Ad Hoc Networks 12 (2014) 243–249

Contents lists available at SciVerse ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Finding available parking spaces made easy ☆

Andreas Klappenecker [1], Hyunyoung Lee *,[2], Jennifer L. Welch [3]

Department of Computer Science and Engineering, Texas A& M University, College Station, TX 77843, USA

A R T I C L E   I N F O

Article history:
Received 7 March 2011
Received in revised form 1 January 2012
Accepted 8 March 2012
Available online 17 March 2012

Keywords:
Continuous-time Markov chains
Matrix exponentials
Parking availability
Vehicular ad hoc networks

A B S T R A C T

We discuss the problem of predicting the number of available parking spaces in a parking lot. The parking lot is modeled by a continuous-time Markov chain, following Caliskan, Barthels, Scheuermann, and Mauve. The parking lot regularly communicates the number of occupied spaces, capacity, arrival and parking rate through a vehicular network. The navigation system in the vehicle has to compute from these data the probability of an available parking space upon arrival. We derive a structural result that considerably simplifies the computation of the transition probabilities in the navigation system of the vehicle.

1. Introduction

A navigation system in a car allows the driver to find the destination with ease. Nowadays, most car manufacturers offer the option to install a navigation system, and aftermarket solutions are available as well. In addition to giving driving directions, navigation systems typically also provide a surprisingly accurate estimate of the remaining travel time. Upon arrival at the destination, the search for an available parking space begins.

In many urban or metropolitan areas it can be quite time consuming to find an available parking space. However, the parking problem is more than a nuisance. A recent article in the Washington Post [8] points out that:

> "Hunting for parking produces more than frustration. Shoup studied a 15-block business district in Los Angeles and determined that cruising about 2.5 times around the block for the average of 3.3 min required to find a space added up to 950,000 excess miles traveled, 47,000 gallons of gas wasted and 730 tons of carbon dioxide produced in the course of a year."

The recent advances in sensing and communication technology prompt the question whether vehicular networks can help reduce the time in the search for available parking spaces. There already exist various systems that give information about currently available parking spaces, see e.g. [1,7,10,11,15,16]. Furthermore, there exist pilot systems for metered street side parking [8,13], which monitor the occupancy of each parking space by a sensor. The high costs for sensor deployment and maintenance are recovered by aggressive automated ticketing of parking violations.

The monitoring of each single parking space is not economically sensible in parking lots of airports or malls. However, it is quite feasible to monitor the flow of entering and exiting vehicles to a parking lot of an airport or a mall. We envision a deployment scenario in which comparatively fewer sensors will provide information about the capacity, current occupancy, arrival rates, and parking rates. The information can be distributed through a vehicular ad hoc network or a mobile cellular network.

☆ This is an extended version of a paper that was published in the proceedings of DIALM-POMC'10.

* Corresponding author.

E-mail addresses: klappi@cse.tamu.edu (A. Klappenecker), hlee@cse.tamu.edu (H. Lee), welch@cse.tamu.edu (J.L. Welch).

[1] The research by A.K. was supported in part by NSF Grants CCF 0622201 and CCF 1018500.

[2] The research by H.L. was supported in part by NSF Grant CNS-0614929.

[3] The research by J.W. was supported in part by NSF Grant 0500265 and Texas Higher Education Coordinating Board Grant ARP-00512-0007-2006.

1570-8705/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.adhoc.2012.03.002

A driver might have the following questions when approaching a parking lot:

(a) Will a parking space at a particular parking area be available when I get there in t minutes?
(b) If there is no parking space available, how long do I need to wait until a parking space becomes available?

One important criterion to consider in finding answers to the above questions is the timing factor when the spot is available. Obviously the data gathered using the vehicular network are always from the past. Thus, a natural question is: How can one *predict* the availability of some parking space in a reasonably correct and efficient way? We propose an online scheme that answers this question with a very small communication overhead.

2. Related work

There have been numerous works studying the problem of finding available parking spaces using a vehicular ad hoc network; those closely related to our work are [2,3,10–12].

Mathur et al. [11] motivate the problem of finding available parking spaces, discuss research challenges and propose some possible solutions. They discuss a centralized architecture and a distributed architecture for their solutions. In the centralized solution, some cars are equipped with ultrasonic sensors, which drive past the parking spaces to collect the occupancy data, and upload the data to the centralized database. The cars that need to park simply query the centralized database. The idea of this centralized architecture is deployed and experimented with in ParkNet [10]. The distributed architecture uses car-to-car communication and Mathur et al. [11] address the importance of having such a distributed architecture.

A network model that consists of onboard units on vehicles and static road side units is proposed by Panayappan and Trivedi [12] with an emphasis on secure communication between these units. In the paper by Caliskan, Graupner and Mauve [3], the parking area is modeled as a grid, and schemes for information aggregation and dissemination over the grid are proposed.

Our work studies how to use the information collected by such vehicular ad hoc networking systems, as the ones proposed above. Caliskan et al. [2] introduced a simple and elegant parking lot model that we adopt here. We review this model and give some background in Section 3. We then show how to explicitly calculate the transition probabilities, which was claimed to be a difficult problem by Caliskan et al. [2]. Our main result provides a factorization of the transition probability matrix that can be easily evaluated. The chief benefit of our solution is that it is suitable for implementation in the navigation system of vehicles.

3. The parking lot model

In this section, we review the parking lot model that was introduced by Caliskan et al. [2] and give the necessary background on continuous-time Markov chains.

Suppose that we are interested in parking at a parking lot with n parking spaces. Since we are only concerned with predicting whether or not the parking lot is full, it suffices to model the number of occupied parking spaces. For all times $t \geqslant 0$, let $X(t)$ denote a random variable with values in $\{0, 1, \ldots, n\}$. We denote by $\Pr[X(t) = u]$ the probability that at time t there are precisely u parking spaces occupied. The future availability of the parking spaces depends on the present occupancy, but not really on past occupancies. Therefore, one can model the changes of the parking spots by a continuous-time Markov chain. By definition, the stochastic process $\{X(t) | t \geqslant 0\}$ is a continuous-time Markov chain if and only if

$$\begin{aligned}\Pr[X(t+s) = u | X(s) = v, X(s_1) = w_1, \ldots, X(s_n) = w_n] \\ = \Pr[X(t+s) = u | X(s) = v] \end{aligned} \tag{1}$$

holds for all $n \geqslant 0$ and for all occupancy numbers u, v, $w_1, \ldots, w_n$ in $\{0, 1, \ldots, n\}$ and all nonnegative real numbers $s_1 < s_2 < \cdots < s_n < s$ and t. This formalizes the notion that the future occupancy of the parking spots depends on the present occupancy but not on past occupancies. We will make a small excursion and introduce some terminology from the theory of continuous-time Markov chains.

The Markov chain is called homogeneous if the right-hand side of Eq. (1) does not depend on s. For simplicity, we will assume that our Markov chains are homogeneous. Furthermore, we will assume that the transition probabilities $\Pr[X(t+s) = u | X(s) = v]$ are right-continuous at $t = 0$, meaning that

$$\lim_{t \to 0+} \Pr[X(t+s) = u | X(s) = v] = \begin{cases} 1 & \text{if } u = v, \\ 0 & \text{if } u \neq v. \end{cases}$$

This assumption ensures that the Markov chain has with probability 1 only a finite number of state changes in small time intervals.

Let us define the probability transition matrices

$$P(t) = (\Pr[X(t+s) = u-1 | X(s) = v-1])_{1 \leqslant u, v \leqslant n+1}$$

for all real numbers $t \geqslant 0$. Notice that the indices of matrices and vectors consistently range from 1 to $n+1$ in this paper.

Obviously, the behavior of the Markov chain is completely determined by the initial probability distribution on its states and the probability transition matrices. Some basic properties of the probability transition matrices are:

(a) $P(t)$ is a stochastic matrix for all $t \geqslant 0$, that is, each row of a probability transition matrix sums to 1.
(b) $P(r+s) = P(r)P(s)$ holds for all real numbers $r, s \geqslant 0$.
(c) $\lim_{h \to 0+} P(h) = P(0)$ is the identity matrix.

Therefore, $\{P(t) | t \geqslant 0\}$ is a continuous semigroup (recall that a semigroup is a set that is equipped with a binary associative operation). The continuity of this semigroup has the somewhat unexpected consequence that $P(t)$ is infinitely differentiable with respect to $t > 0$, see [9, p. 164]. Therefore,

$$Q = \lim_{h \to 0+} \frac{1}{h}(P(h) - I)$$

is a well-defined $(n+1)\times(n+1)$ matrix, called the *generator* or *transition matrix* of the Markov chain.

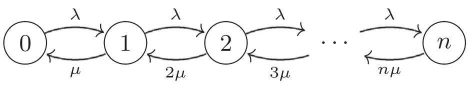
The transition matrix figures prominently in the Chapman-Kolmogorov differential equation

$$P(t)' = \lim_{h\to 0}\frac{P(t+h)-P(t)}{h} = \left(\lim_{h\to 0}\frac{P(h)-I}{h}\right)P(t) = QP(t)$$

of the transition semigroup. It follows that $P(t) = e^{tQ}$ is the unique solution to this system of linear differential equations with initial condition $P(0) = I$. Thus, a continuous-time Markov chain is completely specified by its transition matrix Q and the probability distribution on the states at the time $t = 0$.

We now return from our excursion and detail the Markov-chain model of a parking lot that was introduced by Caliskan et al. [2]. The arrival of vehicles at the parking lot can be modeled by a Poisson process with arrival rate λ. It is assumed that each vehicle occupies a single parking space. Furthermore, it is assumed that the time a parking spot is occupied can be modeled by an exponential distribution with parameter μ. The parameter μ will be called the parking rate. If the parking lot is full (meaning it is occupied by n vehicles), then any arriving vehicle will be rejected.

Therefore, the state transitions of this parking lot model are given by



If we express this in the form of a transition matrix Q, we obtain

$$Q = \begin{pmatrix} -\lambda & \lambda & & & & \\ \mu & -(\lambda+\mu) & \lambda & & & \\ & 2\mu & -(\lambda+2\mu) & \lambda & & \\ & \ddots & \ddots & \ddots & & \\ & & & (n-1)\mu & -(\lambda+(n-1)\mu) & \lambda \\ & & & & n\mu & -n\mu \end{pmatrix} \quad (2)$$

In other words, the transition matrix $Q = (q_{ij})_{1\leqslant i,\ j\leqslant n+1}$ is a tridiagonal $(n+1)\times(n+1)$ matrix with entries

(i) $q_{i,i+1} = \lambda$ for $1 \leqslant i \leqslant n$ on the superdiagonal,
(ii) $q_{ii} = -(\lambda + (i-1)\mu)$ for $1 \leqslant i \leqslant n$ and $q_{n+1,n+1} = -n\mu$ on the diagonal, and
(iii) $q_{i+1,i} = i\mu$ for $1 \leqslant i \leqslant n$ on the subdiagonal.

In the language of queuing theory, the parking lot is modeled by an $M/M/n/n$ queue.

4. The parking availability algorithm

Currently, the navigation system of a vehicle can estimate the time of arrival at a destination. If the destination is, say, the parking lot of a mall or of an airport terminal, then it would be helpful to estimate the probability that a parking space will be available.

Suppose that the parking lot has n parking spaces, monitors the number u of occupied parking spaces, estimates the arrival rate λ, and the parking rate μ at time s. If the parking lot regularly disseminates these five parameters (s, n, u, λ, μ) in a vehicular ad hoc network, then the navigation system can in principle estimate the probability of the availability of parking spot. Indeed, let $\pi(s)$ be the vector of length $n+1$ that has an entry 1 at position $u+1$ and zeros everywhere else. The vector $\pi(s)$ represents the probability distribution that the occupancy number is in state u with probability 1. If the navigation system estimates that the vehicle will arrive at this parking lot at time $t+s$, then the probability distribution of the occupancy number is given by $\pi(s+t) = \pi(s)e^{tQ}$. Therefore, the probability that the parking space is not full at the time of arrival is $1-\pi(s+t)_{n+1}$, since the $(n+1)$-th component of the probability distribution $\pi(s+t)$ of the occupancy number is the probability that the parking lot is fully occupied at time $s+t$.

However, Caliskan, Barthels, Scheuermann and Mauve caution us that "Like evaluating the matrix exponential operator in general, calculating $\pi(s)e^{tQ}$ is non-trivial", see [2, p.279] . In this section, we solve the problem of evaluating the matrix exponential function e^{tQ} by investigating the structure of the matrix Q.

Theorem 1. *The probability transition matrix P(t) for the generator matrix Q given in (2) is given by*

$$P(t) = e^{tQ} = D^{-1}B^T e^{tE} BD,$$

where D and E are diagonal matrices and B an orthogonal matrix. These matrices will be explicitly defined in the proof below.

Proof. Let us define the diagonal matrix

$$D = \mathrm{diag}(d_1, d_2, \ldots, d_{n+1})$$

with entries given by

$$d_k = \begin{cases} 1 & \text{if } k = 1, \\ \frac{1}{\sqrt{k-1}} d_{k-1}\sqrt{\lambda/\mu} & \text{if } k \in \{2,3,\ldots,n+1\}. \end{cases}$$

Then $R = DQD^{-1} = (r_{ij})_{1\leqslant i,j\leqslant n+1}$ is tridiagonal matrix with entries

$$\begin{aligned} r_{ii} &= d_i q_{ii} d_i^{-1} = q_{ii} && \text{for } 1 \leqslant i \leqslant n+1, \\ r_{i,i+1} &= d_i q_{i,i+1} d_{i+1}^{-1} = \sqrt{i\lambda\mu}, && \text{for } 1 \leqslant i \leqslant n, \\ r_{i+1,i} &= d_{i+1} q_{i+1,i} d_i^{-1} = \sqrt{i\lambda\mu} && \text{for } 1 \leqslant i \leqslant n. \end{aligned}$$

Therefore, R is a real symmetric tridiagonal matrix with the same entries $a_i := r_{ii} = q_{ii}$ on the diagonal as Q and the entries $b_i := r_{i,i+1} = r_{i+1,i} = \sqrt{i\lambda\mu}$ on the super- and subdiagonals

$$R = DQD^{-1} = \begin{pmatrix} a_1 & b_1 & 0 & \cdots & 0 \\ b_1 & a_2 & b_2 & \ddots & \vdots \\ 0 & b_2 & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & b_n \\ 0 & \vdots & 0 & b_n & a_{n+1} \end{pmatrix}.$$

Our next goal is to explicitly construct an orthogonal matrix B with rows given by eigenvectors of R and a diagonal matrix E with the corresponding eigenvalues as diagonal entries such that

$$R = B^T EB. \tag{3}$$

This will allow us to write the generator matrix Q in the form

$$Q = D^{-1}B^T EBD. \tag{4}$$

As a consequence, the probability transition matrix $P(t) = e^{tQ}$ can be factorized in the form

$$P(t) = D^{-1}B^T e^{tE} BD, \tag{5}$$

as claimed. This factorized form of $P(t)$ allows one to easily calculate $P(t)$, since the matrix exponential e^{tE} of a diagonal matrix $E = \mathrm{diag}(\lambda_1, \ldots, \lambda_{n+1})$ is simply given by exponential functions of the diagonal entries of E, namely

$$e^{tE} = \mathrm{diag}(e^{t\lambda_1}, \cdots, e^{t\lambda_{n+1}}).$$

Thus, it remains to find the matrix factorization (3).

We relate the matrix R to a sequence of polynomials $p_{-1}(x)$, $p_0(x)$, $p_1(x), \ldots, p_{n+1}(x)$ in the ring $\mathbf{R}[x]$ of polynomials in the variable x with real coefficients (cf. [5]). It turns out that these polynomials are orthogonal with respect to some positive Borel measure. This allows us exploit the theory of orthogonal polynomials to explicitly construct the eigenvectors of R, which form the rows of the matrix B.

Thus, let us associate to the matrix R the family

$$\{p_k(x) | -1 \leqslant k \leqslant n+1\}$$

of polynomials in $\mathbf{R}[x]$ such that $p_{-1}(x) = 0$, $p_0(x) = 1$, and satisfying the three-term recurrence relation

$$b_{k+1}p_{k+1}(x) = (x - a_{k+1})p_k(x) - b_k p_{k-1}(x) \tag{6}$$

for $0 \leqslant k \leqslant n$, where $b_0 = b_{n+1} = 1$ and $b_i = \sqrt{i\lambda\mu}$ for $1 \leqslant i \leqslant n$. Thus, the polynomial $p_k(x)$ is of degree k for $1 \leqslant k \leqslant n+1$. By Favard's theorem [6,14], there exists a probability distribution $F(x)$ such that the polynomials p_k satisfy the orthogonality relation

$$\int_{-\infty}^{\infty} p_j(x)p_k(x)dF(x) = 0$$

whenever the indices j and k are distinct. It follows from this orthogonality relation that the polynomial $p_k(x)$ has k distinct real roots [14]. Furthermore, the roots of the polynomial $p_{k-1}(x)$ lie strictly between the roots of $p_k(x)$, a property known as interlacing, see [14].

The next property will make it clear why the polynomials $p_k(x)$ were introduced in the first place. Keeping in mind that $b_{n+1} = 1$, the recurrence (6) yields

$$xV(x) = RV(x) + p_{n+1}(x)e_{n+1},$$

where $V(x)$ is the vector $V(x) = (p_0(x), p_1(x), \ldots, p_n(x))^T$ of polynomials and $e_{n+1} = (0, \ldots, 0, 1)^T$. If we substitute any root λ of $p_{n+1}(x)$ into this equation, then $V(\lambda) \neq 0$, because of the interlacing property. Thus, we find that $V(\lambda)$ is an eigenvector to the eigenvalue λ, since

$$\lambda V(\lambda) = R\,V(\lambda) + p_{n+1}(\lambda)e_{n+1} = R\,V(\lambda)$$

holds. Let $\{\lambda_k | 1 \leqslant k \leqslant n+1\}$ be the set of roots of the polynomial $p_{n+1}(x)$. Since the roots are pairwise distinct, the $n+1$ vectors $V(\lambda_k)$ with $1 \leqslant k \leqslant n+1$ are $n+1$ eigenvectors of R. These eigenvectors of the real symmetric matrix R are orthogonal, since their eigenvalues are distinct.

Let $v_k = \frac{1}{\|V(\lambda_k)\|}V(\lambda_k)^T$ be the row-vectors obtained from $V(\lambda_k)$ by normalization to unit length and transposition. If we define $B = (v_k)_{1 \leqslant k \leqslant n+1}$, then this is an orthogonal matrix. Let $E = \mathrm{diag}\,(\lambda_1, \ldots, \lambda_{n+1})$. Then $R = B^T EB$, and it follows that $P(t) = e^{tQ} = D^{-1}B^T e^{tE}BD$, as claimed. □

The previous theorem asserts in particular that the transition matrix Q given in (2) is diagonalizable. One can follow the construction of the orthogonal matrix B given in the proof by calculating the polynomials $p_k(x)$ and determining the roots of $p_{n+1}(x)$, but one might as well use efficient algorithms to find eigenvalues and eigenvectors of a tridiagonal matrix.

The previous theorem can be used to calculate the probability distribution of the occupancy number at a given time. Let us assume a floating point implementation. We assume constant complexity for elementary floating point operations such as multiplication, exponentiation, forming square roots, and raising to a power.

Corollary 2. *Given the data (s, n, u, λ, μ) of a parking lot with n parking spaces that has arrival rate λ, parking rate μ, and u occupied parking spaces at time s, there exists an $O(n^2)$ algorithm that computes the probability distribution $\pi(s+t)$ of the occupancy number at time $s+t$ for any $t \geqslant 0$.*

Proof. Given λ and μ, one can construct the matrices D and D^{-1} in linear time directly from the definition. Furthermore, the tridiagonal matrix Q can be constructed in sparse form with $O(n)$ operations. Therefore, the matrix $R = DQD^{-1}$ can be formed in $O(n)$ time as a product of sparse matrices. One can compute the matrices B^T, E, and B in $O(n^2)$ time by eigenvector and eigenvalue calculations using Dhillon's algorithm [4]. Given $E = (\lambda_1, \ldots, \lambda_{n+1})$, the vector $e^{tE} = (e^{t\lambda_1}, \ldots, e^{t\lambda_{n+1}})$ can be obtained in $O(n)$ time.

The probability distribution $\pi(s)$ of the occupancy number is given by the vector in $\mathbf{R}^{n+1}$ that is 1 at position $u+1$ and zero otherwise, since the occupancy number has been observed at time s. Thus, we can calculate the probability distribution $\pi(s+t) = \pi(s)D^{-1}B^T e^{tE}BD$ using standard vector–matrix multiplication in $O(n^2)$ time. □

The previous corollary allows one to answer the question (a) that we raised in the introduction: Will a parking space be available at a parking area when I get there in t minutes? An answer to the question (b) of the introduction is given in the next proposition.

Proposition 3. *Suppose that a parking lot with n parking spaces is occupied. If the parking rate is μ, then the expected time for a vehicle to leave the parking lot is $1/(n\mu)$.*

Proof. For $1 \leqslant k \leqslant n$, let X_k be the random variables measuring the time until the vehicle in parking space k leaves the parking lot. These random variables are exponentially distributed with parameter μ. The time until the next parking space becomes available is thus given by $Y = \min\{X_1, \ldots, X_n\}$, which is an exponentially distributed random variable with parameter $n\mu$. Therefore, $E[Y] = (n\mu)^{-1}$ is the expected time for the next vehicle to leave the parking lot. □

5. Examples

In this section, we illustrate the theorems of the previous section by giving two different examples. In the first example, we have the situation that the parking lot is already quite full and new cars arrive with high frequency. This could for instance be the parking lot of a mall just at the beginning of lunch time. The second example is a parking lot that is currently full, but cars are now arriving less frequently. In each case, we give a few sample probability distributions of the parking lot occupancy after a few minutes. We have chosen the large capacity $n = 1000$ just to demonstrate that calculating the matrix exponential of 1001×1001 matrices can be easily done.

Example 1. Let us suppose that a parking lot at our destination has a capacity of $n = 1000$ parking spaces. The parking lot communicates the information

$$(s, n, u, \lambda, \mu) = (s, 1000, 900, 1000/3060, 1/3060).$$

In other words, at time s there were $u = 900$ occupied spaces. The average parking time is $1/\mu = 51 \times 60 = 3060$ s, and the arrival rate is $\lambda = 1000/3060$ or approximately 0.3268 cars arrive per second. If the arrival rate and parking rates stay this way, then by Little's law we can expect to have in the long run on average $\lambda/\mu = 1000$ cars parked in the parking lot, that is, the parking lot is expected to fill up completely. So the real question is: can we hope to still get a parking spot if we arrive there soon?

Figs. 1–3 depict the probability distributions of the occupancy vector π for 1, 4, and 16 min after time s, respectively. Suppose that the information is received by the vehicle less than a minute after time s. Suppose that we will arrive at the destination in 15 min or less. From Fig. 3, one can see that there is still a good chance that we will be able to get a parking spot. In fact, the probability that the parking lot is full at time s + 16 min is predicted to be less than 10^{-4}.

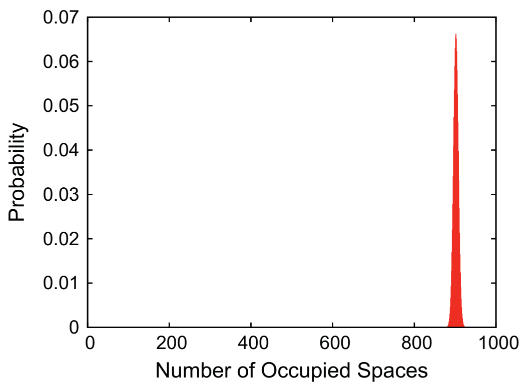


Fig. 1. Probability distribution π of the occupancy number of Example 1 at time $s + 60$ s.

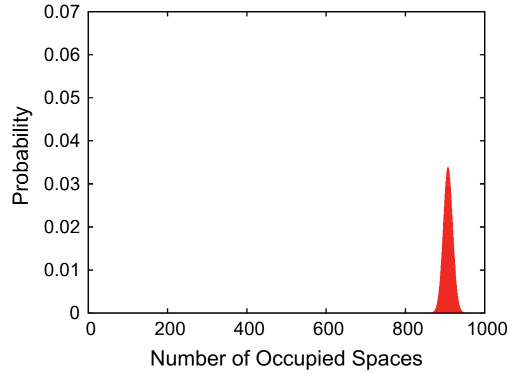


Fig. 2. Probability distribution π of the occupancy number of Example 1 at time $s + 4 \times 60$ s.

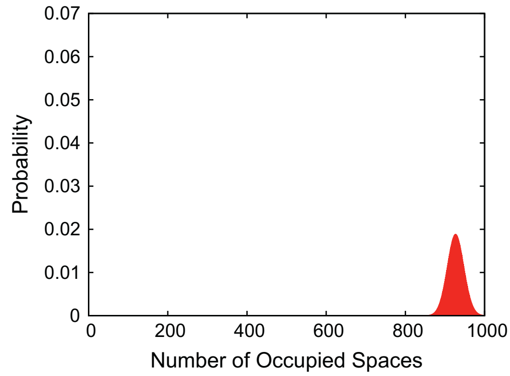


Fig. 3. Probability distribution π of the occupancy number of Example 1 at time $s + 16 \times 60$ s.

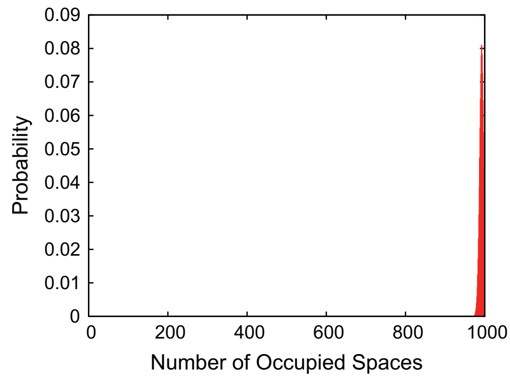


Fig. 4. Probability distribution π of the occupancy number of Example 2 at time $s' + 60$ s.

Example 2. Let us consider the same parking lot at a different point in time. The parking lot communicates the data

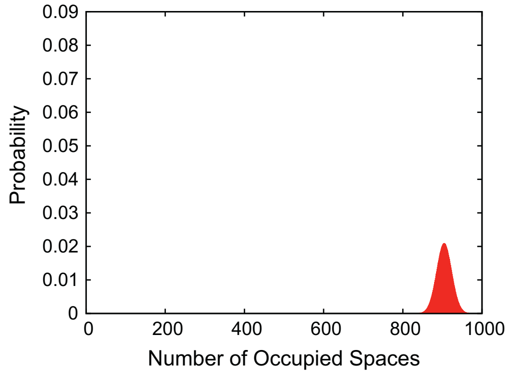


Fig. 5. Probability distribution π of the occupancy number of Example 2 at time $s' + 16 \times 60$ s.

$$(s', n, u, \lambda, \mu) = (s', 1000, 1000, 650/3060, 1/3060).$$

Thus, the parking lot was full at time s' with $n = 1000 = u$. However, fewer cars arrive now per second than in Example 1. An immediate consequence is that one minute after s', the probability that the parking lot is full is already less than 0.025, even though this might not be apparent from Fig. 4. Sixteen minutes after s', the probability that the parking lot is full is negligible, see Fig. 5.

6. Conclusions

We investigated the parking lot model that was introduced by Caliskan et al. [2]. This model aptly describes parking lots at malls and airports that have an admission control such as a gate and monitor the occupancy of the parking lot. If the admission is strictly enforced then the continuous-time Markov model studied here appears to be a good fit. We showed that communicating a minimal amount of data over a vehicular ad hoc network is sufficient to make efficient prediction of future occupancies. Gathering data from various alternative parking locations could lead to an informed decision, saving gasoline and time.

One referee suggested to go a bit further and take into account how far the parking lot is from the destination. This way, the sum of driving time, parking time, and time to walk to the destination can be used to decide which choice of parking lot is optimal.

Our method can be applied in different contexts as well. For example, the method can be used, essentially without change, for the prediction of space in drive-in fast food restaurants. However, for other types of restaurants, this will not work without modification. Nevertheless, we expect that a similar approach can be used.

Acknowledgments

We thank the referees for their insightful comments, corrections, and suggestions for further research.

References

[1] V.G. Ashok, A. Gupta, S. Shiva, H. Iyer, D. Gowda, A. Srinivas, A novel parking solution for metropolitan parking garages, in: The 3rd WSEAS International Conference on Urban Planning and Transportation (UPT'10), 2010, pp. 153–159.

[2] M. Caliskan, A. Barthels, B. Scheuermann, M. Mauve, Predicting parking lot occupancy in vehicular ad-hoc networks, in: 65th IEEE Vehicular Technology Conference, VTC 2007, IEEE Press, Dublin, Ireland, 2007, pp. 277–281.

[3] M. Caliskan, D. Graupner, M. Mauve, Decentralized discovery of free parking places, in: The 3rd Intl. Workshop on Vehicular Ad Hoc Networks, VANET 2006, ACM, Los Angeles, California, USA, 2006, pp. 30–39.

[4] J.S. Dhillon, A New $O(n^2)$ Algorithm for the Symmetric Tridiagonal Eigenvalue/Eigenvector Problem, Ph.D. thesis, UC Berkeley, Berkeley, CA, 1997.

[5] M. Elouafi, A.D. Aiat Hadj, A Takagi factorization of a real symmetric tridiagonal matrix, Appl. Math. Sci. 46 (2008) 2289–2296.

[6] J. Favard, Sur les polynômes de tchebicheff, C.R. Acad. Sci. Paris 200 (1935) 2052–2053.

[7] S. Funck, N. Mohler, W. Oertel, Determining car-park occupancy from single images, in: Proceedings of IEEE International Symposium on Intelligent Vehicles (IVS'04), 2004, pp. 325–328.

[8] A. Halsey III, DC tests new parking technology to help drivers find space, pay more easily, The Washington Post, June 29 2010. <http://www.washingtonpost.com/wp-dyn/content/article/2010/06/28/AR2010062804850.html>.

[9] M. Kijima, Markov Processes for Stochastic Modelling, Chapman & Hall, 1997.

[10] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrashekharan, W. Xue, M. Gruteser, W. Trappe, ParkNet: drive-by sensing of road-side parking statistics, in: The 8th Annual Intl. Conf. on Mobile Systems, Applications and Services, MobiSys 2010, ACM, San Francisco, California, USA, 2010, pp. 1–14.

[11] S. Mathur, S. Kaul, M. Gruteser, W. Trappe, ParkNet: a mobile sensor network for harvesting real time vehicular parking information, in: MobiHoc S3 2009, ACM, New Orleans, Louisiana, USA, 2009, pp. 1–3.

[12] R. Panayappan, J.M. Trivedi, VANET-based approach for parking space availability, in: The 4th Intl. Workshop on Vehicular Ad Hoc Networks, VANET 2007, ACM, Montreal, Quebec, Canada, 2007, pp. 75–76.

[13] W. Roush, Streetline unveils iphone parking app, seeks to take guesswork out of finding a spot, Xconomy, San Francisco, <http://www.xconomy.com/> (22.12.10).

[14] G. Szegö, Orthogonal Polynomials, rev. ed., Colloquium Publications, American Mathematical Society, New York, 1959.

[15] V.W.S. Tang, Y. Zheng, J. Cao, An intelligent car park management system based on wireless sensor networks, in: The 1st International Symposium on Pervasive Computing and Applications, 2006, pp. 65–70.

[16] J. Wolff, T. Heuer, H. Gao, M. Weinmann, S. Voit, U. Hartmann, Parking monitor system based on magnetic field sensors, in: IEEE Conference on Intelligent Transportation Systems, 2006, pp. 1275–1279.

Andreas Klappenecker received his Ph.D. in Computer Science from the University of Karlsruhe in Germany in 1998. He joined Texas A&M University in 1999 as a Visiting Assistant Professor of Mathematics, and worked shortly as a Research Associate at the University of Karlsruhe before accepting his position in Computer Science and Engineering at Texas A&M University, where he is now an Associate Professor. His research interests include the design and analysis of algorithms, in particular quantum algorithms, signal and image processing, and cryptography.

Hyunyoung Lee received the BS and MS degrees from Ewha University in 1987 and 1992, respectively, the MA degree from Boston University in 1998, and PhD degree from Texas A&M University in 2001, all majoring in Computer Science. She worked as a systems programmer at Korean Air, as a member of technical staff at Bellcore (now Telcordia), and as an assistant professor of computer science at the University of Denver, and currently, is a research assistant professor of computer science and engineering at Texas A&M University. Her main area of research is distributed computing.

Jennifer L. Welch received the BA degree from the University of Texas at Austin in 1979 and the SM and PhD degrees from the Massachusetts Institute of Technology in 1984 and 1988, respectively. She has been on the faculty at Texas A&M University since 1992, where she is now a professor of computer science and engineering. Her research interest includes distributed computing.