Intersection Management With Constraint-Based Reservation Systems

Tsz-Chiu Au[1], Shun Zhang[2], and Peter Stone[3]

[1] School of Electrical and Computer Eng.
Ulsan National Institute of Sci. and Tech.
Ulsan, South Korea 689-798
chiu@unist.ac.kr
[2] Dept. of Computer Science
The Univ. of Texas at Austin
Austin, TX 78712 USA
jensen.zhang@utexas.edu
[3] Dept. of Computer Science
The Univ. of Texas at Austin
Austin, TX 78712 USA
pstone@cs.utexas.edu

Abstract. Autonomous Intersection Management (AIM) is a reservation-based intersection control protocol that leverages the capacities of autonomous vehicles to dramatically reduce traffic delay at intersections. AIM was designed for the time when all, or most, of the vehicles on the road are *fully* autonomous. However, we anticipate that there will be a long transition period during which many cars are still driven by human drivers and/or most vehicles have some but not all capabilities of fully autonomous vehicles. In order to accommodate this transition, this paper introduces a new multiagent protocol called *Semi-Autonomous Intersection Management* (SemiAIM), which allows vehicles with partially-autonomous features such as adaptive cruise control to make reservations in AIM. We propose a method for vehicles with limited autonomy to make reservations to enter an intersection in an AIM-like style and conduct extensive experiments in simulation to evaluate its effectiveness. Our results show that the delay of semi-autonomous vehicles in SemiAIM can be greatly reduced compared to human-driven vehicles.

Keywords: Autonomous vehicles, multiagent systems, coordination

1 Introduction

Recent robotic car competitions and demonstrations have convincingly shown that autonomous vehicles are feasible with the current generation of hardware [4]. Looking ahead to the time when autonomous cars will be common, Dresner and Stone proposed a new intersection control protocol called *Autonomous Intersection Management* (AIM) and showed that by leveraging the control and network capabilities of autonomous vehicles it is possible to design an intersection control protocol that is much more efficient than traffic signals [6]. By removing human factors from control loops, autonomous vehicles, with the help of advanced sensing devices, can be safer and more reliable than human drivers. The AIM protocol exploits the fine control of autonomous vehicles to allow more vehicles simultaneously to cross an intersection, thus effectively reducing the delay of vehicles by orders of magnitude compared to traffic signals [7].

AIM is designed for the time when vehicles are autonomous. We, however, anticipate that there will be a long transition period during which most vehicles have some but not all capabilities of fully autonomous vehicles. In fact, this transition period has already begun. Since the late 1990s, adaptive cruise control systems and lane departure warning systems have become widely available as optional equipment on luxury production vehicles. Today's automatic parking systems such as those in the Toyota Prius and various BMW models can perform parallel parking with little or no human intervention. While AIM provides a significant efficiency improvement at intersections when all cars are autonomous, the benefits are minimal even when as few as 10% of the vehicles are driven by humans (Figure 16 in [6]). The requirement that most, if not all, vehicles are fully autonomous is a key obstacle to the adoption of AIM-like intersection control when most vehicles are not fully autonomous.

The National Highway Traffic Safety Administration acknowledges that fully autonomous vehicles represent just the top level in five levels of vehicle automation [11]. Indeed, they define a level below this top level with vehicles that have limited self-driving automation. The main motivation of this paper is to propose a new intersection control system called *Semi-Autonomous Intersection Management (SemiAIM)* that can accomodate both fully autonomous vehicles and *semi-autonomous* vehicles with limited self-driving automation. There is a high likelihood that human-driven vehicles, semi-autonomous vehicles, and fully autonomous vehicles will *coexist* on the road in the future. SemiAIM takes advantages of this trend and allows autonomous intersections to handle a traffic mixture with different types of vehicles.

The main contributions of this paper are 1) the introduction of the concept of SemiAIM; 2) a full specification of the reservation requests for SemiAIM; and 3) detailed empirical results demonstrating the effectiveness of this protocol, especially in moderate traffic levels with a mix of human-driven, semi-autonomous, and fully autonomous vehicles. The remainder of the paper is organized as follows. Section 2 outlines the architecture of AIM which forms the basis of SemiAIM. Section 3 categorizes semi-autonomous vehicles that work with SemiAIM. Section 4 discusses the design of the user interface for drivers on semi-autonomous vehicles to interact with SemiAIM. Section 5 describes the constraint-based reservation system, a generalization of AIM for semi-autonomous vehicles. Section 6 presents the results of the simulation experiments we used to evaluates SemiAIM. The related work and the conclusion are given in Section 7 and 8, respectively.

2 Autonomous Intersection Management

The AIM protocol is based on a *reservation* paradigm, in which vehicles "call ahead" to reserve space-time in the intersection [6]. The system assumes that computer programs called *driver agents* control the vehicles, while an arbiter agent called an *intersection manager* (IM) is placed at each intersection. The *driver agents* attempt to reserve a block of space-time in the intersection. The intersection manager decides whether to grant or reject requested reservations according to an *intersection control policy*. In brief, the paradigm proceeds as follows.

- An approaching vehicle announces its impending arrival to the IM. The vehicle indicates its predicted arrival time, velocity, acceleration, and arrival and departure lanes.
- The IM simulates the vehicle's path through the intersection, checking for conflicts with the paths of any previously processed vehicles.
- If there are no conflicts, the IM issues a reservation. It then becomes the vehicle's responsibility to arrive at, and travel through, the intersection as specified.
- The car may only enter the intersection once it has successfully obtained a reservation.

The prototype intersection control policy called *First Come, First Served* (FCFS) operates by dividing the intersection into a grid of *reservation tiles*. When a vehicle approaches the intersection, the intersection manager uses the data in the reservation request regarding the time and velocity of arrival, vehicle size, etc. to simulate the intended journey across the intersection. At each simulated time step, the policy determines which reservation tiles will be occupied by the vehicle. If the vehicle's space-time request has no conflict, the reservation is successful; otherwise, the request will be rejected.

Empirical results in simulation demonstrated that the reservation system with FCFS can dramatically improve the intersection efficiency when compared to traditional intersection control mechanisms [6]. Overall, by allowing for much finer-grained coordination, the simulation-based reservation system can dramatically reduce per-car delay by two orders of magnitude in comparison to traffic signals and stop signs. This reduction of delays can translate into less traffic congestion [2, 12], which in turn leads to better fuel efficiency and lower emissions.

A drawback of FCFS is that it is designed for fully autonomous vehicles only—it does not work with any vehicles that require human operations. Dresner and Stone did propose a hybrid protocol called FCFS-Signal (a.k.a. FCFS-Light) which allows human-driven vehicles to share the road with autonomous vehicles at intersections [5]. However, their approach has two major drawbacks. First, the benefits of FCFS-Signal over conventional traffic signals are relatively small until 90-95% of the vehicles on the road are autonomous. Second, and most important for this paper, FCFS-Signal cannot take advantage of the limited autonomous capabilities of semi-autonomous vehicles. There are useful features that human-driven vehicles could have, that AIM managers have overlooked. To remedy these issues, this paper introduces a new autonomous intersection system called *SemiAIM* that works not only with autonomous vehicles but also with human-driven vehicles and semi-autonomous vehicles.

3 Semi-Autonomous Vehicles

Dresner and Stone proposed the FCFS-signal policy which makes AIM compatible with human drivers [5]. According to their study, when more than 5-10% of traffic is comprised of human-driven vehicles, the average delay time of all vehicles increases significantly. The reason for this phenomenon is twofold. First, human-driven vehicles cannot

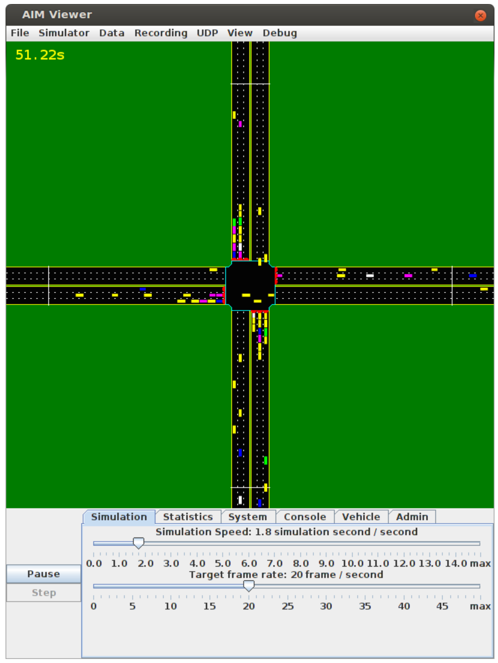

Fig. 1. A screenshot of the simulator we developed for the experiments on SemiAIM.

enter the intersection during red signal phases, because they are *not allowed* to make any reservation. Second, as a consequence of the first, autonomous vehicles often fail to obtain reservations when they are behind the human-driven vehicles stalled at red signals. These issues stem from the fact that FCFS-signal only allows fully autonomous vehicles to enter intersections during red signal phases. SemiAIM sets out to overcome these issues by allowing *semi-autonomous vehicles* to use the reservation system to enter an intersection during red signal phases.

In this paper, we use the term *semi-autonomous vehicles* to refer to vehicles with limited autonomous driving and wireless communication capabilities. While these vehicles are not fully autonomous, they are assumed to be able to follow a *limited* number of predictable trajectories at intersections more precisely than human drivers. This ability allows them to utilize our constraint-based reservation system to make reservations in the same manner as fully autonomous vehicles.

Our proposed reservation system is general enough to accept reservation requests from *any* semi-autonomous vehicles that are capable of following some trajectories and communicating with the IM. To facilitate our discussion, we will focus on semi-autonomous vehicles which use the following set of equipment that is readily available today.

- **Communication device (Com)**: a component in a vehicle's on-board electronic system that enables the vehicle to wirelessly communicate with the transportation infrastructure including the IM. The communication is bidirectional: the messages sent

Table 1. Features of semi-autonomous vehicles.

Vehicle Type	Communication Device	Cruise Control	Adaptive Cruise Control
SA-ACC	X	X	X
SA-CC	X	X	
SA-Com	X		

from the IM is presented to the human driver on the LCD screen of an on-board navigation system or on a smartphone, and the human driver makes decisions on the user interface of the device. The device is also hooked up with the odometer, GPS, and other sensors such that it can send these sensing information along with the request messages to the IM.

- **Simple Cruise control (CC)**: An optional speed control subsystem in vehicles' drivetrain that automatically controls the vehicle speed by taking over the throttle of the vehicles. With the help of cruise control systems, vehicles can maintain a steady constant velocity more precisely than human drivers can manually.
- **Adaptive cruise control (ACC)**: an advanced cruise control system that automatically adjusts the speed of a vehicle in order to maintain a certain distance from vehicles ahead. To achieve this car-following maneuver, ACC uses on-board distance sensors coupled with cruise control in a feedback loop.

All of this equipment gives semi-autonomous vehicles *some* of the functionality of autonomous vehicles, though human drivers still retain some control of the vehicles. We can equip a semi-autonomous vehicle with more than one of these devices. Next, we introduce three types of semi-autonomous vehicles that we envision utilizing this equipment.

- **Type SA-ACC Vehicles**: Utilizing adaptive cruise control to enter an intersection by either moving straight through the intersection or following another vehicle.
- **Type SA-CC Vehicles**: Using simple cruise control only to enter an intersection at a constant velocity in a straight line.
- **Type SA-Com Vehicles**: Reserve an entire lane in an intersection such that the human driver can get through the intersection without the help of any autonomous control device; thus only the communication device is needed.

Table 1 summarizes the equipment being used on different types of these semi-autonomous vehicles. As can be seen, a vehicle can utilize several devices at the same time. In general, having additional equipment provides additional potential advantages over manually driven cars. Since adaptive cruise control can work as simple cruise control, Type SA-ACC vehicles subsume Type SA-CC vehicles, which in turn subsume Type SA-Com vehicles. Of course, all these vehicles can be manual-driven, because the human driver can always opt to simply follow the traffic signals without using cruise control, especially when engaging the equipment is too distracting.

Figure 1 shows a screenshot of the SemiAIM simulator we developed to simulate semi-autonomous vehicles at intersections.[4]. In this figure, the yellow vehicles are fully

[4] Modified from the open-source AIM4 simulator: `http://www.cs.utexas.edu/~aim`

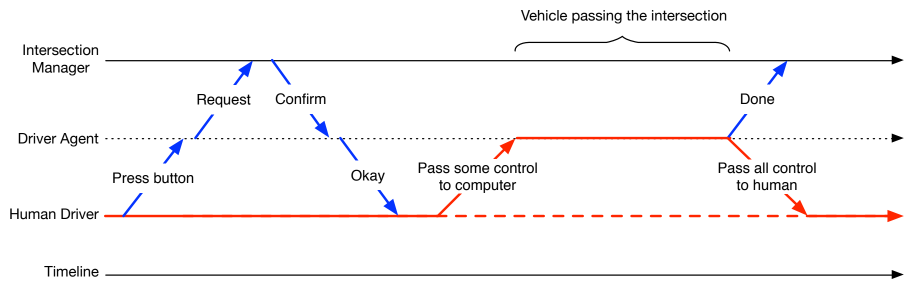

Fig. 2. The interaction between human drivers, driver agents, and the IM. The blue lines are message passing, and the red lines are transfer of control. Note that human drivers retain some control of the vehicle inside the intersection (the dashed red line).

autonomous, the green vehicles are Type SA-ACC, the blue vehicles are Type SA-CC, the white vehicles are Type SA-Com, and the magenta vehicles are human-driven vehicles. Although the traffic signals at the intersection are red (see the red dots at the edges of the intersection), fully autonomous vehicles can still enter the intersection. Actually, all vehicles except the human-driven vehicles are eligible to enter the intersection as long as they get reservations from the IM.

4 Interaction Model

Safety is a main concern when involving human drivers in the control loop of semi-autonomous vehicles. For the semi-autonomous vehicles defined above to be able to go through an intersection safely, we need to define a simple and clean interface for negotiating with the IM and passing control between the human driver and the driver agent. In this section, we describe how our proposed protocol can be realized safely by having an *interaction model* between human drivers and driver agents that simplifies the task of the human drivers.

Figure 2 summarizes the interaction model between human drivers, driver agents, and the IM. We require the inclusion in the vehicle of a single button that signals the driver agent to ask for a reservation. After pressing the button, the driver agent will automatically send a request message to the IM on behalf of the human driver. It is also important that there is a clear "Okay" indicator (such as a green light) installed in the car which indicates when the request has been confirmed. After seeing the okay signal, the driver would have to actively pass control to the driver agent, again by pressing a single button. This way the driver will not be surprised by any sudden autonomous actions of the vehicle.

Keep in mind that the only thing a human driver needs to do is to first press a single button to request a reservation and then press a single button to accept it (possibly the same button). All of the negotiation involving arrival times and trajectories is completed automatically by the driver agent and IM in a manner similar to how it is done in AIM.

Moreover, before entering the intersection the human driver can choose to *opt out* without entering the intersection. All he or she needs to do is to touch the brakes and stop before the intersection. [5] The human driver will regain control, and then have to drive as done today and respect the traffic signals even if they have gotten a confirmation from the IM.

After passing control to the driver agent, the human driver still needs to maintain limited control of the vehicle because the vehicle is not fully autonomous. The question of how to achieve this human-computer cooperative driving depends on the type of semi-autonomous vehicle as well as the type of intersection. In case of Type SA-ACC vehicles, the human driver must keep following the vehicle ahead by steering the vehicle in the correct direction. At intersections where the vehicle ahead must make a sharp turn, the IM would deny the reservation to the vehicle. In the case of SA-CC vehicles, the human driver simply holds the steering wheel to make the vehicle goes straight through the intersection. In Type SA-Com vehicles, no control is passed to the driver agent, and the IM will reserve the entire lane for the vehicle for a period of time. The human driver has to make sure that the vehicle goes through the intersection within that period of time, perhaps with the help of a virtual traffic signal on the computer screen.

This interaction model only requires the human driver to perform relatively simple driving maneuvers such as holding the steering wheel at a certain angle (for Types SA-ACC and SA-CC vehicles) or driving as if under a traffic signal (for Type SA-Com vehicles). These tasks are much simpler than other maneuvers such as lane changing and passing other vehicles, and thus should not be taxing to experienced human drivers.

5 Constraint-based Reservation Systems

SemiAIM extends AIM by allowing human-driven vehicles and semi-autonomous vehicles to make reservations in the same way as fully autonomous vehicles. The key idea of SemiAIM is to turn AIM into a *constraint-based reservation system*, which allows vehicles to make reservations in terms of constraints over 1) their driving profiles such as their arrival time and arrival velocity, and 2) the relationships with other vehicles.

In AIM, a reservation request is a 5-tuple $\langle l_1, l_2, t_0, v_0, p \rangle$, where l_1 is the entry lane, l_2 is the exit lane, t_0 is the arrival time, v_0 is the arrival velocity, and p is the physical characteristics of the vehicle. This information allows the IM to compute the exact trajectory of the vehicle and reserve tiles for the vehicle on the trajectory. However, this computation assumes the vehicle can be controlled *precisely* in the intersection so that it can meet the reservation constraints exactly. Human drivers cannot control their vehicles as precisely, and semi-autonomous vehicles may only be able to control certain aspects of their trajectories. Therefore, we need a new kind of reservation requests that do not rely on this assumption.

[5] More precisely, if a human driver chooses to opt out, he must touch the brake before the *point of no return* beyond which it would be too late for the vehicle to stop. The position of the point of no return depends on the speed of the vehicle.

5.1 Maneuverability Profiles

Semi-autonomous vehicles only have limited self-driving automation. For example, adaptive cruise control can only provide autonomous longitudinal control, but still depends on the human driver to use the steering wheel to maintain the lateral control. Hence the longitudinal control of Type SA-ACC vehicles can be very precise, but the lateral control cannot be as precise. Furthermore, the human driver cannot make a sharp turn while the adaptive cruise control is active, because the human must slow down the vehicle a lot in order to be able to make a sharp turn precisely. In our system, we use a *maneuverability profile* to encode the limitations of the control of human drivers when the automation devices are activated. For simple cruise control, the maneuverability profile is written in Lisp syntax as follows:

```
(cc-profile (v verror angle)
  (is-auto-speed-control)
  (not is-auto-steering)
  (< velocity (+ v verror))
  (> velocity (- v verror))
  (< steer-angle angle) (> steer-angle -angle))
```

where `v` is the target velocity, `verror` is the maximum error of the target velocity, and `angle` is the maximum feasible steering angle for the human driver when the cruise control is turned on. The maneuverability profile for adaptive cruise control is

```
(acc-profile (vin d derror angle)
  (is-auto-speed-control)
  (not is-auto-steering)
  (< (dist-from vin) (+ d derror))
  (> (dist-from vin) (- d derror))
  (< steer-angle angle) (> steer-angle -angle))
```

where `vin` is the vehicle id of the vehicle ahead, `d` is the target distance from the vehicle ahead, `derror` is the maximum error for acc to maintain the target distance from `vin`, and `angle` is the maximum feasible steering angle for the human driver.

Other types of semi-autonomous vehicles have their own maneuverability profiles. When a driver agent makes a reservation request, it is his or her responsibility to ensure that the trajectories encoded by the reservation request *satisfy* all constraints in the maneuverability profile.

5.2 Constraint-Based Requests

Different maneuverability profiles can have different sets of constraints. To facilitate communication with all kinds of semi-autonomous vehicles, SemiAIM uses a *unified* language for vehicles to express their constraints in the same format in their requests. We define *constraint-based* reservation requests as follows. A request message consists of four components:

1. **Intention**: The direction in which the vehicle intends to move.
2. **Vehicle Type**: The type of vehicle.
3. **Entry Condition**: The condition under which the vehicle will enter the intersection.

4. **Acceleration Profile List**: The list of possible acceleration schedules from among which the vehicle will choose one to follow during the traversal of the intersection.

The intention of a vehicle is the direction in which the vehicle wants to exit from an intersection. The intention is expressed as an *intention statement*, which is formally a disjunction of lane and road identifiers: $(l_1 \vee l_2 \ldots \vee r_1 \vee r_2)$, where l_i is an exit lane and r_i is an exit road. For every lane l_i, there exists only one r_j such that $l_i \in r_j$. Examples of legal intentions are $(l_1 \vee l_3)$ or (r_1).[6] This feasibility facilitates different path planning strategies the vehicle might use.

The vehicle type is the information with which the IM can determine how the vehicle will move inside an intersection. Different types of vehicles have different sizes, shapes, and kinematics. For example, the motion of a bus is very different from the motion of a passenger car. By knowing the type of the vehicle, the IM will be able to compute the trajectories of the vehicle under various conditions such as the arrival times and velocities. We assume the IM maintains a database of vehicle types.

The entry condition is the condition under which a vehicle enters an intersection. An *entry statement* is used to describe the entry condition. An entry statement consists of three parts: an arrival lane condition, an arrival time constraint, and an arrival velocity constraint. An *arrival lane condition* states the possible lanes from which the vehicle will enter the intersection. It is a disjunction of labels: $(l_1 \vee l_2 \vee \ldots \vee l_n)$ where l_i is a possible lane from which the vehicle enters the intersection. An *arrival time constraint* $[t_1,\ t_2]$ states the time interval during which the vehicle will arrive at the intersection. An *arrival velocity constraint* $[v_1,\ v_2]$ states that the arrival velocity of a vehicle will be between v_1 and v_2. An entry statement is a 3-tuple $\langle (l_1 \vee l_2 \vee \ldots \vee l_n), [t_1,\ t_2], [v_1,\ v_2] \rangle$.

An *acceleration profile* is the acceleration schedule the vehicle will use to accelerate through the intersection on a trajectory. An acceleration profile is a list of pairs $\langle (t_1, a_1), (t_2, a_2), \ldots, (t_n, a_n) \rangle \in A$, where A denotes the set of possible acceleration profiles, and a_i is the acceleration the vehicle will use from time t_i until time t_{t+1}. Note that the vehicle may or may not provide the acceleration profile of all possible trajectories in a request message. If the acceleration profile is missing, the IM will generate an acceleration profile based on a simulation of the movement of the vehicle given the vehicle type and the entry condition.

Each vehicle may have its own algorithm to generate constraint-based requests that satisfy its maneuverability profile. In this paper, we do not fully explore the space of how vehicles could generate such requests. However, we define the following two requests that will be used by Type SA-CC and Type SA-Com vehicles, respectively:

- A **constant-velocity request** is $\langle \mathsf{Intent}, \mathsf{Type}, \mathsf{Entry}, \mathsf{AP} \rangle$, where $\mathsf{Intent} = (l_1 \vee l_2 \vee \ldots \vee l_n)$ in which l_i is a possible lane from which the vehicle exits the intersection; Type is the vehicle type; $\mathsf{Entry} = ((l'_1 \vee l'_2 \vee \ldots \vee l'_n), [t_1,\ t_2], [v_1,\ v_2])$ is the entry statement; and $\mathsf{AP} = (\langle (t_1, 0) \rangle)$ is the acceleration profile list. Since the acceleration in AP is always zero, the vehicle will move at a constant velocity.

[6] An intention in the form of $(r_i \vee r_j)$ is also possible, especially for multiple-intersection management, which involves path planning. In this case, the vehicle is proposing two directions to go and the IM will respond with a confirmation message for either one. We will leave this case for future work.

- A **whole-row request** is $\langle \mathsf{Intent}, \mathsf{Type}, \mathsf{Entry}, \mathsf{AP} \rangle$, where $\mathsf{Intent} = (l_1 \vee l_2 \vee \ldots \vee l_n)$ in which l_i is a possible lane from which the vehicle exits the intersection; Type is the vehicle type; $\mathsf{Entry} = ((l'_1 \vee l'_2 \vee \ldots \vee l'_n), [t_1,\ t_2], [v_1,\ v_2])$ is the entry statement; and AP is the acceleration profile list. In order to reserve the entire row in an intersection, the difference between t_1 and t_2 as well as the difference between v_1 and v_2 must be large enough to cover the entire row, regardless of the acceleration profiles.

Unlike AIM's reservation requests which always correspond to a particular trajectory, a constraint-based request is an *incomplete* description of the trajectory. Therefore, the IM interprets a constraint-based request as a description of a set of *possible* trajectories that the vehicle may follow, and reserves all the tiles on the possible trajectories. Once the IM receives a request, it conducts an internal simulation of the vehicle movement, as described in [6], to determine the tiles on the possible trajectories, and determine whether the tiles are available.

Our hypothesis is that more restrictive constraint-based requests will perform better because SemiAIM reserves fewer tiles for them, meaning that some types of semi-autonomous vehicles will work better with SemiAIM. We test this hypothesis in Section 6.

5.3 Anchor Requests

Semi-autonomous vehicles with adaptive cruise control can use a special constraint-based request called *anchor requests* to make reservations. An anchor request is $\langle \mathsf{Type}, \mathtt{vin}, d \rangle$, where Type is the vehicle type, `vin` is the vehicle id of the vehicle ahead, and d is the following distance from the vehicle of `vin`. For example, an anchor request $\langle \mathtt{truck\text{-}with\text{-}acc}, \mathtt{GXC345}, \mathtt{5m} \rangle$ states that the vehicle will follow the vehicle with a vehicle id `GXC345` and maintain a following distance of $5m$. This request is different from the previous types because it depends on the confirmed reservation of the vehicle v with id `vin`. If v does not have a reservation, the IM will reject the anchor request; otherwise, the IM will copy the request of v and modify the request with following distance d.

Currently anchor requests are designed specifically for semi-autonomous vehicles with adaptive cruise control. However, this formulation is general enough to enable some sophisticated cooperative maneuvers such as platooning [13].

6 Experimental Evaluation

To demonstrate the feasibility of SemiAIM as well as evaluate the hypothesis that SemiAIM can offer substantial improvements over traffic signals and FCFS-Signal, we conducted a series of experiments with the modified AIM4 simulator that simulates the behavior of vehicles using the constraint-based reservation system. The implementations of fully autonomous vehicles (Type A) and human-driven vehicles (Type H) are as provided with the simulator and described in [6]. The concrete implementations of the semi-autonomous vehicles, as described in Section 3, are as follows.

- **Type SA-ACC Vehicles**: If there exists a vehicle ahead which is either autonomous or semi-autonomous and is going in the same direction, it sends an anchor request

to the IM. If the request is confirmed, the vehicle follows the vehicle ahead and enters the intersection. If the request is denied or there is no such vehicle ahead, the vehicle sends a constant-velocity request to the IM instead. If it can enter the intersection by keeping the current velocity, the request is confirmed; otherwise, the vehicle resends the constant-velocity request after a few seconds, and at the same time decelerates enough to be able to stop before the intersection. If it keeps failing to obtain a reservation it stops at the intersection and returns control to the human driver, who must then obey the traffic signal.
- **Type SA-CC Vehicles**: Sends a constant-velocity request to the IM when it is close enough to the intersection. If it can enter the intersections by keeping the current velocity, the request is confirmed; otherwise, the vehicle resends the request after a few seconds, and at the same time decelerates enough to be able to stop before the intersection. If it keeps failing to obtain a reservation it stops at the intersection and returns control to the human driver, who must then obey the traffic signal.
- **Type SA-Com Vehicles**: Sends a whole-row request to the IM. If the entire lane is available, the request is confirmed; otherwise, the vehicle resends the request after a few seconds, and at the same time decelerates enough to be able to stop before the intersection. If it keeps failing to obtain a reservation it stops at the intersection and returns control to the human driver, who must then obey the traffic signal.

Our experiments were conducted in a 3x3 intersection (3 lanes in each direction) as shown in Figure 1. Unless otherwise specified, the simulator spawns vehicles in each lane according to a Poisson distribution with an expectation of 360 vehicles per hour. We denote this setting as traffic level = 360 vehicles/hour/lane. We chose this traffic level as being heavy enough to cause significant delays at signals, but light enough to allow for benefits even if cars are not precisely controlled. Figure 7 examines higher traffic levels.

The performance of an intersection is measured by the *average delay*, where delay is computed as the increase in travel time compared to traversing the intersection without slowing down at all, as if no other vehicles were on the road. Thus, lower delays correspond to better performance. This measure is the same one used in [6]. For all the vehicles, both the static buffer and the edge time buffer are 0.25 meters (see [6] for details), when auto-controlled.

Traffic signals are used as a fallback strategy when vehicles cannot get a reservation. We used an *optimized* 6-phase signal schedule that includes relatively long green times (roughly 35 seconds) for East-West and North-South phases, and shorter phases for each of the four single direction phases (roughly 9 seconds). It was optimized using SYNCHRO, a commercial traffic optimization package.

6.1 Experiment 1: Technology Penetration

Our first experiment studies the effect of an increasing penetration rate of vehicular automation technology. Suppose human-driven vehicles are gradually replaced by a particular type of semi-autonomous vehicle or fully autonomous vehicle until all vehicles become that type. We examine how much benefit SemiAIM provides during the transition period.

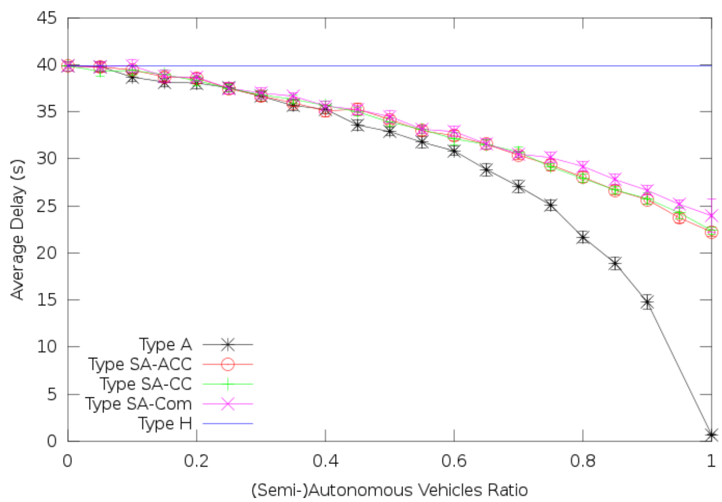

Fig. 3. (Semi-)Autonomous vehicles vs. Human-Driven vehicles. Traffic level = 360 vehicles/lane/hour.

In this experiment, the traffic consisted of one of the three types of vehicles we defined in Section 3 as well as Type A and Type H vehicles. We measured the traffic delay as we gradually increased the ratio of (semi-)autonomous vehicles to human-driven vehicles (Type H) while keeping the traffic level at 360 vehicles/hour/lane. As an example, consider the ratio of Type SA-ACC vehicles. At the beginning, there are 0% Type SA-ACC vehicles and 100% human-driven vehicles, and the ratio is 0. As the number of Type SA-ACC vehicles increases, the ratio increases and eventually becomes 1, which means there are 100% Type SA-ACC vehicles and 0% human-driven vehicles. We repeated the simulation 30 times for 1800s during each time. For each run, we measured the average delay of all vehicles. The average delays are shown in Figure 3. Each data point in the figure is an average of 30 values, and the error bar is the 95% confident interval of the average delay.

According to Figure 3, the performance of semi-autonomous vehicles is very similar to fully autonomous vehicles when the ratio to human-driven vehicles is below 40%. However, when the ratio increases beyond 40%, fully autonomous vehicles increasingly outperform semi-autonomous vehicles. Previous studies showed that FCFS-Signal needs at least 90% of fully autonomous vehicles in the traffic in order to be fully effective [5]. We successfully replicate the result, observing that the average delay drops rapidly when the traffic has more than 90% fully autonomous vehicles, and approaches zero when all vehicles are fully autonomous. Semi-autonomous vehicles cannot achieve the same dramatic decrease in traffic delay, but they, with the help of SemiAIM, manage to reduce the delay by 46% (from 39.9s to 22.4s) compared to human-driven vehicles. As expected, in the presence of semi-autonomous vehicles, SemiAIM provides significant advantages even when there are no fully autonomous vehicles on the road.

Another observation is that both Type SA-ACC and Type SA-CC vehicles have a significantly lower average delay than Type SA-Com vehicles. The difference is consistent with our hypothesis that the use of more constrained requests can increase the performance of intersections since the footprints of the vehicles are smaller and more vehicles can enter the intersection at the same time. Nonetheless, the difference is small because the simulated human drivers in the simulator can control their vehicles quite well.

6.2 Experiment 2: Incremental Deployment

Our second experiment considers realistic scenario of *incremental deployment* of autonomous vehicle technology. We believe that the adoption of semi-autonomous vehicles will be much faster than fully autonomous vehicles since the cost of ownership of semi-autonomous vehicles are lower. Thus we anticipate there will be a period of time during which semi-autonomous vehicles dominate the roads. However, this domination will be short-lived—eventually fully autonomous vehicles will displace semi-autonomous vehicles due to their greater convenience. This section studies the effects of different levels of incremental deployment of (semi-)autonomous vehicles over time.

We consider two plausible deployment schedules, one for the period of time before the domination of semi-autonomous vehicles, and the other for the subsequent period. In the first deployment schedule (Table 2), Type H vehicles on the road are gradually displaced by (semi-)autonomous vehicles. Assume two out of three car buyers who abandoned their vehicles for newer technology will buy a semi-autonomous vehicles rather than a fully autonomous vehicle. Then eventually about two thirds of vehicles on the road will be semi-autonomous, and one third will be fully autonomous. Table 3 shows the second deployment schedule under which semi-autonomous vehicles are gradually replaced by fully autonomous vehicles. We assume that even when most vehicles are fully autonomous, a small number of drivers still prefer driving themselves.

Figures 4 and 5 show the results of the simulations according to these deployment schedules. The simulation setup is exactly the same as in Experiment 1. For each vehicle distribution, we ran the simulation 30 times for 1800s each run. Thus, each data point in the figures is an average of 30 delay times. The 95% CIs of the average are shown as the error bars in the figures. According to these figures, the average delay of the vehicles keeps decreasing as (semi-)autonomous vehicles are deployed. In Figure 4, the (semi-)autonomous vehicles contribute to reduce the average delay time significantly, even though the fully autonomous vechiles (Type A) only make up one third of them. In Figure 5, we observe the advantage of fully autonomous vehicles over semi-autonomous ones, though such substitution doesn't contribute as much as that in Figure 4. Namely, incremental development of (semi-)autonomous vehicles (Figure 4) contributes more to the incremental development of fully autonomous vehicles (Figure 5). Ultimately, the average delay depends on the number of drivers who insist on driving themselves.

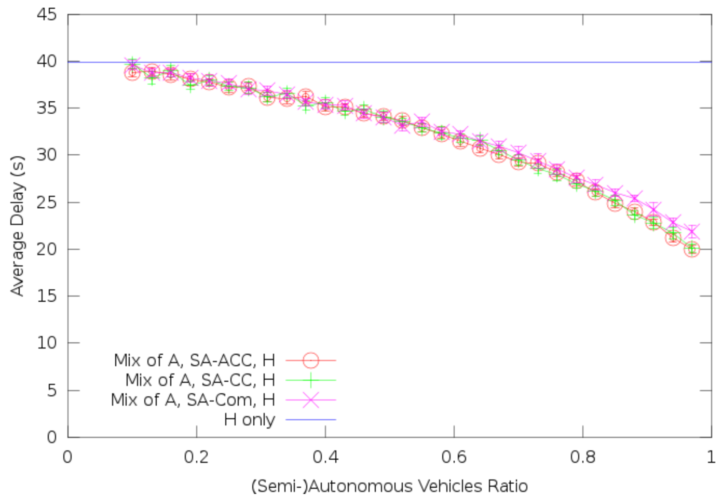

Fig. 4. The average delay according to the deployment schedule in Table 2. Traffic level = 360 vehicles/lane/hour.

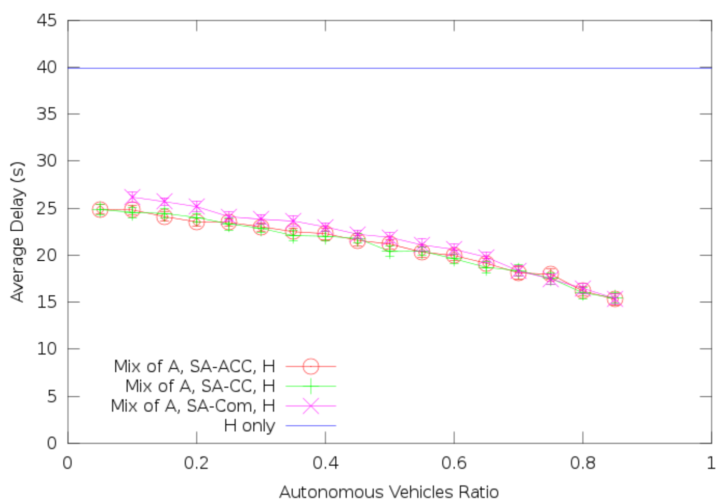

Fig. 5. The average delay according to the deployment schedule in Table 3. Traffic level = 360 vehicles/lane/hour.

Table 2. The deployment schedule in Figure 4.

Type H	Type SA-ACC / SA-CC / SA-Com	Type A
90%	9%	1%
87%	11%	2%
84%	13%	3%
...	...	...
0%	69%	31%

Table 3. The deployment schedule in Figure 5.

Type H	Type SA-ACC / SA-CC / SA-Com	Type A
10%	85%	5%
10%	80%	10%
10%	75%	15%
...	...	...
10%	5%	85%

6.3 Experiment 3: Request Types

We anticipate that the three types of requests we introduced in Section 5 will be common if SemiAIM is deployed in the real world. Our third experiment compares the performance of different types of requests and checks the hypothesis that requests that reserve fewer tiles will perform better. The setup of this experiment is the same as Experiment 1 except that 1) the architecture of the driver agents are the same in all simulations; and 2) two traffic levels are examined. The three request types we considered are: anchor requests, constant-velocity requests, and whole-row requests. As a comparison, we included *AIM requests* which are used by fully autonomous vehicles to make reservations under AIM. Each request type needs a (semi-)autonomous vehicle as a substrate. Anchor requests require vehicles to be equipped with ACC; constant-velocity requests need vehicles with simple cruise control; whole-row requests need vehicles with some communication devices; and AIM requests require the vehicle to be fully autonomous. Note that these vehicles are not exactly SA-ACC, SA-CC, and SA-Com, since they use the same fallback mechanism. In particular, unlike SA-ACC, the vehicles that make anchor requests will not issue any constant-velocity requests if it keeps failing to obtain a reservation.

The experiment was run twice at two traffic levels: 360 vehicles/lane/hour and 540 vehicles/lane/hour, and the results are shown in Figures 6 and 7, respectively. As a baseline, the average delay of human-driven vehicles under traffic signals is shown as the “No request” lines in these figures. Figure 6 is similar to Figure 3 except that the average delays of semi-autonomous vehicles is larger. This difference arises because vehicles can mostly make one type of requests, thus losing some opportunities to enter an intersection during red signal phases. Among the three request types, the whole-row requests performed best this time. This result is unexpected and is different from the result with Type SA-Com vehicles in Figure 3. Since the whole-row requests need more tiles than the other two requests, our hypothesis that requests that reserve less tiles will perform better does not hold. However, at a higher traffic level, the whole-row requests perform worst for the highest technology penetration rate (see Figure 7). This result implies that apart from the number of tiles being reserved, there are other factors that affect the performance of a request type.

7 Related Work

The main context of our work is an extension to the FCFS policy proposed by Dresner and Stone [6]. Their experimental results indicated that a mixture of human-driven vehicles and autonomous vehicles is possible, and leads to better performance than having all human-driven vehicles, which is the current status quo. However, their experiments indicated that the impact of autonomous vehicles is expected to be relatively small until almost all (90-95%) of the vehicles on the road are autonomous. Our extension to embrace semi-autonomous vehicles shows significant performance benefits all along the technology penetration curve.

Our work is similar to the analysis of adaptive cruise control performance by Jerath and Brennan, who showed that by introducing adaptive cruise control vehicles into traffic, the vehicles would have a more *condensed* formation, thus increasing the efficiency of traffic [8]. When compared to our work, there are two key differences. First, we are focusing on the intersection—a place more critical both from the points of view of congestion and safety, while the analysis of Jerath and Brennan is focused on highways. Second, we study five types of vehicles, compared to the three types considered by Jerath and Brennan.

Although we are not aware of any other work that is directly concerned with the interaction of semi-autonomous and autonomous vehicles, research on vehicular autonomy in general has made significant progress in recent years. This was in part due to a series of robotic car competitions such as the *DARPA Grand Challenges* [1]. These competitions accelerated the development of autonomous vehicles to the point where the technical challenge of open-road autonomous driving is considered by some to be essentially solved [6]. When pushed to extremes, autonomous vehicles can even outperform many human drivers in carrying out intricate maneuvers [15]. The non-technical barrier for the adaptation of autonomous vehicles are largely traffic laws and regulations, though progress is also being made in these areas [3].

The vast majority of research on autonomous vehicles focuses on how to ensure they run on existing road infrastructure; there is limited literature on understanding changes to road infrastructure that can facilitate vehicular autonomy. One such project on jointly optimizing autonomous vehicles and road infrastructure is the PATH program, which relies on magnetic markers in the roadway for measuring steering angle and vehicle movements [14]. The AIM protocol [6, 7, 12] is a vehicle-to-infrastructure (V2I) mechanism in which vehicles request space-time in the intersection for their trajectories prior to arriving at the intersection; a server at the intersection handles these requests, granting or rejecting reservations using a grid-based collision detection scheme. This protocol is enhanced to reduce network traffic and increase safety using spatial-temporal buffers surrounding the vehicles [7].

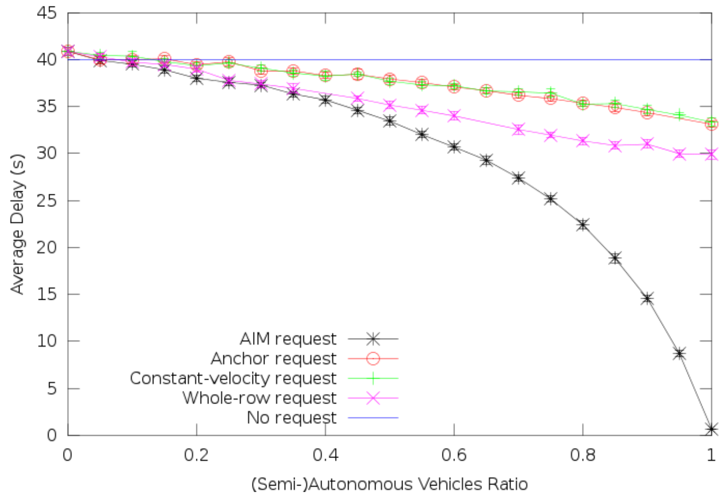

Fig. 6. Comparisons of request types. Traffic level = 360 vehicles/lane/hour.

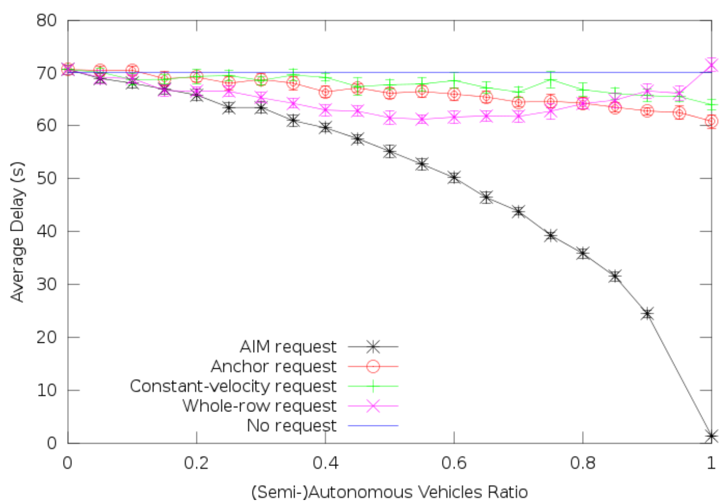

Fig. 7. Comparisons of request types. Traffic level = 540 vehicles/lane/hour.

Vehicle-to-Vehicle (V2V) forms of autonomous intersection management have also been investigated [10, 16]. In this form, no centralized server is required (i.e., there is no single point of failure) and vehicles coordinate in a peer-to-peer fashion when crossing the intersection. Naumann *et al.* investigated a distributed policy that uses virtual "tokens" that a vehicle must possess to cross certain contested areas of the intersection [10] and formally evaluated it using petri-nets. VanMiddlesworth *et al.* developed an AIM-inspired protocol that enables vehicles to "call ahead" to reserve space-time in the intersection [16]. Their protocol outperformed the traditional stop sign in light traffic. The anchor requests introduced in this paper can be implemented using V2V communications.

Other researchers have investigated autonomous intersections using real systems involving multiple mobile vehicles. For example, Kolodko and Vlacic used golf-cart-like Imara vehicles in evaluating an autonomous intersection [9]. In their study, all vehicles must come to a complete stop at the intersection irrespective of traffic conditions. Our work differs from this, and all other previous research by incorporating semi-autonomous vehicles into the mix, along with autonomous and human-driven cars.

8 Conclusions and Future Work

This paper introduces SemiAIM, a new multiagent constraint-based autonomous intersection management system that enables human-driven vehicles and semi-autonomous vehicles, in addition to fully autonomous vehicles, to make reservations and enter an intersection within the AIM paradigm. To the best of our knowledge, SemiAIM is the first multiagent protocol to enable smooth interactions between human-driven, fully autonomous, and semi-autonomous vehicles. Our experimental results showed that our system can greatly decrease traffic delay when most vehicles are semi-autonomous, even when few (if any) are fully autonomous.

This study opens up several interesting directions for future work. For example, an open question is how to design better constraint-based reservation requests using more accurate profiling of the vehicles' physical behavior. It will also be important to study in detail the performance of SemiAIM under a variety of different, or varying, traffic levels, and with different amounts of traffic traveling in different directions.

References

1. DARPA grand challenge. `http://en.wikipedia.org/wiki/DARPA_Grand_Challenge`.
2. T.-C. Au and P. Stone. Motion planning algorithms for autonomous intersection management. In *AAAI 2010 Workshop on Bridging The Gap Between Task And Motion Planning (BTAMP)*, 2010.
3. R. Calo. Nevada bill would pave the road to autonomous cars. `http://cyberlaw.stanford.edu/node/6663`, April 2011.
4. DARPA. DARPA Urban Challenge. `http://www.darpa.mil/grandchallenge`, 2007.
5. K. Dresner and P. Stone. Sharing the road: Autonomous vehicles meet human drivers. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.

6. K. Dresner and P. Stone. A multiagent approach to autonomous intersection management. *Journal of Artificial Intelligence Research (JAIR)*, March 2008.
7. D. Fajardo, T.-C. Au, S. T. Waller, P. Stone, and C. Y. D. Yang. Automated intersection control: Performance of a future innovation versus current traffic signal control. *Transportation Research Record : Journal of the Transportation Research Board*, (2259):223–232, 2012.
8. K. Jerath and S. N. Brennan. Adaptive cruise control: Towards higher traffic flows, at the cost of increased susceptibility to congestion. In *AVEC'10*, 2010.
9. J. Kolodko and L. Vlacic. Cooperative autonomous driving at the intelligent control systems laboratory. *Intelligent Systems, IEEE*, 18(4):8 – 11, jul-aug 2003.
10. R. Naumann and R. Rasche. Intersection collision avoidance by means of decentralized security and communication management of autonomous vehicles. In *Proceedings of the 30th ISATA - ATT/IST Conference*, 1997.
11. NHTSA. Preliminary statement of policy concerning automated vehicles. Technical report, National Highway Traffic Safety Administration, 2013.
12. M. Quinlan, T.-C. Au, J. Zhu, N. Stiurca, and P. Stone. Bringing simulation to life: A mixed reality autonomous intersection. In *IEEE/RSJ International conference on Intelligent Robots and Systems*, 2010.
13. S. Sheikholeslam and C. A. Desoer. Longitudinal control of a platoon of vehicles. In *American Control Conference*, pages 291–296, 1990.
14. S. Shladover, C. Desoer, J. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown. Automated vehicle control developments in the path program. *IEEE Transactions on Vehicular Technology*, 40(1):114–130, 1991.
15. C. Squatriglia. Audi's robotic car drives better than you do. `http://www.wired.com/autopia/2010/03/audi-autonomous-tts-pikes-peak`, March 2010.
16. M. VanMiddlesworth, K. Dresner, and P. Stone. Replacing the stop sign: Unmanaged intersection control for autonomous vehicles. In *AAMAS Workshop on Agents in Traffic and Transportation*, pages 94–101, Estoril, Portugal, May 2008.