Pricing of Parking for Congestion Reduction*

Daniel Ayala, Ouri Wolfson, Bo Xu, Bhaskar DasGupta
University of Illinois at Chicago
Department of Computer Science
{dayala, wolfson, boxu, bdasgup}@uic.edu

Jie Lin
University of Illinois at Chicago
Department of Civil and Materials Engineering
janelin@uic.edu

ABSTRACT

The proliferation of mobile devices, location-based services and embedded wireless sensors has given rise to applications that seek to improve the efficiency of the transportation system. In particular, new applications are already available that help travelers to find parking in urban settings by conveying the parking slot availability near the desired destinations of travelers on their mobile devices.

In this paper we present two notions of parking choice: the optimal and the equilibrium. The equilibrium describes the behavior of individual, selfish agents in a system. We will show how a pricing authority can use the parking availability information to set prices that entice drivers to choose parking in the optimal way, the way that minimizes total driving distance by the vehicles and is then better for the transportation system (by reducing congestion) and for the environment. We will present two pricing schemes that perform this task. Furthermore, through simulations we show the potential congestion improvements that can be obtained through the use of these schemes.

Categories and Subject Descriptors

K.4.4 [**Computers and Society**]: Electronic Commerce—*payment schemes*

General Terms

Algorithms, Economics

Keywords

Parking, Pricing scheme

1. INTRODUCTION

*This work was supported in part by the NSF IGERT program under grant DGE-0549489 and NSF grants IIS-1213013 and CCF-1216096.

ACM SIGSPATIAL GIS '12 November 6-9, 2012. Redondo Beach, CA USA
Copyright 2012 ACM ISBN 978-1-4503-1691-0/12/11 ...$15.00.

Cruising for parking by driving around an urban area looking for available parking slots has been shown to be a major cause of congestion in urban areas. For example, in [16], studies conducted in 11 major cities revealed that the average time to search for curbside parking was 8.1 minutes and cruising for these parking slots accounted for 30% of the traffic congestion in those cities. This means that each parking slot would generate 4,927 vehicle miles traveled (VMT) per year [17]. That number would of course be multiplied by the number of parking slots in the city. For example, in a big urban city like Chicago with over 35,000 curbside parking slots [18], the total number of VMT becomes 172 million VMT per year due to cruising while searching for parking. Furthermore, this would account for waste of 8.37 million gallons of gasoline and over 129,000 tons of CO_2 emissions.

The proliferation of mobile devices, location-based services and embedded wireless sensors has given rise to applications that seek to improve the efficiency of the transportation system. In particular, new applications are becoming available to help travelers find parking in urban settings. For example, wireless sensors embedded on parking slots are used to detect the availability of slots across some area, and the locations of currently available parking slots are disseminated to the mobile devices of users that are looking for parking in the area. A prime example of this application is SFPark [7] that uses sensors embedded in the streets of San Francisco. When a user is looking for parking in some area of the city, the application shows a map with the marked locations of the open parking slots in the area. Though the primary motivation for these applications is to help the individual users to find open parking slots quicker [1, 2], they can also be used to reduce congestion by decreasing the total distance traveled by vehicles looking for parking.

This paper deals with this second objective of congestion reduction. Specifically, we propose to achieve the reduction by focusing on a conceptual gap between two notions of parking, which is just a spatio-temporal matching between mobile agents (drivers) and resources (parking slots). The two matching notions are optimality and equilibrium. Ideally, we would like the matching to be optimal, i.e. the total distance driven to park by all vehicles, to be minimum. However, achieving this optimality requires a central authority that can dictate the slot in which a driver should park, even if that driver can do better. Fig. 1 illustrates this point by an example. Suppose that the edge labels represent distances in one-hundredths of a mile, *i.e.*, v_1 is 0.1 miles from s_1, 0.2 miles from s_2 and so forth. To achieve minimum total driving distance, i.e. System Optimum cost, v_1 will have to

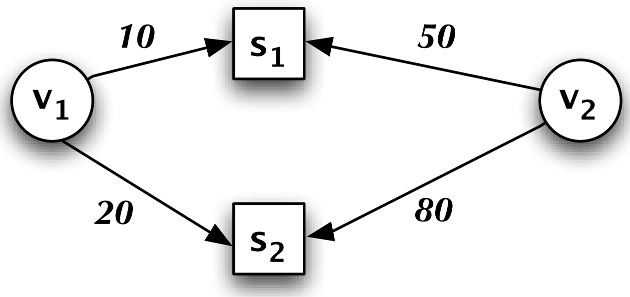

Figure 1: An example for the parking pricing problem.

park in s_2, and v_2 will have to park in s_1. The total driving distance of this parking assignment, called SO, is 0.7 miles. However, this requires v_1 to drive to a farther slot, s_2, i.e. inferior from her point of view because s_1 is closer. There is no central authority that can dictate this parking assignment to v_1. If v_1 drives to s_1 (and captures it since she is closer than v_2), then v_2 must settle for s_2. The total driving distance (cost) of this parking assignment, called NE, is 0.9 = 0.8 + 0.1 miles, i.e., higher than that of SO. However, the property of the NE assignment is that no driver d can unilaterally deviate from NE and improve d's cost. The NE assignment is the Nash Equilibrium, and it is assumed that without a central authority to dictate parking assignments the system will naturally settle into it. The reason for this is that drivers act selfishly, and will lower their cost if they can do so. But this means that the total driving distance, and therefore congestion, will be higher.

This paper is motivated by the fact that although there is no authority to dictate parking, municipalities have the authority to *price* on-street parking slots. Thus, we study the question: Is it possible to convert the NE parking assignment into the SO parking assignment by appropriately pricing the slots? In other words, is it possible to price the parking slots in such a way that when considering both, price and driving distance, the NE assignment becomes the SO assignment in terms of driving distance? If so, then the selfish drivers will be incentivized into the SO assignment. To demonstrate this point by continuing the above example, assume that on average driving 1 mile has a total \$-cost of 50 cents (including gas, driver-time, vehicle wear-and-tear, etc.). Then, if the authority prices s_1 at 15 cents and leaves s_2 free, it will become better for v_1 to park in s_2, leaving s_1 available for v_2. Thus, the NE assignment in terms of \$-cost + driving-distance, is the SO assignment is terms of driving distance. This means that when considering both \$-cost + driving-distance, if each driver drives to the slot assigned to her by the SO assignment (i.e. the assignment that minimizes the total driving distance), she cannot unilaterally change her slot and improve her cost. Thus, by pricing, the authority made it worthwhile for the drivers to travel a shorter distance in total.

The above pricing scheme works for the toy example of Fig. 1. What about an arbitrary configuration, i.e. initial locations of vehicles and slots? In this paper we propose an algorithm that prices the parking slots in such a way that when considering both, price and driving distance, the NE assignment becomes the SO assignment in terms of total driving distance. This algorithm is given as input the driving-distances between parking slots and vehicles. Observe that the algorithm can be generalized to take into consideration other factors such as walking distances from the slot to the destination of each driver, safety of the area around the parking slot; these may in turn depend on the weather, e.g., in the rain the importance of walking cost may be higher.

Several questions arise at this point. First, what if the parking slots already have a price? Moreover, this price may be time-dependent, e.g. \$2/hour. Then the price computed by our algorithm will be a flat fee in addition to the normal parking fees. Second, how the pricing authority would know the configuration of vehicles and slots? In other words, the pricing scheme depends on the distances between slots and vehicles, and computing these would necessitate knowing the locations of slots and vehicles. How would the authority know these? The answer is that as explained previously, the locations of available slots are already known based on sensors; and vehicles can announce their locations when starting to look for parking. Finally, how should the algorithm be adapted to a dynamic situation in which vehicles and slots enter the matching-game at different times and different locations? The answer is that every time a new slot becomes available or a new vehicle is looking to park, new prices are computed. This in turn may change the target slots for the vehicles. However, this is expected since such events change the optimum parking assignment, and adjustments need to be made. Furthermore, we assume that the target slot for a vehicle is computed automatically by some Parking Navigator that makes the best choice for the driver, and the driver may not even be aware of the price changes. A driver is only aware of the price of a slot s that exists at the time she reaches s. This is the price she pays, and is guaranteed that she cannot unilaterally improve upon it. The Parking Navigator is an add-on to a car navigation system, or an app in a smartphone, that guides the driver to a parking slot in the same sense that a Car Navigation System guides the driver to a destination.

This paper considers an additional pricing problem, called vs-pricing, which is distinguished from the previous one by the fact that rather than pricing individual slots, the authority prices pairs of (vehicle, slot); in other words, different vehicles may pay different prices for the same slot. The objective is again to convert NE assignments to SO assignments. Vs-pricing strategies are more flexible in that they can be revenue-neutral. In contrast, observe that in the regular pricing problem discussed above, the pricing authority has a revenue which can be considered as an additional tax. Vs-pricing strategies show how to distribute this additional revenue back to the drivers, so that drivers which drive longer than necessary in the SO assignment that is eventually achieved, are compensated. Moreover, drivers are guaranteed not to pay more in terms of \$-cost + driving-distance than they do in the NE assignment, upon which they cannot unilaterally improve.

Continuing with the above example from figure 1, assume again that on average driving 1 mile has a total \$-cost of 50 cents. Then to enforce a vs-pricing scheme, the pricing authority can set vehicle-slot for v_1 as an extremely large quantity for s_1 and \$0 for s_2. For v_2 the pricing authority will set the prices as 0.3 miles for s_1 and an extremely large quantity for s_2. That 0.3 miles that v_2 will have to pay converts to 15 cents. Then by setting these vehicle-slot prices, the pricing authority incentivizes the vehicles to choose slots according to the SO assignment (v_1 to s_2 and v_2 to s_1). By paying this price, *i.e.* 15 cents, v_2 actually ends up paying the same amount (by adding \$-cost and driving distance) that she would've paid had she gone to s_2, as in the NE assignment. v_1 pays more for her trouble, but the pricing authority can compensate her from the money that was collected from v_2. In this paper we will propose the vs-pricing scheme that computes the vehicle-slot pair prices in a way where every vehicle will:

- Choose parking according to the SO assignment.
- Pay the same total cost (\$-cost + driving cost) as in the NE assignment.
- Receive extra compensation from the pricing authority.

Besides these benefits for the drivers, we will show that the pricing authority will still be able to make a profit for managing these transactions.

Finally, in this paper we compute by simulation the average difference in total driving distance between the SO and NE matchings. The results showed that on average the driving distance used on the NE assignment is up to 1.3 times bigger than the distance traveled with the SO assignment. This leads to an improvement on of 23% by using our pricing schemes. Then the distances traveled on average could be potentially reduced by 23% and this leads to a 23% reduction in congestion, gas consumption and gas emissions that affect the environment.

In summary, this paper presents three contributions:

1. A pricing scheme that computes prices of parking slots; the scheme that converts the NE assignment into the SO assignment.
2. A vs-pricing scheme that assigns vehicle-slot pair prices and not only converts the NE assignment into the SO assignment, but also guarantees that no driver will pay a higher total cost than the one she can obtain by behaving selfishly.
3. The computation, through simulations, of the practical impact that our pricing schemes have.

The Roadmap The rest of the paper is organized as follows. In Section 2 we survey prior relevant work related to this research. In Section 3 we state the general setup of the related transportation problem with appropriate notations that are followed in the rest of the paper. In Section 4 we present the two possible models for the parking congestion reduction problem and related game-theoretic concepts. In Section 5 we present a pricing scheme based on an auction algorithm and prove its correctness. In Section 6 we present the vs-pricing scheme that is vehicle-dependent in the sense that the price of a slot may be different for different vehicles. In Section 7 we present simulations results to determine the empirical effectiveness of our pricing schemes. Finally, in Section 8 we present some concluding remarks.

2. PRIOR RELATED WORKS

For the purpose of this paper, we assume that all vehicles can receive information about open parking slots at any time. Such information can indeed be obtained by already published and existing research works and technologies on monitoring and sensing open parking slots. Examples of research works dealing with detection of open parking slots include the usage of ultrasonic sensor technology that is used to determine the spatial dimensions of open parking slots [14], and usage of wireless sensors that are used to track open parking slots in a parking facility [13]. Beyond simple detection of slots, Mathur *et al.* [10] show how to couple detection with sharing of the parking slot information in a mobile sensor network by presenting a methodology for vehicles driving past curbside parking slots to detect open ones, as opposed to having to spend on equipping each parking slot with wireless sensors for monitoring. These mobile sensors generate a map of parking slot availability.

In [1, 2], the authors of this paper introduced the so-called Parking Slot Assignment Games (PSAG) to analyze various parking related problems in competitive settings. The parking problem was studied in a centralized context as well as in the context of a distributed model with individual selfish agents, and a relationship between the Nash equilibrium and stable marriage assignments [6] was established in [2]. When drivers are selfish and cannot be controlled by a central authority, it is well accepted that the overall system converges to the Nash equilibrium since it describes a situation where they cannot improve on their incurred costs.

Pricing of resources to obtain some system-wide objectives as studied in this paper has been studied in the past in other contexts for transportation applications. In the transportation literature this is commonly known as "congestion pricing" [19]. The most common type of congestion pricing is that of toll-like prices assessed on major urban areas or major roads to decrease the demand of entering to these areas and roads, and pricing strategies of similar type has been famously implemented in the central business district of Singapore [15] and in other major cities across the world.

This paper investigates the pricing problems in the context of algorithmic game theory which has a rich history, see textbooks such as [12] for further details.

3. BASIC DEFINITIONS AND NOTATIONS

The general setup of our parking problem is as follows:

- We have a set of n vehicles $V = \{v_1, v_2, \dots, v_n\}$ and a set of n parking slots $S = \{s_1, s_2, \dots, s_n\}$.
- Each vehicle makes an independent choice of its parking slot, and thus competes with other vehicles for the parking slots contained in S.
- d_{ij} denotes the driving distance between $v_i \in V$ and $s_j \in S$.
- Each vehicle is assumed to be moving independently of all other vehicles at a fixed velocity. Without loss of generality, we assume that the speeds of all vehicles are the same[1].

[1]Otherwise, we simply need to rescale the distances for each vehicle in our algorithmic strategies.

- An *assignment* of vehicles to parking slots is a function $g : V \to S$, where $g(v_i)$ is the parking slot that is assigned to vehicle v_i.[2] We call this matching of vehicles to slots an assignment but it doesn't mean that there is a central authority assigning the vehicles to slots. These assignments are just matchings of vehicles and slots.

- There is a *natural cost* a_{ij} associated with a vehicle v_i and a slot s_j. This natural cost may, for example, be the distance from the vehicle to the slot, or a weighted average of the driving distance from the vehicle to the slot and the walking distance from the slots to the vehicle's *ultimate* destination.

- The *obtained natural cost* $C_g(v_i)$ of a vehicle v_i, based on an assignment g, is defined as:

  $$C_g(v_i) = \begin{cases} a_{ij} & \text{if } v_i \text{ is closest of all vehicles assigned to } g(v_i) \\ \alpha & \text{otherwise} \end{cases}$$

  Here α is a *penalty* for not obtaining its assigned slot, and is a large number, *e.g.* equal to the sum of all the natural costs between all vehicles and slots.

- We will setup a price p_j for each parking slot s_j when we consider the pricing problem for parking. The *priced cost* between a vehicle v_i and a slot s_j is defined as $a_{ij} + p_j$.

- The *obtained priced cost* $C_g^p(v_i)$ of a vehicle v_i based on an assignment g is defined as:

  $$C_g^p(v_i) = \begin{cases} a_{ij} + p_j & \text{if } v_i \text{ is closest of all vehicles assigned to } g(v_i) \\ \beta & \text{otherwise} \end{cases}$$

  Here β is a *penalty* for not obtaining its assigned slot, and is a large number, *e.g.* equal to the sum of all the priced costs between all vehicles and slots.

- We assume that one has a "conversion factor" between the natural costs (a_{ij}'s) and monetary prices. So when we talk about combining the prices (p_j's) and the natural costs into the obtained priced cost ($a_{ij} + p_j$), we assume that the natural cost is in terms of dollars. For example, if the natural cost is simply the driving *time* to the slot, then one would need a suitable conversion of time into money. We will assume that this conversion factor will be chosen by the pricing authority (whoever has the right to set the prices).

4. THE PARKING PROBLEM

In a parking problem, we wish to compute an assignment g of vehicles to slots that will optimize some objective. One natural way of modeling the problem is in a centralized manner in which a central authority will assign vehicles to slots in order to optimize some *system-wide* objectives. Another way of modeling the problem is to let the vehicles choose their slots independently and selfishly, and study the competition between the vehicles for the slots. In this section we will present solutions for both these models.

[2]Based on this definition, there is a difference between where a vehicle is assigned and where a vehicle parks. If more than one vehicle is assigned to the same slot, then the vehicle closest to the slot will park there and the other vehicles are left without parking.

4.1 Centralized Model – Optimizing Social Welfare

In a centralized model, a central authority is in charge of assigning vehicles to slots with the goal of optimizing some system-wide objectives ("social welfare"). In the transportation literature this is usually called a *system optimal assignment* in which the total obtained natural cost of the assigned vehicle-slot pairs is minimized, *i. e.*, the assignment g minimizes the following objective function:

$$\sum_{i=1}^{n} C_g(v_i)$$

In [1], we showed how such a system optimal assignment can be computed in polynomial time by posing it as a minimum-cost network flow problem on a bipartite graph [4]. Even though this centralized model shows good computational properties, it is difficult to justify in real life to distributed mobile users that make their own choices, *e.g.*, optimizing social welfare may imply that some travelers will incur a greater cost for the good of others.

4.2 Distributed Model – Nash Equilibrium

In a distributed model there is no centralized authority that guides drivers and each user makes an independent and selfish choice for its parking slot. These choices are used to determine the vehicle-slot assignment g. Viewed in another equivalent way, such a model can be viewed as a *game* where the strategies of the players (vehicles) are the available parking slots, and the payoff (cost) for the players are their obtained natural cost (or obtained priced cost when there are slot prices). The well-known Nash equilibrium [11] is a standard desired strategy that is used to model the individual choices of selfish players in a game by defining a situation in which no player can decrease its cost by changing strategy unilaterally. The standard definition of Nash equilibrium translates to the following definition for our distributed parking slot assignment model:

DEFINITION 1. (Nash Equilibrium for distributed parking slot assignment model) *Let g be an assignment of vehicles to slots that represents the strategy choices of the players in the distributed parking slot assignment model. Then, g is a Nash equilibrium strategy for all the players if and only if, for every index i and for every assignment g' that is identical to g except that $g(v_i) \neq g'(v_i)$, it holds that*

$$C_g(v_i) \leq C_{g'}(v_i)$$

In [2], we have shown that the Nash equilibrium strategy for parking slot assignment games is actually a stable marriage [6] between the vehicles and the slots in the following manner. For this stable marriage, the preferences of the vehicles are determined by their natural cost (a_{ij}) and the preferences of the slots are determined by the distances to the vehicles (d_{ij}). When the parking slots have prices assigned to them, the preferences of the vehicles will be determined by their obtained priced cost ($a_{ij} + p_j$), and this equilibrium can then also be computed in polynomial time using the Gale-Shapley deferred acceptance algorithm [6, 12].

The Nash equilibrium exemplifies the behavior of a system with individual selfish agents. This applies to real-world applications because travelers in the real-world are selfish and look to minimize their own costs rather than those of the system. Nevertheless, the system optimal assignment is the one that minimizes congestion caused by vehicles cruising for parking since vehicles travel less as a whole. This system optimal assignment also has greater benefits for the environment. This is why we want to design a pricing scheme that makes the system optimal assignment be equal to the Nash equilibrium assignment. With this pricing scheme, the selfish vehicles will travel in accordance with the system optimal assignment.

5. EQUILIBRIUM PRICING SCHEME

We will denote a parking pricing scheme as a set of prices that are assigned to each available parking slot and represent it as a n-tuple $\mathcal{P} = (p_1, p_2, \ldots, p_n)$. We will say that $\mathcal{P}$ is an *equilibrium pricing scheme* if it makes the Nash equilibrium assignment identical to the system optimal assignment. In this section we will present an algorithm that computes such a desired equilibrium pricing scheme. Our algorithm is based on the auction algorithm that was created to compute optimal assignments [3].

5.1 Algorithmic Preliminaries and Notations

In this section we describe the intuition behind an algorithm that computes the desired equilibrium pricing scheme through an iterative process that simulates an auction among the vehicles.

If a vehicle v_i is assigned and obtains slot s_j, then the obtained priced cost for v_i is $a_{ij} + p_j$. Ideally, v_i would want to be assigned to a slot s_j such that:

$$a_{ij} + p_j = \min_{1 \le k \le n} \left\{ a_{ik} + p_k \right\}$$

If this is possible, then this vehicle v_i would have no incentive to deviate to another strategy and thus this strategy would be an equilibrium strategy for v_i. If this condition holds for all vehicles with their assigned and obtained slots, then the assignment is an equilibrium.

Let ε be called the “minimum bidding increment” for the algorithm (we will soon define what this means). Then, define a vehicle v_i as being in *almost at equilibrium* with an assignment and a set of prices if the value of its assigned slot s_j is within ε of being minimal, *i.e.*,

$$a_{ij} + p_j \le \min_{1 \le k \le n} \left\{ a_{ik} + p_k \right\} + \varepsilon$$

We will say that an assignment and a set of prices are *almost at equilibrium* (or at ε-approximate equilibrium) when the above condition holds for *all* vehicles with their assigned and obtained slots. We will now present an algorithm that will compute prices that make an ε-approximate equilibrium also be an optimal solution within a factor of ε. The algorithm will compute the desired prices in a way such that vehicles will be almost at equilibrium when assigned with a system-optimal assignment. We note that similar definitions of ε-approximate equilibrium have been used before in the algorithmic game theory community (*e.g.*, see [5, 8, 9]).

5.2 The Auction Algorithm for Pricing Parking Slots

The algorithm executes in “rounds” or iterations starting with an arbitrary assignment and an arbitrary set of prices. We will assume that we start with all prices set at 0. There is an assignment and a set of prices at the end of each iteration. If all the vehicles are at almost equilibrium with their assigned parking slot at the end of any round then the algorithm terminates. Otherwise, any vehicle that is not almost at equilibrium, say vehicle v_i, is selected. Let s_j be the slot that has minimal cost for v_i, *i.e.*:

$$j = \operatorname*{argmin}_{1 \le k \le n} \{ a_{ik} + p_k \}$$

Then the following steps are executed:

1. v_i exchanges slots with the vehicle assigned to s_j at the beginning of the round.

2. v_i sets the price of his/her best slot s_j to the level at which he/she is indifferent between s_j and his/her second most preferred slot in the following manner. Let $x_i = a_{ij} + p_j$ be his/her obtained cost for the most preferred slot, and let $w_i = \min_{k \ne j} \left\{ a_{ik} + p_k \right\}$ be his/her obtained cost for the second most preferred slot. Then p_j is set to:

$$p_j + \gamma_i,$$

where $\gamma_i = w_i - x_i + \varepsilon$. Basically, γ_i is the highest value that s_j's price can be increased while still being v_i's preferred slot. Also notice that the minimum value for γ_i is ε (the minimum bid increment).

This algorithm continues in a sequence of rounds until all vehicles are at almost equilibrium. The iterative approach can be viewed as an *auction* where v_i raises the price of his bid on slot s_j by the bidding increment γ_i.

Primal-dual interpretation of the auction algorithm.

Readers familiar with the primal-dual approach for solving linear programs by iteratively satisfying complementary slackness conditions [4] will realize that the above auction algorithm can be interpreted as a primal-dual schema in the following manner: start with a feasible (not necessarily optimal) solution of the dual linear program for the parking problem [12, Section 11.3.1] by, for example, setting all the dual variable (prices) to zeroes and iteratively increase dual variables until all complementary slackness conditions are satisfied.

5.3 Proof of Correctness

The auction algorithm as described in the previous section is guaranteed to terminate in a finite number of steps that depends on ε [3]. Since the algorithm terminates when all vehicles are almost at equilibrium, obviously it computes an ε-approximate equilibrium in terms of the priced cost. Thus, what remains to be done is to show the relationship of this assignment to the system optimal assignment in terms of the natural cost. The following theorem is a special case of a more general result on the existence of an ε-approximate competitive equilibrium of ascending item-price auctions [12, Theorem 11.30].

THEOREM 2 (SEE ALSO [12, THEOREM 11.30]). *The obtained priced cost of the assignment computed by the auction algorithm for parking pricing is within an additive factor of $n\varepsilon$ of the minimum value of the total obtained natural cost.*

PROOF. Let $h \colon V \to S$ be an arbitrary assignment, and let g be the assignment computed by the auction algorithm resulting in the pricing scheme $\mathcal{P} = (p_1, p_2, \ldots, p_m)$. According to the auction algorithm and with subsequent algebraic manipulation, we get the following inequalities:

$$
\begin{aligned}
& a_{ig(v_i)} + p_{g(v_i)} \leq a_{ih(v_i)} + p_{h(v_i)} + \varepsilon \\
\Rightarrow \quad & \sum_{i=1}^{n} (a_{ig(v_i)} + p_{g(v_i)}) \leq \sum_{i=1}^{n} (a_{ih(v_i)} + p_{h(v_i)} + \varepsilon) \\
\equiv \quad & \sum_{i=1}^{n} a_{ig(v_i)} + \sum_{i=1}^{n} p_{g(v_i)} \leq \sum_{i=1}^{n} a_{ih(v_i)} + \sum_{i=1}^{n} p_{h(v_i)} + n\,\varepsilon \\
\Rightarrow \quad & \sum_{i=1}^{n} a_{ig(v_i)} \leq \sum_{i=1}^{n} a_{ih(v_i)} + n\,\varepsilon
\end{aligned}
$$

where the last implication follows from the fact that

$$\sum_{i=1}^{n} p_{g(v_i)} = \sum_{i=1}^{n} p_{h(v_i)} = \sum_{j=1}^{n} p_j$$

□

Thus, we can use the auction algorithm to compute the desired prices that ensure that the ε-approximate equilibrium assignment is identical to the system optimal assignment within a factor of ε. In this way, vehicles looking for parking are incentivized to act in a way that benefits the system by reducing the total time traveled to find parking. The potential benefits of the algorithm are studied empirically through simulations in Section 7.

5.4 Illustrative example of the pricing scheme

We illustrate the pricing approach using an example shown in Fig. 1. Suppose that the numbers represent one-hundredths of a mile, *i.e.*, v_1 is 0.1 miles from s_1, 0.2 miles from s_2 and so forth. The Nash equlibrium assignment g_{ne} for this situation is given by $g_{ne}(v_1) = s_1$ and $g_{ne}(v_2) = s_2$, and the system optimal assignment g_{so} is given by $g_{so}(v_1) = s_2$ and $g_{so}(v_2) = s_1$.

If we were to run our auction algorithm, with starting prices at 0, for this parking situation, we would get that s_1 will have a price of $0.3 + \varepsilon$ miles and s_2 will be free. The monetary amount that these $0.3 + \varepsilon$ miles will eventually translate to, depends on the conversion factor. This conversion factor is set by the pricing authority based on an estimate of what is the cost of driving for an average user (based on gas expenditure, time, etc.). Suppose that $\varepsilon = 0$ for this example and that users feel that one mile of travel is worth 50¢. Then the price of parking in s_1 would be 15¢. This price of 15¢ will make s_2 more valuable for v_1 and would make the equilibrium assignment identical to the system optimal one.

5.5 Illustrative example of the pricing scheme with general costs

Take the example shown in Fig. 1. Now we add walking times by defining the time to walk from slot $s_j \in S$ to the destination of $v_j \in V$ as w_{ij}. For this example, let $w_{11} = 30$, $w_{12} = 18$, $w_{21} = 42$, and $w_{22} = 6$. Let the weights in Fig. 1 now represent travel times as well. Then we can combine the walk times and the driving times into a more general cost with the cost values defined as: $a_{11} = 40$, $a_{12} = 38$, $a_{21} = 92$, and $a_{22} = 86$.

If we were to run our auction algorithm, with starting prices at 0, for this parking situation, we would get that s_2 will have a price of $6 + \varepsilon$ seconds and s_1 will be free. Again, the monetary amount that these $6 + \varepsilon$ seconds will eventually translate to, depends on the conversion factor between seconds (time) and money. This pricing scheme will lead to v_1 preferring slot s_1 and v_2 going to slot s_2, which is the system optimal assignment when minimizing this general cost.

6. VEHICLE-SLOT (VS) PRICING SCHEME

6.1 Formal Definitions

In the previous scheme, prices were set to each available parking slot *regardless* of who was going to park there, *i.e.*, the prices did not depend on the assigned vehicles. One could also consider a vs-pricing scheme in which the assigned price *depends* on the vehicle that wants to park there. Thus, instead of having a pricing scheme $\mathcal{P} = (p_1, p_2, \ldots, p_n)$ like before, we now have a vs-pricing scheme $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_n)$, where each $\mathcal{P}_i = (p_{i1}, p_{i2}, \ldots, p_{in})$ for $1 \leq i \leq n$, and each p_{ij} represents the price that vehicle v_i would have to pay to park in slot s_j. This is a *vehicle-dependent pricing scheme*. The prices in this scheme will again be designed to incentivize drivers into making parking slot choices in a system optimal manner.

6.2 Computing the pricing scheme

THEOREM 3. *There exists a vs-pricing scheme $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_n)$ that converts the system optimal assignment of the vehicles and slots to a Nash equilibrium assignment considering the priced costs.*

PROOF. Let $g_{so} \colon V \to S$ be a system optimal assignment between the vehicles and slots and let g_{ne} be the Nash equilibrium assignment. If $g_{ne}(v_i) = s_j$ then let $j_i^{ne} = j$, and similarly if $g_{so}(v_i) = s_j$ then let $j_i^{so} = j$. Set p_{ij} as:

$$p_{ij} = \begin{cases} \max\left\{0,\, a_{ij_i^{ne}} - a_{ij_i^{so}}\right\} & \text{if } g_{so}(v_i) = s_j \\ \beta & \text{otherwise} \end{cases} \tag{1}$$

Recall (from section 3) that β is a sufficiently large quantity and p_{ij} is the price that vehicle v_i would have to pay if it wants to park in slot s_j. Since β is sufficiently large, each v_i will have incentive to only choose the strategy $g_{so}(v_i)$. Then, $g_{so}(v_i)$ is the equilibrium strategy for v_i as well as the system optimal strategy (by definition). Thus, if each $\mathcal{P}_i = (p_{i1}, p_{i2}, \ldots, p_{in})$ is computed using Equation (1), then the pricing scheme $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_n)$ makes the Nash equilibrium assignment be identical to the system optimal assignment. □

With the vs-pricing scheme presented in the above proof, no vehicle will pay more than the price dictated by the original Nash equilibrium assignment g_{ne}. If a vehicle travels less than what it would have originally traveled according to original Nash equilibrium then it has to pay a \$-cost of $a_{ij_i^{ne}} - a_{ij_i^{so}}$ and additionally it incurs the cost of travel which is $a_{ij_i^{so}}$. Therefore such a vehicle ends up paying exactly $a_{ij_i^{ne}}$, which is the natural cost it would have paid originally with the assignment g_{ne}. However, if a vehicle travels more than originally stipulated by the Nash equilibrium then it does not have to pay an extra \$-cost according to equation 1. But it travels more and therefore it ends up paying $a_{ij_i^{so}} > a_{ij_i^{ne}}$, a driving cost. The next paragraph explains how to compensate such an unhappy vehicle.

Let us call a vehicle *happy* if it pays the original cost stipulated by the equilibrium. Suppose the pricing authority pays back to an unhappy vehicle the amount of $a_{ij_i^{so}} - a_{ij_i^{ne}}$. Then the user of this vehicle will end up paying a total of $a_{ij_i^{so}} - (a_{ij_i^{so}} - a_{ij_i^{ne}}) = a_{ij_i^{ne}}$, which is the natural cost the user would have paid originally with the Nash equilibrium assignment g_{ne}. This would make the user *happy.*

Thus, it follows that with this scheme there is a potential for everyone to be happy because all users will simply pay the same costs that they would have paid originally with the Nash equilibrium assignment. The only question that is left to be answered is if the pricing authority will also make a profit as well.

THEOREM 4. *The vs-pricing scheme described above yields a non-negative profit for the pricing authority.*

PROOF. Using simple algebraic manipulation, the total profit for the pricing authority can be ultimately written as:

$$\sum_{i:\, a_{ij_i^{ne}} > a_{ij_i^{so}}} \left[a_{ij_i^{ne}} - a_{ij_i^{so}}\right] - \sum_{i:\, a_{ij_i^{so}} \geq a_{ij_i^{ne}}} \left[a_{ij_i^{so}} - a_{ij_i^{ne}}\right]$$

$$= \sum_{i:\, a_{ij_i^{ne}} > a_{ij_i^{so}}} \left[a_{ij_i^{ne}} - a_{ij_i^{so}}\right] + \sum_{i:\, a_{ij_i^{so}} \geq a_{ij_i^{ne}}} \left[a_{ij_i^{ne}} - a_{ij_i^{so}}\right]$$

$$= \sum_{i=1}^{n} \left[a_{ij_i^{ne}} - a_{ij_i^{so}}\right] = \sum_{i=1}^{n} a_{ij_i^{ne}} - \sum_{i=1}^{n} a_{ij_i^{so}} \geq 0$$

where the last inequality follows since the total cost of g_{ne} is by definition greater than the total cost of g_{so}. $\square$

Thus, the pricing authority could act as a broker where:

- it collects payments from the vehicles that travel *less* with the system optimal assignment,
- it pays an amount to the vehicles that travel *more* with the system optimal assignment, and
- it makes a profit.

Since the vehicles pay exactly what they would have paid originally with the Nash equilibrium and the pricing authority is also making a profit, everybody is "happy". Furthermore, the pricing authority could potentially make the users even *happier* by splitting a fraction of the profits with them and still keeping the other fraction of the profits for itself. In this case each vehicle will pay a total cost that is lower than the one that they would have paid by being selfish.

It should also be noted that this pricing scheme could also work for the more general case where the number of vehicles is not equal to the number of slots. If there are more vehicles than slots, the vehicles that would have been left unassigned in the system optimal assignment would be charged really high prices for all the slots. This would incentivize them to not park so as to not pay such a high price for parking. We have not yet shown that the pricing scheme presented in section 5 could work with differing amounts of vehicles and slots.

6.3 Illustration of vs-pricing scheme

In this section we again analyze the parking situation of Fig. 1, but now with this new vs-pricing scheme. With this new scheme, the prices of s_1 and s_2 for v_1 are $p_{11} = \beta$ and $p_{12} = 0$, respectively, and the prices of s_1 and s_2 for v_2 are $p_{21} = 0.3$ and $p_{22} = \beta$, respectively. This means that the pricing authority will charge 0.3 miles to v_2 so that it can get s_1. If the conversion factor is again 50¢ per mile then v_2 will have to pay 15¢. Then, the pricing authority will pay back 0.1 miles $\equiv$ 5¢ to v_1, and will make a profit of 10¢ as a broker. These 10¢ could potentially be used to further compensate the drivers for driving optimally by giving them a fraction of these 10¢ and keeping the other fraction as profit.

7. SIMULATION AND RESULTS

In this section we use simulations to determine empirically, on average, the benefits of using the pricing schemes.

7.1 Simulation Setup

The goal of our simulation is to test the solution concepts presented in Section 4 to empirically ascertain how much better off is the transportation system and the environment by using our proposed pricing schemes.

Our simulations were run on a road network (grid) that was embedded in an Euclidean space. The positions of the vehicles, the motion of the vehicles and locations of the slots were restricted to be on the network. The number of vehicles and slots (n) was a parameter of the simulation. The values of n that were tested were $n = 25k$ for $k = \{1, 2, \ldots, 12\}$. A system optimal assignment and a Nash equilibrium were computed for each configuration of vehicles and slots, and the total distance traveled by all the vehicles based on these two assignment was saved. This means that the natural cost (a_{ij}) for these simulations was simply the driving distance (d_{ij}) between the vehicles and the slots. This choice was made to determine what are the environmental benefits that are obtained from the pricing scheme.

We also tested a varying number of *competitive ratios.* Say now that there are m available slots and n vehicles. Then define the competitive ratio as n/m. The higher the competitive ratio, the bigger the competition for the available slots.

We also tested a varying number of regional skews of the locations of slots. In reality, available parking slots are not uniformly distributed across a road network. Thus, we generate skewness as follows. The map is partitioned into 16 equal-sized square regions. A random permutation of the regions is generated (uniform distribution) and is used as the ranking of the popularity of each region for available slots. To choose the location of each of the available slots, first a random number is generated to determine in which region to place the slot. The Zipf distribution with its skew parameter and the regional popularity previously generated are used to generate this random number. Then a random position in the grid (uniform) is chosen from the region denoted by the Zipf number. The n vehicles' initial positions are generated using the uniform distribution on the grid.

For each value of simulation configuration under consideration, 1000 different simulation runs were tested and averaged. Each test was done to compute what is basically an average *price of anarchy* (PoA). PoA is the ratio of the total cost in the Nash equilibrium assignment to the total cost of the system optimal assignment [12]. In [1] we showed that *in general* the PoA of the parking problem is *unbounded.* In these simulations we computed the average PoA to determine the average benefit of using our proposed parking pricing schemes for the system.

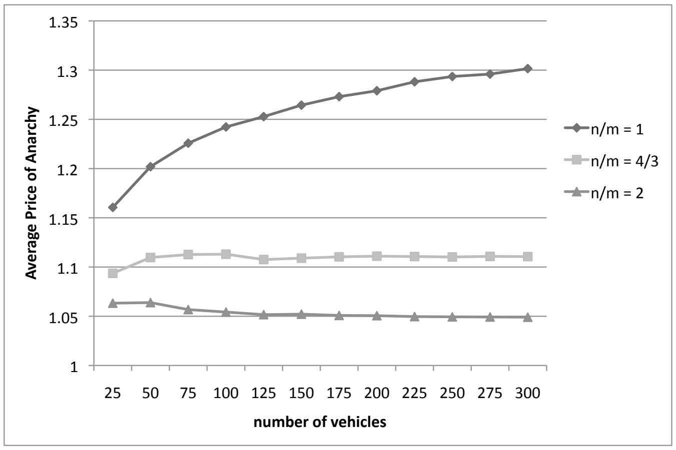

Figure 2: Average Price of Anarchy with varying values of n and varying competitive ratio (skew = 0).

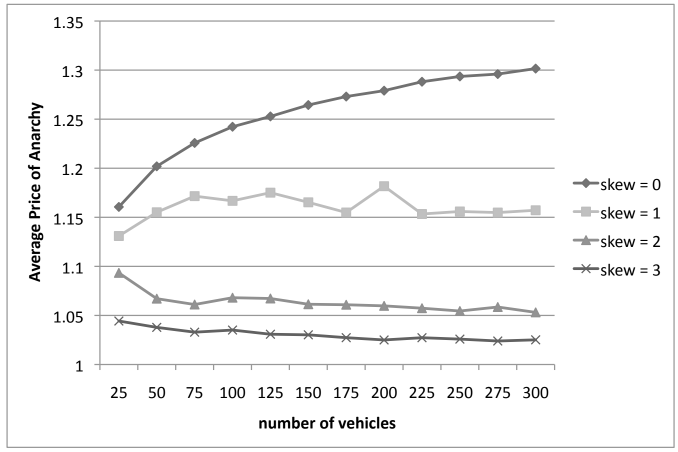

Figure 3: Average Price of Anarchy with varying values of n and varying regional skew (n/m = 1).

The parameters for the simulation are:

- n - the number of agents.
- n/m - the competitive ratio between vehicles and slots.
- $skew$ - the regional skew of the Zipf distribution.

The values that were tested for each parameter are detailed in table 1. For each configuration of the parameters, 100 different simulation runs were generated and tested.

Parameter	Symbol	Range
Vehicles	n	{25,50,...,275,300}
Competitive Ratio	n/m	$\{2,\frac{4}{3},1\}$
Regional Skew	-	{0, 1, 2, 3}

Table 1: Parameters tested on Simulation

7.2 Simulation Results

Fig. 2 shows the results for the average PoA computation for various values of n. It also shows the results for varying values of competitive ratio (n/m).

We can see that the highest values of average PoA were attained when the competitive ratio was lowest ($n/m = 1$). This means that as the availability of slots increases, so does the congestion cost incurred by the system based on the Nash equilibrium assignment compared to the System optimal assignment.

We can see that at $n = 300$ and $n/m = 1$, the average PoA that was obtained was around 1.3. This was the highest PoA obtained in all simulations. This means that for every mile traveled by each vehicle with the system optimal assignment, the vehicles will travel 1.3 miles when using the Nash equilibrium assignment on an average. That in turn means that the percent improvement of the system optimal assignment can be up to $1 - {}^{1}/_{1.3} \approx 23\%$.

Figure 3 shows some results with varying skew. We can also see that the highest values of average PoA were obtained when the skew was 0, *i.e.* slots were distributed uniformly across the road network.

According to the analysis presented in Section 1, cruising for parking accounts for 172 million vehicle miles traveled (VMT) per year in a big urban city like Chicago. Furthermore, this accounts for waste of 8.37 million gallons of gasoline and over 129,000 tons of CO_2 emissions per year. By using our proposed pricing scheme and incentivizing vehicles to use the system optimal assignment to choose their parking slots, we can get an improvement of up to 23% which leads to a reduction of up to 39 million VMT per year year in a city like Chicago. This would account for a reduction of 1.9 million gallons of gasoline and over 29,000 tons of CO_2 emissions per year.

8. CONCLUSIONS AND FUTURE WORK

In this paper we studied the problem of assigning prices to available parking slots to incentivize travelers to choose parking slots in a way that helps the system and environment. We presented the system optimal assignment and the Nash equilibrium assignment of vehicles to parking slots. In general, the system optimal assignment is more economical in terms of the total distance traveled by all the vehicles. However, due to the lack of an authority to impose parking assignments, drivers act selfishly leading to the Nash equilibrium. We introduced a pricing scheme, based on the auction algorithm [3], that makes the Nash equilibrium assignment and the system optimal assignment be approximately identical. Additionally, we presented a vs-pricing scheme that not only has the desired property that the vehicles will move in accordance with the system optimal assignment but also keeps all vehicles happy by guaranteeing that each vehicle pays a cost equal to its (selfish) equilibrium cost. According to simulations that were run, this type of pricing scheme can lead to improvements in total distance traveled of up to 23%. In a big city like Chicago this leads to improvements of up to 39 million vehicle miles traveled, 1.9 million gallons of gasoline and over 29,000 tons of CO_2 emissions per year.

The pricing schemes presented here work in an offline setting. The number of vehicles and resources are known and don't change. However, the parking problem in real life works in an online setting since vehicles and slots come in to the system at future times. One could recompute the prices every time a new object arrives to the system. Nevertheless, the effectiveness of this idea would need to be proven for this online framework. Future work for this research includes modeling the parking pricing problem in an online setting and finding a suitable pricing scheme that adapts to changes in the vehicles and parking slots while still manages to move vehicles in a system optimal fashion in the online sense.

9. REFERENCES

[1] D. Ayala, O. Wolfson, B. Xu, B. Dasgupta, and J. Lin. Parking slot assignment games. In *Proc. of the 19th Intl. Conf. on Advances in Geographic Information Systems (ACM SIGSPATIAL GIS 2011)*, Chicago, IL, November 2011.

[2] D. Ayala, O. Wolfson, B. Xu, B. DasGupta, and J. Lin. Parking in competitive settings: A gravitational approach. In *Proc. of 13th Intl. Conf. on Mobile Data Management (MDM)*, Bengaluru, India, July 23-26 2012.

[3] D. Bertsekas. The auction algorithm for assignment and other network flow problems: A tutorial. *Interfaces*, 20(4, The Practice of Mathematical Programming):133–149, July 1990.

[4] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver. *Combinatorial Optimization*. John Wiley & Sons, 1998.

[5] E. Even-Dar, A. Kesselman, and Y. Mansour. Convergence time to nash equilibria. In *Proc. of the 30th Int. Conference on Automata, Languages and Programming*, pages 502–513, 2003.

[6] D. Gale and L. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[7] http://sfpark.org/.

[8] M. Kearns and Y. Mansour. Efficient nash computation in large population games with bounded influence. In *18th Conference in Uncertainty in Artificial Intelligence*, pages 259–266, 2002.

[9] R. Lipton, E. Markakis, and A. Mehta. Playing large games using simple strategies. In *4th ACM Conference on Electronic Commerce*, pages 36–41, 2003.

[10] S. Mathur, T. Jin, N. Kasturirangan, J. Chandrashekharan, W. Xue, M. Gruteser, and W. Trappe. Parknet: Drive-by sensing of road-side parking statistics. In *MobiSys*, San Francisco, CA, June 2010.

[11] J. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.

[12] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA, 2007.

[13] B. Panja, B. Schneider, and P. Meharia. Wirelessly sensing open parking spaces: Accounting and management of parking facility. In *AMCIS 2011 Proceedings*, 2011.

[14] W. Park, B. Kim, D. Seo, D. Kim, and K. Lee. Parking space detection using ultrasonic sensor in parking assistance system. In *IEEE Intelligent Vehicles Symposium*, Eindhoven, The Netherlands, June 2008.

[15] S.-Y. Phang and R. S. Toh. Road congestion pricing in singapore: 1975 to 2003. *Transportation Journal*, 43(2):16–25, 2004.

[16] D. Shoup. *The High Cost of Free Parking*. American Planning Association, Chicago, IL, 2005.

[17] D. Shoup. Cruising for parking. *Transport Policy*, 13:479–486, 2006.

[18] Transportation Alternatives (www.transalt.org), New York, NY. *Pricing the Curb: How San Francisco, Chicago and Washington D.C. are reducing traffic with innovative curbside parking policy*, July 2008.

[19] B. Wie and R. Tobin. Dynamic congestion pricing models for general traffic networks. *Transportation Research Part B - Methodological*, 32(5):313–327, 1998.