To appear in International Conference on Intelligent Robots and Systems (IROS),
San Francisco, September 2011.

Autonomous Intersection Management: Multi-Intersection Optimization

Matthew Hausknecht, Tsz-Chiu Au, and Peter Stone
Department of Computer Science
The University of Texas at Austin
Austin, TX 78712
{mhauskn,chiu,pstone}@cs.utexas.edu

***Abstract*— Advances in autonomous vehicles and intelligent transportation systems indicate a rapidly approaching future in which intelligent vehicles will automatically handle the process of driving. However, increasing the efficiency of today's transportation infrastructure will require intelligent traffic control mechanisms that work hand in hand with intelligent vehicles. To this end, Dresner and Stone proposed a new intersection control mechanism called *Autonomous Intersection Management* (AIM) and showed in simulation that by studying the problem from a multiagent perspective, intersection control can be made more efficient than existing control mechanisms such as traffic signals and stop signs. We extend their study beyond the case of an individual intersection and examine the unique implications and abilities afforded by using AIM-based agents to control a network of interconnected intersections. We examine different navigation policies by which autonomous vehicles can dynamically alter their planned paths, observe an instance of Braess' paradox, and explore the new possibility of dynamically reversing the flow of traffic along lanes in response to minute-by-minute traffic conditions. Studying this multiagent system in simulation, we quantify the substantial improvements in efficiency imparted by these agent-based traffic control methods.**

INTRODUCTION

Increasing numbers of vehicles burden today's transportation infrastructure, often resulting in substantial transportation delays. As roads become increasingly over-utilized, new solutions to old traffic problems are in high demand. Simultaneously, as advances in Autonomous Vehicles and Intelligent Transportation Systems begin to lay the foundation for a future of computer-controlled vehicles, so too must we work to improve the mechanisms which coordinate the flow of vehicles on existing networks of roads.

To this end, Dresner and Stone introduced a multiagent systems approach to traffic intersection management called *Autonomous Intersection Management* (AIM) in which vehicles *(driver agents)* call ahead to a reservation manager agent at the intersection to reserve conflict-free trajectories in intersection space-time [1]. In comparison to today's intersection managers (traffic signals and stop signs), AIM dramatically improves traffic throughput in the case of a *single* intersection. However, little is known about coordinating multiple intersections using AIM in order to optimize traffic flow through a network of interconnected roads.[1]

On the other hand, research in the traffic and transportation literature has focused extensively on the interactions between multiple intersections using traditional control mechanisms such as stop signs and especially traffic signals. A current focus area is on dynamic traffic assignment (DTA) whereby cars adapt their routes, and signals adapt their timings, based on current traffic conditions [2], [3], [4], [5].

This paper pushes the boundaries of what can be done with a multiagent-based approach to autonomous control of multiple intersections. More than just applying DTA to AIM-based intersections, we consider the unique properties of AIM that enable novel, fine-grained, dynamic control of autonomous vehicles through a grid of intersections. Specifically, we demonstrate how individual vehicles can leverage the information already available through the AIM protocol to predict travel times along various routes and dynamically change their planned paths. Next, we observe an instance of Braess' paradox whereby opening additional travel options for the vehicles reduces the efficiency of all vehicles in the system. In response, we explore the possibility of dynamically redirecting lanes in response to minute-by-minute traffic conditions. Throughout the paper we assume that each car is an autonomous agent interested in minimizing its own individual travel time. At no point do the cars accept redirection from any outside decision-maker. We examine the resulting multiagent system and, consistent with the original AIM results, demonstrate the significant potential improvements compared to current control mechanisms when taking advantage of the capabilities of autonomous vehicles.

AUTONOMOUS INTERSECTION MANAGEMENT (AIM)

This multiagent systems-based intersection management strategy, introduced by Dresner and Stone, employs a reservation protocol in which vehicles (*driver agents*) who wish to cross an intersection will contact the *Intersection Manager (IM)* agent responsible for managing that particular intersection and reserve a trajectory through intersection space-time [1], much like a landing aircraft will contact an air traffic control tower to reserve space along a certain runway at a certain time for its landing. The intersection manager decides whether to grant or reject requested reservations according

This work has taken place in the Learning Agents Research Group (LARG) at UT Austin. LARG research is supported in part by NSF (IIS-0917122), ONR (N00014-09-1-0658), and FHWA (DTFH61-07-H-00030).

[1]The few studies of multiple intersections in AIM have essentially just strung individual intersections together, without considering their interactions.

to an *intersection control policy*. A typical interaction goes as follows:

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

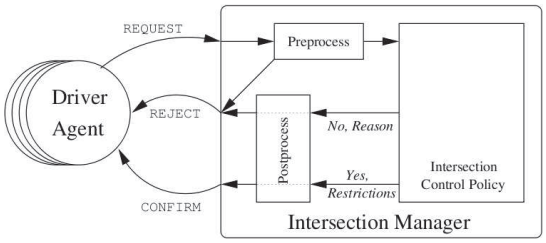
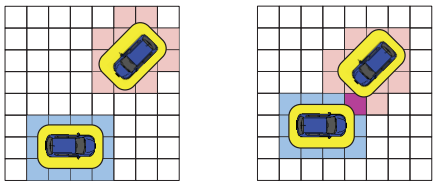


Fig. 1: Diagram of the intersection system.



(a) Successful (b) Rejected

Fig. 2: (a) The vehicle's space-time request has no conflicts at time t. (b) The black vehicle's request is rejected because at time t of its simulated trajectory, the vehicle requires a tile already reserved by another vehicle. The shaded area represents the *static buffer* of the vehicle.

Figure 1 diagrams the interaction between driver agents and an intersection manager. A key feature of this paradigm is that it relies only on vehicle-to-infrastructure (V2I) communication[2]. In particular, the vehicles need not know anything about each other beyond what they need for local control (e.g. to avoid running into the car in front). The paradigm is also completely robust to communications disruptions: if either the intersection manager or the vehicle drops a message, delays may increase, but safety is not compromised. AIM also guarantees safety when both autonomous and manual vehicles operate at intersections. The intersection efficiency will increase with the ratio of autonomous vehicles to manual vehicles in such scenarios.

The prototype intersection control policy divides the intersection into a grid of *reservation tiles*, as shown in Fig. 2.

[2]V2V was later introduced for low-traffic intersections.

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 the vehicle will occupy. If at any time during the trajectory simulation the requesting vehicle occupies a reservation tile that is already occupied by another vehicle, the policy rejects the driver's reservation request, and the intersection manager communicates this to the driver agent. Otherwise, the policy accepts the reservation and reserves the appropriate tiles. The intersection manager then sends a confirmation to the driver. If the reservation is denied, it is the vehicle's responsibility to maintain a speed such that it can stop before the intersection. Meanwhile, it can request a different reservation.

Empirical results in simulation demonstrate that the proposed reservation system can dramatically improve the intersection efficiency when compared to traditional intersection control mechanisms. To quantify efficiency, Dresner and Stone measure *delay*, defined as the amount of additional travel time incurred by a vehicle as a result of passing through the intersection under external traffic load compared to travel time of the vehicle passing through the same intersection without any other cars. According to their experiments, the reservation system performs very well, nearly matching the performance of the optimal policy, which represents a lower bound on delay should there be no other cars on the road. 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. Due to space constraints, we cannot fully describe the details of the AIM protocol. For a complete specification, including how it handles human drivers, pedestrians, dropped messages, and accidents in the intersection, refer to [1].

AIM has been studied extensively in the case of single intersections. However, we now shift focus from prior work to our own contributions relating to extending the AIM protocol from a single intersection to multiple intersections.

Navigation Policies

Transitioning from a single intersection to a grid of intersections requires each *driver agent* to have a *navigation policy* which determines the route the vehicle takes towards its destination. Vehicles are responsible for determining their own paths through a network of roads and are required only to make local reservations with the IM of the next intersection they plan to traverse. This property allows agents to revise their planned paths dynamically in light of new information about arising traffic conditions.

Intersection Delay Estimation

In order for Intersection Managers to keep current information about incoming traffic, we modify the basic AIM protocol by requring vehicles to send a *notification message* before calling for a reservation. A vehicle sends this message as soon as it enters a road leading to an intersection

and comes within DSRC *(Dedicated Short-Range Communication)* range, approximately 300 meters. This message, typically sent long before a reservation request, informs the intersection manager that a vehicle will eventually request a reservation and cross the intersection. Additionally, it conveys to the IM the road currently occupied by the vehicle (e.g. the road which enters the intersection) as well as the exit road the vehicle plans to depart on. Keeping track of incoming notification messages, intersection managers now possess the required information to estimate and publish the *expected traversal times* of incoming vehicles. (These delay estimates form the foundation of the time-based navigation policy discussed later.) In this case, traversal time equals the time elapsed between when the vehicle sent a notification message and the time that the vehicle successfully exits the intersection. In order to accurately estimate traversal time of a *querying vehicle*, intersection managers consider the following features:

1) *Total Active Vehicles (TAV)*: The total number of vehicles actively interacting with this intersection (vehicles are considered active after they send a notification message and before they finish traversing the intersection). Since AIM protocol requires vehicles only to make reservations for the most proximate intersection they plan to traverse, this quantity represents a rough estimate of instantaneous total demand for this intersection.
2) *Path-Active Vehicles (PAV)*: The number of active vehicles along the querying vehicle's planned (entry road, departure road) trajectory. More specialized than the last, this metric gives an estimate of the instantaneous demand along the vehicle's planned path through the intersection. Intuitively, a small number of path-active vehicles may indicate low delays along a certain path despite a large number of total active vehicles.
3) *Oldest Path-Active Vehicle (oPAV)*: The time at which the oldest active vehicle along the querying vehicle's planned (entry road, departure road) trajectory notified the intersection manager of its presence. In steady state traffic (e.g. delays are neither systematically increasing or decreasing), the traversal time of the oldest active vehicle closely approximates the traversal time of a new vehicle along the same path.
4) *Temporal Window Average (TWA)*: The average traversal time along the querying vehicle's planned trajectory of all vehicles who successfully crossed the intersection within the last n seconds. In our experiments, using a temporal window over the last $n = 50$ seconds allowed traversal time averages to adapt reasonably quickly to influxes of traffic while not being overly sensitive to minute fluctations and noisy data.

Intersection managers calculate traversal time estimates from the following linear combination of features:

$$estTT = .09\text{TAV} + .83\text{PAV} + .25\text{oPAV} + .25\text{TWA} + 2.26 \quad (1)$$

We developed Equation 1 by simulating a single intersection for 2000 seconds with 9297 incoming vehicles/hour. As every vehicle traversed the intersection, the value of each of the above features was recorded as well as the actual traversal time of that vehicle. We derived the weights in Equation 1 via linear regression on the resulting dataset, which yielded a mean absolute prediction error of less than seven seconds. Using this equation, intersection managers answer queries from incoming vehicles about expected traversal times.

Notably, the weights in Equation 1 reflect the flow-pattern of incoming traffic. The flow pattern used here mimicked an intersection under an average, evenly distributed load. As this flow changes, we expect corresponding changes in traversal times. Adapting the weights in Equation 1 to changing traffic-flow-patterns remains a subject of future work.

*Time-based A**

Leveraging the expected traversal time estimates provided by intersection managers, the time-based A* navigation policy performs an A* search [6] starting from the vehicle's current location and terminating at the vehicle's destination. Each possible path through the grid of intersections is sorted not on the distance of that path, but instead on the path's expected travel time. The expected travel time of a path is calculated by summing the traversal time estimates provided by each intersection manager incident along the path. Upon termination, the search returns the route which minimizes expected traversal time. Since intersection managers provide no guarantees against over-estimating traversal times, the heuristic we propose is not strictly *admissible* (e.g. if intersection managers provide poor estimates, A* is not guaranteed to return the minimum time path). Despite this, empirically it has performed well.

Since AIM protocol requires vehicles to send a notification message only to the most proximate intersection manager, they are free to dynamically re-plan routes using updated delay estimates as frequently as desired. We hypothesize that driver agents employing the time-based A* navigation policy will adapt far better to changing traffic conditions than agents following distance-based shortest path navigation policies.

Experimental Evaluation

We empirically evaluate the time-based A* navigation policy on a 2×2 grid of intersections shown in Figure 3. Persistent traffic with flow 4500 vehicles/hour enters the grid at point A and exits at point D ($A \rightarrow D$), utilizing either path $\{A, 0, 2, 3, D\}$ or $\{A, 0, 1, 3, D\}$. We designate this flow the *Traffic of Interest (ToI)* and study how well the individual driver agents constituting this flow can adapt their paths to avoid sporadic influxes of heavy traffic.

The scenario subjects the traffic of interest to two different influxes of sporadic traffic ($B \rightarrow E$ and $C \rightarrow E$), with flow 9000 vehicles/hour. Each influx of sporadic traffic lasts for 100 seconds and is followed by the alternate influx after a cooldown period of 100 seconds. Finally, we add random source, destination traffic to the scenario with a flow of 900 vehicles/hour on all roads except A, B, and C (as these spawn other flows). The goal of the scenario is for the ToI to leverage delay estimates provided by intersection managers

in order to dynamically adjust to the influxes of sporadic traffic and utilize routes of minimum traversal time through the road network.

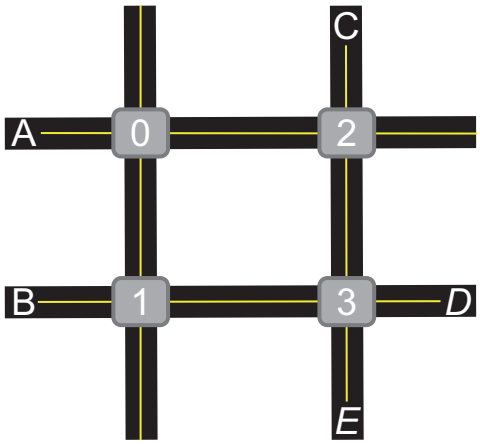


Fig. 3: 2×2 grid used for testing navigation policies. Intersection Managers are outlined in gray. The Traffic of Interest (ToI) travels from A to D along paths $\{A, 0, 2, 3, D\}$ and $\{A, 0, 1, 3, D\}$. Sporadic traffic flows alternate between entering at B and C and both exit at destination E. Paths taken by sporadic traffic typically follow the direct route (e.g. $\{B, 1, 3, E\}$ and $\{C, 2, 3, E\}$) but in some cases may take the longer routes $\{B, 1, 0, 2, 3, E\}$ and $\{C, 2, 0, 1, 3, E\}$. Random source-destination traffic enters on all roads except A, B, and C.

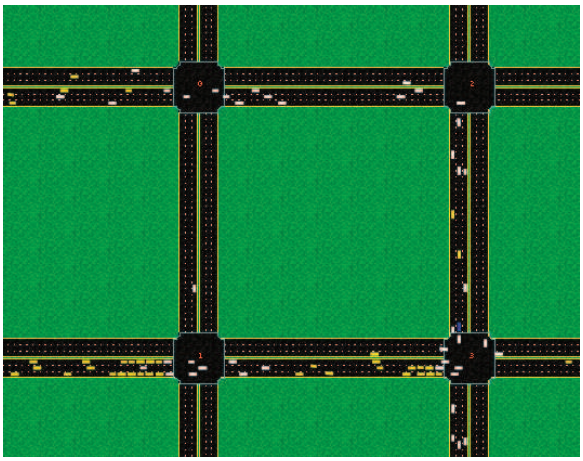


Fig. 4: Screenshot of the AIM GUI showing experimental scenario.

Using this scenario, we evaluate the following navigation policies for 1000 seconds of simulated time:

1) *Spatial A**: All vehicles navigate using a distance based A* search, in which the shortest distance routes are favored without regard to traffic congestion or delays. Ties between equi-distant routes are broken arbitrarily.
2) *ToI Time-based A**: Vehicles belonging to the traffic of interest (ToI) utilize the time-based A* navigation policy. All others navigate using Spatial A*.
3) *All Time-based A**: All vehicles navigate using the time-based A* algorithm.
4) *Omniscient*: In the omniscient scenario, the ToI has access to perfect future knowledge about incoming traffic flows, which is not available in any other scenario. This serves as an unachievable performance bound as it allows driver agents to instantaneously adapt to the influxes of sporadic traffic, exclusively utilizing the route of minimum conflict at all times.
5) *Non-Adaptive*: Intersection delay estimates are held constant, causing the ToI to fail to adapt to sporadic traffic. All other cars navigate with Spatial A*.

Navigation Policy	Global Thrghpt (cars)	ToI Thrghpt (cars)	Global Avg Delay (seconds)	ToI Avg Delay (seconds)
Non-Adaptive	2349 $\pm$ 12	733 $\pm$ 8.9	69.7 $\pm$ .65	133.5 $\pm$ 1.9
Spatial A*	2439 $\pm$ 11	980 $\pm$ 8.5	53.3 $\pm$ 1.4	55.1 $\pm$ 3.0
ToI Time-based A*	2591 $\pm$ 12	1008 $\pm$ 9.2	40.5 $\pm$ 1.6	26.9 $\pm$ 2.1
All Time-based A*	2550 $\pm$ 12	981 $\pm$ 7.1	50.3 $\pm$ 1.6	46.4 $\pm$ 2.3
Omniscient	2616 $\pm$ 14	1020 $\pm$ 7.9	36.8 $\pm$ 1.2	26.2 $\pm$ 2.1

TABLE I: Delay and throughput under different navigation policies. *Throughput* indicates the total number of cars who have reached their destinations before the simulation ends. *Average Delay* is the average, per-vehicle delay experienced. *Global* indicates the metric was applied globally while *ToI* designates the metric was applied only to the Traffic of Interest. $\pm$ denotes a 95% confidence interval.

Table I depicts the throughput and delays experienced by the traffic of interest as well as the overall system when driver agents employ different navigation policies. The time-based A* navigation policies significantly[3] reduce delays, both global and local to the ToI, in comparison to the Spatial A* navigation policy. Throughput increases as well. The non-adaptive policy performs far worse than others on account of its failure to adapt beyond the initial influx of traffic (e.g. use of a very high conflict route). These results in AIM highlight the ability of the intersection managers to dynamically influence driver agents' routes by providing up-to-date information about expected delays.

Traffic Signals

To compare the efficiency of multi-intersection AIM to more traditional traffic control mechanisms, we benchmark each of the navigation policies against an idealized implementation of traffic signals. These traffic signals compute the *cumulative wait* of cars on all incoming roads and give a green light to the incoming road with the highest cumulative wait (as well as to its *dual* – the adjacent, parallel road running in the opposite direction). As Equation 2 indicates, the cumulative wait of cars on a road r equals the sum of their individual waits, where the wait of a vehicle is the elapsed time since its arrival on r.

$$Cumulative\ Wait(r) = \sum_{vehicle_i \in r} wait(vehicle_i) \quad (2)$$

We design these idealized traffic signals to represent an upper bound on the efficiency of any real-world signal. Right turns on red were also allowed to increase traffic efficiency.

Table II depicts the results from the same set of experiments run in Table I only with optimal traffic signals replacing autonomous intersections. Side-by-side comparison of these tables indicates that delays, both global and local

[3]Throughout this paper, 30 trials were used for each experiment and significance is measured based on an unpaired t-test yielding two-tailed P values less than 0.0001

Navigation Policy	Global Thrghpt (cars)	ToI Thrghpt (cars)	Global Avg Delay (seconds)	ToI Avg Delay (seconds)
Non-Adaptive	2288 ± 15	983 ± 11	86.6 ± 2.1	60.6 ± 4.1
Spatial A*	2388 ± 13	909 ± 11	82.7 ± 1.7	93.3 ± 2.7
ToI Time-based A*	2396 ± 13	973 ± 9	75.2 ± 1.8	67.4 ± 2.3
All Time-based A*	2433 ± 11	929 ± 8	81.7 ± 1.9	85.1 ± 3.0
Omniscient	2420 ± 12	934 ± 12	79.9 ± 1.7	89.2 ± 2.8

TABLE II: Navigation Policies evaluated under traffic signals rather than autonomous intersections.

to the traffic of interest, rise significantly[3] in all cases (except Non-Adaptive) when optimal traffic signals replace autonomous intersections. Real-world traffic signals would almost certainly incur much higher delays.

Effects of Network Topology

As we have demonstrated, both the navigation policies of agents and the traffic control mechanism governing the intersections affect delays in a road network. An additional factor which we held constant throughout these experiments was the topology of the road network. One might hypothesize that once traffic reaches an equilibrium, adding additional roads to a network would reduce overall delays. While this is true in most cases, Braess' Paradox is a well-known phenomenon whereby opening additional travel options for the vehicles reduces the efficiency of the system as a whole. In the next section we observe an instance of this phenomenon associated with the experimental scenario discussed above.

Formally, Braess' Paradox is a phenomenon in which adding additional capacity to a network, when moving entities selfishly choose their route, can in some cases reduce the overall performance [7]. Historically, this phenomenon has been observed in South Korea, Germany, and New York City where new roads increased delays for all vehicles [8], [9], [10]. We observe an instance of Braess' paradox on the 2×2 grid depicted in Figure 3, in which the opening of the Northbound road connecting intersection 1 to intersection 0 increases global delays.

Similar to the scenario described previously, persistent traffic flows from A to D along paths $\{A, 0, 2, 3, D\}$ and $\{A, 0, 1, 3, D\}$ with rate 4500 vehicles/hour. Heavy traffic (9000 vehicles/hour) flows from B to E. When the Northbound $\{1, 0\}$ road is closed this traffic must flow along path $\{B, 1, 3, E\}$, but when this road is opened, traffic has the additional option of taking the longer $\{B, 1, 0, 2, 3, E\}$ route.

Braess' Paradox occurs when 1) despite the fact that cars minimize personal delay by selfishly choosing their routes, 2) overall efficiency declines when additional capacity is added to the network. To address the second point, we reference Table III which shows the significantly[3] increased delays and decreased throughput which result from opening the Northbound $\{1, 0\}$ road. Intuitively, when the Northbound road is opened, some vehicles belonging to the heavy, $B \rightarrow E$ traffic flow divert from the direct $\{B, 1, 3, E\}$ path to the longer $\{B, 1, 0, 2, 3, E\}$ path. This results in a high-conflict trajectory through intersection 3. Congestion forms first at the Westbound entrance to intersection 3, creating an incentive for more drivers to use the longer $\{B, 1, 0, 2, 3, E\}$ path. This results in even higher congestion through intersection 3. The cycle continues until an equilibrium is reached with roughly equal traversal times on both the long and short paths and high delays for all.

Road Status	Throughput (cars)	Avg Delay (seconds)
Closed	2021 ± 6.7	43.28 ± .24
Open	1821 ± 5.4	64.91 ± 1.22

TABLE III: Increased delays result from opening an additional road.

It might seem that Condition 1 wouldn't hold since selfish agents rarely forgo direct routes to their destinations. However, empirical evidence shows that the first cars to take the direct $\{B, 1, 3, E\}$ path deal with a delay of 52.35 seconds while their counterparts who take the longer but less-trafficked $\{B, 1, 0, 2, 3, E\}$ path experience average delays of only 38.64 seconds.

To summarize, in this section we have demonstrated that when driver agents selfishly choose their routes and additional capacity is added to the network in Figure 3, overall efficiency can suffer. Braess' paradox presents an interesting challenge to traffic engineers because it highlights the need for dynamic reconfiguration of network topology in response to changing traffic conditions. In the next section we explore one possible approach to this issue by introducing a methodology for reconfiguring network topology by dynamically reversing the flow of traffic along lanes.

Dynamic Lane Reversal

The reversal of traffic and reconfiguration of lanes via movable barriers has been implemented on several highways and bridges throughout the US, Canada and UK. Typically separation is achieved through barriers or markers which must be reconfigured to open and close lanes, often requiring considerable effort and time. As a result, lanes are usually only reconfigured twice a day to handle rush hour traffic. Barrier transfer machines (also known as zipper machines) can reduce the manual effort required to reconfigure a lane but are expensive, move at slow speeds (3-4 mph), and still require human drivers.

More recently, reversible lanes have been implemented using overhead signs which indicate the direction of a lane. Tested in Atlanta and Washington, DC, these overhead signs have potential to reverse the direction of lane far faster than barrier transfer mechanisms, but little is known about the safety of such quick reversals. Furthermore there have been reports of extensive driver confusion and malfunctions [11].

The AIM protocol affords the ability to safely reverse traffic along lanes on a fine-grained timescale. Without reliance on a physical barrier, an intersection manager may reverse the flow of traffic along a lane in approximately 15 seconds, allowing the road network to adapt quickly to unforeseen fluctuations in traffic.

In our prototype implementation of this idea, an intersection manager polls the demand on connected outlet roads (e.g. all traffic exiting on these roads must pass through the intersection) every two seconds and reverses the flow of traffic on a lane l belonging to an outlet road r if it detects enough demand detected on r_{dual}, the road flowing in the

Lane Reversal Status	Global Throughput (cars)	ToI Throughput (cars)	Global Avg Dly (seconds)	ToI Avg Dly (seconds)
Inactive	4300 ± 26	1431 ± 12	263.89 ± 4.4	348.23 ± 8.8
Active	4303 ± 24	1434 ± 12	163.34 ± 5.2	113.01 ± 4.1

TABLE IV: Reduced delays resulting from Dynamic Lane Reversal.

opposite direction. For example if l is an Eastbound lane, and the intersection manager determines the total demand for Westbound lanes is some constant factor N larger than the total demand for Eastbound lanes, it converts l into a Westbound lane (experimentally we used $N = 1.5$). Equation 3 expresses the condition to expand the capacity of r by reversing the flow on a lane belonging to its dual, r_{dual}.

$$demand(r) \geq N * demand(r_{dual}) \quad (3)$$

An intersection manager determines the demand for an incident road r by counting the number of vehicles who have expressed intent (in the form of a notification message or reservation) to cross the intersection and exit on road r.

After deciding to reverse traffic on a lane l, the IM adds l to a list of *clearing lanes*, and waits for all vehicles currently occupying l to clear. At this point the IM rejects all attempts to make reservations through the intersection which result in a vehicle entering l. When the intersection manager detects that all traffic on l has cleared, it removes l from the list of clearing lanes and begins accepting westbound reservations on l. The length of time required for l to clear depends on the length of the lane as well as traffic conditions at the exit of l, but typically requires less than 20 seconds.

To test the efficacy of dynamic lane reversals, we revisit the experimental scenario described previously (section *Experimental Evaluation*), this time increasing the number of incoming cars along all roads by changing the number of lanes on external road segments (e.g. $\{A, 0\}$,$\{B, 1\}$,$\{D, 3\}$, etc) from 3 to 5 in each direction (a 67% increase in traffic flow). Everything else is preserved between the two scenarios, including the number of internal road segments (e.g. those between intersections), which still consist of 6 total lanes. Thus, to accommodate 5 lanes of heavy traffic, two of the 3 lanes in the opposing direction should have their flow reversed, resulting in 5 open lanes in the direction of heavy traffic and 1 open lane in the opposite direction.

Table IV shows results with and without dynamic lane reversals active on a 1000-second simulation. While throughputs are comparable in both cases, the ability of Autonomous Intersection Managers to effectively adapt to incoming flows of traffic and open additional lanes results in significantly[3] reduced delays, suggesting that dynamic lane reversals, coordinated by Autonomous Intersection Managers, may enable existing road networks to successfully handle far more vehicles while maintaining resonable delays.[4]

CONCLUSION

As traffic congestion continues to increase in metropolitan areas throughout the world, the need for more effective ways to utilize current transportation infrastructure grows. In this work we explored, in simulation, several methods for avoiding congestion and reducing traffic delays which prove effective under a road network utilizing Autonomous Intersection Managers. Specifically, we examine methods, both large and small, of optimizing traffic flow through networks of autonomous intersections. On the micro level, we investigate different navigation strategies an individual vehicle could employ to minimize its delay through a transportation network and quantify the effect of each strategy on the overall flow of traffic. On the macro level, we recreate and explore Braess' Paradox, the implications of which highlight the need for newer and more intelligent traffic managers. To this end, we investigate the effects of dynamically reversing the flow of traffic along certain lanes in a network of roads and describe methods by which autonomous intersections could successfully identify which lanes need to undergo this change. Unlike prior work on contraflow network reconfiguration, which involves reversing entire roads in predominantly uni-directional flow scenarios such as city-wide evacuations [12], AIM grants the ability to quickly reverse individual lanes on the fly in response to rapidly changing traffic conditions. Consistent with the original AIM results, we demonstrate significant potential improvements compared to current control mechanisms when taking advantage of the capabilities of autonomous vehicles. In future work, we expect the implicitly parallel nature of the AIM framework to yield efficient solutions to the computational challenges inherent in larger road networks.

[3]Throughput is nearly unchanged in our experiments because we introduce a constant number of cars into the system. When we saturate the networks, throughput is significantly higher with dynamic lane reversal, but delays are similar (high in both cases).

REFERENCES

[1] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–656, March 2008.

[2] D. K. Merchant and G. L. Nemhauser, "A model and an algorithm for the dynamic traffic assignment problems," *Transportation Science*, vol. 12, no. 3, pp. 183–199, 1978.

[3] A. K. Ziliaskopoulos, S. T. Waller, Y. Li, and M. Byram, "Large-scale dynamic traffic assignment: Implementation issues and computational analysis," *Journal of Transportation Engineering*, vol. 130, 2004.

[4] S. Peeta and A. Ziliaskopoulos, "Foundations of dynamic traffic assignment: The past, the present and the future," *Networks and Spatial Economics*, vol. 1, pp. 233–265, 2001, 10.1023/A:1012827724856.

[5] S. T. Waller and A. K. Ziliaskopoulos, "A chance-constrained based stochastic dynamic traffic assignment model: Analysis, formulation and solution algorithms," *Transportation Research Part C: Emerging Technologies*, vol. 14, no. 6, pp. 418 – 427, 2006.

[6] P. Hart, N. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *Systems Science and Cybernetics, IEEE Transactions on*, vol. 4, no. 2, pp. 100 –107, july 1968.

[7] D. Braess, "Über ein Paradoxon aus der Verkehrsplanung," *Unternehmensforschung*, vol. 12, pp. 258–268, 1969.

[8] D. Easley and J. Kleinberg, *Networks*, p. 71, 2008.

[9] W. Knödel, *Graphentheoretische Methoden und ihre Anwendungen*. Springer-Verlag, 1969.

[10] G. Kolata, "What if they closed 42d street and nobody noticed?" *New York Times*, 1990.

[11] [Online]. Available: http://www.ajc.com/news/gwinnett/reversible-lane-on-u-124429.html

[12] S. Kim, S. Shekhar, and M. Min, "Contraflow transportation network reconfiguration for evacuation route planning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, pp. 1115 – 1129, 2008.