a $(m+1) \times (m+1)$ matrix where each element is defined as the derivative of the transition probability from i to j. This results in the following Q matrix with a tri-diagonal pattern:

$$Q = \begin{pmatrix} -\lambda & \lambda & & \cdots & & \\ \mu & -(\lambda+\mu) & \lambda & \cdots & & \\ & 2\mu & -(\lambda+2\mu) & \cdots & & \\ & & 3\mu & \cdots & & \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ & & & \cdots & \lambda & \\ & & & \cdots & -(\lambda+(m-1)\mu) & \lambda \\ & & & \cdots & m\mu & -m\mu \end{pmatrix}$$

Steady State

Assuming the knowledge of the street capacity, along with the arrival and parking rates, the matrix Q enables to compute the probabilities being in some states in the future starting from an initial state distribution $\pi^0 = \pi(0)$. One quantity of particular interest is the *average probability to park*, that is, the probability that at least one parking place is free upon arrival once the system has reached a stationary distribution. In the case of our street model, all states of the Markov chain can communicate, which ensure a unique stationary distribution $\pi^\infty = \lim_{t\to\infty} \pi(t)$. This one can be found by solving equation $\pi^\infty . Q = 0$. For that matter, it is possible to rearrange the Q-matrix so that the system to resolve becomes $\pi^\infty . R = 0$ with:

$$R = \begin{pmatrix} -\lambda & & \cdots & & \\ \mu & -\lambda & \cdots & & \\ & 2\mu & \cdots & & \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ & & \cdots & -\lambda & \lambda \\ & & \cdots & m\mu & -m\mu \end{pmatrix}$$

Looking at the last two columns of R makes it clear that the system has infinity of solutions. We just have to find one and normalize it to get a valid probability distribution. It turns out that a (not normalized) solution to this system is given by:

$$\forall\, 0 \le n \le m, \qquad \pi_n^\infty = \frac{\left(\frac{\lambda}{\mu}\right)^n}{n!}$$

Figure 3 illustrates different probability density functions (PDF) at various equilibria. This solution means that the stationary distribution only depends on the ratio $r = \lambda/\mu$ of the arrival and parking rates, which is also the number of cars parked in the street at equilibrium. The corresponding steady probability to park in a street of capacity m is thus given by:

$$P_{park}(m,r) = 1 - \frac{r^m/m!}{\sum_{n=0}^{m} r^n/n!}$$

It is possible to find an upper bound to P_{park} when r rises above street capacity m. To show it, we first need to rewrite P_{park} as:

$$P_{park}(m,r) = 1 - \frac{1}{\sum_{n=0}^{m}\left({}^{1}\!/_{r^n} . {}^{m!}\!/_{(m-n)!}\right)}$$

while this latter is bounded by:

$$P_{park}(m,r) \le 1 - \frac{1}{\sum_{n=0}^{m} {}^{m^n}\!/_{r^n}} \cong m/r$$

The probability to park evolves asymptotically in the inverse of $m.\mu/\lambda$, as depicted in Figure 4. That means that for low probabilities to park, those odds are simply defined by

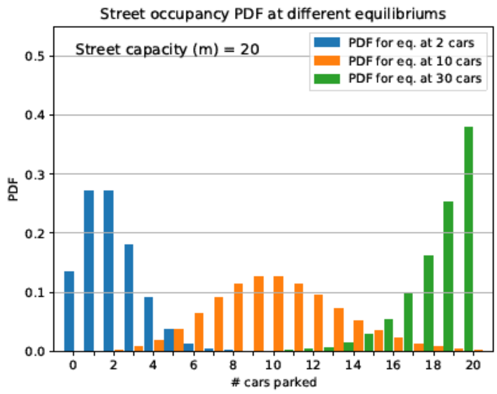

Figure 3: Street occupancy PDF at different equilibria.

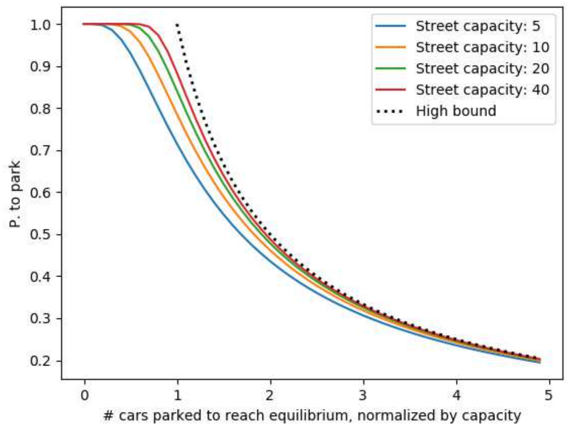

Figure 4: Probability to park in a street as a function of different arrival and departure rate ratios.

the ratio between the departure and arrival frequencies, as we can expect.

Transient Phase

The stationary solution found in previous paragraph gives the distribution of parked cars in a street we can expect if we only know the ratio of arrival and departure frequencies, without any additional observation. Of course, this distribution changes as soon as we can get some insight. For example, if we observe a street is full at time t_0, then the corresponding distribution becomes $\pi^0 = [0 \quad \cdots \quad 0 \quad 1]$.

Now, we would like to turn our attention to the transient evolution of $\pi(t)$ after such observation. As time progresses, it becomes more likely that a parking spot becomes available again as the initial distribution changes and tends gradually towards π^∞. The solution of this initial value problem is given by $\pi(t) = \pi^0 . e^{Q.t}$, while the Q-matrix eigenvalues correspond to the time constants driving the speed of the recovery. This problem is not trivial in general as it requires to compute the generator matrix exponential, which implies the usage of any dedicated numerical algorithm. Anyway, full accuracy may not be needed if we just want a realistic approximation of the PDF relaxation. In this case, we propose a simple algorithm to simulate the transient phase after an observation of the number of cars parked in the street.

First, we need a reasonable estimation $\overline{\tau}$ of the PDF relaxation time. This one is impacted by all the $(m+1)$ eigenvalues of Q, but we know that those are spread around a mean time constant that depends on the trace of Q:

$$1/_{\overline{\tau}} = \frac{-tr(Q)}{m} = \lambda + \mu.\frac{m+1}{2}$$

Then, we can choose some constant N to define a reasonably small time step $dt = \overline{\tau}/N$, in order to catch the transient phase thanks to an approximation of $e^{Q.dt}$, as follows:

$$e^{Q.dt} \cong T_{dt} = \begin{pmatrix} 1-\lambda.dt & \lambda.dt & \cdots & \\ \mu.dt & 1-(\lambda+\mu).dt & \cdots & \\ & 2\mu.dt & \cdots & \\ \vdots & \vdots & \cdots & \vdots \\ & & \cdots & \lambda.\mathrm{dt} \\ & & \cdots & 1-m\mu.dt \end{pmatrix}$$

Finally, the street occupancy PDF at the successive time steps is estimated by multiplying the initial distribution with this elementary transition matrix:

$$\pi(k.dt) = \pi^0.{T_{dt}}^k$$

Figure 5 illustrates the comparison of the proposed approximation against the exact solution when $N = 10$. The three curves are plotted for a period of $5.\overline{\tau}$ in the case of a street capacity of 20 cars and with a mean parking time $\mathbf{1}/\mu$ of 1000s.

3. Street Parking Simulator

Previous section brought us the different mathematical pieces needed to build a realistic parking simulator, based on the reasonable assumption that arrival and departure rates follow a Poisson process. Of course, those rates are difficult to measure and that is why a simulation is interesting. Its goal is not really to simulate a city centre perfectly, but rather to set up a reasonably realistic configuration for the study of our winning strategy. Because we fully know the simulation parameters, it is then possible to evaluate the strategy sensitivity to an imperfect knowledge of its environment.

This simulator must support the following testing methodology:

1. Setup a realistic map background;
2. Configure the map with realistic driving speeds and probabilities to park;
3. Clone the map, then corrupt its configuration somehow;
4. Choose a starting point and goal point at random;
5. Elaborate a parking route over the true map, according to a given strategy;
6. Elaborate a parking route over the corrupted map, according to this same strategy;
7. Compute performance indicators of the route found at step 5;
8. Compute performance indicators of the route found at step 6 while applying it to the true map (instead of the corrupted one);
9. Repeat at step 4 in order to average the indicators on different runs.

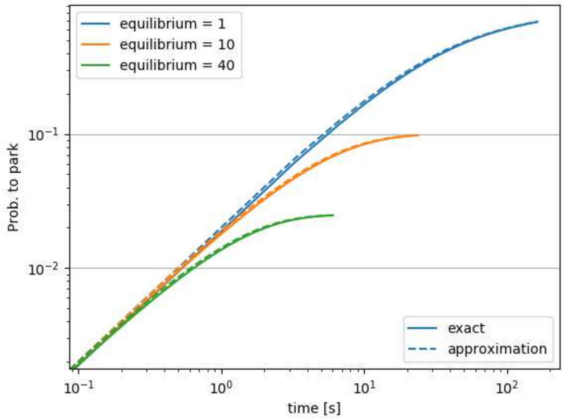

Figure 5: Approximation of the relaxation phase for different equilibria.

The idea behind this methodology is to understand how much we can corrupt the underlying map attributes until the strategy stops to propose useful routes. The more these attributes are different from those considered when elaborating a parking route, the less the gain for the driver should be. However, we need a comparison point in order to compute how much time the strategy allows to save.

For that purpose, we use a dumb strategy σ_{rw} based on a random walk in order to get a worst time to park bound. It is intended to simulate a highly inefficient (anti) strategy one could follow if he has absolutely no idea of where the parking spots are, or what the topology of its surroundings is. Its principle is as follows: the driver drives straight on its destination goal, without trying to park if it would have been possible to. Then, he starts searching randomly until any free spot is encountered: if the driver could not park in the current street, he selects the next one at random and repeats the process.

3.1. Map Simulation and Initialisation

The map is a small collection of edges extracted from a real map database and located around a city centre. For our work, we selected an area inside Brussels. The size of this area does not have to be particularly big but must be at least the quadruple of the maximum walking distance tolerated when parking around a destination. In all our test cases, we took a walking speed w of $3\ km/h$ and a maximum walking time of $1000\ s$ that results in a map of $4\ km$ wide.

According to this study [10], the average driving speed for a trip inside Paris intra muros is around $15\ km/h$. Unless stated otherwise, we used this default value to compute the traversal time of every street in the map.

In this other study [11], it is possible to get some insights about the number of parking places available on-street in a municipality of Brussels. The typical density is between 10 and 20 spots by $100\ m$. However, we still need a means to convert the number of parking places in a street into an effective probability to park. Considering an average occupancy rate $\mathcal{O}$, one possibility consists in using a numerical solver to find out $r = \lambda/\mu$ so that $\sum_n n.\pi_n^{\infty} = \mathcal{O}.m$, then to derive the corresponding P_{park}.

For the second option, we start from a *binomial* distribution, assuming a uniform repartition of the occupied spots. Put-in other words, we ignore any correlation effects between nearby parking places that could result from attractive spots in the city. Given a street with capacity m, the probability to park in this street becomes $1 - \mathcal{O}^m$ and the complete PDF is given by:

$$\pi_n^{bin} = C_n^m.(1 - \mathcal{O})^{m-n}.\mathcal{O}^n$$

Unfortunately, even if this distribution has the same general shape than our street probabilistic model, it overestimates P_{park}, as illustrated in Figure 6. We found out that a more accurate approximation of the probability to park resulting from a given average occupancy rate is adequately given by:

$$\hat{P}_{park}(m, \mathcal{O}) = \frac{1 - \mathcal{O}^m}{1 - \mathcal{O}^m + \mathcal{O}^{m/2}}$$

Figure 7 compares the exact and the approximate probability to park for different street capacities and occupancy rates.

3.2. Evaluation of a Parking Route

Given a road network graph $G = (V, E)$, a path of G is a finite sequence $\rho = (\rho_0 \cdots \rho_n) \in E^*$ such that for all $0 \leq i \leq n$ we have $\rho_i = (., v_i)$ and $\rho_{i+1} = (v_i, .)$, i.e consecutive edges are connected by the same vertex. In this work, we consider a parking strategy that defines the path a driver must follow from a given start location and leading around a target destination. This path is augmented with flags $\mathfrak{f}$ indicating to the driver if he should or should not park at each edge (street) along the path.

More formally, given a start vertex $v_s \in V$ and a target destination $v_t \in V$, a *parking strategy* is a function

$$\sigma: (E \times \{0,1\})^* \to E \times \{0,1\}$$

such that for all

$$\big((\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n)\big) \in (E \times \{0,1\})^*$$

we have

$$\begin{cases} \sigma\big((\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n)\big) = (\rho_{n+1}, \mathfrak{f}_{n+1}) \\ \rho_n = (., v_n) \\ \rho_{n+1} = (v_n, .) \end{cases}$$

The first edge of this path always starts at v_s, i.e. $\rho_0 = (v_s, .)$. However, none of the route edges has to contain the target vertex v_t inevitably. Indeed, this latter is always joined on foot once a free parking spot is found and may be purposely avoided by the strategy.

If the driver follows the parking strategy σ from v_s, then σ provides a unique flagged path $\omega = (\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n)$ that we call the *parking path*. The flags indicate the behaviour the driver should follow when any free spot is found along the path, as it may be interesting to get closer to the destination to minimize the total time to reach it. For example, the dumb strategy σ_{rw} sets those flags to zero up to v_t, then put them all to one.

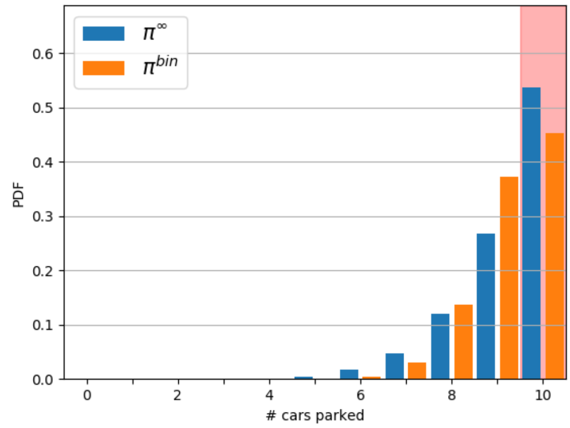

Figure 6: Comparison of PDF resulting from the street probabilistic model and the binomial distribution.

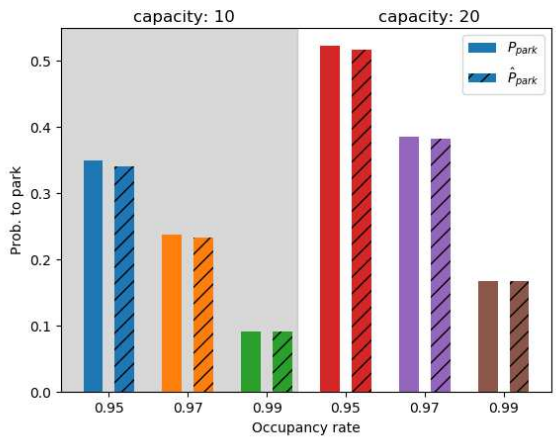

Figure 7: Comparison of the exact and approximated probability to park.

As parking is probabilistic, without any guarantee of success, it is potentially a never ending process. Therefore, we need a criterion to stop expanding a path once it has reached a sufficient probability threshold to get parked. For that purpose, we start defining the success probability of any edge along the route. Let us first define some notations.

Notations: Given a parking path $\omega = (\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n) \in (E \times \{0,1\})^*$ *of* $G = (V, E)$, *let* t_{ρ_i} *be the time of arrival at* ρ_i *and let* $p_{\rho_i}^\infty$ *be the steady probability to park on* ρ_i. *In addition, let's define* dt_{ρ_i} *the time elapsed since the last parking trial on the edge* ρ_i*:*

$$dt_{\rho_i} := \begin{cases} \min\limits_j \left(t_{\rho_i} - t_{\rho_j}\right) & \text{if } \{j < i | \rho_i = \rho_j \text{ and } \mathfrak{f}_j = 1\} \neq \emptyset \\ \infty & \text{else.} \end{cases}$$

Definition 1: *Let* $\rho = (\rho_0 \cdots \rho_n) \in E^*$ *be a finite path of* $G = (V, E)$. *For* $0 \leq i \leq m$, *if* Q_{ρ_i} *denotes the generator matrix of edge* ρ_i , *then the occupancy PDF on the corresponding edge is given by:*

$$\pi_{\rho_i}(t_{\rho_i}) := (0, \cdots 0,1).e^{Q_{\rho_i}.dt_{\rho_i}}$$

And the associated probability to park depends on the last element of this vector:

$$p_{\rho_i}(t_{\rho_i}) = 1 - \left(\pi_{\rho_i}(t_{\rho_i})\right)_{[m_{\rho_i}]}$$

where $[.]$ denotes the indexing operator and m_{ρ_i} is the capacity of the edge corresponding to ρ_i. This probability is unconditional in the sense that it does not depend on the route strategy flag $\mathfrak{f}_i$. Next definition takes this information into account.

Definition 2: *Let $\omega = (\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n) \in (E \times \{0,1\})^*$ be a finite parking path of $G = (V, E)$. Then, for each $0 \le i \le n$ the effective probability to park on ρ_i conditioned on the route strategy ω is given by:*

$$p_{\rho_i}^{\omega} = \begin{cases} p_{\rho_i}(t_{\rho_i}) & \text{if } \mathfrak{f}_i = 1 \\ 0 & \text{if } \mathfrak{f}_i = 0 \end{cases}$$

Using the definitions above, we can now define the success probability of a complete route:

Definition 3: *Let $\omega = (\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n) \in (E \times \{0,1\})^*$ be a finite parking path of $G = (V, E)$. The success probability of ω, denoted $\mathbb{P}_\sigma(\omega)$, is defined as the complementary event of not finding any suitable parking spot along ω:*

$$\mathbb{P}_\sigma(\omega) \coloneqq 1 - \left(\prod_{i=0}^{n} 1 - p_{\rho_i}^{\omega}\right)$$

Thanks to this definition, we can stop expanding a route once its chance of success rises above a threshold that, in this work, we fix to 99%. Different routes resulting from different strategies can then be elaborated and compared on the same level. However, we now need some criterion to evaluate a route efficiency, that is, the expected total time to reach the target destination:

Definition 4: *Let $v_t \in V$ be a target vertex and let $\omega = (\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n) \in (E \times \{0,1\})^*$ be a finite parking path of $G = (V, E)$. For $0 \le i \le n$, we denote by $\rho_{\ge i}$ the partial path $(\rho_i \cdots \rho_n)$. Then, the expected total time to reach v_t from ρ_i is defined recursively by:*

$$\mathbb{E}_{v_t}(\rho_{\ge i}) = p_{\rho_i}^{\omega}.\left(t_{\rho_i} + t_{\rho_i}^{v_t}\right) + \left(1 - p_{\rho_i}^{\omega}\right).\mathbb{E}_{v_t}(\rho_{\ge i+1})$$
$$\mathbb{E}_{v_t}(\rho_{\ge n+1}) = 0$$

Where $t_{\rho_i}^{v_t}$ denotes the time needed to reach the destination vertex v_t from edge ρ_i on foot. It can be computed knowing a defined walking speed and thanks to the distance function $d_{v_t}(v)$. In other words, the expected total time is the sum of the time needed to reach a given road and park, then to walk to the destination, weighted by the respective probabilities to park.

This expected total time is, however, of limited interest since it includes a route of arbitrary length depending on the distance between the start and target vertices. In order to compare different paths with each other, even when they don't share the same locations, we need to decompose it into its different subparts: the time to get close enough to the destination, the time spent searching for a parking place and the final time on foot. For that matter, we first define the time the parking strategy actually begins:

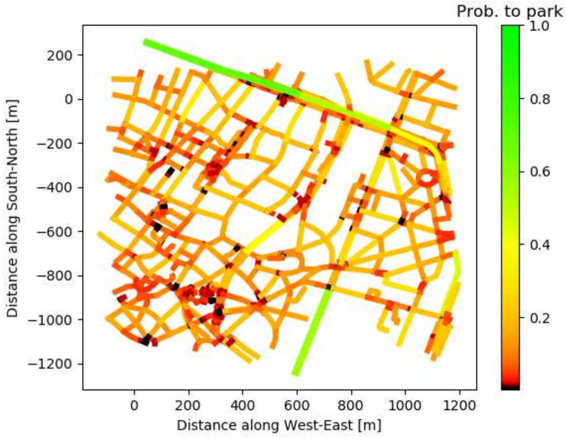

Figure 8: Map background (zoom) located in Belgium at coordinates($\mathbf{4.36}$°; $\mathbf{50.85}$°), with occupancy rate set to 98%.

Definition 5: *Let $\omega = (\rho_0, \mathfrak{f}_0) \cdots (\rho_n, \mathfrak{f}_n) \in (E \times \{0,1\})^*$ be a finite parking path of $G = (V, E)$. Then, the starting time $t^{start}(\omega)$ of the parking path ω is defined by the time of arrival of the first street where the driver should park :*

$$\begin{cases} t^{start}(\omega) \coloneqq t_k \\ k \coloneqq \operatorname*{argmin}_i(\mathfrak{f}_i \mid \mathfrak{f}_i = 1) \end{cases}$$

With this later definition, it is now possible to rewrite the expected total time to park as follows:

$$\mathbb{E}_{v_t}(\omega) \equiv t^{start}(\omega) + \mathbb{E}_{v_t}^{park}(\rho_{\ge 0}) + \mathbb{E}_{v_t}^{walk}(\rho_{\ge 0}),$$

where the first term simply enables to forget the first part of the route that is more related to classical routing than to the strategy itself. Then, the remaining time is split in two parts to differentiate how long we expect the driver to look for a parking place from how long we expect him to walk:

$$\mathbb{E}_{v_t}^{park}(\rho_{\ge i}) = p_{\rho_i}^{\omega}.\left(t_{\rho_i} - t^{start}\right) + \left(1 - p_{\rho_i}^{\omega}\right).\mathbb{E}_{v_t}^{park}(\rho_{\ge i+1})$$
$$\mathbb{E}_{v_t}^{walk}(\rho_{\ge i}) = p_{\rho_i}^{\omega}.t_{\rho_i}^{v_t} + \left(1 - p_{\rho_i}^{\omega}\right).\mathbb{E}_{v_t}^{walk}(\rho_{\ge i+1})$$

4. Insights

In the previous section, we described a methodology to evaluate the efficiency of parking routes returned by a parking strategy, along with the outline of the simulator needed to put it to work. We now turn our attention to the proper analysis of our parking strategy presented in [5]. We begin with a comparison against a worse case bound described earlier, that is, a dumb strategy based on a random walk. The idea is to get a clear understanding of how much time our winning strategy enables us to save in various situations, so that we can understand when it is worthwhile.

In a second step, we study the sensitivity of this strategy to the incomplete knowledge of the parameters regulating the street dynamic. A clear illustration of this situation is the great difficulty to know the probability to park in a street when it tends towards zero, as it requires a huge amount of observations to discriminate between possible occupancy rates that are very close to each other. Therefore, the question that can be asked is how much this information is critical to not lose the benefit of the parking strategy.

4.1. Winning Strategy Benefits

Here, we first look at the winning strategy pure benefits by comparing its expected time to park against the random walk, considering a perfect knowledge of the street occupancy rates and other parameters. This analysis is performed with a simulator working as described in section 3. For each setup, the test is repeated 30 times with random starting and destination locations and the average result is reported.

The background map of the simulation is the one given in Figure 8, whose centre coordinates are $(4.36°;\ 50.85°)$. For simplicity, we consider a uniform occupancy rate across all the streets and a uniform repartition of the parking spots. Because different streets have different lengths, and thus different capacities, the resulting probabilities to park are proportional to those lengths.

As described above, a strategy returns a route long enough to reach 99% probability to park. The simulator plays this route until its end, as if the driver never managed to park, and considers the parking flags to update the probabilities to park in the traversed streets: if a flag is set to true, the probability of the corresponding street is dropped to zero as soon as the car comes out. Then, each probability recovers to its initial value as time passes, depending on each street's dynamic.

It should be noted that considering uniform occupancy rate and parking spots distribution is conservative because at the advantage of the random walk strategy. Indeed, this strategy is a non-informed one and could not benefit of the actual parking distribution if it was not evenly scattered.

Concretely, those 4 different parameters are needed to configure the simulator:

1. The *occupancy rate*, which enables to derive a probability to park from a given street capacity, and thus the ratio λ/μ ;
2. The *mean parking time* of a single car $1/\mu$;
3. The *driving speed* which, combined with the street length, gives its traversal time;
4. The *parking spots density* which, combined with the street length, gives its capacity.

Figure 9 and Figure 10 illustrate the kind of parking routes returned by the winning strategy and by the random walk, respectively. As we can see, the former expands the search around the target location up to some distance and then tries to take advantage of resources reappearance at shorter distances. In addition, it tries to park early without even visiting the target location. On the other hand, the random walk drives straight to the destination, then, it tries its luck randomly. This behaviour is quite bad and we expect a sane driver to do better, but it offers a practical point of comparison.

We compared those two strategies over this set of simulator parameters:

1. Uniform occupancy rate $O \in \{0.95; 0.97; 0.99; 0.995\}$;
2. Uniform mean parking time $1/\mu = 1.5h$;
3. Uniform driving speed of 15 km/h;

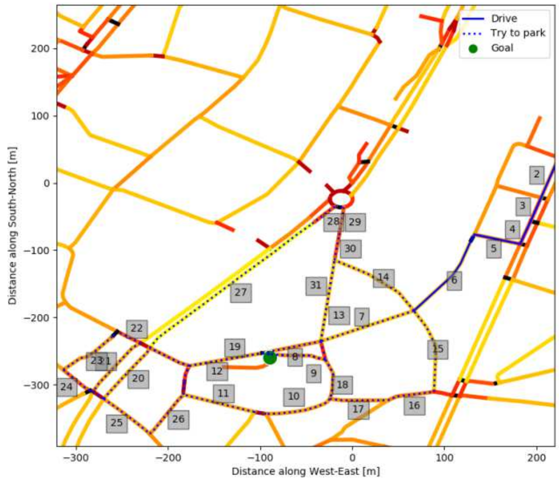

Figure 9: Route generated by the winning strategy.

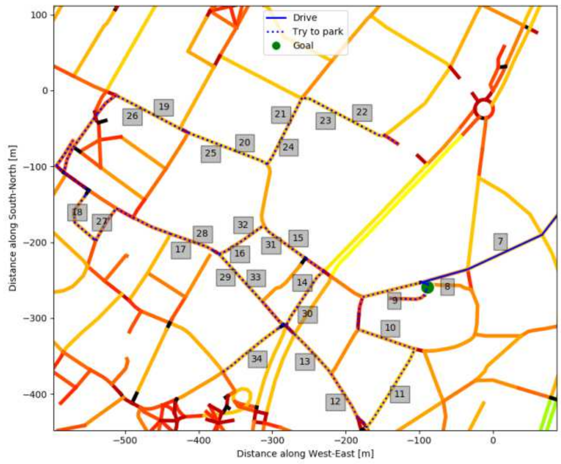

Figure 10: Route generated by a random walk once destination is reached.

4. Uniform parking density of 1 spot by 10 m,

and reported the results in Table 1. For a fair comparison, it should be noted that given a start and end location, two different strategies don't start searching for a parking place at the same time. In other words, $t_{\sigma_{rw}}^{start} \neq t_{\sigma_{ws}}^{start}$. To deal with this shift, we simply match the time zero reference with the most anticipative $t_{\sigma_i}^{start}$ and add an offset penalty to the searching time of the other strategy.

As we could expect, the winning strategy does always better than the random walk. However, what is interesting to look at is the amount of time it finally allows the driver to spare. We can notice that with an occupancy rate as high as 95% the saving is only about 40 seconds, which is quite neglectable. The reason is simply because it is still easy to park in such circumstances and, therefore, there is not a lot to gain. However, the situation is much different in case of very rare parking availability. If a free spot could only be found every 200 parked cars, then the winning strategy offers an interesting saving of about 5 minutes.

Of course, this conclusion is only as strong as the validity of the underlying assumptions, especially the one considering the uniform distribution of the different parameters. Anyway, the difference between those two strategies can only increase if one can benefit from additional knowledge like where the parking spots are or what are the true street travel times due to traffic. However, this is not the primary goal of this study, but rather to understand how strong are the savings given above under imperfect knowledge of the street parameters.

4.2. Winning Strategy Sensitivity

The previous sections gave us an idea of how much time a good parking strategy enables to save in various circumstances. Now, we can focus on the final goal of this study, that is, to understand how much such strategy withstands the partial and erroneous knowledge of the dynamic influencing the probability to park in streets.

The idea to conduct this study consists in computing the parking strategy on a corrupted map, then by analysing its performance (i.e. the expected parking and walking time) on the true original map. If the strategy is near optimal, then, any degradation of the information used to build the parking routes should result in worse expected times once applied back to the ground truth.

Occupancy Rate

The first parameter that we studied is the estimated occupancy rate that directly drives the probabilities to park in the different streets and the result is depicted in Figure 11. The four graphs show the level of expected time to park and walking time as a function of the estimated occupancy rate. The four cases correspond to different ground truth regarding the real occupancy rate. Each time, the performance of the random walk is given by the red line and the other parameters remain fixed.

Let's look at the upper left graph for a real occupancy rate of 95%. What we can notice is that the minimum of the total time to destination occurs in the case the guess is correct. Overestimating the difficulty to park lowers the driving time as the parking strategy anticipates too much and tries to park too far from the goal. As a result, the walking time explodes and may finally ruin any benefit as the total time may become worse than the random walk. In addition, we expect this situation to be very frustrating for the driver as in general one prefers to walk as few as possible.

The situation is quite different for the lower right graph with a real occupancy rate of 99.5%. Underestimating the difficulty results in a situation where the strategy tries to park too close to the goal, which lowers a bit the walking time but increases quite much the parking time. However, even if the total saved time is less, the strategy remains always better than the random walk.

Those graphs also show that a rough and easy guess of the occupancy rates (and thus of the real probabilities to park) may be enough for a parking strategy to operate successfully. For example, estimating the occupancy rate to be 97% everywhere would not degrade the strategy too much when parking is easy (i.e. $\mathcal{O}$ less than 95%), but would still enable to save a good share of the potential gain when parking is hard. In others words, we suggest that knowing the street capacities, along with a rough guess of the *average occupancy rate*

Table 1: Comparison of expected parking time with perfect knowledge of the environment, separated in *search* time and *walking* time, in seconds.

$\mathcal{O}$	Winning Strategy			Random Walk		
	srch.	*wlk.*	***tot.***	*srch.*	*wlk.*	***tot.***
95%	63	98	**161**	79	125	**204**
97%	102	117	**219**	139	170	**309**
99%	284	150	**434**	403	254	**657**
99.5%	540	159	**699**	738	294	**1032**

enables to derive probabilities good enough for a parking strategy like the winning strategy to perform well.

Mean Parking Time

We saw in section 2 that, knowing a street capacity, the steady probability to park only depends on the ratio $r = \lambda/\mu$, that is, the ratio of demand over supply. However, the exact scaling of one of those parameters fixes the churn rate, that is, how often parking events occur in a given time interval. Of both, this is the mean parking time $1/\mu$ that we choose to fix as it has a simple meaning: how long does a single car stay parked in average? Studies show that this value revolves around 1.5 hour, but we can expect it to change depending on the parking context. For example, it may be closer to half an hour around a shopping mall while it should last several hours in a company parking or in a residential area during the night. Anyway, the range of reasonable values is well delimited and our goal is now to determine how much guessing this value right is important for a parking strategy.

Figure 12 illustrates how much the total parking time varies with the estimations, for different real average parking times. The four graphs correspond to an occupancy rate of 99% and the random walk performance is once again depicted by the red line. As expected, the best results always happen when the guesses match the ground truth. If we look at the upper left graph, we see that overestimating the parking time results in a worse exploitation of the parking spots reappearance. The walking time is a bit greater as the strategy prefers to look further from the destination once the nearby streets have been tested.

On the other hand, the lower right graph illustrates the impact of underestimating the parking time. In such a case, the strategy is reluctant to explore too far from the destination, still counting on a good chance to find a free spot nearby because of the advantageous expected churn rate. Then, the search time increases more than the final gain on the walking time as this probability is not that high.

In any case, we can see that this parameter can really have a low impact provided that we chose it wisely. Estimating this value to be around 2 hours enables to benefit from the churn rate without impacting too much the cases where it is not as favourable.

Road Traversal Times

Up to now we have always considered uniform distributions of the parameter we wanted to study. We take here another approach to look at the influence of the traversal time of the map different streets, or equivalently, to the driving speed. Because of inherent imbalances and maladjustments in the road network, time lost in the traffic is far from uniform, even if the average driving speed is quite stable on longer distances. To reflect this situation, we now consider that the average driving speed is constant and well known, but that the precise speed of each street may randomly vary is some defined interval.

The result of this setup is given in Figure 13 for an average driving speed that is always fixed to 15 km/h. The four graphs correspond to different occupancy rates and the curves are plotted for different $[min;\ max]$ speed intervals. The green curve relates to a winning strategy informed of the exact travel time of each street, while the purple one considers only the mean travel speed to elaborate its parking path. As before, the red curve relates to the random walk.

We can see that even the informed winning strategy is affected by the increase of speed dispersion around the average. This seems natural as this modification simply adds new constraints to take into account during the determination of the optimal path. However, this increase remains reasonable and is less than 1.5 minute in the 99.5% occupancy rate case.

In the other hand, both the uninformed winning strategy and the random walk suffer equally well of this difference between the ground truth and their expectation. This first means that the winning strategy remains just as interesting as before, even without the exact knowledge of the travel time, but it also means that knowing the traffic conditions may considerably improve its performances. This situation is even more important that it is very frustrating to get caught in a traffic jam when trying to find a parking place.

Wrapping up

In the previous paragraphs, we showed various graphs giving how much additional time is required if our winning strategy has not a perfect knowledge of the parking conditions. Any deviation deteriorates the potential benefits but not all parameters are equally important. To support this statement, Table 2 gives the impact of the different approximations studied so far, for various occupancy rates. Those numbers correspond to the additional time it would be possible to save when parking if the strategy would have access to the complete information instead to a rough guess. They are only valid in the limited context of those simulations but they highlight nonetheless an important trend.

The mean parking time, directly connected to the churn rate, is maybe the more difficult to collect without metered parking spots. Fortunately, this is also the one showing the lowest sensitivity and we saw that considering an intermediate value around 2 hours give satisfactory results.

The mean occupancy rate, directly connected to the individual street probabilities, proves to have more impact. In addition, a pernicious effect of the winning strategy is that overestimating the difficulty to park can deteriorate the gain by an excess of anticipation, resulting in parking too far from the driver destination. However, this effect can be easily mitigated by the proper choice of an intermediate value. For example, the maximum expected gain in the 99.5 % case is around 6 minutes, but $3/4$ of this saving can be easily achieved just by considering a uniform distribution around 97% occupancy. Off course, a better knowledge of the real probabilities to park enables to improve the gain further, but this moderate increase comes at the cost of difficult means to collect this information. This is indeed an uneasy variable to measure as it requires a global solution to count, either the number of vacant parking spots across the different streets of a city, or the ratio of successful and unsuccessful parking attempts from different drivers. Both approaches are challenging.

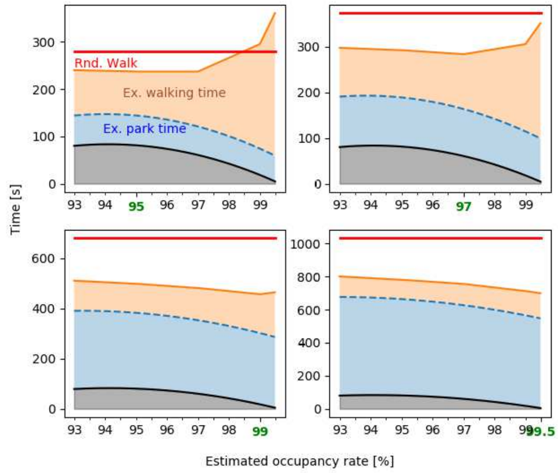

Figure 11: Sensitivity of the winning strategy regarding the estimation of the occupancy rate, for 4 different ground truths (95; 97; 99 and 99.5%).

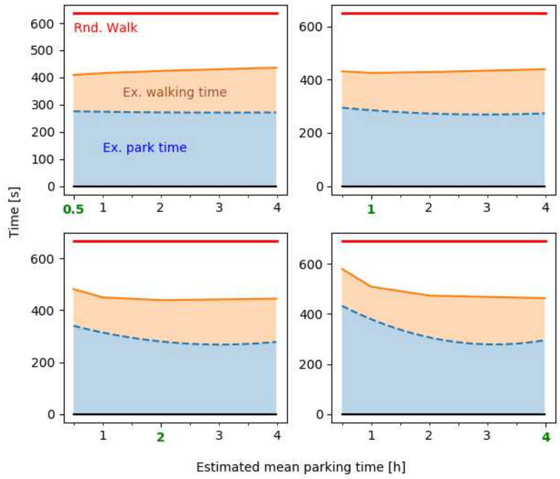

Figure 12: Sensitivity of the winning strategy regarding the estimation of the mean parking time, for 4 different ground truths (0.5; 1; 2 and 4 hours).

Finally, an important information to consider is the traffic influencing the travel time in each street. The average driving speed itself does not impact too much the parking path returned by the winning strategy. However, the dispersion of those speeds among congested and uncongested streets can considerably deteriorate the final savings when the strategy is not aware of this information. Typically, this degradation in our simulations can be four times more important that ignoring the exact probabilities to park. Traffic is thus first choice information to consider to increase the quality the parking strategy.

5. Conclusion

In this paper, we investigate the sensitivity of a probabilistic parking strategy regarding the inaccurate knowledge of the different parameters affecting the way an optimal path is computed. We aim to understand how much knowing the probabilities to park and the other factors impacting the efficiency of a parking strategy is important to preserve the final gain of time experienced by the driver. Because this kind of information is difficult to collect, we propose a complete setup to reasonably simulate various parking situations that enable to fix a ground truth and test a parking path on it. This simulator allows configuring various parameters and supports short and long-term observations regarding the probability to park.

Based on this environment, we propose a methodology to measure the expected time needed to reach a suitable parking spot and to walk to a destination, knowing a parking path elaborated by a parking strategy. A first comparison with a dumb random walk strategy enables to get an idea of how much time this path is efficient and allows the driver to spare time during its journey.

We finally study the sensitivity of one parking strategy, namely the winning strategy, regarding the erroneous estimation of different parameters. More particularly, we run various simulations to understand the impact of inaccurate estimates on the occupancy rates, the mean parking times and on the road traversal times. Thanks to the parking simulator and the test methodology we are able to weight the influence of all those variables regarding the behaviour of the studied strategy. All imprecise estimates degrade the final gain by some factor, but the important finding is the quantification of the lost. We demonstrate that the knowledge of the probability to park and the churn rate provide a lower improvement that taking the traffic and the parking spot locations into account. As this latter information is far easier to collect, we think that it can constitute a first viable step towards the implementation of a helpful parking strategy inside navigation systems.

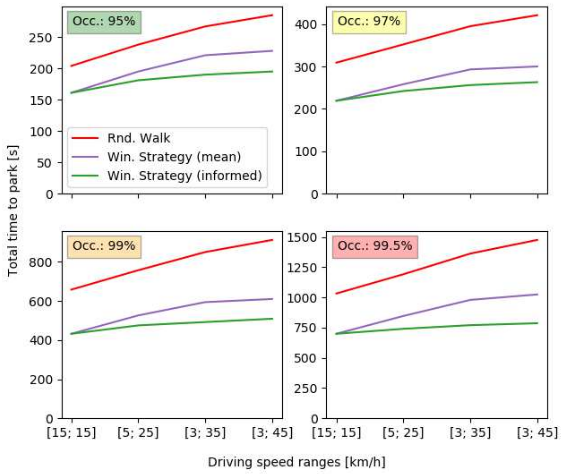

Figure 13: Sensitivity of the winning strategy regarding the driving speed distribution, for an average mean speed of 15km/h.

Table 2: Expected additional parking time in case of various approximations, in seconds, for different occupancy rate conditions.

	Occupancy rate			
Approximation:	**95**	**97**	**99**	**99.5**
Unif. mean parking time of $2h$	5	10	15	19
Unif. occ. rate of 97%	0	0	25	57
Unif. driving speed of $15km/h$	33	37	100	238

6. References

[1] ARMONK, “IBM Global Parking Survey: Drivers Share Worldwide Parking Woes,” IBM, 28 September 2011. [Online]. Available: https://www-03.ibm.com/press/us/en/pressrelease/35515.wss. [Accessed 17 January 2019].

[2] “SF Park,” December 2017. [Online]. Available: http://sfpark.org/. [Accessed January 2019].

[3] “Find & Pay For The Perfect Spot,” Inrix , 2018. [Online]. Available: https://www.parkme.com. [Accessed January 2019].

[4] V. Verroios, V. Efstathiou and A. Delis, “Reaching Available Public Parking Spaces in Urban Environments using Ad-hoc Networking,” *IEEE 12th International Conference on Mobile Data Management,* 2011.

[5] J.-S. Gonsette and N. Meunier, “A Winning Strategy to Park,” in *ITS World Congress 2017* , Montreal, 2017.

[6] M. Caliskan, A. Barthels, B. Scheuermann and M. Mauve, “Predicting Parking Lot Occupancy in Vehicular Ad Hoc Networks,” *2007 IEEE 65th Vehicular Technology Conference - VTC2007-Spring,* pp. 277-281, 2007.

[7] G. Jossé, K. A. Schmid and M. Schubert, “Probabilistic Resource Route Queries with Reappearance,” EDBT, 2015.

[8] T. Arndt, D. Hafner, T. Kellermeier, S. Krogmann, A. Razmjou, M. S. Krejca, R. Rothenberger and T. Friedrich, “Probabilistic Routing for On-Street Parking Search,” in *24th Annual European Symposium on Algorithms (ESA 2016)*, 2016.

[9] I. Bogoslavskyi, L. Spinello, W. Burgard and C. Stachniss, “Where to Park? Minimizing the Expected Time to Find a Parking Space,” *IEEE International Conference on Robotics and Automation (ICRA),* 2015.

[10] K. Lebrun, M. Hubert, P. Huynen, A. D. Witte et C. Macharis, «Les pratiques de déplacement à Bruxelles,» *Cahiers de l’Observatoire de la mobilité de la Région de Bruxelles-Capitale,* 2013.

[11] “Etat de la question en ce qui concerne le stationnement et la politique en matière de stationnement en Région de Bruxelles-Capitale”.

[12] M. Bouleau, “La circulation routière en ïle-de-France en 2010,” 2013.