Collaboration and Shared Plans in the Open World: Studies of Ridesharing

Ece Kamar *
SEAS, Harvard University
Cambridge, MA 02138
kamar@eecs.harvard.edu

Eric Horvitz
Microsoft Research
Redmond, WA 98052
horvitz@microsoft.com

Abstract

We develop and test computational methods for guiding collaboration that demonstrate how shared plans can be created in real-world settings, where agents can be expected to have diverse and varying goals, preferences, and availabilities. The methods are motivated and evaluated in the realm of ridesharing, using GPS logs of commuting data. We consider challenges with coordination among self-interested people aimed at minimizing the cost of transportation and the impact of travel on the environment. We present planning, optimization, and payment mechanisms that provide fair and efficient solutions to the rideshare collaboration challenge. We evaluate different VCG-based payment schemes in terms of their computational efficiency, budget balance, incentive compatibility, and strategy proofness. We present the behavior and analyses provided by the ABC ridesharing prototype system. The system learns about destinations and preferences from GPS traces and calendars, and considers time, fuel, environmental, and cognitive costs. We review how ABC generates rideshare plans from hundreds of real-life GPS traces collected from a community of commuters and reflect about the promise of employing the ABC methods to reduce the number of vehicles on the road, thus reducing CO_2 emissions and fuel expenditures.

1 Introduction

We investigate challenges with the generation of efficient collaborative plans for self-interested people based on their availabilities and preferences, and with providing fair incentives to promote collaboration. We frame and motivate the development of methods with the real-world challenge of generating shared transportation plans, referred to commonly as *ridesharing*. Rideshare plan generation is an interesting and representative open-world collaboration problem because of the varying goals, diverse preferences, and changing locations and availabilities of actors. Beyond the intriguing technical challenges, the pursuit of effective rideshare plan generation is also motivated by the potential value of implementing wide-scale ridesharing for the environment and economy. Transportation is a significant source of CO_2 emissions [e Sustainability Initiative, 2008]; ridesharing has been proposed as a promising means for reducing these emissions as well as fuel expenditures. Although making use of unfilled seats in cars can deliver personal and more global value, participants in a carpool can incur costs with ridesharing, including fuel and time costs associated with the lengthening of a commute containing new waypoints, and a shifting of the departure and arrival times to match the needs of others.

We present a formal methodology for generating ideal ridesharing plans for a community of users. We pursue incentive mechanisms that provide fair and efficient solutions while respecting the privacy of users and promoting truthful behavior. The coordination machinery generalizes to other settings where agents need to collaborate on joint plans to minimize their cumulative cost, and obtain fair payments as incentives. Beyond usage in online systems for generating batch and real-time rideshare plans, the methods provide tools for exploring the influence of changing parameters, such as the cost of fuel and time and numbers of participants, on the overall efficiencies and savings achieved.

We shall discuss several phases of analysis for constructing collaborative plans for ridesharing, including the acquisition of changing user costs and preferences for ridesharing, the solution of computationally expensive optimization problems and the use of VCG-based payment schemes in a dynamic setting. We explore different mechanism design ideas and discuss the goals of computational efficiency, budget balance, incentive compatibility, and strategy proofness, while addressing the computational limitations of a dynamic mechanism. We review the operation of a prototype and experimental platform called the Agent-Based Carpool (ABC) system. We describe how ABC generates rideshare plans from hundreds of real-life GPS traces, collected from a community of commuters over five years. The empirical results indicate significant reductions on number of commutes and on total cost of transportation, and show promise for generating efficiency by bringing self-interested agents together. In addition, they highlight challenges and tradeoffs that arise from the coordination of self-interested agents in real-life domains.

There has been growing interest in the use of the web and

*Ece Kamar performed this research during an internship at Microsoft Research.

computing methods in assisting with ridesharing. A number of online rideshare services[1] offer users varied experiences from simple to complicated. Nuride, Zimride, Craigslist and mailing groups provide social networks where users can manually arrange carpools. More sophisticated ridesharing services such as iCarpool and CarpoolWorld help users with online trip matching. These services also motivate users to carpool with mile-based rewards, or interfaces for negotiation on payments. The systems typically serve as platforms that bring users together, rather than as active mechanisms that generate rideshare plans and provide fair payments.

Altruism and reciprocity have been proposed as strategies to explain and promote cooperation among people when they interact repeatedly [Bowles and Gintis, 2005]. However, in dynamic domains we consider in this work, collaborative carpool plans may change with respect to changing preferences of users (e.g., trip start times, meeting schedules, cost of delay), and drivers may be paired with different set of passengers each day. Therefore, cooperation strategies such as altruism and reciprocity that require continuous interactions are not typically valid for dynamic domains such as ours.

Coalescing rational agents into groups of participants in rideshare plans is similar to the *initial-commitment decision problem* (ICDP) proposed by [Hunsberger and Grosz, 2000], as both problems aim to determine the set of tasks that agents need to commit in a collaboration. The methods employed in ABC system are an extension to prior work on ICDP as a payment mechanism is included, which provides a rationale and incentives for self-interested agents to collaborate.

Several previous studies on set-cover optimization problems focus on mechanism design for cost sharing [Devanur *et al.*, 2005; Li *et al.*, 2005]. The cost sharing problem focuses on dividing the cost of a service among self-interested agents in a fair manner, where the cost is independent of agents' preferences. The optimization used in ABC makes use of greedy optimization procedures similar to the approach taken in the earlier set-cover optimization efforts. However, the payment mechanisms employed in the past are not suitable for collaboration among self-interested agents. We don't have a distinction between service providers and receivers, and the cost is not independent of agents' preferences.

Mechanism design has been applied to the coordination of self-interested robots in sequential decision-making scenarios [Cavallo *et al.*, 2006]. Our work differs from the prior work in that both the joint plans and payments are based on combinations of dynamic and changing preferences of people about their daily habits including time, fuel, cognitive costs and travel preferences. We also present detailed analysis of the optimization and payment mechanisms with respect to the computational issues with real-life data in a dynamic domain.

2 Methodology and Architecture

Computing ideal ridesharing plans is a challenging problem as the solution must consider the varied and dynamically changing preferences of self-interested agents, must provide compelling and fair incentives, and must be easy to use. The ABC prototype addresses these challenges by creating personalized rideshare plans while minimizing the cumulative cost of transportation. The system has three main components that embody separate but interrelated reasoning methodologies: a user-modeling component that accesses and represents the preferences of agents, an optimization component that generates collaborative rideshare plans, and a payment component that provides incentives to agents to collaborate.

The user-modeling component is responsible for identifying the preferences of agents about their desired trips, and for passing the preferences into the optimization and payment components. It gathers information about agents' individual commute plans, including their origin, destination, timing of a trip, and preferences about a return trip. A destination analyzer accesses or infers the intended destination of a mobile user under uncertainty [Krumm and Horvitz, 2006]. To perform cost-benefit analysis of a ridesharing plan, the user-modeling component models agent-specific costs for driving, delaying a trip, diverting an ideal route to pick up or drop off other agents, and changing stop points. Capturing these costs in a dynamic manner is crucial for the success of the ridesharing system, as the system needs to adapt to different and changing preferences of agents. For example, an agent may be willing to wait and pick up other agents on the way when the cost of time is low, but not on a rainy day when the cost of time is high.

Time is an important resource and is one of the major factors influencing the cost of different commute plans. The user-modeling component employs a probabilistic time-cost model. The model considers as input the time of day, day of week, and sets of attributes about agents' commitments drawn from an online appointment book. Probabilistic models for the cost of time and for the commitment to attend events are learned from user annotated training data via a machine-learning procedure based on Bayesian structure search. Similar predictive models of the cost of time and meeting commitments have been used in other applications, including mobile opportunistic planning [Horvitz *et al.*, 2007; Kamar *et al.*, 2008], meeting coordination [Horvitz *et al.*, 2002] and the triaging and routing of communications [Horvitz *et al.*, 2005]. See [Horvitz *et al.*, 2005] for additional details about the machine learning and reasoning about the cost of time in different settings and the evaluated performance of the predictive models of the context-sensitive cost of time. For each agent, the user modeling component constructs a time-cost function T to estimate the cost of the time spent travelling between the start time (t_s) and end time (t_e) of a trip, and the additional cost for delaying the start time of a rideshare trip from the initial start time t_s^o to t_s. T is captured with respect to the nearest deadlines drawn from the agent's calendar. Given that the set of calendar items fall between $[t_s, t_e]$ is M, $m \in M$ is a calendar item, the start time of m is t_s^m, the end time of m is t_e^m, c_n is the minute time cost for travelling, c_m is the additional cost for missing a minute of m, c_d is the minute cost for delay; T is defined as,

$$\begin{aligned} T(t_s, t_e) = \quad & ((t_e - t_s) \times c_n) + (|t_s - t_s^o| \times c_d) \\ & + (\sum_{m \in M} (\min(t_e^m, t_e) - \max(t_s^m, t_s)) \times c_m) \end{aligned}$$

[1] www.nuride.com, www.carpoolworld.com, www.zimride.com, sfbay.craigslist.org/sfo/rid/, www.icarpool.com

3 Rideshare Optimization

The optimization component groups agents together and generates a collection of rideshare plans that maximizes the efficiency of transportation. The component acquires private user preferences from the user modeling component, combines them with global contexts to capture the collaborative value of a rideshare plan. The optimization component has the following properties that make it difficult for agents to find out about other agents in the system and thus collude in the mechanism; the component combines multiple user preferences and contextual factors to determine the best possible plan, agents do not get to know about other users' preferences or rideshare plans that they are not involved in. The optimization component takes in the set of individual desired commute plans as inputs and solves two difficult optimization problems to generate a collection of collaborative rideshare plans. The two optimizations are: (1) generating rideshare plans for groups of agents and (2) clustering agents into rideshare groups (see Figure 1).

3.1 Rideshare Plans

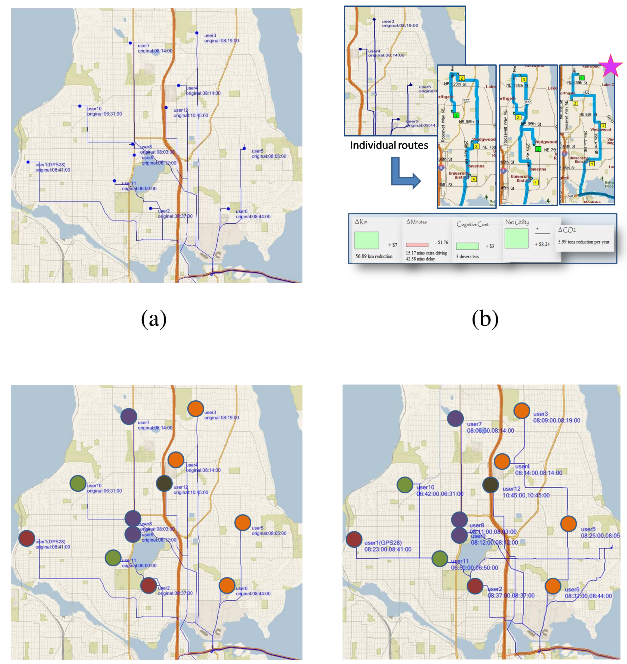
Choosing the best possible rideshare plan with respect to agent preferences is a large search problem where the system explores possible combinations of trip start times, stop orderings, stop locations, trip durations, and possible routes among stop points to generate a plan with highest possible cumulative value. Let P be the set of all agents in rideshare system, $S \subseteq P$ a rideshare group, $\mathcal{C}(\mathcal{S})$ the universe of all possible rideshare plans for S. A rideshare plan $C_i \in \mathcal{C}(\mathcal{S})$ is defined by the following attributes:

- $S = \{p_h, \ldots, p_q\}$, the set of agents of the rideshare group; $p_d \in S$, the assigned driver for the rideshare plan; $S_{-d} = S \setminus \{p_d\}$.
- $\mathcal{L}_{-d} = \{\ell_{h,s}, \ell_{h,e}, \ldots, \ell_{q,s}, \ell_{q,e}\}$, the set of start\end (stop) locations of agents in S_{-d}, where p_i's start location is $\ell_{i,s}$, the end location is $\ell_{i,e}$. For all $p_i \in S_{-d}$, $\ell_{i,s}$ and $\ell_{i,e}$ are located in a radius of $\ell^o_{i,s}$ and $\ell^o_{i,e}$ – the initial start\end locations for p_i's individual commute plan. $\mathcal{L}$, the complete set of start\end locations, is the combination of $\mathcal{L}_{-d}$ with the start\end locations of p_d: $\mathcal{L} = \mathcal{L}_{-d} \bigcup \{\ell_{d,s}, \ell_{d,e}\}$, where $\ell_{d,s} = \ell^o_{d,s}$, $\ell_{d,e} = \ell^o_{d,e}$.
- Θ_{-d}, the commute chain excluding p_d, is any ordering of $\mathcal{L}_{-d}$ such that for all $p_i \in S_{-d}$, $index(\ell_{i,s}) < index(\ell_{i,e})$ (i.e., any agent's start location precedes the end location in Θ_{-d}). $\Theta = \ell_{d,s} \circ Q_{-d} \circ \ell_{d,e}$ is the commute chain for S.
- t_s, the start time of the rideshare plan. $t(l)$, the scheduled time of stop location l is defined as below, where $\Delta t(\ell_j, \ell_{j+1})$ is the estimated travel duration between two consecutive stop locations $\ell_j, \ell_{j+1} \in \Theta$:

$$t(\ell) = \begin{cases} t_s & \ell = \ell_{d,s} \\ t_s + \sum\limits_{j<index(\ell)} \Delta t(\ell_j, \ell_{j+1}) & otherwise \end{cases}$$

3.2 Value of Shared Plans

Although reduction in fuel costs and personal or organizational (e.g., per the goals of an employer) goals of reducing CO_2 emissions from vehicles are the primary motivations for bringing self-interested agents to collaborate in rideshare plans, the additional time and travel required for adding new stops to a trip, or having fewer numbers of agents driving in heavy traffic can play an important role in the willingness of agents to participate. We define a personal inconvenience cost that captures several agent-specific cost factors. The personal inconvenience factors are composed to yield the cumulative value of a rideshare plan.



(a) (b) (c) (d)

Figure 1: ABC rideshare optimization. (a) Input of individual commute plans. The initial segments of the individual plans are drawn as blue lines originating from the start positions of agents indicated by blue dots. Each start position is labeled by the username and original start time. (b) Rideshare plan optimization. Individual plans acquired from four users are shown by top left. Middle right images display three different rideshare plans generated for users. The economical analysis of a rideshare plan in terms of fuel, time, cognitive costs and CO_2 emissions is illustrated on the bottom. (c) Rideshare group optimization. Users assigned to the same rideshare groups are labeled with the same color. (d) Rideshare plans generation. Shared transportation plans are drawn with blue lines. Each circle represents the rideshare group that a user is assigned to, and is labeled with the username, and with the updated and original start times.

A model for the cost of personal inconvenience combines the cost of a lengthening of the duration of the commute and of shifts in leaving and arrival times with gains in the savings of fuel and reduction in the cognitive costs of driving a vehicle, to yield an estimate of the net value of an agent becoming associated with a trip. The user modeling component provides the probabilistic time-cost function, $T_i(t_s, t_e)$. The fuel cost in dollars for one mile is represented as c_g. The inconvenience model combines the input from the user modeling component with traffic prediction services and public contexts (e.g., daily events that may affect the traffic) to construct a cognitive cost model for an agent. $CC_i(\ell_s, \ell_e)$ represents the predicted cognitive cost of p_i for driving between the given stops. The optimization engine makes calls to Microsoft Mappoint services to estimate the travel duration. $\Delta t(\ell_i, \ell_j)$ represents the duration of travel between stops ℓ_i and ℓ_j, whereas $\Delta d(\ell_i, \ell_j)$ represents the distance to be travelled between these stops.

The initial inconvenience cost of agent p_i, $PC^o(p_i)$, represents the cost for following the individual trip that would be created between initial start\end locations of p_i in the absence of ridesharing, where the start time of the individual trip is $t^o_{i,s}$.

$$\begin{aligned} PC^o(p_i) &= T_i(t^o_{i,s}, t^o_{i,e}) + \Delta d(\ell^o_{i,s}, \ell^o_{i,e}) \times c_g + \\ &\quad CC_i(\ell^o_{i,s}, \ell^o_{i,e}) \\ t^o_{i,e} &= t^o_{i,s} + \Delta t(\ell^o_{i,s}, \ell^o_{i,e}) \end{aligned}$$

A fuel and cognitive cost is incurred if an agent is assigned as the driver in a given trip. Let $\ell_j, \ell_{j+1} \in \mathcal{L}$ be consecutive stop locations in commute chain Θ, $PC(p_d, C)$ is the inconvenience cost of the driver for rideshare plan C.

$$\begin{aligned} &PC(p_d, C) = T_d(t(\ell_{d,s}), t(\ell_{d,e})) + \\ &\quad \sum_{\ell_j, \ell_{j+1}} (\Delta d(\ell_j, \ell_{j+1}) \times c_g + CC_d(\ell_j, \ell_{j+1})) \end{aligned}$$

The passengers of a rideshare are only subject to time costs for the period of time for travel between their scheduled start and end locations. $PC(p_i, C)$ is the inconvenience cost of passenger $p_i \in S_{-d}$ for rideshare plan C.

$$PC(p_i, C) = T_i(t(\ell_{i,s}), t(\ell_{i,e}))$$

$v_i(C)$ represents the value of agent p_i for rideshare plan C. The cumulative value of a rideshare plan, $V(C)$, represents the value of agents in rideshare plan C for switching to collaborative plan C from their individual plans.

$$\begin{aligned} v_i(C) &= PC^o(p_i) - PC(p_i, C) \\ V(C) &= \sum_{p_i \in S} v_i(C) \end{aligned}$$

Before leaving our discussion of preferences, we note that there are subtle, yet potentially powerful psychological and social costs and benefits associated with sharing rides with others in the open world. We see an opportunity to assess and smoothly integrate key psychosocial factors as additional costs into the optimization used for generating plans. For instance, participants can be offered the option of providing preference functions that yield estimates of the cost of traveling with one or more people based on an established reputation, and on social or organizational relationships. As examples, preferences can be captured with utility functions that specify the costs with including people in a shared plan that are related to the participant via different types of organizational links or via increasing graph distances in a social network. Such additional costs would likely influence individual objective functions, and thus the overall behavior of the system, leading to modifications in the rideshare plans generated as compared to the output system that does not consider the psychosocial issues.

3.3 Plan Optimization as Search

Rideshare plan optimization seeks to identify the shared transportation plan for a group of agents S with the highest cumulative value. This optimization problem is a search problem over the universe of rideshare plans $\mathcal{C}(\mathcal{S})$ available for S, where the search dimensions of $\mathcal{C}(\mathcal{S})$ are the set of possible commute chains, set of possible stop locations for the passengers, trip start times, and potential routings between stop points. The optimization component performs geospatial search over the feasible paths that satisfy the constraints of a rideshare plan for S. Given the start\end locations of the assigned driver, the optimizer considers sets of updated routes by adding potential passenger stop points as waypoints and performing A^* search. The set of potential passenger stop points are selected from a radius around the initial stop points of the passenger. The magnitude of the radius is limited by the maximum distance the passenger is willing to diverge from the initial stop location to have a more efficient rideshare. The engine searches for the start time of the rideshare plan that minimizes the total cost.

The plan optimizer selects the plan $C^*(S)$ that offers the maximum cumulative value to agent set S, among all possible plans $\mathcal{C}(\mathcal{S})$. It provides $C^*(S)$ to the rideshare group optimizer.

$$C^*(S) = \arg\max_{C_j \in \mathcal{C}(\mathcal{S})} V(C_j)$$

3.4 Group Assignment as Set Cover

Given a set of agents P in the rideshare system, the rideshare group optimization finds the set of subset of P that covers all agents in P by offering the highest cumulative value. Thus, this optimization is identical to the well-known NP-hard set-cover problem.

Let us consider a set of agents, $P = \{p_1, \ldots, p_n\}$ willing to collaborate in a rideshare system. k is the capacity of a single vehicle, thus the maximum size of a collaborative rideshare group. A set cover for $SC_i = \{S_h, \ldots, S_m\}$ for agent set P is a set of subsets of P, such that for all subsets S_j; $|S_j| \leq k$, $\bigcup_{S_j \in SC_i} S_j = P$, and for any $S_j, S_k \in SC_i$ $S_j \bigcap S_k = \emptyset$. Thus, a set cover SC_i in rideshare system represents a collection of rideshare groups, and their best possible rideshare plans that cover all agents in the ridesharing system without exceeding the capacity of a transportation vehicle. $\mathcal{SC}(\mathcal{P}) = \{SC_1, \ldots, SC_r\}$ is defined to be the universe of all set covers for set of agents P.

We define a valuation function $V(S_j)$, which corresponds to the value generated by the best possible rideshare plan for bringing agents S_j together. The value of a set cover SC_i, which is also a collective rideshare plan for P is calculated as given below:

$$
\begin{aligned}
V(S_j) &= \begin{cases} 0 & |S_j| \leq 1 \\ V(C^*(S_j)) & otherwise \end{cases} \\
V(SC_i) &= \sum_{S_j \in SC_i} V(S_j)
\end{aligned}
$$

A set-cover solver returns the optimal set cover $SC^* = \arg\max_{SC_i \in \mathcal{SC}(\mathcal{P})} V(SC_i)$. The dynamic, open-world nature of the domain requires the optimization to run efficiently because agents may unexpectedly arrive, leave, or change preferences which may result in running the optimization multiple times. However, optimal set cover solver takes exponential time in practice. Additionally, the optimization calls expensive online traffic prediction and routing services to evaluate the value of each set cover which makes the optimization more expensive. Thus, the optimal set-cover solver is infeasible to apply in open-world settings, and we implement an approximate, greedy set-cover algorithm to generate the rideshare groups [Li *et al.*, 2005].

The rideshare optimization system ensures that no rideshare group is worse off by engaging in the process. The rideshare group generator includes single-item subsets as well as rideshare groups in the set-cover optimization, thus selects individual (initial) trips for some of the agents rather than assigning them into carpools should no beneficial rideshare plan be available. Thus, any rideshare group generated by the optimizers offers non-negative cumulative utility to the agents. However, ensuring non-negative utility does not guarantee individual rationality or fairness between agents in the rideshare system. The system may incur additional costs to the assigned driver for a group while generating benefit for the other passengers. In the next section, we investigate payment mechanisms that can fairly divide the collaborative benefit generated by the rideshare optimization component among participants.

4 Mechanism Design for Collaboration

The payment mechanism is a crucial component of ABC's operations as it promotes collaboration among people, and directly influences the user behavior and the efficiency of the system. Sharing fuel costs among passengers is a simple but widely used payment mechanism in ridesharing. However this simple payment scheme is not suitable for a personalized ridesharing system, because it does not consider varying user costs in payment calculation. Using such a payment scheme in ABC would make the system vulnerable to deceptive reporting of needs by individual agents with the goal of biasing carpool plans to satisfy their preferences.

Designing the payment component of a dynamic and personalized ridesharing system is a challenging problem. As stated by the impossibility theorem, no exchange mechanism can be efficient, budget balanced and individually rational [Myerson and Satterthwaite, 1981]. Moreover, computationally expensive payment calculations may not be feasible for a dynamic system. In this work, we focus on VCG-based payments as they promote truthful behavior and individual rationality, and adapt to the changing preferences of users, in contrast to simple payment methods such as basic cost sharing. We shall present our initial VCG-based payment mechanism, and then explore the tradeoffs with applying the mechanism within the ABC prototype in terms of efficiency, computational complexity, budget balance and individual rationality.

4.1 VCG Payments for ABC

ABC's payment mechanism distributes VCG-based payments to promote truthful behavior, to ensure fairness and the ultimate sustainability of the system, while maximizing cumulative value of the collaboration [Vickrey, 1961; Groves, 1973; Clarke, 1971]. ρ_i, agent p_i's VCG payment to the system, is calculated as below, given that V^*_{-i} is the collaborative value of the collection of rideshare plans SC^* to all agents except p_i, $(V_{-i})^*$ is the value of the collection of rideshare plans generated when p_i is excluded from the ABC system:

$$\rho_i = (V_{-i})^* - V^*_{-i}$$

If the rideshare policy calculated by the optimization component is optimal, the VCG payment mechanism is efficient–its output maximizes social value, is individual rational–all agents have positive utility by participating, and strategy proof–truth-telling is a dominant strategy.

The VCG payment component does not burden people by inquiring about the utility of each potential rideshare assignment. Instead, valuations are generated by the system based on acquired preferences.

4.2 Tradeoffs on VCG Based Payments

Pursuing the use of VCG payments to ridesharing optimization immediately faces several challenges. First, the VCG payment mechanism is not budget balanced, and may return a loss. Secondly, calculating VCG payments in a dynamic mechanism is computationally expensive. Third, VCG mechanisms require the computation of optimal outcomes to ensure truthfulness. The ABC system calculates VCG-based payments based on an approximate optimization of rideshare assignments and routes. Thus, agents are not necessarily incented to be truthful [Nisan and Ronen, 2007].

We modify the VCG payment scheme to adapt it to the dynamic requirements of the open-world ridesharing problem. To simplify the analysis, we make the assumption that removing one agent from a carpool group does not affect the rideshare allocation of agents outside of that group. We calculate local VCG-based payments, which computes the VCG payment of agent p_i only among the agents that share the same carpool as p_i. This assumption makes payment calculations significantly more efficient, as carpool optimizations for payment calculations are done over a small subset of all agents. Calculating VCG payment locally offers an alternative for efficient calculation of VCG-based payments, by pointing out an important tradeoff for implementing expensive payments efficiently. However, the assumption we make to calculate VCG payments locally is not fundamental, does not affect the collaborative rideshare plans, and can be ignored if sufficient computational power is provided to compute payments globally.

We tested the local VCG-based payment scheme on a large dataset of GPS trails that we describe in more detail in Section 5. The experimental results show that value distribution with local payments maintains 99.7% to 100% of individual-rationality among agents with varying fuel and time costs. However, the evaluation highlights the prospect of incurring a deficit with VCG-based payments. In our study, we found that the system pays drivers more than it collects from the passengers. To sustain the carpooling system with local VCG-based payments, the system needs to distribute 55% to 79% of the cumulative value generated with carpools back to agents as payments. The deficit of the system grows proportional to the average time costs of the agents, as it gets harder to bring self-interested agents together when time cost is high.

Given the challenge with balancing the budget, we experimented with an alternate VCG-centric scheme, based on previous work proposing a threshold-based mechanism that enforces budget-balance as a hard constraint on payment calculation [Parkes *et al.*, 2001]. We modify the local VCG-based payment scheme with the threshold rule specified by Parkes, et al. to eliminate deficit where V^* is the cumulative value of rideshare plans. $\Delta_{vick,i}$ represents the non-negative portion of VCG payments which is called Vickery discount.

$$\Delta_{vick,i} = V^* - (V_{-i})^*$$

For some parameter $C \geq 0$, we define threshold discounts $\Delta^t_{vick,i}$, and redefine payments ρ^t_i based on $\Delta^t_{vick,i}$. The threshold parameter C is calculated with linear programming based on local VCG-based payments and $\Delta_{vick,i}$ values.

$$\begin{aligned} \Delta^t_{vick,i} &= max(0, \Delta_{vick,i} - C) \\ \rho^t_i &= v_i(SC^*) - \Delta^t_{vick,i} \end{aligned}$$

Studies with the real-world commute dataset using the local VCG-based payments with the threshold rule demonstrate that the revised mechanism was able to eliminate the deficit for a range of time and fuel cost values. The mechanism does not negatively influence the individual rationality nor the efficiency of the ABC system.

With threshold-based payments and suboptimal outcomes, our mechanism is not guaranteed to be truthful. Investigating the effect of using the local payments and threshold on the truthfulness of agents will require a deeper analysis on the system. Parkes et al., states that the threshold-based payment scheme has better incentive properties than other rules proposed in their work. Our threshold-based local VCG payments promote truthful behavior as an agent's payment does not directly depend on its preference revelation. We believe that the payment scheme is hard to manipulate by bounded-rational agents given the incomplete information available to agents about other agents and the indirect affect of an agent's preferences on outcomes.

5 Real-World Trip Dataset and Studies

The ABC prototype provides options for offline, batch optimizations as well as for real-time simulations of incoming ride requests based on the dynamic queuing of travel needs and preferences. Statistics are maintained on multiple dimensions of cost and savings for gaining insights into the operation and sensitivity of the plan generation to different workloads and assumptions. The system also provides visualizations of routes and route plans on a city map.

We ran studies based on the driving trip data gathered from 215 subjects over a five-year period [Krumm and Horvitz, 2005]. These subjects included Microsoft employees and spouses who volunteered to place GPS receivers with logging in their cars over several weeks in return for participating in a random drawing for a prize. Nearly all the subjects live in the Seattle, WA USA area. The GPS receivers were programmed to record GPS data only when the users are in motion. The dataset contains a total of 1,434,308 (latitude, longitude) points for an average of 6,671 points per participant.

As the initial goal of this research is to generate carpool plans for daily commutes of users, we segmented the dataset into discrete trips. We identified any two consecutive GPS points that are either 5 minutes or more than 7 kilometers apart as two separate trips. The trips that are shorter than a threshold are eliminated, which resulted in 7,377 individual trips. For each user, we selected a pair of morning and evening trips that appear to capture daily commute patterns of the users by having the following properties: (1) the regularity of the commutes on trip data of the user, (2) minimum divergence of the selected commutes from a round trip. 215 morning\evening commute patterns were extracted with an average duration of 26 mins for morning, 29 mins for evening, and average distance of 21km for morning and 24 km for the evening.

We tested the ABC prototype on commute patterns extracted from the trip dataset. The results of the rideshare system are evaluated in terms of the *efficiency on number of commutes* (i.e., the reduction ratio on total number of commutes), *efficiency on cost* (i.e., the reduction ratio on total cost), and the reduction on CO_2 emissions. We explored the sensitivity of the analyses to variations in the fuel costs (i.e., from \$0.035/mile to \$0.14/mile) and the average costs of time (i.e., from \$0/hour to \$9.6/hour).

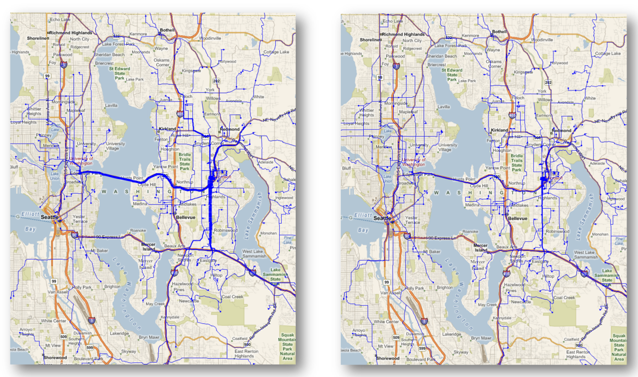


Figure 2: Seattle area map displaying the commute routes of study participants (thicker blue lines represent more crowded routes) Left: Morning trips without ABC system, Right: Morning trips with ABC system. The ridesharing system reduces the number of cars in the traffic significantly.

Figure 2 compares the individual commute plans with the collection of rideshare plans generated by the system. The thinner blue color on main highways indicates the positive effect of ridesharing on the morning commute traffic in the Seattle region. When the fuel cost is set to $0.07/mile[2], and average time cost is set to $4.8/hour, ABC system is able to achieve 41% efficiency on number of commutes, 14% efficiency on total cost of transportation which results in 84.16 tons of CO_2 reduction per year.

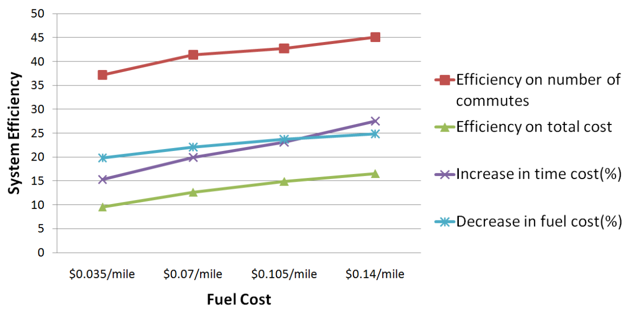


Figure 3: Effect of fuel cost on the efficiency of ABC system. Increasing fuel costs improves the efficiency of ridesharing.

To investigate the influence of the cost of fuel on the efficiency of the rideshare optimization, we tested ABC over a range of fuel costs. As shown in Figure 3, the efficiency of the carpooling system on both the number of commutes and the total cost improves significantly with increases in the cost of fuel. These results indicate that increasing fuel costs can provide higher incentives for agents to collaborate, and we expect the willingness of agents to carpool to grow as fuel costs increase. The reduction on CO_2 emissions increases 25% as fuel costs increases from 0.035/mile to $0.14/mile.

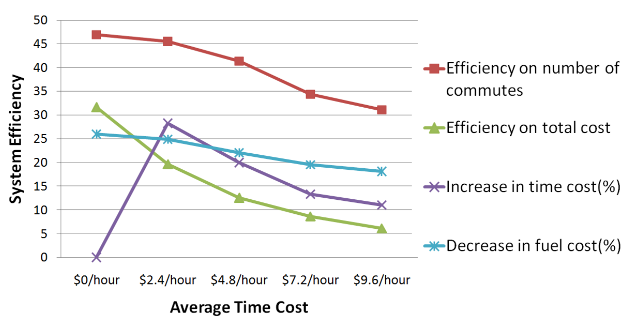


Figure 4: Influence of the average time cost on the efficiency of ABC planning. Increasing time costs decreases the efficiency of ridesharing.

We investigated the influence of changes in the cost of time on the efficiency of the rideshare system by varying the average time costs of users as shown in Figure 4. As the costs of time increase, the efficiency of the optimization with regard to the number of commutes and total costs incurred drops significantly. The reduction on CO_2 emissions decreases 29.6%. Increasing time costs reduces the incentive of agents to collaborate in ridesharing.

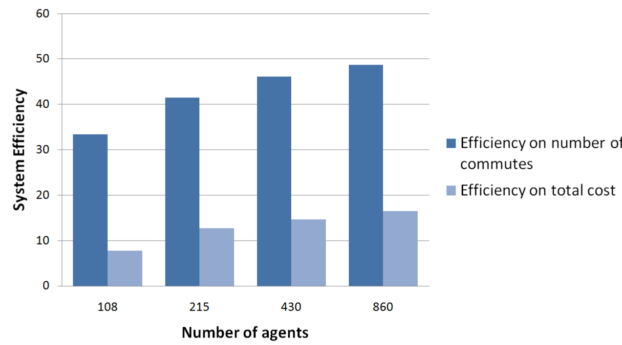


Figure 5: Effect of the size of agent set on the efficiency of ABC planning. The efficiency of ridesharing improves with the number of users in the system.

We sought to understand how the density of actors might change the behavior and overall savings generated by the system. To simulate the effect of increasing the number of agents in the system, we populated commute patterns with randomly created artificial commute patterns. The synthetic commuting requests are generated by pairing randomly selected start/end points from the trips dataset, with trip start times taken from a Gaussian distribution representing the start times of the commute patterns in the data. As displayed in Figure 5, the efficiency of the system grows as the logarithm of the number of the agents in the system. With more agents, the system is more likely to find better matches for the users. Thus, we can expect that the performance of the ridesharing system will improve with increasing numbers of users.

6 Summary and Conclusions

We reviewed research on reasoning and optimization for generating shared transportation plans in a real-world setting. We explored the problem as an agent collaboration challenge and developed extensions to prior work on coordination among multiple agents and market-based incentives to solve key challenges. We constructed a prototype and explored the performance of the system with a dataset of real-world trips collected over five years. Our studies included sensitivity analyses that enable us to explore the influence on the behavior of the system of changing such variables as the costs of fuel and number of participants. In ongoing work, we are investigating new applications of the mechanisms and ideas, including within and beyond the transportation domain. Within the transportation domain, we are exploring the use of the methods to inform such decisions as to where to locate park and ride facilities and additional practical issues with the deployment of a dynamic version of the system in a running online service that serves an organization or a larger city region [3].

[2] $0.07/mile is stated to be the per mile cost of driving by http://www.commutesolutions.org/calc.htm.

[3] More detailed presentation of the ideas investigated in this work and the ongoing work on dynamic optimization, park-and-ride cen-

Other directions include assessing psychosocial factors and investigating how including these considerations in the larger utility function influences the rideshare plans generated. We believe that the ideas presented on generating shared plans for self-interested agents extend to numerous real-life domains where individual preferences and goals must be considered in the generation of valuable collaborations. We hope that the challenges highlighted in our pursuit of methods for promoting collaboration in real-world settings will motivate further research. Turning to the domain, we believe that there are opportunities for analogous applications of learning, inference, optimization, and market mechanisms to address difficult challenges with the environment and economy.

References

[Bowles and Gintis, 2005] S. Bowles and H. Gintis. Can Self-interest Explain Cooperation? *Evolutionary and Institutional Economics Review*, 2(1):21–41, 2005.

[Cavallo *et al.*, 2006] R. Cavallo, D.C. Parkes, and S. Singh. Optimal Coordination of Loosely-Coupled Self-Interested Robots. *AAAI*, 2006.

[Clarke, 1971] E.H. Clarke. Multipart pricing of public goods. *Public Choice*, 1971.

[Devanur *et al.*, 2005] N.R. Devanur, M. Mihail, and V.V. Vazirani. Strategyproof cost-sharing mechanisms for set cover and facility location games. *Decision Support Systems*, 2005.

[e Sustainability Initiative, 2008] Global e Sustainability Initiative. SMART 2020: Enabling the low carbon economy in the information age. 2008.

[Groves, 1973] T. Groves. Incentives in Teams. *Econometrica*, 1973.

[Horvitz *et al.*, 2002] E. Horvitz, P. Koch, C.M. Kadie, and A. Jacobs. Coordinate: Probabilistic forecasting of presence and availability. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI*, 2002.

[Horvitz *et al.*, 2005] E. Horvitz, P. Koch, R. Sarin, J. Apacible, and M. Subramani. Bayesphone: Precomputation of Context-Sensitive Policies for Inquiry and Action in Mobile Devices. *UM*, 2005.

[Horvitz *et al.*, 2007] E. Horvitz, P. Koch, and M. Subramani. Mobile Opportunistic Planning: Methods and Models. *UM*, 2007.

[Hunsberger and Grosz, 2000] L. Hunsberger and B.J. Grosz. A combinatorial auction for collaborative planning. In *ICMAS*, 2000.

[Kamar and Horvitz, 2009] E. Kamar and E. Horvitz. Generating Shared Transportation Plans Under Varying Preferences: Ridesharing Models and Mechanisms. Technical report, MSR-TR-2009-2011, Microsoft Research, 2009.

ter placement, and admitting future methods of transportation, such as shared vehicles, are presented in [Kamar and Horvitz, 2009].

[Kamar *et al.*, 2008] E. Kamar, E. Horvitz, and C. Meek. Mobile opportunistic commerce: mechanisms, architecture, and application. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pages 1087–1094, 2008.

[Krumm and Horvitz, 2005] J. Krumm and E. Horvitz. The Microsoft Multiperson Location Survey. Technical report, MSR-TR-2005-103, Microsoft Research, 2005.

[Krumm and Horvitz, 2006] J. Krumm and E. Horvitz. Predestination: Inferring Destinations from Partial Trajectories. *Ubicomp*, 2006.

[Li *et al.*, 2005] X. Li, Z. Sun, and W. Wang. Cost sharing and strategyproof mechanisms for set cover games. In *STACKS*, 2005.

[Myerson and Satterthwaite, 1981] RB Myerson and MA Satterthwaite. Efficient Mechanisms for Bilateral Trading. *Journal of Economic Theory*, 29(2), 1981.

[Nisan and Ronen, 2007] N. Nisan and A. Ronen. Computationally Feasible VCG Mechanisms. *JAIR*, 2007.

[Parkes *et al.*, 2001] D.C. Parkes, J. Kalagnanam, and M. Eso. Achieving Budget-Balance with Vickrey-Based Payment Schemes in Exchanges. In *IJCAI*, 2001.

[Vickrey, 1961] W. Vickrey. Counterspeculation, Auctions, and Competitive Sealed Tenders. *The Journal of Finance*, 1961.