MODELING PARKING SEARCH ON A NETWORK USING STOCHASTIC SHORTEST PATHS WITH HISTORY DEPENDENCE

Shoupeng Tang*
PhD candidate
The University of Texas at Austin
Department of Civil, Architectural and Environmental Engineering
301 E. Dean Keeton St. Stop C1761
Austin, TX 78712
Tel: (512) 471-4622, Fax: (512) 475-8744
Email: sptang@utexas.edu

Tarun Rambha
PhD candidate
The University of Texas at Austin
Department of Civil, Architectural and Environmental Engineering
301 E. Dean Keeton St. Stop C1761
Austin, TX 78712
Email: tarun.1988@gmail.com

Reese Hatridge
Undergraduate Research Assistant
The University of Texas at Austin
Department of Civil, Architectural, and Environmental Engineering
301 E. Dean Keeton St. Stop C1761
Austin, TX 78712
Email: reesehatridge@gmail.com

Stephen D. Boyles
Assistant Professor
The University of Texas at Austin
Department of Civil, Architectural and Environmental Engineering
301 E. Dean Keeton St. Stop C1761
Austin, TX 78712
Email: sboyles@mail.utexas.edu

Avinash Unnikrishnan
Assistant Professor
Room 621 ESB, PO Box 6103
Department of Civil and Environmental Engineering
West Virginia University
Morgantown, WV 26506
Email: Avinash.Unnikrishnan@mail.wvu.edu

Submitted for Presentation at the Transportation Research Board (TRB) 93nd Annual Meeting and Publication in the Transportation Research Record
First submitted on August 1, 2013
Revised and submitted on November 15, 2013

5641 + 4 Figures+2 TABLES = 7141 words total

* Corresponding author

ABSTRACT

A substantial amount of urban traffic is related to drivers searching for parking. This paper develops an online stochastic shortest path (SSP) model to represent the parking search process in which, drivers must choose whether to park at an available space or continue searching for a space closer to the destination. Existing online shortest path algorithms have been formulated for the "full reset" or "no reset" assumptions on revisiting links. As described in this paper, neither is fully suitable for the parking search process. Accordingly, this paper proposes an "asymptotic reset" model which generalizes both the "full reset" and "no reset" cases and uses the concept of "reset rate" to characterize the temporal dependence of parking probabilities on earlier observations. In this model, drivers try to minimize their expected travel cost, which includes driving cost and the cost of walking from a parking spot to the actual destination conditioned on the parking availability on m most recently traversed links. The problem is formulated as a Markov decision process and is demonstrated using a network representing the neighborhood of the University of Wyoming campus. The case study successfully shows the "extra time" used by drivers to cruise for an acceptable parking space and also illustrates the impact of m on the computational effort required to compute an optimal policy.

Keywords: Markov decision process, parking search, stochastic shortest path, history dependence

1. INTRODUCTION

With vehicle ownership rates peaking in the U.S. and rising in most other countries across the globe (*1*), the impact of parking on urban transportation networks, urban development, the environment, and the lives of individuals are becoming more pronounced. In urban centers where space is limited and cars are prevalent, parking shortages cause drivers to "cruise" for parking, a frustrating endeavor both for drivers and city planners(*2*). By averaging the results of ten different international studies, Shoup found that approximately 34% of congestion in urban areas consists of people "cruising" for parking (*1*). Analyzing the problem from a driver's perspective, a study in Frankfurt showed that searching for a parking spot during peak hours accounted for as much as 40% of the total travel time for journeys to central urban areas (*3*).

Clearly, there is a need for city planners to better manage parking in their jurisdictions – steps which are possible due to the recent trend by youth away from personal vehicles (*4*, *5*). While recent programs in San Francisco (*6*), Boston (*7*), Seattle, Washington D.C., and elsewhere (*8*) can provide much-needed data, sophisticated and accurate parking models are needed to help forecast the effects of changes in capacity, pricing, and strategies aimed at reducing the number of single person trips by car. Similarly, more accurate parking guidance systems would significantly reduce congestion, enabling the rest of the network to run more efficiently.

Unfortunately, existing parking search models usually make extreme simplifying assumptions and may not be able to capture real parking behavior. For instance, based on the two parking regimes they propose, Arnott and Inci (*9*) assume that if vacant spaces are present, drivers will not cruise for parking. Leurent and Boujnah (*10*) assume that if a driver's first choice for parking is not available, they will continue transferring to other lots based on discrete-choice derived probabilities, implying an absence of circling. The two papers present the typical assumptions on parking search models. They assume that travelers will follow the shortest path to their ideal parking spot closest to the destination, and if no space is available, travelers continue searching until they find one. Under these assumptions, the probabilities of finding parking at any location do not depend on past parking availability history.

In reality, since drivers are unaware of the exact likelihood of finding parking near their destination and circling wastes time, they will typically pay attention to availability as they approach the area. Their parking choice will then be a memory-influenced decision, where they circle back to a previously seen spot if necessary. To model this reality, this paper proposes a stochastic parking search approach with recourse which incorporates memory of parking availability on recently-traversed links. Drivers may use this information to reevaluate their route and parking choice every time they get to a new network link. In our model drivers seek to minimize their total expected travel time, driving plus walking, though this disutility function could easily be adapted to incorporate fees, search time, or a host of other characteristics. We name our approach an "asymptotic reset" model, as it generalizes the "reset" and "no reset" formulations identified by Provan (*11*).

The subsequent sections of this paper are organized as follows. Section 2 discusses previous research on parking and the other modeling apporaches. Section 3 presents our model and solution methods. Section 4 provides a case study, using the University of Wyoming campus network. It details how to apply our model in practice and explains how drivers optimize their decision making. In Section 5, concluding remarks and suggestions for future work are provided.

2. OVERVIEW OF EARLIER STUDIES

Recognizing the impact of parking searches on urban congestion, many studies have been conducted using economic, statistical, and optimization frameworks on various aspects of parking. Based on the approach taken, these models can be classified into three groups: discrete choice based approaches, simulation based approaches, and network assignment based approaches.

Both discrete choice and simulation approaches examine parking choice explicitly. Discrete choice models work at the macro level, using random utility theory to understand parking choice as a function of various driver and parking facility attributes. Such models differ in their complexity, with some utilizing the multinomial logit model (*12, 13, 14*), and others using mixed multinomial logit (*15, 16*), or nested logit models (*17*). However, by neglecting the network structure, discrete choice models are unable to model the stochastic and adaptive nature of the parking search process, as drivers sequentially traverse roadway links which may or may not have available parking.

In contrast with discrete choice models, simulation models try to capture the parking search at the micro level. Thompson and Richardson (*18*) developed an analytical model to mimic the search process where the disutility of a car park location was defined as a function of in-vehicle travel time, in-car park search time, waiting time, fees, fines, and walk time. Other researchers (*19, 20, 21, 22*) have adopted agent-based approaches where the behavioral and parking decision making rules were assigned to the drivers. At issue with micro-simulation models however, is that their size must be restrained due to computational complexity and to date none fully address the dynamic effects that congestion and parking choices have on one another (*22*).

Also working at the macroscopic level, network approaches based in equilibrium assignments are regarded for their ability to successfully model the interaction between road traffic and parking choices (*23*). Like non-network models, network approaches are predicated on individuals choosing parking locations which maximize their utility or minimize travel costs, though they try to simulate the parking choice implicitly (*24*). Hall (*25*) developed a recursive algorithm to solve shortest paths in networks where arc costs were random and time-dependent. Introducing the concept of recourse, Polychronopoulos and Tsitsiklis (*26*) proposed a formulation where arc costs are learned progressively as an end-node of an arc is visited, enabling policies to include cycling and corrective actions where information gathering is beneficial. Waller and Ziliaskopoulos (*27*) analyzed networks with spatial dependence and temporal dependence of arc costs in further detail, showing that "online optimum paths outperform offline shortest paths by up to 40% under certain conditions." Provan (*11*) provided a polynomial-time algorithm for solving shortest paths with recourse where arc lengths are determined by a Markov process and reset upon each traversal. Even these models (*9, 10, 11, 25, 26, 27*) which capture the unknown and stochastic nature of arc costs do not incorporate any kind of memory-based decisions, a key feature of the individual parking search, and typically assume either "full reset" or "no reset" conditions, to use the language of Provan (*11*). In the context of parking, "full-reset" implies that the probability of finding parking on a link remains the same every time a driver visits the link, even if he/she previously observed the parking availability on that link. "No reset", on the other hand means that if a driver cannot park a link, he/she would *never* find a parking space on that link, no matter how long before returning. Clearly neither assumption well-characterizes the parking search problem.

Therefore, the contribution of this paper is to include a "memory" for the traveler, in which the probability of finding parking after traversing a link gradually resets to an *a priori* probability as the time since traversal increases, a formulation we term "asymptotic reset". This formulation generalizes the "full reset" and "no reset" formulations, which can be obtained as special cases. Incorporating the concepts discussed above, our model treats the individual parking search as a Markov decision process (MDP), examining whether a node has been visited previously, and then using that information to influence the probability of finding parking at that link in the near future.

3. METHODOLOGY

Notation and Problem Description

Let $G^o = (N^o, A^o)$ be an undirected graph/network, where N^o represents the set of nodes and A^o consists of arcs/links. The set N^o is defined as $N_r^o \cup {N_d}^o$, where N_r^o is the set of actual intersections in the network and N_d^o denotes a set of dummy nodes. Similarly, A^o is comprised of actual roadway links (A_r^o) and a set of dummy links (A_d^o). The construction of dummy nodes and links, and their function will become evident in the example discussed later. Assume $G = (N, A)$ represents the dual graph of G^o, i.e., $N = A^o$ and a link $a = \{i, j\} \in A$, where $i, j \in N = A^o \Leftrightarrow$ i and j are arcs in $G^o$with a common end point. In other words, the nodes in G represent arcs in G^o and the arcs in G represent turn movements in G^o. We also refer to G^o as the original graph/network and refer to the dual graph simply as graph/network. Although the parking search process can be formulated on the original network, the dual graph lets us think of decisions as being made at nodes rather than on links.

The cost of an arc $i \in A_r^o$ is denoted by c_i^o and is static and deterministic. The corresponding node in the dual network is assumed to be equidistant from its end-points. With the exception of the dual nodes and arcs created from the dummy nodes, and links of the original network, the cost of an arc $a = \{i, j\} \in A$ is defined as $c_a = (c_i^o + c_j^o)/2$. The choice of the dummy link costs will be explained later using an example. Let the walking travel time from node $i \in N$ to the destination be w_i. The prior probability of finding parking at a node $i \in N$ is also assumed to be known and is denoted by ρ_i.

As drivers travel through the network, they are assumed to remember if a previously traversed node had parking (P) or not (NP). However, this ability to retain information is also assumed to be limited to the m nodes (excluding the current node of the traveler) that were most recently visited. The value of m, also called the *memory limit*, is a parameter of the model. Let the set $\Omega = \{P, NP\}$ represent the parking conditions at a node. The state of a driver s is then defined using an $m+1$-dimensional vector of ordered pairs $\big((i_1, p_1), (i_2, p_2), \ldots, (i_{m+1}, p_{m+1})\big)$, where $i_1, i_2, \ldots, i_{m+1} \in N$, $p_1, p_2, \ldots, p_{m+1} \in \Omega$ and $i_1 - i_2 - \ldots - i_{m+1}$ represents a path (with cycles allowed) of the most recent m-nodes visited. The current node at which the traveler is present, i_1, is also represented by $\eta(s)$; and the current parking availability, p_1, is denoted by $\pi(s)$. The set of all states or the state space is denoted by S. At each state the driver may choose to park (only if $\pi(s) = P$) or continue to drive. For each decision at s, the driver may find himself/herself at a subset of states with some known probability. The parking search problem can thus be formulated as an MDP in which drivers use an adaptive strategy or policy that is conditional on their current state. The objective of the problem is to find a policy that minimizes the expected cost of reaching the destination. The list of symbols used is shown in Table 1.

TABLE 1 List of Symbols

Symbol	*Description*
N	Set of nodes in the dual network
A	Set of links in the dual network
S	State space
c_a	Cost of travel on link a, where $a \in A$
ρ_i	Prior probability of finding parking at node i, where $i \in N$
w_i	Walk time to destination from node i, where $i \in N$
$\eta(s)$	The current node at which the traveler is present in state $s \in S$
$\pi(s)$	Parking availability at node $\eta(s)$, where $s \in S$
$N(i)$	Adjacency list of node i (nodes that are directly connected to i), where $i \in N$
X(s)	Decision space at state s, where $s \in S$
λ_i	Reset rate at node i, where $i \in N$
V(s)	Value function of state s, where $s \in S$

Assumptions

In modeling the parking search process as an MDP the following assumptions are made:

1. The network is undirected and ρ_i represents the probability of finding parking along either sides of a link in the original network. This assumption is not restrictive and extending the proposed model to directed network is straightforward.
2. The traveler is experienced enough to have a knowledge of the arc costs and prior probabilities of finding parking.
3. The model does not capture the effect of parking fee. However, this can be easily incorporated by introducing node costs and minimizing the expected generalized cost of travel.
4. Travelers are willing to walk to the destination from any node in the network. In practice, we may select a small sub-network around the destination only considering nodes for which walking is feasible or alternately, set the walking costs to a sufficiently large value.
5. The walking distance from a link in the original network to the destination does not depend on the location at which the traveler parks. This simplification is reasonable unless links in the original network are extremely long. However, such an assumption is not binding as we can split a long link into shorter ones and model the search process on the resulting network.
6. The transition probabilities depend on whether a node that a driver considers to visit features in his/her current state vector, the time elapsed since it was last visited, and a reset rate parameter (λ_i). The reset rate is a measure of how quickly parking probabilities reset to their priors. In practice, this rate depends on the arrival and departure rates of travelers which in turn depend on factors such as land use of the neighborhood, parking meter rates and presence of special events. This assumption is a key characteristic of this paper and is a generalization of the full reset and no-reset versions of SSP problems. A more detailed description of the transition probabilities will follow later.

An Example

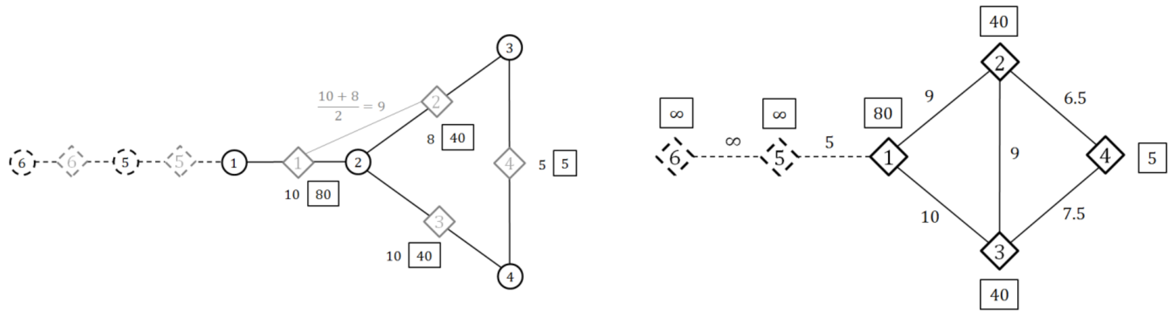

FIGURE 1 An example of the parking search process.

The following example illustrates the MDP formulation of the parking search problem. Consider the network shown in Figure 1. The left panel contains the original network. The dual nodes (shown in grey) are superimposed on the arcs of the original network. Assume that the traveler departs from node 1. The dummy nodes and arcs are shown in dotted lines. The travel cost of each arc and the walking time to the actual destination (shown in boxes) are displayed on the links. We create $m + 1$ dummy nodes and arcs (m is assumed to be 1 in this example) in 'series' and connect it to the source node, i.e., node 1. These dummy nodes and links are constructed because when a traveler starts at the origin, he/she has no memory of links traversed. In order to define the state of the traveler in such situations we assume that he/she had traversed these dummy links before arriving at node 1.

The resulting dual network is shown in the right panel of Figure 1. Arc $\{1,2\}$ in original network is node 1 in the dual network; arc $\{2,3\}$ is node 2, and so on. Thus, the dual nodes created from A_r^o are $1, 2, 3$ and 4. The origin node of the traveler in the dual network is node 5. The cost of arc $\{5,1\}$ in the dual network is set to $10/2 = 5$ as dual node 1 is assumed to be located midway of arc $\{1,2\}$ in the original network. The cost on all remaining dummy links in the dual network (i.e., ones not connected to the dual nodes created from A_r^o) are set to ∞. Also, since the dummy nodes in the dual network are non-existent in reality, the walking distances to the destination and the parking probabilities are set to ∞ and 0 respectively.

Assume that the probability of finding parking at nodes $1, 2$ and 4 in the dual network is 0.5 and the probability of finding parking at node 3 is 0.8. In the full reset case, the traveler's state is defined using his/her current node and the availability of parking at it. The optimal solution under the full reset version prescribes a traveler to first choose path $5 - 1 - 2 - 4$ and park at node 4 if possible and walk to the destination. However, if parking is unavailable at node 4, the policy suggests to cycle between nodes 4 and 2 until parking is found at node 4. The optimal expected cost of the policy was found to be 38.5 units. Although this variant of the parking search problem is easy to solve, the probabilities of finding parking reset to the prior probabilities each time the driver revisits a node. This is unrealistic as travelers would update their beliefs of finding parking based on their previous experiences.

Hence, we propose an asymptotic reset version which assumes that the probability of finding parking depends on the state vector. For instance, if a driver cannot park at node 4; when at node 2, the probability of finding parking at node 4 is updated to a value less than 0.5 since he/she could not find parking at an earlier point in time. This value is modeled to depend on the time taken to travel back and forth between node 4, i.e., 13 units, and the reset rate at node 4. However if node 4 does not appear in the current state of the traveler, even if it was previously visited (i.e., the traveler forgets having visited node 4), the probability of finding parking at node

4 is reset to 0.5. The optimal expected cost for the asymptotic reset version was found to be 46.78 units and can be obtained from the value of any one of the states $\big((5, p_1), (6, p_2)\big)$, where $p_1, p_2 \in \Omega$. Unlike the full-reset case the optimal policy is harder to describe as it explores node 3 and also suggests parking on nodes 2 and 3 if parking was unavailable at node 4.

In this section, we discuss the components of the Markov decision process used to model the parking search problem: the state space, decision space, transition probabilities and the value functions. The value iteration algorithm used to compute the optimal policy is also explained.

State Space

As discussed earlier, the state space consists of the most recently visited nodes and the parking availability at each of these nodes. Populating the state space requires enumeration of all paths containing m arcs. An efficient way to enumerate such paths using repeated breadth first search (BFS) is outlined in the following pseudocode. Let $\Gamma_i(k)$ represent the set of nodes which can be reached from node i by traversing at most k arcs. Suppose the set of all paths of size m is denoted by Γ. Without being mathematically rigorous, the state space can be expressed as $\Gamma \times \Omega^{m+1}$. Note that an element of $\Gamma \times \Omega^{m+1}$ is of the form $\big((i_1, i_2, \dots, i_{m+1}), (p_1, p_2, \dots, p_{m+1})\big)$, where as an element of S is of the form $\big((i_1, p_1), (i_2, p_2), \dots, (i_{m+1}, p_{m+1})\big)$.

Psuedocode *State space construction*

for all $i \in N$ **do**
    Perform BFS with i as the origin
    Store the BFS distance labels (shortest number of arcs required to reach each node)
    for $j = 0$ to m **do**
        Populate $\Gamma_i(k)$ using the BFS labels
    end for
    Set $\Gamma_i = \times_{k=0}^{m} \Gamma_i(k)$
    Scan each element of Γ_i and discard infeasible paths
end for
$\Gamma = \cup_i \Gamma_i$

Decision Space

The set of decisions available at state s, denoted by $X(s)$, can be defined as follows:

$$X(s) = \begin{cases} N\big(\eta(s)\big) \cup \{Destination\} & if\ \pi(s) = P \\ N\big(\eta(s)\big) & otherwise \end{cases}$$

In the first case, the driver can park and walk to the destination or continue to drive to one of the adjacent nodes. However, if parking is unavailable at the current state (second case), the driver has no option but to drive to a node in $N\big(\eta(s)\big)$.

Transition Probabilities

Given a state $s = \big((i_1, p_1), (i_2, p_2), \dots, (i_{m+1}, p_{m+1})\big)$, and a decision $i \in N\big(\eta(s)\big)$, the transition probabilities, denoted by $\Pr(s'|s, i)$, specifies the probability of reaching the state $s' = \big((i, P), (i_1, p_1), \dots, (i_m, p_m)\big)$. Notice that the probability of reaching state $s'' = \big((i, NP), (i_1, p_1), \dots, (i_m, p_m)\big)$, $\Pr(s''|s, i)$, is $1 - \Pr(s'|s, i)$. $\Pr(s'|s, i)$ depends on whether node i appears in s and if it does; it is a function of the time elapsed (t) between recent revisits to node i which is equal to the cost of path $i - i_1 - i_2 - \ldots - i_k = i$, where $k \in \{2, \ldots, m+1\}$ and $i_l \neq i \; \forall \, l \in \{2, \ldots k-1\}$. Let $t = c_{\{i,i_1\}} + c_{\{i_1,i_2\}} + \cdots + c_{\{i_{k-1},i\}}$ represent the cost of this path. The transition probabilities are assumed to be governed by the following equations:

$$\Pr(s'|s,i) = \begin{cases} \rho_i\left(1 - e^{-\lambda_i t}\right) & if\ p_k = NP \\ \rho_i + (1-\rho_i)e^{-\lambda_i t} & if\ p_k = P \end{cases}$$

Figure 2 shows the variation of the transition probability with time. The probability of finding parking is reset to the prior probabilities in an asymptotic manner. As mentioned earlier, these equations help us formulate an intermediate version of the full and no-reset SSP models. In fact, the full and no-reset versions are special cases of this model and can be solved by setting λ to ∞ and 0 respectively. If node i does not appear in state s, the conditional probability of finding parking $\Pr(s'|s,i)$ is simply assumed to be ρ_i.

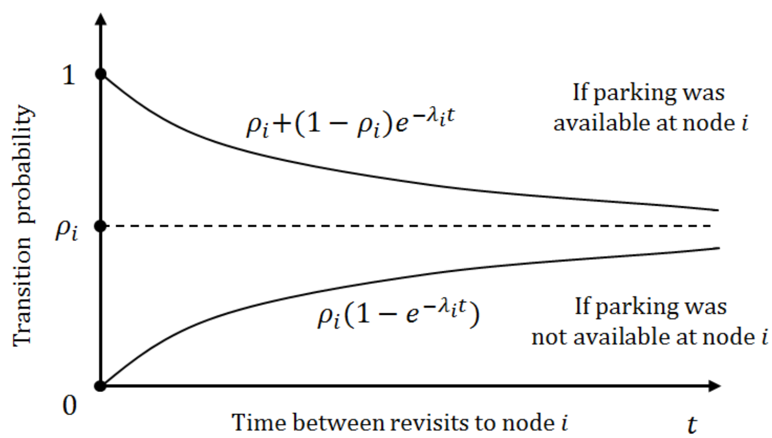

FIGURE 2 Asymptotic reset of transition probabilities.

Value Functions

The optimal value function $V(s)$ is the least expected cost of reaching the destination from s. Using the notation defined in the previous section, Bellman's equations can be written as follows:

$$If\ \pi(s) = NP, V(s) = \min_{i \in N(\eta(s))} \left\{ c_{\{\eta(s),i\}} + \Pr(s'|s,i)\, V(s') + \Pr(s''|s,i)\, V(s'') \right\}$$

$$If\ \pi(s) = P, V(s) = \min\left\{ \min_{i \in N(\eta(s))} \left\{ c_{\{\eta(s),i\}} + Pr(s'|s,i)\, V(s') + Pr(s''|s,i)\, V(s'') \right\}, w_{\eta(s)} \right\}$$

The optimal values of the states can be computed using the Gauss-Seidel variant of value iteration, a pseudo code for which is presented below. The algorithm iteratively updates the values of each state using the above optimality criteria and terminates if the change in values across successive iterations is less than a given tolerance level ϵ. The algorithm is guaranteed to converge and upon termination the value functions can be used to construct an ϵ-optimal policy (see Bertsekas(*28*)).

Psuedocode *Value Iteration – Gauss-Seidel Method*

Step 0: Initialization

$V^0(s) = w_{\eta(s)}\ \forall\ s{:}\,\pi(s) = P$

$V^0(s) = 0\ \forall\ s{:}\,\pi(s) = NP$

$terminate = 1$

$k = 1$

Step 1:

for all $s \in S$ **do**

  if $\pi(s) = NP$ **then**

$$V^k(s) = \min_{i \in N(\eta(s))} \{c_{\{\eta(s),i\}} + \Pr(s'|s,i)\, V^{k-1}(s') + \Pr(s''|s,i)\, V^{k-1}(s'')\}$$

    if $|V^k(s) – V^{k-1}(s)| > \epsilon,$ **then** $terminate = 0$

    $V^{k-1}(s) = V^k(s)$

  else

$$V^k(s) = \min\left\{ \min_{i \in N(\eta(s))} \{c_{\{\eta(s),i\}} + Pr(s'|s,i)\, V^{k-1}(s') + Pr(s''|s,i)\, V^{k-1}(s'')\}, w_{\eta(s)} \right\}$$

    if $|V^k(s) – V^{k-1}(s)| > \epsilon,$ **then** $terminate = 0$

    $V^{k-1}(s) = V^k(s)$

  end if

end for

Step 2:

If $terminate = 1$, then terminate the algorithm. $V^k(s)$ is the value of state s under the ϵ-optimal policy. Else update $k = k + 1$, terminate = 1 and go to Step 1.

4. CASE STUDY

This section contains a case study of the parking search model. A network representing the main campus of the University of Wyoming (UW), Laramie, WY, was used for this demonstration. The network consists of 34 nodes and 56 arcs (see Figure 3), and the destination is closest to node 31(represented by a black circle). All results are discussed using the original network and not it's dual. A graduate student estimated the prior parking probabilities on these arcs over 10 days, which suffices for the purpose of this demonstration. A constant reset rate λ was used for all links in the network. The implementation was carried out in C++ (using the g++ compiler with -O3 optimization flags) on a Linux machine with a 4 core Intel Xeon processor (3.47 GHz) and 12 MB Cache. A tolerance (ϵ) of 10^{-4} was used in this case study.

Excess Cost of Parking

Most transportation models assume that trips begin and end at nodes and parking is not explicitly modeled. However, in reality one is likely to drive around the destination until a suitable parking spot is found resulting in longer trip travel times. In this section, we study the expected increase in trip duration by comparing the shortest path cost of reaching the destination and the expected cost of the optimal policy. The values of m and λ were set to 4 and 1 respectively.

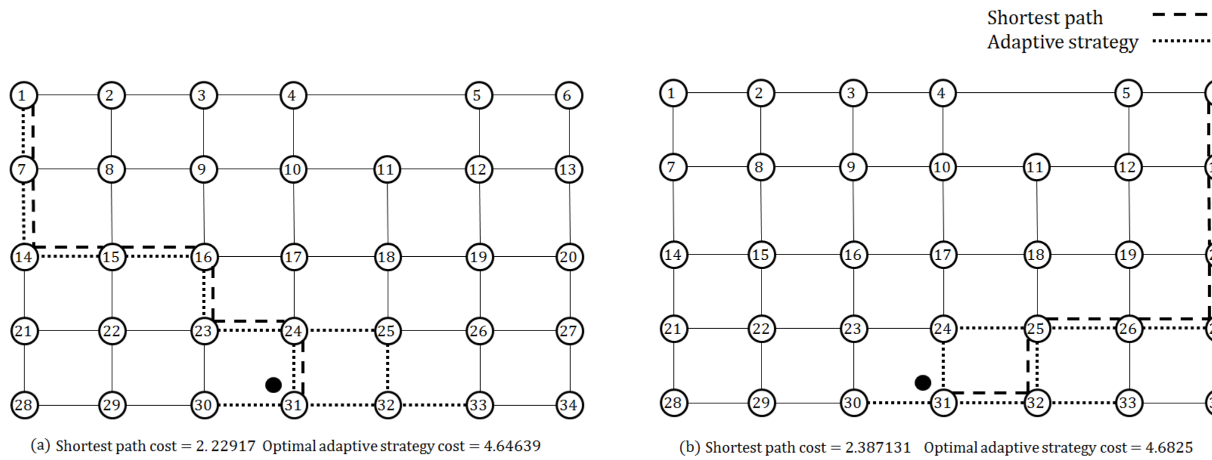

FIGURE 3 Impact of parking on trip costs.

Figure 3 shows the links used in the shortest path and the optimal policy for two origins (1 and 6). As expected, the optimal policy explores more links either because of the lack of parking or due to the anticipation of finding a better parking spot. The expected cost of the adaptive strategy was found to be approximately twice as much as the shortest path cost. Notice that most of the arcs that are revisited are centered around the destination.

Effect of Memory

From a theoretical standpoint, it would be ideal to compute a policy based on an infinite memory. One could assume that the driver is assisted by a navigation system which keeps track of parking conditions on all traversed links. However, as the memory limit is increased, the size of the state space grows exponentially and the problem ends up being computationally intractable. For instance, as can be seen from Table 2, the size of the state space for a memory limit of 5 was found to be nearly 6.5 million and the wall clock time for computing the optimal solution was approximately 7 minutes. The performance of the Gauss-Seidel method was also tested by imposing an ordering (in Step 1 of the algorithm) in which, using the BFS labels, the states whose current nodes were closer to the destination were scanned first. The results (shown in Table 2) indicate a marginal improvement in the number of iterations required for convergence.

We explored the performance of the ϵ-optimal policy under different memory limits in an "infinite memory setting" using Monte Carlo (MC) simulations. Specifically, at each state, the probabilities of finding parking were drawn from a distribution that is a function of the infinite memory (which comprises of the parking conditions on nodes visited since the start of the trip), but the policy used prescribes a decision only based on the state of the traveler (with finite memory). The following table shows the expected cost of the ϵ-optimal policy for memory limits 1 through 5 and an estimate of expected cost of the policy under the infinite memory setting. The origin for this example was node 1. A sample size of 10^4 was used for the MC simulations. The confidence intervals for the estimated expected costs are reported. The computational time for the MDP in seconds and the sizes of the state space are also shown. As the memory limit increases, the gap between the optimal solution and the MC estimate of the optimal strategy decreases as expected and captures the trade-off between the optimal solution and its computational cost.

TABLE 2 Results of the MDP for Different Memory Limits

m	ϵ- optimal solution	MC estimate of the ϵ- optimal policy	Estimate of stdev	95% CI	Computation time (in sec) w/ordering	State space size	# iterations w/ ordering	# iterations w/o ordering
1	4.06966	7.05037	5.85829	(6.88799,7.21275)	0.225	556	62	67
2	4.55313	5.52018	3.49695	(5.42325,5.61711)	0.346	5640	62	64
3	4.55313	5.52805	3.54122	(5.42989,5.62621)	3.067	58352	62	64
4	4.64683	5.24363	3.08402	(5.15815,5.32911)	32.095	611424	60	69
5	4.72728	5.00112	2.27721	(4.93800,5.06424)	442.869	6457664	70	71

Location of trip ends

Lack of parking at a parking spot close to the destination may force drivers to park on nearby links and walk. Figure 4 shows the percentage of trips that end on links in the network computed by simulating the policy in an infinite memory setting for two values of λ. It is interesting to note that as λ increases, travelers are more likely to park closer to the destination. This is because for higher λ, the probabilities reset to their prior values faster and thus the search process mimics the full-reset version. Hence, it is advantageous to revisit links that are closer to the destination.

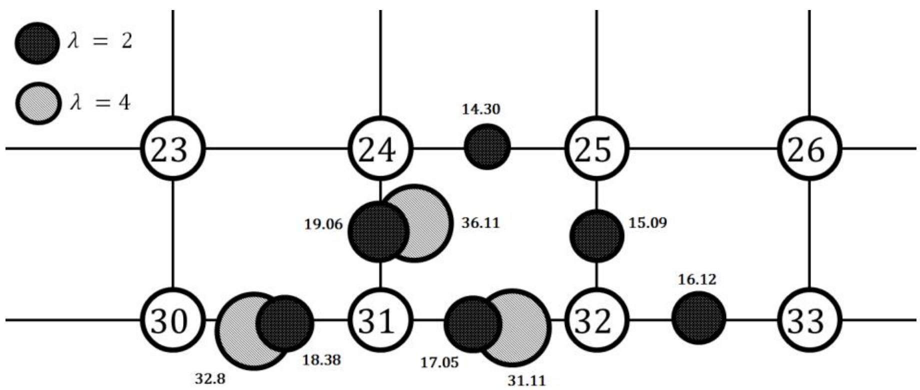

FIGURE 4 Percentage distribution of trip ends.

5. CONCLUSIONS

This paper formulates the parking search process as an online shortest path problem, developing the “asymptotic reset” model to incorporate memory of the parking status of links visited before. This online shortest path problem identifies a routing policy specifying whether drivers will choose to park at an available space or continue to search. Unlike previous research in this area, this approach simultaneously recognizes the stochasticity inherent in the parking search process, and represents the spatiotemporal characteristics of the underlying network structure. The case study demonstrates this model in a network representing a neighborhood near the University of Wyoming, analyzing the sensitivity of the solution to memory size and the value of the “reset rate” parameter.

This paper lays the foundation for future research in several directions. First, spatial correlations can be accounted for, in that parking availability or lack of availability on particular links provides partial information on the likely parking availability on other links. More sophisticated cost functions could account for parking fees, consecutive parking time limitations, and other factors. A flow dependent or time dependent a priori probability of the parking availability can be introduced in future study. For instance, this stochastic shortest path formulation may be usable as the basis for an equilibrium algorithm involving many drivers, in which the probability of finding parking on a link depends on the search patterns used by all drivers.

6. ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant Nos. 1069141 and 1157294 (Collaborative Research: Stochastic and Dynamic Hyperpath Equilibrium Models). The authors thank Sadeq Safaripoor for discussions on the modeling approach and assistance with data collection. Any error or omission related to this research paper is the sole responsibility of the authors.

7. REFERENCES

1. Shoup, D. *The High Cost of Free Parking*. Chicago, IL: Planners Press, American Planning Association, 2011.
2. Feeney, B. P. A Review of the Impact of Parking Policy Measures on Travel Demand. *Transportation Planning and Technology*, Vol. 13, Issue 4, April 1989, pp. 229-244.
3. Axhausen, K. W., J. W. Polak, M. Boltze, and J. Puzicha. Effectiveness of the Parking Guidance System in Frankfurt am Main. *Traffic Engineering and Control*, Vol. 35, Issue 5, 1994, pp. 304-309.
4. Davis, B., T. Dutzik, and P. Baxandall. Transportation and the New Generation: Why Young People Are Driving Less and What It Means for Transportation Policy. *Frontier Group*; *U.S. PIRG Education Fund*, April 2012. http://www.frontiergroup.org/reports/fg/transportation-and-new-generation. Accessed July 26, 2013.
5. Yen, H. and K. Wyatt. Cities Grow More than Suburbs, First time in 100 Years. *Associated Press*, June 28, 2012. http://www.today.com/id/47992439/ns/today-money/t/cities-grow-more-suburbs-first-time-years/#.UfNMLG3ODW6. Accessed June 26, 2012.
6. Pierce, G. and D. Shoup. Getting the Prices Right: An Evaluation of Pricing Parking by Demand in San Francisco. *Journal of the American Planning Association*, Vol. 79, Issue 1, May 9, 2013, pp. 67-81.
7. Ross, C. Boston Limiting New Parking as Number of Residences Soars. *The Boston Globe*, July 5, 2013. http://www.bostonglobe.com/business/2013/07/04/boston-limiting-new-parking-number-residences-soars/kYMnkSr6l55wBMgH4d7VKP/story.html?s_campaign=sm_rd. Accessed July 5, 2013.
8. Greenberg, A. *Contemporary Approaches to Parking Pricing: A Primer.* Publication FHWA-HOP-12-026, FHWA Office of Operations, U.S. Department of Transportation, May 2012.
9. Arnott, R. and E. Inci. The Stability of Downtown Parking and Traffic Congestion. *Journal of Urban Economics,* Vol. 68, Issue 3, Nov. 2010, pp. 260-276.
10. Leurent, F. and H. Boujnah. Traffic Equilibrium in a Network Model of Parking and Route Choice, with Search Circuits and Cruising Flows. Presented at the Proceedings of EWGT2012 - 15th Edition of the EURO Working Group on Transportation, Paris, September 2012.
11. Provan, J. S. A Polynomial-Time Algorithm to Find Shortest Paths with Recourse. *Networks*, Vol. 41, Issue 2, 2003, pp. 115-125.

12. Van der Goot, D. A Model to Describe the Choice of Parking Places. *Transportation Research Part A*, Vol. 16, 1982, pp. 109-115.
13. Axhausen, K. and J. W. Polak. Choice of Parking: Stated Preference Approach. *Transportation*, Vol. 18, No. 1, 1991, pp. 59-81.
14. Lambe, T. A. Driver Choice of Parking in the City. *Transportation Research Part B*, Vol. 30, 1996, pp. 207-219.
15. Hess, S. and J. W. Polak. Mixed Logit Estimation of Parking Type Choice. Presented at the 83rd Annual Meeting of the Transportation Research Board, Washington, D. C., 2004.
16. Hess, S. and J. W. Polak. An Analysis of Parking Behaviour Using Discrete Choice Models Calibrated On SP Datasets. *Proceedings of the European Regional Science Association*, Vienna, Austria, 2004.
17. Hensher, D. A., and J. King. Parking Demand and Responsiveness to Supply, Pricing, and Location in the Sydney Central Business District. In *Transportation Research A*, Vol. 35, 2001, pp. 177–196.
18. Li, Z. C., H. J. Huang, W. H. K. Lam, and S. C Wong. A Model for Evaluation of Transport Policies in Multimodal Networks with Road and Parking Capacity Constraints. *Journal of Mathematical Modeling and Algorithms*, Vol. 6, No. 2, 2007, pp. 239–257.
19. Thompson, R. G. and A. J. Richardson. A Parking Search Model. *Transportation Research Part A*, Vol. 32, No. 3, 1998, pp. 159-170.
20. Benenson, I., K. Martens, and S. Birfir. PARKAGENT: An Agent-based Model of Parking in the City. *Computers, Environment, and Urban Systems*, Vol. 32, 2008, pp. 431-439.
21. Martens, K. and I. Benenson. Evaluating Urban Parking Policies with Agent-Based Model of Driver Parking Behavior. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2046*, 2008, pp. 37-44.
22. Dieussart, K., K. Aerts, T. Steenberghen, S. Maerivoet, and K. Spitaels. SUSTPARK: An Agent-based Model for Simulating Parking Search. 12th AGILE International Conference on Geographic Information Science, Hannover, Germany, 2009.
23. Waraich, R. A. and K. W. Axhausen. Agent-based Parking Choice Model. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2319*, 2012, pp. 39-46.
24. Dell'Orco, M., M. Ottomanelli, and D. Sassanelli. Modeling Uncertainty in Parking Choice Behavior. Presented at the 82nd Annual Meeting of the Transportation Research Board, Washington, D.C., 2003.
25. Hall, R. W. The Fastest Path through a Network with Random Time-Dependent Travel Times. *Transportation Science*, Vol. 20, Issue 3, 1986, pp. 182-188.
26. Polychronopoulos, G. H. and J. N. Tsitsikilis. Stochastic Shortest Path Problems with Recourse. *Networks*, Vol. 27, Issue 2, 1996, pp.133-143.
27. Waller, S. T. and A. K. Ziliaskopoulos. On the Online Shortest Path Problem with Limited Arc Cost Dependencies. *Networks*, Vol. 40, Issue 4, 2002, pp. 216-227.
28. Bertsekas, D.P. Dynamic Programming and Optimal Control, 4th edition, Vol 2. Athena Sientific, Belmont, MA, 2012.