2011 IEEE International Symposium on Computer-Aided Control System Design (CACSD)
Part of 2011 IEEE Multi-Conference on Systems and Control
Denver, CO, USA. September 28-30, 2011

Dynamic Resource Allocation in Urban Settings: A "Smart Parking" Approach

Yanfeng Geng and **Christos G. Cassandras**
Division of Systems Engineering
and Center for Information and Systems Engineering
Boston University
Brookline, MA 02446
gengyf@bu.edu, cgc@bu.edu

***Abstract*— We propose a "smart parking" system for an urban environment based on a dynamic resource allocation approach. The system assigns and reserves an optimal resource (parking space) for a user (driver) based on the user's objective function that combines proximity to destination with parking cost, while also ensuring that the overall parking capacity is efficiently utilized. Our approach solves a Mixed Integer Linear Program (MILP) problem at each decision point in a time-driven sequence. The solution of each MILP is an optimal allocation based on current state information and subject to random events such as new user requests or parking spaces becoming available. The allocation is updated at the next decision point ensuring that there is no resource reservation conflict, that no user is ever assigned a resource with higher than the current cost function value, and that a set of fairness constraints is satisfied. We add an event-driven mechanism to compensate for users with no assignment that are close to their destinations. Simulation results show that using this "smart parking" approach can achieve near-optimal resource utilization and significant improvement over uncontrolled parking processes or state-of-the-art guidance-based systems.**

***Index Terms*— Smart Parking, Resource Allocation, MILP**

I. Introduction

The motivation for this paper is provided by the need to reduce traffic in urban settings. On a daily basis, it is estimated that 30% of vehicles on the road in the downtown area of major cities are cruising for a parking spot and it takes an average of 7.8 minutes to find one [4]. This causes not only a waste of time and fuel for drivers looking for parking, but it also contributes to additional waste of time and fuel for other drivers as a result of traffic congestion. For example, it has been reported [16] that over one year in a small Los Angeles business district, cars cruising for parking created the equivalent of 38 trips around the world, burning $47,000$ gallons of gasoline and producing 730 tons of carbon dioxide.

During the past two decades, traffic authorities in many cites have started to inform and guide drivers to parking facilities with real-time information such as the number of available parking spaces; this information may be displayed on variable-message sign (VMS) at major roads, streets, and intersections, or it may be disseminated through the Internet [17]. This parking guidance information (PGI) system is based on the development of autonomous vehicle detection and parking spot monitoring, typically through the use of sensors placed in the vicinity of parking spaces for vehicle detection and surveillance [13]. Using a PGI system, e-parking is a platform which allows drivers to obtain parking information before or during a trip, and reserve a parking spot via phone or Internet [10], [14].

Although current parking guidance systems increase the probability of finding vacant parking spots, they have several shortcomings. First, drivers may not actually find vacant parking spots by merely following the guidance. In essence, such systems change driver behavior from searching to competing for parking. Second, even if a driver is successfully guided to a parking spot, such a system encourages increasing the probability of finding *any* parking spot at the expense of missing the opportunity for a *better* spot. Third, parking space utilization becomes imbalanced: parking spaces for which information is provided are highly utilized and cause higher traffic congestion nearby, while other parking spaces may be routinely left vacant. In general, guidance systems do not solve the basic parking problem. In fact, system-wide reductions in travel time and vehicle benefits may be relatively small [18]. Even worse, they may cause new traffic congestion in areas where parking spaces are monitored.

In this paper, we propose a new concept for a "smart parking" system. The basic idea is described as follows. Drivers who are looking for parking spots send requests to an allocation center. A request is accompanied by two requirements: a constraint (upper bound) on parking cost and a constraint (upper bound) on the walking distance between a parking spot and the driver's actual destination. The center collects all driver requests over a certain time window and makes an overall allocation at decision points in time seeking to optimize a combination of driver-specific and system-wide objectives. If a driver is satisfied with the assignment, he has the choice to reserve that spot. Once a reservation is made, the driver still has an opportunity to obtain a better parking spot before the current assigned spot is reached. If a driver is not satisfied with the assignment, he has to wait until the next decision point. Observe that this system explicitly *allocates*

The authors' work is supported in part by NSF under Grant EFRI-0735974, by AFOSR under grant FA9550-09-1-0095, by DOE under grant DE-FG52-06NA27490, and by ONR under grant N00014-09-1-1051.

and reserves a parking spot to a driver, as opposed to simply *guiding* him to a space that may not be available by the time it is reached.

The realization of such a "smart parking" system mainly relies on the implementation of a reservation that guarantees a specific parking spot to a driver. This is achievable through wireless technology interfacing a vehicle with hardware that makes a spot accessible only to the driver who has reserved it. Examples include gates, "folding barriers," and obstacles that emerge from and retract to the ground under a parking spot; these are wirelessly activated by devices on-board vehicles, similar to mechanisms for electronic toll systems. A "softer" scheme is to use a red/green light system placed at each parking spot, where red indicates that the spot is reserved and only the vehicle assigned to it may switch it back to green (a vehicle parked when the light is red is fined). In what follows, we will concentrate on the methodology that enables us to make optimal parking space allocations.

In our problem, a key feature is that each driver has specific requirements and only a subset of resources (parking spots) satisfy them. This is similar to the Skills-Based Routing (SBR) problem encountered in telephone call centers where calls are routed based on the skills required for a server to respond to the call [1], [6]. Related research has focused on various forms of approximations to bypass the high dimensionality involved in determining optimal routing policies via dynamic programming, e.g., using Approximate Dynamic Programming (ADP) [11] and limiting problems to a heavy traffic regime [5], [7], [8], [9]. However, all such methods assign a single user to one server/resource at a time and aim to minimize a user delay cost metric. In our problem, we allocate multiple users to multiple resources and a key objective is the average minimum user cost.

We view the "smart parking" process as a sequence of Mixed Integer Linear Programming (MILP) problems solved over time at specific decision points subject to suitably designed fairness constraints. We will discuss the choice of decision points comparing a time-driven approach (where they are made periodically) to an event-driven scheme where a new allocation is made with every event occurrence.

The rest of the paper is organized as follows. In Section II, we describe the dynamic resource allocation model and formulate the MILP problem solved at every decision point. In Section III, we address the issues of allocation feasibility and fairness. Simulation results are given in Section IV where we provide empirical evidence of the significant benefits, in terms of several performance metrics, of "smart parking" over uncontrolled settings or guidance-based systems. We conclude and discuss future work in Section V.

II. Dynamic Resource Allocation Model

For the sake of generality, we will employ the term "user" when referring to drivers or vehicles and the term "resource" when referring to parking spots. We adopt a queueing model for the problem as shown in Fig. 1, where there are N resources and every user arrives randomly and independently to join an infinite-capacity queue (labeled WAIT) and waits to be assigned a resource if possible. At the kth decision point, the system makes allocations for all users in both the waiting queue and the queue (labeled RESERVE) of users who have already been assigned and have reserved a resource from a prior decision point. If a user in WAIT is successfully assigned a resource, he joins the RESERVE queue, otherwise he remains in WAIT. A user in RESERVE may be assigned a different resource after a decision point and returns to the same queue until he can physically reach the resource and occupy it. A user leaves the system after occupying a resource for some amount of time at which point the resource becomes free again.

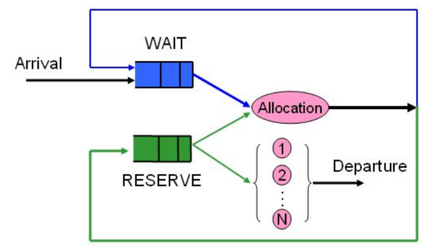

Fig. 1. Queueing Model for Dynamic Resource Allocation

At the kth decision point we define the state of the allocation system, $X(k)$, and the state of the ith user, $S_i(k)$ as explained next. Note that a user is designated by a unique positive integer $i = 1, 2, \ldots$ First, we define

$$X(k) = \{W(k), R(k), P(k)\} \tag{1}$$

where $W(k) = \{i : \text{user } i \text{ is in the WAIT queue}\}$, $R(k) = \{i : \text{user } i \text{ is in the RESERVE queue}\}$, and $P(k) = \{p_1(k), ..., p_N(k)\}$ is a set describing the state of the jth resource, $j = 1, \ldots, N$, defined as follows:

$$p_j(k) = \begin{cases} -1 & \text{if resource } j \text{ is occupied} \\ 0 & \text{if resource } j \text{ is free} \\ i & \text{if resource } j \text{ is reserved by user } i \end{cases} \tag{2}$$

We assume that each resource has a known location associated to it denoted by $y_j \in Z \subset \mathbb{R}^2$ in a two-dimensional Euclidean space. We also define

$$S_i(k) = \{z_i(k), r_i(k), q_i(k), \Omega_i(k)\} \tag{3}$$

where $z_i(k) \in Z \subset \mathbb{R}^2$ is the location of user i, $r_i(k) \in \mathbb{R}^+$ is the total time that user i has spent in the RESERVE queue up to the kth decision point ($r_i(k) = 0$ if $i \in W(k)$), and $q_i(k)$ is the reservation status of user i:

$$q_i(k) = \begin{cases} 0 & \text{if } i \in W(k) \\ j & \text{if } i \in R(k),\ p_j(k) = i \end{cases} \tag{4}$$

Clearly, if $p_j(k) = i$ we must have $q_i(k) = j$ and vice versa. Finally, $\Omega_i(k)$ is a feasible resource set for user i, i.e., $\Omega_i(k) \subseteq \{1, \ldots, N\}$ depending on the requirements set forth by this user regarding the resource it requests. In general, $\Omega_i(k)$ may be a set specified by each user upon arrival at the system; however, for the specific parking problem we are interested in, we will define $\Omega_i(k)$ in terms of *attributes* associated with user i and defined as follows.

We associate two attributes to user i. The first, denoted by D_i, is an upper bound on the distance between the resource that the user is assigned and his actual destination $d_i \in Z \subset \mathbb{R}^2$. If the user is assigned a resource j located at y_j, let $D_{ij} = \|d_i - y_j\|$ where $\|\cdot\|$ is a suitable distance metric. Then, the constraint

$$D_{ij} \leq D_i \tag{5}$$

defines a requirement that contributes to the determination of $\Omega_i(k)$ by limiting the set of feasible resources to those that satisfy (5). If the requirement is expressed in terms of a *time* D_i, then the constraint is simply rewritten as $\|d_i - y_j\|/V \leq D_i$, where V is a given speed parameter (e.g., an average walking speed).

The second attribute for user i, denoted by M_i, is an upper bound on the cost this user is willing to tolerate for the benefit of reserving and subsequently using a resource. The actual cost depends on the specific pricing scheme adopted by the allocation system and may include a flat fee for reserving a resource, a fee dependent on the total reservation time, and subsequently a fee for occupying the resource. Our approach does not depend on the specific pricing scheme used, but we will assume that each user cost is a monotonically nondecreasing function of the total reservation time $r_i(k)$, as well as a function of the traveling time from the user location at the kth decision time, $z_i(k)$, to a resource location y_j. Let $s_{ij}(k) = \|z_i(k) - y_j\|$ be this distance, and define the traveling time $t_{ij}(k) = f(s_{ij}(k), \omega)$, where ω denotes all random traffic conditions. We use $M_{ij}(r_i(k), t_{ij}(k))$ to denote the total expected cost for using resource j, evaluated at the kth decision time. Comparing $M_{ij}(r_i(k), t_{ij}(k))$ to M_i, leads to the constraint

$$M_{ij}(r_i(k), t_{ij}(k)) \leq M_i \tag{6}$$

This defines a second requirement that contributes to the determination of $\Omega_i(k)$ by limiting the set of feasible resources to those that satisfy (6). In order to fully specify $\Omega_i(k)$, we further define

$$\Gamma(k) = \{j : p_j(k) \neq -1,\ j = 1, \ldots, N\}$$

to be the set of free and reserved resources at the kth decision time and set

$$\Omega_i(k) = \{j : M_{ij}(k) \leq M_i, D_{ij} \leq D_i,\ j \in \Gamma(k)\} \tag{7}$$

where, for simplicity, we have written $M_{ij}(k)$ instead of $M_{ij}(r_i(k), t_{ij}(k))$. Note that this set allows the system to allocate to user i any resource $j \in \Omega_i(k)$ which satisfies the user's requirements even if it is currently reserved by another user (i.e., if $p_j(k) = m \neq i$). Thus, a resource j may be dynamically re-allocated to different users at each decision point until $p_j(k) = -1$, signaling that it has become physically occupied by a user.

We can now concentrate on defining an objective function which we will seek to minimize at each decision point by allocating resources to users. We use a weighted sum to define user i's cost function, $J_{ij}(k)$ if he is assigned to resource j, as follows:

$$J_{ij}(k) = \lambda_i \frac{M_{ij}(k)}{M_i} + (1 - \lambda_i)\frac{D_{ij}}{D_i} \tag{8}$$

where $\lambda_i \in [0,1]$ is a weight that reflects the relative importance assigned by the user between *cost* and *resource quality*. In the case of parking, resource quality is measured as the walking distance between the parking spot the user is assigned and his actual destination.

To capture the essence of "smart parking," the objective of the system is to make allocations for as many users as possible and, at the same time, to achieve minimum user cost as measured by $J_{ij}(k)$. Define binary control variables x_{ij}:

$$x_{ij} = \begin{cases} 0 & \text{if user } i \text{ is not assigned to resource } j \\ 1 & \text{if user } i \text{ is assigned to resource } j \end{cases} \tag{9}$$

We can now define the allocation problem **(P1)** at the kth decision point as follows:

$$\min \sum_{i \in W(k) \cup R(k)} \sum_{j \in \Omega_i(k)} x_{ij} \cdot J_{ij}(k) \tag{10}$$

s.t.

$$\sum_{j \in \Omega_i(k)} x_{ij} = 1, \quad \forall i \in W(k) \cup R(k) \tag{11}$$

$$\sum_{i \in W(k) \cup R(k)} x_{ij} \leq 1, \quad \forall j \in \Gamma(k) \tag{12}$$

$$\sum_{j \in \Omega_i(k)} x_{ij} \cdot J_{ij}(k) \leq J_{iq_i(k-1)}(k), \quad \forall i \in R(k) \tag{13}$$

$$x_{ij} \in \{0,1\}, \quad \forall i \in W(k) \cup R(k), \quad j \in \Gamma(k) \tag{14}$$

In this problem, the objective function focuses on user satisfaction. One can formulate alternative versions that incorporate system-centric objectives such as maximizing resource utilization or total revenue without affecting the essence of our approach which is primarily dependent on the three constraints (11), (12), and (13). In particular, the "request satisfaction" constraints (11) require that every user is allocated to a resource, unless of course $\Omega_i(k) = \varnothing$. The capacity constraints (12) ensure that every resource is occupied by no more than one user. The constraints (13) add a unique feature to our problem by guaranteeing that every user in the RESERVE queue is assigned a resource which is no worse than the one most recently reserved, i.e., $q_i(k-1)$.

Problem **(P1)** is a Mixed-Integer Linear Programming (MILP) problem that can be solved using any of several commercially available software packages (we use ILOG CPLEX in this paper). However, the problem is often infeasible and fails to provide an allocation. Infeasibility arises when the number of available resources is smaller than the number of users who are competing for them, thus violating some of the constraints in (11). If that happens, an auxiliary problem may be defined whereby we seek to pick the maximum number of users which guarantees that the problem becomes feasible and results in minimal cost. In other words, since only constraints in (11) are violated, we should first find maximal Feasible Subsets (MAX FS) of (11) and choose one such subset which generates a minimal cost. However the problem of finding MAX FS is proved to be an NP-hard problem [2], [3], [15]. When the user set is large, determining the MAX FS requires an enormous computational effort and solution time which is not suited to the real-time nature of such a DRA problem. However, we proceed in a different way, as described next, which avoids this complication.

III. Resource Allocation Strategies

Observe that the constraints in (11) apply to users in the set $W(k) \cup R(k)$, thus requiring the system to immediately assign a resource to a new user $i \in W(k)$. This is unnecessarily restrictive given the inherent delay between a user request and actually occupying a resource. Thus, we follow a different direction by replacing the constraints (11) with the following:

$$\sum_{j \in \Omega_i(k)} x_{ij} \leq 1, \quad \forall i \in W(k) \tag{15}$$

$$\sum_{j \in \Omega_i(k)} x_{ij} = 1, \quad \forall i \in R(k) \tag{16}$$

and, at the same time, we add a penalty cost $\sum_{i \in W(k)} (1 - \sum_{j \in \Omega_i(k)} x_{ij})$ to the objective function in (10):

$$\min \sum_{i \in W(k) \cup R(k)} \sum_{j \in \Omega_i(k)} x_{ij} \cdot J_{ij}(k) + \sum_{i \in W(k)} (1 - \sum_{j \in \Omega_i(k)} x_{ij}) \tag{17}$$

Note that, unlike (11), in (15) and (16) constraints are now separately imposed over $W(k)$ and $R(k)$. The constraints (15) indicate that any user in the WAIT queue may be assigned at most one resource but may also fail to get an assignment. On the other hand, (16) still guarantees that each user in the RESERVE queue maintains a resource assignment. If the system fails to allocate a resource to some user i, i.e., $\sum_{j \in \Omega_i(k)} x_{ij} = 0$, a cost of 1 is added to the objective function. Therefore, the added term $\sum_{i \in W(k)} (1 - \sum_{j \in \Omega_i(k)} x_{ij})$ in (17) is the total cost contributed by the number of "unsatisfied" users. Since by its definition in (8) $J_{ij}(k) \leq 1$, the added cost of value 1 is sufficiently large to ensure that a user is assigned to a resource if there are free qualified resources left.

In this formulation, we can easily prove that the problem is always feasible. Indeed, letting the matrix $X \equiv [x_{ij}]$ denote a solution of (17), then the set

$$\{X : \sum_{j \in \Omega_i(k)} x_{ij} = 0, \; x_{mq_m(k)} = 1, \; i \in W(k), \; m \in R(k)\}$$

is always a feasible solution, since it implies that all users in $W(k)$ are not allocated and all users in $R(k)$ simply maintain their previous reservation (assuming that $R(k) \neq \varnothing$).

Fairness. As we can see from (15) and (16), this strategy gives a higher assignment priority to users in the RESERVE queue, which is reasonable because they are already incurring a positive cost. On the other hand, (15) involves no distinctions among the waiting users, regardless of how long they have resided in the WAIT queue or where they are located. This introduces unfairness among waiting users. For example, a waiting user may be located right in front of an available resource which, however, is assigned to another waiting user at a considerably larger distance from it. In order to remove such unfairness, we add the following constraints:

$$\left[\sum_{n \in \Omega_i(k)} x_{in} \right] - x_{mj} \geq 0, \; \forall i, j, m \text{ s.t. } \begin{array}{l} j \in \Gamma(k), \; j \in \Omega_i(k), \\ m \in W(k), \\ t_{mj} > t_{ij} \end{array} \tag{18}$$

These constraints are explained as follows. Consider a resource j which is available for assignment ($j \in \Gamma(k)$) and qualified for user i ($j \in \Omega_i(k)$). If i fails to be allocated any resource, we have $\sum_{n \in \Omega_i(k)} x_{in} = 0$ and (18) requires that $x_{mj} = 0$, i.e., any other waiting user m located farther away from j than user i ($t_{mj} > t_{ij}$) is forbidden from being assigned to j.

Decision points. At this point, the modified problem, which we shall refer to as **(P2)**, uses the objective function (17) and the constraints (12), (13), (14) from the original formulation, along with (15), (16), and (18). Moreover, the existence of a solution is now guaranteed. An important remaining issue concerns the choice of decision points over time or, equivalently, defining appropriate "decision intervals" $\tau(k)$, $k = 1, 2, \ldots$ The simplest idea is to adopt an event-driven approach, i.e., to solve **(P2)** whenever an event is observed in the system. The advantage of this approach is that it provides quick response to users; however, it obviously also entails significant computational burden to the system since the frequency of solving **(P2)** may become high. We adopt a time-driven strategy for decision making. After the $(k-1)$th decision point, the system waits for some duration $\tau(k)$ and then makes a new allocation over all users that arrived during $\tau(k)$ and all previous users residing in either the WAIT or RESERVE queue. Clearly there is a tradeoff: a large $\tau(k)$ may eventually yield a lower cost for all users involved, but it also forces a large number of users to remain in the WAIT queue with no assignment, until it is either too late because a user has reached his destination or has lost patience and searches for resources by himself. In the next section, we shall empirically explore the effect of varying $\tau(k)$ on the performance of the system.

Performance metrics. In solving problem **(P2)** we aim to minimize user costs as defined by (8) at each decision point. In order to assess the overall system performance over some time interval $[0, T]$, we define several appropriate metrics evaluated over a total number of users N_T served over this interval (simulation run length).

From the system's point of view, we consider resource *utilization* as a performance metric and break it down into two parts: $u_r(T)$ is the utilization of resources by reservation (i.e., the fraction of resources that are reserved) and $u_p(T)$ is the utilization by occupancy (i.e., the fraction of resources that are physically occupied by a user).

From the users' point of view, we first define an average satisfaction metric $\bar{J}(T)$ for those users that actually occupy a resource, as defined in (8). Another metric we will use is the *abandonment ratio* $a(T)$ defined as follows. Let

$$A_W(k) = \{i : i \in W(k),\ \ \|z_i(k) - d_i\| < \epsilon\}$$

be the set of users who reach their destination but are still in the WAIT queue at the kth decision point, where $\epsilon > 0$ is a small real number used to indicate that a user is in the immediate vicinity of his destination d_i. Letting k_T denote the last decision point within the time interval of length T, we then define $a(T) = \frac{|A_W(k_T)|}{N_T}$. Finally, we consider the *average time-to-park* $t_p(T)$, which is the time from the instant a user "arrives" to the instant he physically occupies a parking resource.

IV. Simulation Results

In this section, we describe a simulation testing environment to explore the behavior of the proposed "smart parking" system. A small business district map is shown in Fig. 2. In this scenario, there are 4 malls (indicated by red triangles) which are the users' destinations, and 30 parking resources denoted by green squares. The lines that define the map grid are roads, and blue circles represent users. A dotted line connecting a user to a resource represents a reservation. In all simulations, user arrival times are Poisson distributed with rate λ, and uniformly located in the map. The user cost parameter M_i is uniformly distributed in the interval $[M_{min}, M_{max}]$, and the walking-distance parameter D_i is also uniformly distributed in $[D_{min}, D_{max}]$. The resource occupancy time is exponentially distributed with rate μ.

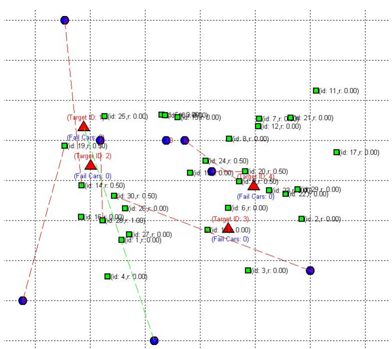

Fig. 2. Simulation Environment

We adopt a pricing scheme based on which the expected cost incurred by user i when assigned resource j at the kth decision point is

$$M_{ij}(k) = e^{\alpha(r_i(k)+t_{ij}(k))} + CT_i \tag{19}$$

where α is a positive constant, $r_i(k)$ is the time already spent at the RESERVE queue, $t_{ij}(k)$ is an estimate of the driving time for i to reach j, and T_i is the expected parking time of user i. We do not consider random traffic events in the simulation, so that $t_{ij}(k)$ is simply estimated by

$$t_{ij}(k) = \|z_i(k) - y_j\|_M / v_i$$

where $\|\cdot\|_M$ denotes the Manhattan distance [12], and v_i is user i's average speed estimate.

TABLE I

"Smart parking" performance with no user requirements

τ	10	15	20	25	30	E
$u_p(T)$	0.73	0.75	0.76	0.75	0.73	0.70
$u_p(T)$-IA	0.80	0.85	0.83	0.79	0.80	0.70
$u_r(T)$	0.09	0.09	0.08	0.08	0.08	0.10
$u_r(T)$-IA	0.09	0.07	0.09	0.08	0.08	0.10
$a(T)$	0.09	0.12	0.15	0.18	0.20	0.04
$a(T)$-IA	0.03	0.05	0.06	0.05	0.07	0.04
$t_p(T)$	43	47	51	54	62	40
$t_p(T)$-IA	42	43	48	46	50	40

The walking-distance cost is defined as $D_{ij} = \beta w_{jd_i}$ where β is a positive constant and w_{jd_i} measures the walking distance from resource j to user i's destination d_i.

In all simulations, we use a constant decision interval $\tau(k) = \tau$, $k = 1, 2, \ldots$ We will also study the effect of τ on our performance metrics. We expect that as τ increases, $a(T)$ should increase. This is because the length of the WAIT queue increases with τ and the number of waiting users that reach their destination before having an opportunity to join the RESERVE queue, i.e., $|A_W(k_T)|$, also increases. To deal with this effect, we adopt the following *Immediate Allocation* (IA) policy: Whenever user i is in the WAIT queue and reaches a location z_i such that $\|z_i - d_i\| \le v_i\tau$, he is placed in an "immediate allocation" queue. If this queue is not empty, then as soon as a user departure makes a resource available the system immediately prioritizes user i over other users in $W(k)$ and assigns him this resource if it is feasible. This "immediate allocation" problem is easy to solve. We define an "urgent" user set

$$I(k) = \{i : i \in W(k),\ \ \|z_i - d_i\| \le v_i\tau\}$$

and, as soon as a resource j becomes free, we allocate it to user i such that $J_{ij} = min_{n\in I(k), j\in\Omega_n(k)} J_{nj}$, if such i exists.

Table I shows results obtained with $1/\lambda = 10$ (time units), $1/\mu = 220$, $M_{min} = M_{max} = \infty$, $D_{min} = D_{max} = \infty$ (in practice, M_{max} and D_{max} are selected as large positive numbers), so that in this case there are no constraints imposed by user requirements. Results are shown over different values of τ, as well as the event-driven decision policy (last column, labeled E). Every result is generated by the average of 5 simulations, with each lasting for $T = 18000$. In each case, we also include results when the IA policy is adopted. Since requirements are set to infinity, an event-based allocation is very similar to the *M/M/n* queueing system for which the average utilization is given by $\bar{u} = \lambda/(N\mu) \approx 0.73$ which is close to $u_p(T)$ over different τ values and generally insensitive to τ. Note, however, that $u_p(T)+u_r(T)$ exceeds 0.80; the $u_r(T)$ utilization component represents added benefit to the system in terms of revenue, while at the same time providing a reservation guarantee for users. As expected, the abandonment ratio $a(T)$ decreases with τ and, for sufficiently low values, it is comparable to the event-driven decision policy. The same is true for the average time-to-park $t_p(T)$ metric.

TABLE III

PERFORMANCE METRICS UNDER DIFFERENT TRAFFIC INTENSITY

$1/\lambda$	7			9			10			12			15		
	SP-E	G	NG	SP-E	G	NG	SP-E	G	NG	SP-E	G	NG	SP-E	G	NG
$u(T)$	0.96	0.92	0.91	0.86	0.82	0.80	0.79	0.75	0.72	0.64	0.60	0.57	0.51	0.48	0.47
$a(T)$	0.62	0.67	0.92	0.38	0.47	0.73	0.25	0.35	0.55	0.11	0.19	0.63	0.08	0.11	0.51
$\bar{J}(T)$	0.531	0.536	0.590	0.514	0.539	0.586	0.504	0.534	0.592	0.482	0.523	0.603	0.497	0.536	0.587
$t_p(T)$	282	387	454	61	101	154	48	108	180	38	38	70	38	42	68

TABLE II

"SMART PARKING", G AND NG PERFORMANCE WITH USER REQUIREMENTS

τ	10	20	30	E	G	NG
$u_p(T)$	0.73	0.75	0.72	0.70		
$u_p(T)$-IA	0.76	0.84	0.75	0.70	0.75	0.72
$u_r(T)$	0.08	0.08	0.07	0.09		
$u_r(T)$-IA	0.09	0.08	0.07	0.09		
$a(T)$	0.23	0.29	0.33	0.25		
$a(T)$-IA	0.19	0.23	0.19	0.25	0.35	0.55
$\bar{J}(T)$	0.499	0.493	0.475	0.504		
$\bar{J}(T)$-IA	0.500	0.496	0.498	0.504	0.534	0.592
$t_p(T)$	58	62	78	48		
$t_p(T)$-IA	54	61	74	48	108	180

Table II shows results when there are user requirements and we set $M_{min} = 0$, $D_{min} = 0$, $M_{max} = 100$, $D_{max} = 100$, $\alpha = 0.025$, $\beta = 1$, $C = 1$. Comparing with Table I, we find that resource utilizations are minimally affected, but $a(T)$ considerably increases as the presence of user requirements limits their feasible options. The average user cost $\bar{J}(T)$ decreases as τ increases since the system gathers more user information and is able to make better overall decisions. This also explains why $\bar{J}(T)$ increases when the IA policy is used, though still outperforming the event-driven decision policy.

We also seek to quantify the improvement of the "smart parking" (SP) approach over an uncontrolled setting where users park without any guidance (NG) and the case of parking with guidance to free parking spaces (G). In both cases, we assume users start to look for parking when they reach regions defined by their walking distance. If there is guidance, users know exactly the location of free resources; otherwise, they search for free resources by themselves. The four performance metrics are shown and we note that SP provides significant benefits over the G approach. From the system point of view, total resource utilization increases by as much as 20% (from 0.75 to 0.92). From a user's point of view, we see decreases in both $a(T)$ and $\bar{J}(T)$, while average time-to-park is reduced by as much as half (from 108 to 54).

In Table III we examine the effect of traffic intensity as we change the value of the interarrival interval $1/\lambda$. Since there is no obvious optimal value for τ, we chose to compare the event-driven decision policy (SP-E) to the G and NG cases. Here $u(T) = u_p(T) + u_r(T)$ is the total utilization. We find that the benefit of the SP-E strategy over the G case is substantial under heavier traffic, as expected.

V. CONCLUSIONS AND FUTURE WORK

We have proposed a "smart parking" system that exploits technologies for parking space availability detection and for driver localization and which allocates parking spots to drivers instead of only supplying guidance to them. Simulation results show significant performance improvements over existing parking behavior, including the use of guidance-based systems. Ongoing research focuses on selecting (possibly state-dependent) decision intervals and on the use of pricing control to adjust parking space prices for different classes of users or other bidding-type mechanisms that can enhance fairness.

REFERENCES

[1] Z. Aksin, M. Armony, and V. Mehrotra. The modern call center: A multi-disciplinary perspective on operations management research. *Production and Operational Management*, 16(5):665–688, 2007.

[2] E. Amaldi and V. Kann. The complexity and approximability of finding maximum feasible subsystems of linear relations. *Theore. Comput. Sci.*, (147):181–210, 1995.

[3] E. Amaldi and V. Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theore. Comput. Sci.*, 209:237–260, 1998.

[4] R. Arnott, T. Rave, and R. Schob. *Alleviating Urban Traffic Congestion*. MIT Press, 2005.

[5] R. Atar, A. Mandelbaum, and M. Reiman. Scheduling a multi-class queue with many exponential servers: asymptotic optimality in heavy-traffic. *Annals of Applied Probability*, (15):2606–2650, 2004.

[6] N. Gans, G. Koole, and A. Mandelbaum. Telephone call centers: tutorial, review and research prospects. *Manufacturing & Service Operations Management*, (5):79–141, 2003.

[7] I. Gurvich and W. Whitt. Service-level differentiation in many server service systems: A solution based on fixed-queue-ratio routing. *Operations Research*, 2007.

[8] S. Halfin and W. Whitt. Heavy-traffic limits for queues with many exponential servers. *Operations Research*, (29):567–588, 1981.

[9] J. Harrison and A. Zeevi. Dynamic scheduling of a multi-class queue in the Halfin-Whitt heavy traffic regime. *Operations Research*, (52):243–257, 2004.

[10] T. B. Hodel and S. Cong. Parking space optimization services, a uniformed web application architecture. *Intelligent Transport Systems and Services, ITS World Congress Proceedings*, pages 16–20, 2003.

[11] G. Koole and A. Pot. Approximate dynamic programming in multi-skill call centers. *Proceedings of the 37th Conference on Winter Simulation*, pages 576–583, 2005.

[12] E. Krause. *Taxicab Geometry*. Dover, 1987.

[13] L. Mimbela and L. Klein. *A Summary of Vehicle Detection and Surveillance Technologies used in Intelligent Transportation Systems*. The Vehicle Detector Clearinghouse, 2000.

[14] C. J. Rodier and S. A. Shaheen. Transit-based smart parking: An evaluation of the San Francisco bay area field test. *Transportation Research Part C*, (18):225–233, 2010.

[15] J. Sankaran. A note on resolving infeasibility in linear programs by constraint relaxation. *Operations Research Letter*, (13):19–20, 1993.

[16] D. Shoup. *The high cost of free parking*. APA Planner Press, 2005.

[17] D. Teodorovic and P. Lucic. Intelligent parking systems. *European Journal of Operational Research*, 175:1666–1681, 2006.

[18] R.G. Thompson and P. Bonsall. Driver's response to parking guidance and information systems. *Transport Review*, 17(2):89–104, 1997.