cARs

# v2v Communication for Augmenting Reality Enabled Smart HUDs to Increase Situational Awareness of Drivers

Work in Progress*

Andrew Dudley, Tathagata Chakraborti, Subbarao Kambhampati
Department of Computer Science, Arizona State University
Tempe, AZ 85281 USA
[addudley,tchakra2,rao]@asu.edu

## ABSTRACT

Inter-car communication has emerged in recent times as a viable solution towards reducing traffic hazards, with the recent US government mandate in favor of vehicle-to-vehicle communication highlighting the movement towards this direction in the automobile industry. However, questions remain as to how information from other cars can be effectively relayed to a driver, especially so as to not overload the driver with too much information. Meanwhile, a parallel thread of development in the space of Smart HUDs has shown the applicability of augmented reality to increase the situational awareness of drivers on the road. In this paper, we build on these threads of work and show how Smart HUDs can be an effective platform for projecting relevant information from surrounding vehicles in real time, and how an onboard AI component can avoid increased cognitive burden on the driver by determining when and what information to project based on its models of the driver and the surrounding environment.

## CCS CONCEPTS

• **Human-centered computing → Mixed / augmented reality**; *Interaction paradigms*; Human computer interaction (HCI); • **Computing methodologies → Multi-agent systems**; *Multi-agent planning*; *Intelligent agents*; *Cooperation and coordination*; Artificial intelligence;

## KEYWORDS

Inter-car Communication, Autonomous Cars, Augmented Reality, Smart HUDs, Mental Modeling, Human-Aware Planning.

**ACM Reference Format:**

Andrew Dudley, Tathagata Chakraborti, Subbarao Kambhampati. 2018. v2v Communication for Augmenting Reality Enabled Smart HUDs to Increase Situational Awareness of Drivers: Work in Progress. In *Proceedings of HRI 2018 Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interactions (VAM-HRI'18).* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

*First two authors contributed equally to the project.


*VAM-HRI'18, March 2018, Chicago, Illinois USA*

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

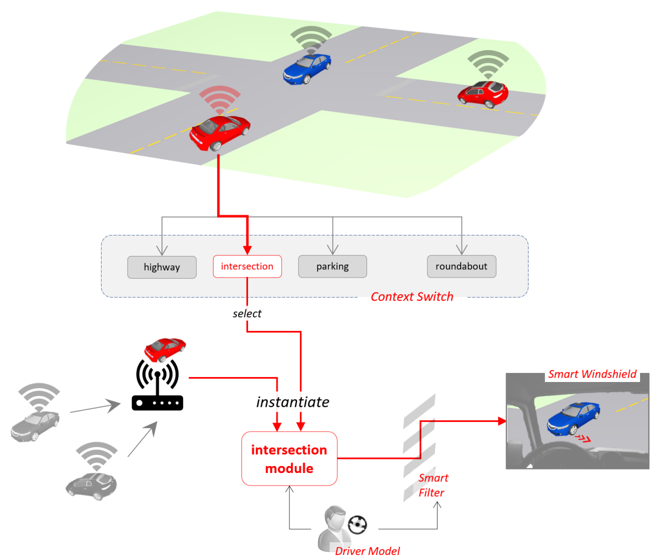


**Figure 1: An overview of the cARs – every car broadcasts their intentions which is processed by the car with the help of its model of the driver and the driving context to determine the relevant content for its DVI.**

## 1 INTRODUCTION

Vehicle-to-vehicle (v2v) communication technology has seen significant interest [12] over the last decade in a bid to make the roads safer both from the perspective of (fleets of) autonomous cars [14] on a shared network or for improving the situational awareness of individual human drivers [16]. This interest has also been reflected in the US government's recent mandate [1] [20] for making v2v communication a mandatory feature of automobiles.

However, v2v communication poses different interfacing challenges depending on the whether the receiving vehicle is autonomous or not. The former is likely to be able to process larger amounts of data in byte form while the latter can be easily overloaded with information and become distracted [1, 21] thus rendering the whole point of v2v communication moot. There has indeed been significant work [17] aimed at different forms of driver-to-vehicle interfaces (DVIs) and driver-to-infrastructure interfaces (DIIs), such as

[1]Although the previous mandate for v2v communication has seen some setbacks [7] of late, the general outlook towards this area remains positive with the Department of Transportation (DoT) [22] continuing to be one of the major players in the field.

with the help of vibrating seats or audio-visual cues [22] to alert the driver of impending danger. As [17] acknowledges, with the continued progress of augmented reality technology, heads-up displays (HUDs) might become the interface of choice for drivers. Recent trends [6, 8, 13, 23] certainly point towards this eventuality.

However, such visual cues are also likely to increase the cognitive burden on drivers if not designed effectively. Further, most of the existing works on v2v communication are aimed at specific kinds of alerts such as while navigating an intersection [2, 9] or when detecting possibility of collisions [8]. As such, the design of a general purpose DVI with v2v information sharing must account for the possibility of cognitive overload of the driver. We posit then that solutions for visual DVIs cannot be purely an exercise in interface design but also a matter of designing higher order reasoning capabilities of the software that drives the DVIs to be able to differentiate between information that is relevant to the driver and those that are not. This problem is exacerbated when all the data for the DVI is not generated in situ (as is the state-of-the-art), but is rather, as mentioned before, accumulated from information sent over from surrounding vehicles.

The scope of this project is thus not to develop efficient or secure protocols for v2v communication or design user-friendly interfaces for smart windshields. Instead, we focus on how the on-board software for the DVI can leverage AI techniques to reason over the information at its disposal and decide on what to make available to the driver, based on its model of the driver and its understanding of the current situation or context. Our starting point is then that v2v communication technologies already exist or are, in most cases, an eventuality as are smart HUDs or visual DVIs. We will build on this to show how such DVIs can benefit significantly from incorporating v2v communication in the most unrestricted sense (we make no assumptions on what information is feasible for real-time transmission with today's technology) and show how the on-board AI engine can reason with this information to realize a smart HUD that is truly smart. To this end, we will discuss the design of a general-purpose architecture for the reasoning or AI engine that drives the DVI, and show demonstrations of the technology on some typical use cases covered in the literature.

### 1.1 Typical Use Cases

We will now envision some scenarios that illustrate the usefulness of combining v2v communication technology with DVI design.

*Intention Projection.* Vehicles currently use a very limited collection of signals to inform surrounding vehicles of their intended actions. Human drivers often augment these signals by using hand gestures to communicate, but even these are not sufficient for many scenarios, and furthermore, are not available for autonomous cars. For instance, there is no unique signal for making a U-turn at an intersection, even though this information would be very relevant to a car in the adjacent street making a right turn, as well as to a car heading the opposite direction on the other side of the intersection making a right turn. Or consider driving down a narrow road when suddenly a car pulls out of parking, and similar issues with maneuvering in blind-sighted corners – this information can be relayed beforehand, even more so if the parked car is autonomous and has thus already decided on its intention to pull out. Perhaps the most well-studied scenario for v2v communication is that of collision avoidance at intersections [15] – while usually this is a simpler problem since it is a matter of following the turn-taking rules as outlined in the driver's manual, the problem is complicated often by errors in human judgment, or for autonomous vehicles, the absence of social cues or even mistakes by the automation. Thus, in navigating an intersection, the DVI might have to reason about what is wrong with the current situation than what is right, in order to alert the driver, thus complicating the problem further. Providing drivers with visual projections of the information (e.g. *intentions* [4] of nearby cars) that is relevant to their intended path could enable them to make better informed decisions, decreasing the risk of a collision without increasing their cognitive load.

*Information Sharing.* Human drivers make decisions based on what they are able to perceive, yet there are several scenarios we encounter while driving where impending dangers are obstructed from view – e.g. a car slamming on its breaks two vehicles ahead, a yellow light hidden behind a semi-truck, a cyclist approaching from a blind spot, etc. With access to information from nearby vehicles, Smart HUDS could augment the vision of the driver with pertinent details of their occluded surroundings.

During the project we explore these scenarios from the perspective of the reasoning capabilities of the AI-enabled DVIs. Of course, these scenarios become more nuanced as we delve into the details of whether the participating cars are autonomous or the details of the driver model itself, as well as the modes of interaction enabled by the environment and supported by the reasoning engine. These are also necessary issues that must be addressed so as to mitigate the additional cognitive burden on the human driver while also improving the safety of driving, which makes this a significant and challenging problem going forward.

## 2 THE cARs SYSTEM

In the following, we describe briefly describe the components of the proposed system cARs (illustrated in Figure 1). Every car, irrespective of whether it is autonomous or not, or whether it has on-board cARs, broadcasts their "state" information (e.g. velocity, location, intended route, etc.) and is received by all the cars within a certain range either via direct peer-to-peer communication or mediated by traffic infrastructure such as smart intersections [3]. This (broadcast followed by processing at the receiving end) is a more practical way of sharing information than, say, determining upfront what information to share with whom. In the case of the latter, a car will need to estimate the state of its surrounding cars to determine relevance of information, which is a much harder task.

The on-board DVI has a *context switch* which it uses based on the current situation the car is in (e.g. maneuvering at an intersection versus a roundabout, or driving in a parking lot). With its current model of the driver (e.g. where they are headed, attention model, etc.) and the incoming information from surrounding vehicles, the DVI spawns an instance of one of the *interaction modules* inside its context switch and solves this instance to get the optimal course of action (CoA) of its parent car as well as the cars around it. This enables the smart filter to process the computed CoA to determine information that is relevant to its parent car. It forwards this filtered content to the DVI (in this case, a smart HUD). This process repeats during the entire driving period or as long as the DVI is enabled.

## 2.1 Assumptions

We will now describe our implementation of cARs (c.f. Figure 3) and assumptions made in that process.

- Context Switch – We assume the context switch can correctly identify the current environment and spawn the correct problem instance. This is not a strong assumption given the state of the art in vehicular perception and v2v communication technology. Thus, in this project, we only concentrate on the smart filter (c.f. Section 2.2) as introduced in Figure 1.
- Discrete / Shared World – We also discretized the environment in order to run the reasoning engine in reasonable time. This is inspired by similar discretization approaches in previous literature [14] to make inter-vehicular information sharing tractable. We also assume that the vehicles have a shared understanding of this map [14] in order to reason over each others' intentions.
- We also assume that the high-level intentions of the cars are known to the car – e.g. if the car is going to take a left turn at an intersection or if the car is going to come out of a parking space, etc. This is, in fact, an interesting assumption (and becomes more nuanced depending on whether a car is autonomous or not) and worthy of further explication –
  - For human-drive car, the intention information is not readily available. However, this can be derived from the status of the car either explicitly (e.g. driver turns on left-turn signal or starts up engine in a parked car) or even implicitly (e.g. driver turns into a left-turn only lane).
  - For an autonomous car, the future intentions are easily derivable from its trajectories. Thus, arguably, the cARs framework could become more and more useful (i.e. it can be wary of more higher level intentions further down the line) as more and more cars become autonomous with respect to the parent car.

  In either case, we assume that the higher-level intent is already provided to the onboard DVI.

## 2.2 The Smart Filter

Given the above information, the smart filter proposed in Figure 1 is realized using the following communication protocol.

- **On every car (broadcast, publisher node)** –

[1] Compile the current feedback from the environment (location of the car, surrounding area, desire/intent) into a planning problem [10].

[2] Solve this, and repeat.

[3] The solution, i.e. plan or aggregated course of action in the next few time steps, is now broadcasted to all surrounding cars within the communication range.

- **On every car (in situ, receiver node)** –

[1] Upon receiving a ping from a nearby car (this contains the intent and relative position on the shared map of the surroundings as computed above on board the car sending the ping) every car compiles these as observations to be included in its own planning problem in the framework of [18, 19].

[2] This compilation produces a new planning problem the solution to which *must enforce the provided observations*. This new planning problem is now solved and the solution is compared (for equality or cost, or specific features as required by the designer) to the one from the previous step where the self-plan was computed without considering the intentions of the cars around the parent car. If they do not match, then these intentions must have been conflicting, and hence may be useful information to the driver, if they are the same then the intentions of the other cars do not matter to the parent car. This is following Theorem 7 in [18].

[3] The step is asynchronous and runs every time a ping is received from a nearby car, whereas the broadcast step repeats continuously. The output of the check about determines if the intent send over from a car needs to be forwarded by the DVI to be displayed on the SmartHUD.

## 2.3 Discussion

It is worth having a discussion at this point about the trade-offs made in the above communication scheme, i.e. *what did we gain out of casting the interaction modules in Figure 1 as planning problems? And what do we lose?*

*Gains.* The representation allows us a slick way to not only formulate different forms of interactions encountered on the road (note: given these are already well established rules, a declarative rule based system provides an ideal approach towards this) but also allows us to call upon off-the-shelf techniques to provide a variety of support functionalities.

- Consider the following domain – https://goo.gl/bUxLQy which fleshes out the full scope of the intersection domain in much more detail, including rules for turn taking, queuing, etc. We can use the exact same framework, and keep adding such newer modules, to enable newer and newer forms of interaction in the cARs framework.
- Further, imagine now that the parent car itself is trying to enter the intersection out out turn. The on-board planner that is computing the intentions of the self, can easily catch this immediately using plan validation [11] techniques. This opens up a whole set of possibilities on projecting intentions that are not only conflicting in the future but are also, for example, against the law – e.g. in the above case, the SmartHUD will display back the driver of the parent car that they should stop. We plan to expand to these scenarios (by integration of the interaction modules with VAL [11]) in the project going forward.

Thus, the planning representation allows us a rich language to reason over the driving domain.

- However, planning (of the form used here) itself is known to be PSPACE-complete and thus we are in danger of taking a computational hit. Luckily, since these interaction modules

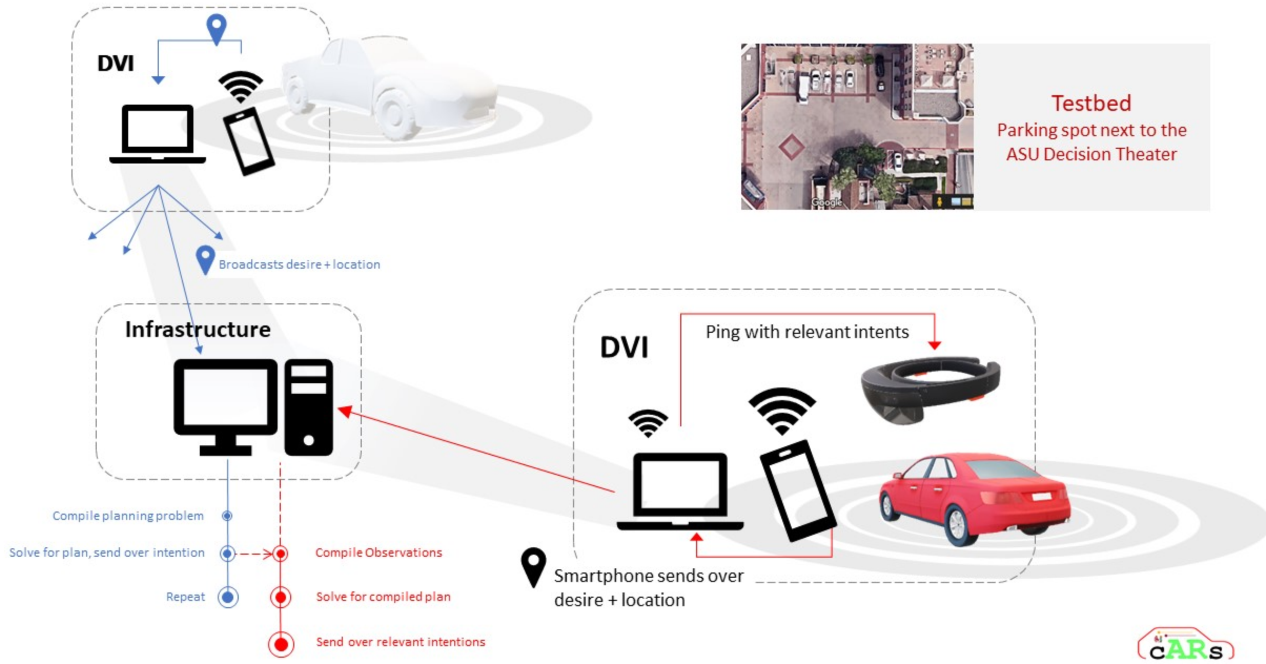


**Figure 2: The flow of control in the current implementation of cARs. Each car runs two parallel threads - one where they compute a limited rollout of their future actions and broadcast it to surrounding cars; the other where they compile incoming intentions from surrounding cars and filter out the ones that are not relevant before forwarding them for display in the heads up display. The details of this process are provided in Section 2.2.**

are small sub-domains (such as roundabout, intersection, lane change, etc.) spawned repeatedly, we avoid getting into the weeds of minimizing computational burden. Indeed, the domains talked about in the paper solve in 1 sec and can thus the planning instances on the self can be re-spawned almost every second. The compiled domains also operate similarly, but are asynchronous. Note here that the compiled domains have to be run for observations from each car, but these can be run parallel on independent threads since they do not affect each other, giving us the same *amortized* runtime.

*Loss.* The classical planning formulation, while allowing for an expressive vocabulary for representing complex interaction constraints, in discrete space, also suffers from the nature of that discretization. In the current form, we have essentially turned the environment into a dynamic grid world shared among the cars. Further, we have assumed communication at the symbolic level which may not always be possible - e.g. intent as a general high level goal is derivable from the left-turn or backing signals switched on in a car, but general trajectories are not. It is, of course, not necessary to account for all sorts of interactions in the same framework, and the proposed solution does account for a wide variety of them.

*Lesson.* One of the lessons learned in dealing with the hardware issues (latency and localization in the real-time / real-world implementation of cARs) is that it may well be more useful to isolate the visualization challenge with the reasoning challenge in so much as the information required for them is concerned. Note that the SmartHUD can only display with respect to what is viewable through the windshield and thus the localization information from the broadcasted pings are not necessarily useful there (might be easier to do this with vision, if the pings can be tokenized to match objects in the scene). This can allow for more flexible representations going forward. Especially, issues with the GPS accuracy and the HoloLens map (which, though to be fair, is not built to work outdoors and is not an issue for the scope of the problem in general since the HoloLens is just a proxy for the SmartHUD) proved to be troublesome during the project.

## 2.4 Implementation

**GPS and location discretization.** The DVI for each vehicle in our experiment was simulated in part by utilizing real-time GPS information from an Android cell phone. We developed an Android application which receives the latitude and longitude coordinates of the device and then publishes them - along with the Cartesian-discretized coordinates - to a centralized RabbitMQ messaging

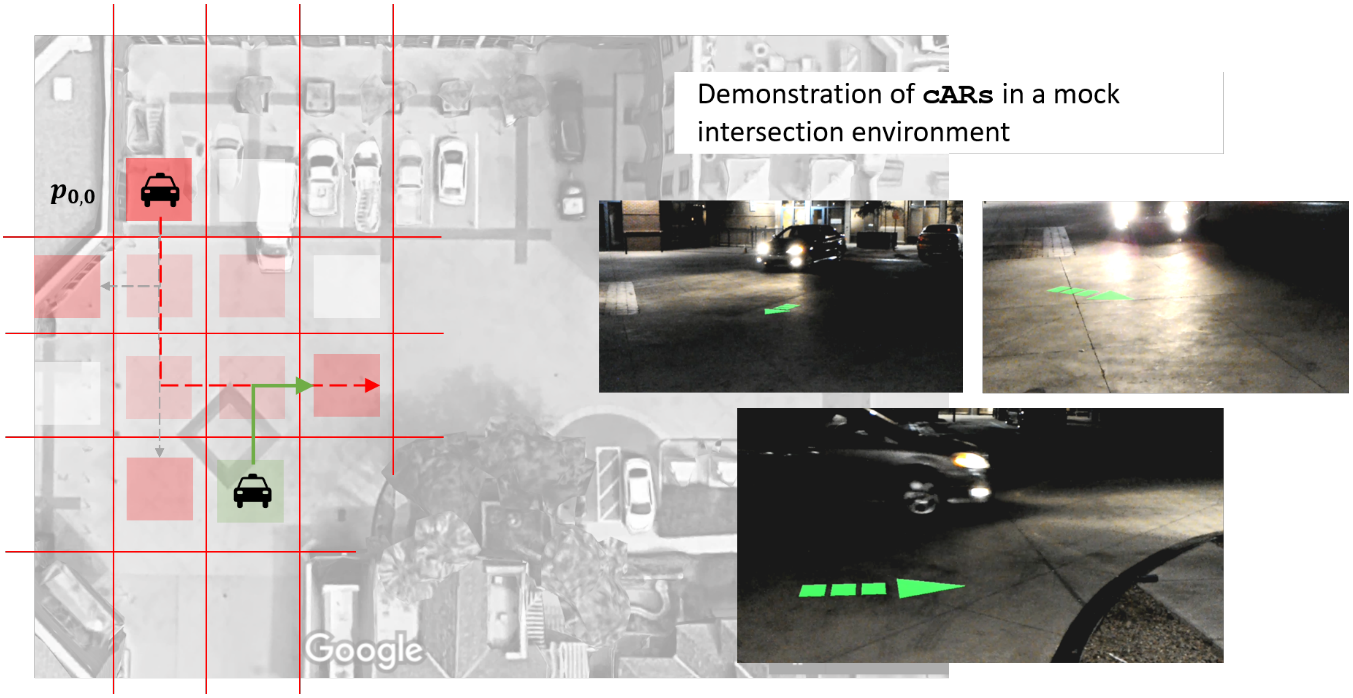


**Figure 3: A demonstration of a simple intersection scene in the ASU parking lot – the car in the green spot has the first person view and is intending to take a right turn. The car in the red spot can have three possible intents – go straight, take a right as well, or make a left turn. Note that only the last intent conflicts with the parent car. Our cARs framework filters out this information, as described in Figure 2, and presents the future intent of the other car to the in the parent car only if the former is going to take a left turn. This is seen here in terms of the projected arrow around Andy pretending to be the other car (a HoloLens simulates the SmartHUD on the parent car). A video of the demonstration can be seen at** https://goo.gl/pAzuzn**.**

server. The discretization can be performed manually by marking the centroid of each location, or by reading in pre-computed mappings created with Google Maps.

**Communication.** RabbitMQ messaging brokers were configured for each vehicle to enable vehicle-to-vehicle communication. Each broker builds a queue of messages from the surrounding vehicles, and then initializes the planning and compilation procedures discussed above. In our final implementation, instead of using separate messaging servers for each vehicle, we decided to use a single centralized server, which receives the locations of - and handles the planning for - all vehicles in each scenario. This was done for convenience, as it enabled us to add more cars to each scenario by only adding more cell phones, without the need of additional laptops. Without the use of ad-hoc communication, we were also limited by the port security configured on the Local Area Network at the scenario site, so using a centralized server greatly simplified the networking requirements. *Note that this design in effect converts our vehicle-to-vehicle solution into a vehicle-to-infrastructure solution.*

**Intention Projection.** Simulation of a Smart HUD was accomplished using a Microsoft HoloLens, which enabled us to augment the driver's view with the relevant intentions of nearby vehicles using a virtual arrow. The information required by the HoloLens to generate each projection is accessed via a RESTful API, which provides a JSON object that encodes each intention of the scenario.

The HoloLens uses a mapping of the environment that corresponds to the same discretized space defined previously in order to properly draw the arrow relative to the driver's position and orientation. This mapping is done separately, and is more typical of how the HoloLens works in practice, more than how a SmartHUD is implemented [5] (where the projection is done in situ relative to the view from the windscreen).

## 3 DEMONSTRATION

In this section, we will provide a demonstration of cARs in action in a mock intersection domain. The files for the interaction module are provided at https://goo.gl/zNYVqj – this is a simplified version of the one described in Section 2.3, without the queuing rules and turn taking. The target situation, illustrated in Figure 3, plays out in the ASU parking lot next to BYAC, as a mock setting for an intersection without traffic lights. Here, the parent car is equipped with the AR-enable SmartHUD (i.e. the HoloLens) and is trying to take a right turn at the bottom right corner of the image. The car in the opposite corner in the intersection can, of course, go either of three ways – i.e. go straight, take a right or take a left. These paths are shown with arrow in Figure 3. Note that only one of these intentions are conflicting.

In the cARs framework, both the cars internally solve their immediate planning problems, compute their intentions, and broadcast them. In the image, these intents are shown with the red and green arrows (the other possible ones from the other car are shown in green). This is the outcome of the first step of the protocol explained in Section 2.2. The on-board DVI, here the server, upon receiving intentions from other cars, compiles these observations into a new planning problem and solves it asynchronously. As explained in Section 2.2 (second asynchronous step), if at any time, the solution from the compiled problem requires the parent car to change its future course of action, that means the intention of the other car is conflicting with its own, and thus must be forwarded to the driver. In the current situation, this is done by placing an arrow next to the car with the conflicting intent, thus illustrating its immediate planned motion, so as to alert the driver of a possible collision.

The insets in Figure 3 provide snapshots of this interaction in progress (with us pretending to be the cars, for now). A video of the same can be viewed at https://goo.gl/pAzuzn. The arrow of course does not show up if the other "car" was taking a right turn or just moving forward on the same road.

## ACKNOWLEDGMENTS

This research is supported in part by the AFOSR grant FA9550-18-1-0067, the ONR grants N00014161-2892, N00014-13-1-0176, N00014-13-1-0519, N00014-15-1-2027, and the NASA grant NNX17AD06G. Chakraborti is also supported by the IBM Ph.D. Fellowship.

## REFERENCES

[1] Roberto Baldwin. 2017. Driving with AR glasses may be information overload. (2017). https://www.engadget.com/2017/09/26/driving-with-ar-glasses/ engadget.

[2] F. Basma, Y. Tachwali, and H. H. Refai. 2011. Intersection collision avoidance system using infrastructure communication. In *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. 422–427. https://doi.org/10.1109/ITSC.2011.6083007

[3] Anthem Test Bed. 2017. Connected Vehicles Program. (2017). http://www.maricopa.gov/640/Connected-Vehicles-Program MCDOT SMARTDrive Program.

[4] Tathagata Chakraborti, Sarath Sreedharan, Anagha Kulkarni, and Subbarao Kambhampati. 2017. Alternative Modes of Interaction in Proximal Human-in-the-Loop Operation of Robots. *CoRR* abs/1703.08930 (2017). arXiv:1703.08930 http://arxiv.org/abs/1703.08930

[5] Daqri. 2017. The Brand New Way to Drive. (2017). https://medium.com/@DAQRI/the-brand-new-way-to-drive-6739408ad1dd Medium.

[6] Daqri. 2018. Smart HUDs. (2018). https://daqri.com/products/smart-hud/

[7] Zac Estrada. 2018. US set to drop proposed vehicle-to-vehicle communications mandate. (2018). https://www.theverge.com/2017/11/1/16592704/vehicle-to-vehicle-communications-mandate-trump Verge.

[8] Ford. 2017. Ford, Virginia Tech go undercover to develop signals that enable autonomous vehicles to communicate with people. (2017). https://media.ford.com/content/fordmedia/fna/us/en/news/2017/09/13/ford-virginia-tech-autonomous-vehicle-human-testing.html

[9] MR Hafner, D Cunningham, L Caminiti, and D Del Vecchio. 2011. Automated vehicle-to-vehicle collision avoidance at intersections. In *Proceedings of world congress on intelligent transport systems*.

[10] Malte Helmert. 2006. The fast downward planning system. *Journal of Artificial Intelligence Research* 26 (2006), 191–246.

[11] Richard Howey, Derek Long, and Maria Fox. 2004. VAL: Automatic plan validation, continuous effects and mixed initiative planning using PDDL. In *Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on.* IEEE, 294–301.

[12] Will Knight. 2017. A simple wireless technology promises to make driving much safer. (2017). https://www.technologyreview.com/s/534981/car-to-car-communication/ MIT Technology Review.

[13] Swapna Krishna. 2017. Self-driving cars are safer when they talk to each other. (2017). https://www.engadget.com/2017/06/24/self-driving-cars-mcity-augmented-reality/ engadget.

[14] Swarun Kumar, Lixin Shi, Nabeel Ahmed, Stephanie Gil, Dina Katabi, and Daniela Rus. 2012. Carspeak: a content-centric network for autonomous driving. In *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication.* ACM, 259–270.

[15] Shlomo Zilberstein Kyle Hollins Wray, Stefan J. Witwicki. 2017. Online Decision-Making for Scalable Autonomous Systems. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*. 4768–4774. https://doi.org/10.24963/ijcai.2017/664

[16] Michael Maile, Qi Chen, G Brown, and Luca Delgrossi. 2015. Intersection collision avoidance: From driver alerts to vehicle control. In *Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st*. IEEE, 1–5.

[17] James A Misener. 2010. *Cooperative intersection collision avoidance system (CICAS): Signalized left turn assist and traffic signal adaptation.* Technical Report.

[18] Miquel Ramırez and Hector Geffner. [n. d.]. Plan recognition as planning.

[19] Miquel Ramırez and Hector Geffner. 2010. Probabilistic plan recognition using off-the-shelf classical planners. In *Proceedings of the Conference of the Association for the Advancement of Artificial Intelligence (AAAI 2010)*. 1121–1126.

[20] Stephen Shankland. 2017. US to push for mandatory car-to-car wireless communications. (2017). https://www.cnet.com/news/us-to-push-for-mandatory-car-to-car-wireless-communications/ CNET.

[21] David L. Strayer. 2015. Up to 27 seconds of inattention after talking to your car or smartphone. (2015). https://unews.utah.edu/up-to-27-seconds-of-inattention-after-talking-to-your-car-or-smart-phone/ University of Utah.

[22] Intelligent Transportation Systems. 2018. Connected Vehicle Videos. (2018). https://www.its.dot.gov/communications/video.htm United States Department of Transportation.

[23] Timothy Torres. 2015. Mini Cooper AR Glasses Give Drivers X-Ray Vision. (2015). https://www.pcmag.com/article2/0,2817,2482386,00.asp pcmag.