Managing Competition in Spatial Computing

Ouri Wolfson,

University of Illinois, Chicago

A lot of work has been devoted to collaboration among distributed computing devices. Sensor networks and crowdsourcing are prominent examples of distributed collaborative computing. Much less attention has been paid to competition among computing devices, particularly competition for space. An example of such competition is drivers, guided by their smartphones, attempting to park; they are competing for a limited number of parking spaces. Similarly, car navigation systems compete for road space, commuters compete for seats on a bus, taxi cabs compete for customers, and customers compete for taxi cabs. In other words, spatial competition abound.

In this paper we use parking to illustrate the challenges of information systems that manage spatial competition. Cruising for parking by driving around an urban area looking for available parking slots has been shown to be a major cause of congestion. For example, 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 (see [1]). This means that each parking slot would generate 4,927 vehicle miles traveled (VMT) per year [2]. 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 [5], the total number of VMT becomes 172 million per year due to cruising 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 to help drivers find parking in urban settings are becoming available. For example, wireless sensors embedded in parking slots are used to detect the availability of slots in some area, and the locations of currently available parking slots are disseminated to the mobile devices of drivers that are looking for parking in the area. A municipality that uses sensors embedded in the streets is San Francisco (see SFPark [3]). 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.

We propose that smartphone apps that navigate drivers to parking slots be developed. Furthermore, they have a focus on a conceptual gap between two notions of parking, which is a spatio-temporal matching between mobile agents (drivers) and spatial resources (parking slots). The two matching notions are *optimality* and *equilibrium.* Ideally, we would like the matching to be optimal, i.e. minimize the total time driven to park by all vehicles. However, achieving this optimality requires a central authority that can eliminate competition by dictating the slot in which a driver should park, even if that driver can do better. Figure 1 below illustrates this point by an example.

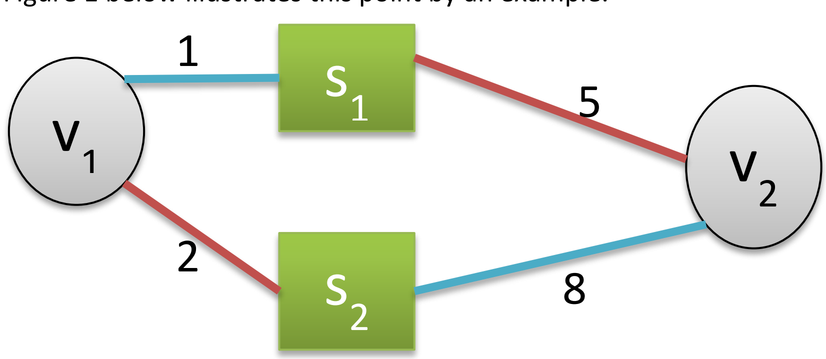

Figure 1: A parking example with two vehicles and two parking slots

Suppose that the edge labels represent travel times in minutes, i.e., vehicle v_1 is 1 minute away from slot s_1, 2 minutes away from s_2 and so forth. To achieve minimum total driving time, v_1 will have to park in s_2, and v_2 will have to park in s_1. This parking configuration, called minimum, has a total time of 7 minutes. However, this requires v_1 to drive to a farther slot, s_2, i.e. an inferior slot from her point of view because s_1 is closer. There is no central authority that can dictate this parking configuration 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 time of this configuration, called equilibrium, is 9 minutes, i.e., higher than the minimum. However, in an equilibrium configuration no driver *d* can unilaterally deviate and improve *d*'s cost. In other words, if there is no competition then v_2's parking app should navigate her to s_1, the closest slot; whereas if there is competition from v_1, the app should navigate her to s_2. This means that the parking app P should navigate differently in the face of competition for spatial resources. Furthermore, P usually does not know whether there is competition, and where the competing devices are located. In [4] we introduced a method of competitive parking, Gravitational Parking, that is provably superior to the one that simply goes to the closest slot, i.e. ignores competition. The superiority reaches 30%. This means that the price of using a noncompetitive algorithm in competitive situations in Chicago would be 25 million Vehicle/miles traveled per year, over 1.2 million gallons of wasted gasoline, and over 19,000 tons of CO_2 emissions.

Gravitational Parking works by having a vehicle be "attracted" to available parking slots, i.e., slots applying a "gravitational force" on vehicles. Each force is represented by a vector, and the vehicle moves in the direction of the vector-sum of the forces. This means that the vehicle *v* does not always pursue an available parking slot because the slot may become unavailable by the time *v* reaches it. Instead, it moves in a general direction that is promising in the sense that contains multiple available slots. Figure 2 shows the gravitational force field generated by five available slots. The arrows represent the direction in which a vehicle will move when it is located at the start point of the arrow, and the small dots represent the slots. This diagram indicates how vehicles move across the map when using Gravitational Parking, and it shows that they will eventually converge to a slot.

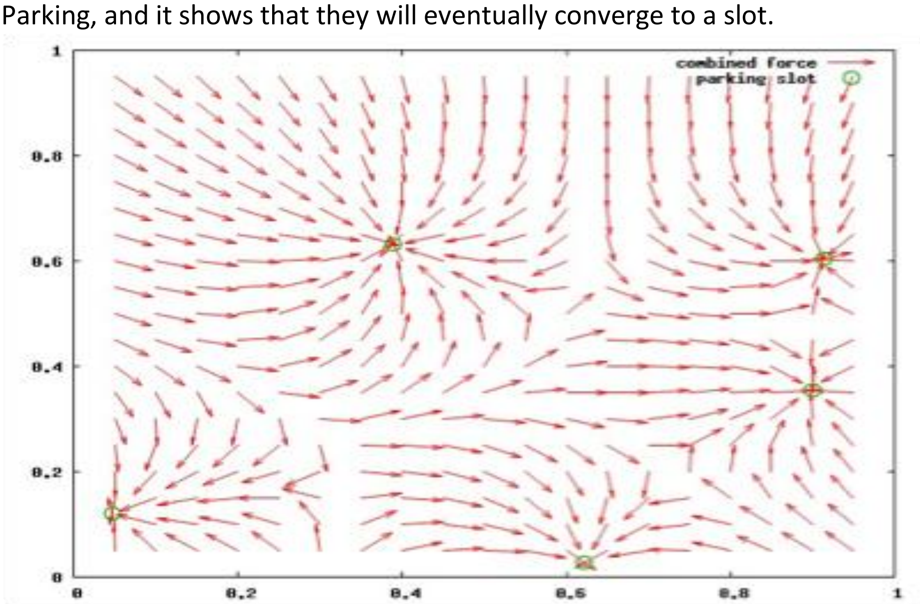

Figure 2: Field force generated by 5 slots

How should Gravitational Parking be extended to account for competition in routing? In other words, the shortest-distance path to a destination is inferior if it is congested. And a shortest-time path will also be inferior if many drivers are led to it by their car navigation systems. This situation becomes increasingly likely as car navigation systems with traffic information proliferate, leading to the phenomenon called herding. Similarly, what is the equivalent of Gravitational Parking in competing for other spatial resources? To address these questions we propose competition management as research direction for the emerging spatial computing community.

References:

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

[2] D. Shoup. Cruising for parking. Transport Policy, 13: 479-486, 2006.

[3] http://sfpark.org/

[4] 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

[5] 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.