A SIMULATOR FOR TRADING TRAFFIC PRIVILEGES BY SELFISH DRIVING CARS

Zhan Tu
Purdue University
W. Lafayette, IN, USA

Anastasios Dimas
Rutgers University
Piscataway, NJ, USA

Mehmet Necip Kurt
Columbia University
New York, NY, USA

Anastasia Mavrommati, Pieter J. Mosterman, Akshay Rajhans, Roberto G. Valenti
Advanced Research & Technology Office
MathWorks
Natick, MA 01760, USA

ABSTRACT

Connected autonomous vehicles are an important class of cyber-physical systems that are expected to have a major impact on society. Connectivity and autonomy in next-generation automobiles can be leveraged to improve safety and efficiency of our transportation systems. This paper presents a novel approach for incorporating individual driving preferences of vehicles in a computational framework to allow dynamic assignment and transfer of right-of-way privileges between cars as they navigate contested road segments. Dynamic priorities based on time of arrival estimates and positions in queues are used to unambiguously identify the owners of right-of-way privileges to conflict zones at any given time. A mechanism for transferring the privileges from a unique rightful owner to another car, possibly incentivized by using a shared currency, is proposed. A simulation framework using MATLAB® is developed to enable rigorous study of this mechanism across tens of thousands of simulations.

Keywords: connected autonomous vehicles, intelligent transportation, intersection management, driving privileges, right of way.

1 INTRODUCTION

Cyber-physical systems (CPS) comprise open and interconnected systems that sense, interpret, and control their physical environment. Connected and autonomous vehicles (CAV) represent a key application domain of CPS given the extent of impact it is expected to have on our society. Connectivity and autonomy are increasingly being used to improve safety and efficiency in the transportation domain. This paper presents one such novel approach to intersection traffic efficiency based on individual objectives. The approach makes use of (i) an unambiguous quantitative notion of right-of-way privileges for accessing an intersection and (ii) a mechanism for trading and transferring these privileges among each other.

1.1 Related Work

Simulation of traffic flow has been extensively studied in the literature. Some representative examples include simulation of human-operated cars in T junctions (Paruchuri et al. 2002, Kamrani et al. 2014), signalized intersections (Ganiyu et al. 2011, Liang et al. 2018), and other road topologies such as merges (Antoniotti et al. 1997, Ito et al. 2018).

The most extensive body of work that leverages the communication ability of CAVs to resolve conflicts and ensure safe intersection traversal takes an optimization approach, that is, it aims to optimize a common goal across a population of vehicles. Examples of such techniques include using nonlinear optimization possibly in combination with heuristics (Kamal et al. 2014, Lee and Park 2012), decentralized model predictive control (Qian et al. 2015), convex optimization from the perspective of the traveller (Dai et al. 2016), and mixed

integer linear programming (Fayazi and Vahidi 2018). Other references formulate a scheduling problem to be solved using branch-and-bound search methods (Li and Zhou 2017), particle swarm optimization (Guney and Raptis 2018), and graph-based modelling for centralized intersection management (Lin et al. 2019). Further related work (Xu et al. 2018) transforms the two-dimensional vehicle clustering problem at an intersection into a one-dimension conflict-free virtual platoon system, while other work (Ci et al. 2009) employs Monte Carlo Markov Chains. These solutions can be thought of as socialist policies that attempt to minimize the *price of anarchy* (Zhang et al. 2018).

The concept of using a shared currency for intersection traffic management has been previously adopted with different purposes. A centralized intersection manager charging vehicles an amount of time-tokens corresponding to their impact on other drivers has been proposed for maximizing social welfare by Sayin et al. (2019). In contrast, the approach presented here considers currency transactions directly between vehicles to transfer privileges. A game-theoretic framework to study Nash equilibria of selfish agents trading the use of resources has been studied by Censi et al. (2019), where ownership resolution is not considered and only pairwise transactions are allowed. The presented work tackles the ownership disambiguation question based on vehicle dynamics and queue priorities, and builds a mechanism that supports trading privileges to an arbitrary number of other vehicles.

1.2 A Selfish Driving Approach to Traffic Management

There are two main challenges with the optimization approaches. First, population-wide optimization requires every vehicle that is part of the population to partake in the optimization. There will, however, be vehicles that lack connectivity in traffic for quite a while and this prevents optimal behavior. Second, people do not always share the common criteria for the optimization objective, and they may change depending on their particular circumstances, e.g., time spent traveling might outweigh fuel consumption when in a hurry.

This work takes an individual tack that can be considered to be based on capitalist principles. Each actor is enabled to optimize their individual objective. Traffic privileges (e.g., the right of way at an intersection, available space for a merge maneuver, and unobstructed use of a highway lane) can be directly traded with other actors. If the trading actors come to an agreement, the privilege is transferred from the owner to the requester. In return, the original owner is acknowledged by transfer of an agreed upon amount of common currency (monetary or otherwise) from the requester (the Buyer) to the original owner (the Seller). As such, actors accumulate a balance of currency that can be used for future situations where priority access to traffic privileges is important to the individual. The documented solution is incremental in that it adds a layer of functionality on top of existing traffic behavior (displayed by humans).

This paper presents a simulation tool developed in order to experiment with the capitalist notion of trading the right of way at a T junction. In the simulated scenarios, vehicles may enter from three locations (without loss of generality called East, West, and South) and enter the respective road segments. The road segments meet at a shared section, the *conflict zone*, where the right of way is assigned according to the rules of an 'all-way stop' (stop signs for each road segment). As such, the car that comes to a stop at the conflict zone first obtains the right of way. When there are lines of cars that have stopped at the junction the cars that are second in each line obtain the right of way in a round robin fashion (i.e., sequentially based on when the lead car traverses the conflict zone).

The objectives of the simulator are threefold:
1. Determine the abstraction and entities that allow a computational form of the problem. This includes the definition of privileges as conflict zones, the notion and extent of priorities, a characterization of individual objectives, vehicle roles, and trade preferences.
2. Create the problem structure as an interaction protocol combined with computational simulations. This includes when to assign priorities, how to assign priority, which actors to assign priority, when to allow priority transfer, and how to support trades.
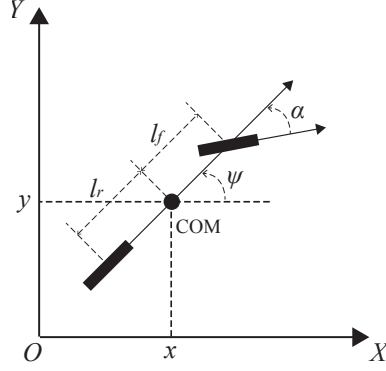
Figure 1: Bicycle model of a car.

3. Provide insight into the emerging behavior. This includes an understanding of viability, whether the priority assignment is intuitive, and analysis of performance simulation results.

Note that in a physical scenario the rules of engagement allow for interpretation. For example, the location where to stop typically differs per driver. Also, there typically is some exceptional behavior that must be treated properly. For example, a driver may not come to a stop at all and mistakenly claim the right of way. Now in violation of the all-way-stop rules, the other actors must adopt exceptional behavior that at the very least is safe.

Section 2 introduces the parameterized mathematical model of car dynamics used in the simulation framework. Section 3 presents a quantitative framework based on priorities to unambiguously and dynamically identify the right-of-way privileges to conflict zones in a deterministic order of precedence. Section 4 proposes a framework for trading and transferring such privileges based on quantified driving preferences and facilitated by the use of a shared currency. Section 5 discusses future opportunities to improve on this work and concludes the paper.

2 SIMULATION OF CARS TRAVERSING A T-JUNCTION INTERSECTION
Car dynamics and control models are essential for the simulator.

2.1 Car Dynamics
The car dynamics are modeled using the well-known bicycle model (see Fig. 1). The model parameters are based on the vehicle dynamics model in the Vehicle Dynamics Blockset™ (MathWorks® 2019b). The angle of the current velocity of the center of mass (COM) with respect to the longitudinal axis of the car is calculated as:

$$\beta = \arctan(\frac{l_r}{l_r + l_f} \tan \alpha), \tag{1}$$

with α the front wheel steering angle, and l_f and l_r the front and rear axle offsets from the COM. The car dynamics are then represented by a system of four first order differential equations:

$$\begin{aligned}
\dot{x} &= v \cdot \cos(\psi + \beta), \\
\dot{y} &= v \cdot \sin(\psi + \beta), \\
\dot{\psi} &= \frac{v}{l_r} \sin \beta, \\
\dot{v} &= a,
\end{aligned} \tag{2}$$

with x and y the coordinates of the COM with respect to an inertial frame (O, X, Y), v the instantaneous velocity of the car along its trajectory, and a the acceleration of the COM along the linear velocity direction.
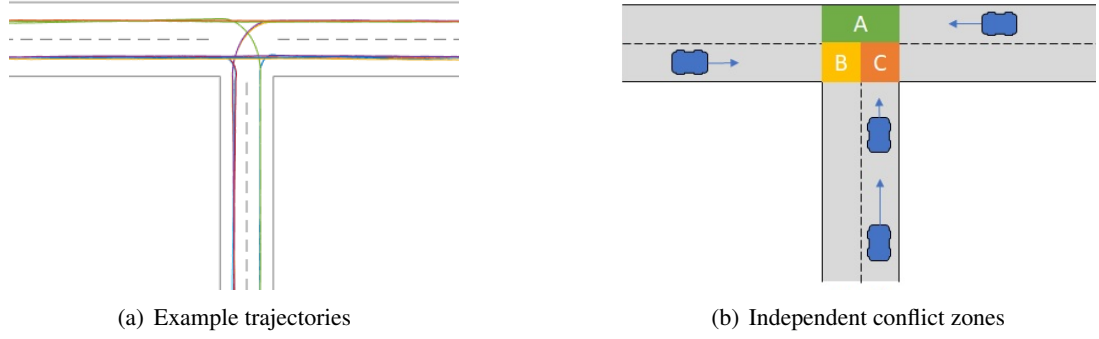
(a) Example trajectories   (b) Independent conflict zones

Figure 2: T junction considered in this paper.

In the above system the steering wheel angle α and the linear acceleration a are the command inputs and therefore used to control the yaw and velocity of the car. Such control is achieved by adopting PID (proportional–integral–derivative) controllers. The PID control inputs are computed with a sample time Δt and remain fixed for the duration of Δt. A variable step solver (`ode45` in MATLAB® (MathWorks® 2019a)) is used for solving the dynamics for each time interval Δt.

2.2 Reference Trajectory Generation

For each of the cars, a heading reference is determined such that the vehicle follows the road from a given origin to a given destination. For a car with its origin in the South and destination in the West (i.e., the car makes a left turn at the junction), the heading reference starts at $\psi_{ref} = \pi/2$, indicating that the trajectory is 90° rotated from horizontal. In the West road segment where the car ends its trajectory, $\psi_{ref} = \pi$ indicating that the positive direction is right to left and opposite of the reference orientation from left to right. The heading in the conflict zone to reach the West road segment from the South road segment is computed as $\psi_{ref}(t) = \pi/2 + \phi(t)$, where ϕ is the desired turning angle. $\phi(t)$ calculated as the ratio $\delta(t)/\rho$, where ρ is the turning radius, and $\delta(t)$ is the distance that the car is into the curve, computed as

$$\delta(t) = \delta(t - \Delta t) + v(t) \cdot \Delta t. \tag{3}$$

The trajectories for vehicles with different source and destination road segments are computed analogously. Figure 2(a) plots such trajectories on the intersection geometry of a T junction.

2.3 Reference Velocity Generation

The reference velocity, v_{ref}, of a car changes at distinct times as it moves along its trajectory from origin to destination. When far away from the conflict zone, the reference velocity is set to a default reference velocity. When the distance to the conflict zone falls below a value that allows coming to a stop given the current velocity ($\frac{v}{0.05} \leq \text{distance} \leq \frac{v}{0.035}$), the reference velocity is reduced to start deceleration. The deceleration profile is determined by first computing a stopping measure

$$\sigma = -\sigma_{min} + \sqrt{x^2 + y^2} - \frac{v}{2}, \tag{4}$$

based on a minimum distance to keep, σ_{min}, a distance norm, and the velocity to decelerate from. A scaling factor is used to then compute the deceleration profile as 0.135σ. Once a car has obtained the right of way, its velocity reference is reset to the default value till the car completes its trajectory.

When a car is following another car, if the ℓ_2 distance between the lead car and the follower car falls below a safe distance, the reference velocity of the follower is adjusted to avoid a rear-end collision. Two cases

are distinguished: (i) the follower car is on a straight road segment and (ii) the follower car is in the conflict zone, making a turn.

In the first case, the stopping measure is computed as

$$\sigma = -(\sigma_{min} + \frac{v}{4} + |p - p_{lead}|), \tag{5}$$

where p is the position of the follower car and p_{lead} is the position of the lead car along the direction of motion. From this stopping measure, the reference velocity of the follower car is computed as

$$v_{ref} = \begin{cases} v_{lead} - 20\sigma^2 & \sigma \leq -1, \\ v_{lead} - 20|\sigma| & \sigma \leq 0, \end{cases} \tag{6}$$

where v_{ref} is the reference velocity of the follower car. In the second case, the reference velocity of the follower car is set to a slightly lower value than the velocity of the lead car $v_{ref} = v_{lead} - 0.1$.

3   RIGHT-OF-WAY DISAMBIGUATION FOR CONFLICT ZONES

For cars to successfully travel from the source to the destination of their prescribed trajectories, they must cross the conflict zone without colliding with other cars.

3.1 Junction Conflict Zones

The first step in formulating the computational problem is to identify areas where collisions must be prevented by restricting access of the area to at most one car at any given time. Restricted access is achieved by assigning the right of way for such areas. Independent possible collision points can be identified in three areas as shown in Fig. 2(b). In area *A* East vehicles may collide with South vehicles that make a left turn. In area *B* West vehicles may collide with East vehicles that make a left turn. In area *C* South vehicles may collide with West vehicles that continue straight. Consequently, the overall conflict zone of the junction is divided into three conflict zones *A*, *B*, and *C* and the right of way for each of these conflict zones can be assigned independently of each other.

3.2 Priorities for Quantification of Right-of-Way Assignment

When human drivers approach an all-way-stop junction, they are expected to come to a full stop. The car that came to a full stop at the intersection before cars on any other road segments obtains the right of way. Only a car with the right of way for a conflict zone is allowed to enter it. To unambiguously identify rights of way to contested conflict zones, the developed simulator uses priorities encoded by numerical values.

3.2.1 `P` *Priorities:* Working priority assignment based on estimated time of arrival

The simulator comprises a central arbitrator that assigns the right of way. The arbitrator makes its P-priority assignments as cars approach the conflict zones. Once a car reaches a perimeter near the conflict zone called a *prioritization perimeter*, the estimated times of arrival at a given conflict zone for this car and any other cars within a wider perimeter called a *reachable perimeter* are computed. The assignment is based on an estimated time of arrival $t_{arrival,est} = \xi / v_{ref}$, where ξ represents the distance to the *conflict zone perimeter* which demarcates the conflict zone.

The P priorities are assigned as follows: All cars outside the reachable perimeter or not on the path to a given conflict zone receives the unassigned priority level of -1. Otherwise in uncontested cases the default priority of 0 is assigned to the only particular car that requires access to a given conflict zone. For all other cases, P priorities are assigned as increasing numbers in the order of estimated time of arrival. If a car approaches close to the intersection such that it cannot safely come to a stop, it receives the highest irrevocable priority of 1 as long as there is no other car already with priority of 1 for that conflict zone.

On the one hand, assigning the right of way based on an estimated time of arrival provides a mechanism to decide an unambiguous order or precedence even before cars have come to a stop. On the other hand, it also has an important implication when there are multiple cars approaching the conflict zone on the same single-lane road segment (see Fig. 2(b)). Not considering passing maneuvers, only the estimated time of arrival for the first car to arrive can be reliably used. For other cars behind that must decelerate to very low speeds (in the worst case completely stopped) near but not at the intersection, their estimated time of arrival is not a good indicator for right of way computations. For such cases, a different set of priorities is used called Q *priorities* based on their positions in the queue of arrival, as explained next.

3.2.2 Q *priorities:* **Working priority assignment based on queues**

For each road segment, the first car to arrive at the conflict zone from a particular approach direction is identified as the *head* of the queue and receives a Q priority of 1. Cars that are not at the head are determined to be queued up behind the head car. If an oncoming car not at the head of the queue reaches a perimeter near the conflict zone called the *stopped perimeter* and has a low-enough velocity (e.g., $v < 0.25$), the car is identified as having *effectively stopped* and is queued. Such cars receive Q priorities of ≥ 2 according to their order in the queue. Whenever a car at the head of a queue (with Q priority 1) exits the conflict zone, the next car behind it with Q priority of 2 becomes the new head of the queue and receives a Q priority 1.

3.2.3 R *priorities* and S *Priorities*: **Merging and compacting of working priorities**

In order to unambiguously determine the right of way to conflict zones, P priorities and Q priorities must be merged into a single list. The merge uses the following sets of cars in overriding levels of importance.

- The most important set is that of all the head cars that have come to a stop and that are waiting to enter the conflict zones.
- Next important set is that of the cars that have effectively stopped but that are not at the head.
- Last is the set of all head cars that have not yet stopped, i.e., are traveling toward the conflict zones.

Within each of the above sets, there can be more than one car at a given time. The relative ordering of individual cars within each set is determined by their P priorities. The merging results in a set of R priorities, which may be further compacted into S priorities so as to use smallest possible numbers.

Table 1 summarizes the different levels of (S) priority that are finally used in the disambiguation. If a car is beyond a *reachable perimeter* for the current planning horizon (i.e., is sufficiently far away) or does not require access to a particular conflict zone given its approach-exit combination, the car receives the 'do not care' priority of -1. If there is only one car inside the reachable perimeter that requires access to a particular conflict zone at a given point in time (i.e., the conflict zone is not contested at that time), that car is assigned a default priority of 0. For example, in Fig. 2(b), if the car that has its origin in the South has its destination in the East, then conflict zone A will not be contentious. In such a case, the car approaching from the East has priority 0 for conflict zone A.

Table 1: Priority encoding for right-of-way disambiguation of a given conflict zone.

Value	Qualification	Meaning
-1	Do not care	no priority assigned
0	Default	(possibly revocable) default right of way; conflict zone is not contentious
1	Highest	irrevocable right of way
2	Lower	no right of way to the conflict zone, cars must not enter

Two additional priority levels are required for contested conflict zones. For a given conflict zone, a car that has the irrevocable right of way to it has the highest priority (encoded by 1). Because of the configuration, there can only be one other car approaching from another direction that also requires access to that conflict zone. If such other car is present it receives a lower priority (encoded by 2).

3.3 A Priority Assignment Example

Consider the scenario depicted in Fig. 2(b) with two cars (id: #1 and id: #2) coming from the South and headed East and West, respectively, one car (id: #3) coming from the East and headed South, and one car (id: #4) from the West headed East. Figure 3 shows the position of the four vehicles along their trajectories over time with the various perimeters demarcated.
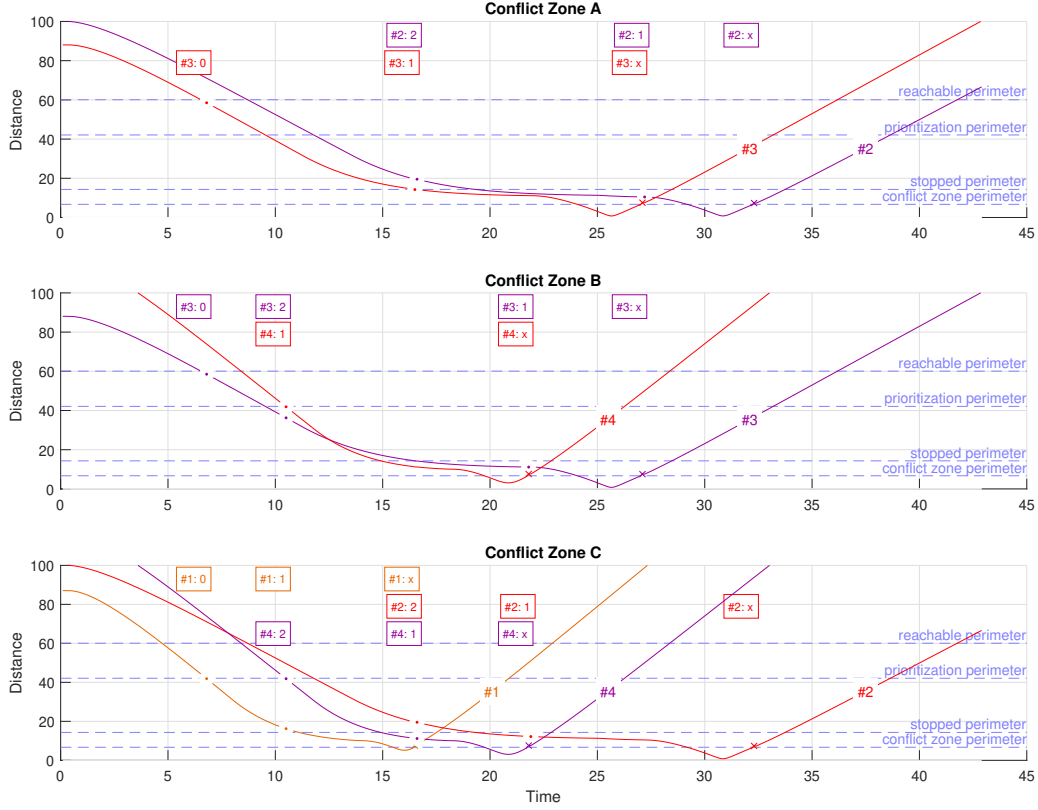

Figure 3: Position along individual trajectories over time.

The heads of the queues are referred to as *S*, *W*, and *E* and the priorities are compactly represented in the format $\langle S, W, E \rangle$. At time 0, all four cars have initial P priority and Q priority values of -1 assigned for all three conflict zones, that is, all priorities are $\langle -1, -1, -1 \rangle$.

At time 6.7 car #1 moves within the prioritization perimeter, which triggers a priority assignment. At this time car #3 is the only other car within the reachable perimeter. None of the zones are contested, so car #3 and #1 are assigned the default priority 0. For zone *A*, car #3 (*E*) is the first and only car to traverse zone *A*, so it receives a P priority of 0 and all others receive a P priority of -1. Since no queues have formed yet, all Q priorities are -1. Merging P and Q priorities sets the corresponding R priority assignment $\langle -1, -1, 0 \rangle$, which is also the compact S priority assignment. Priorities for the other zones *B* and *C* are derived analogously as $\langle -1, -1, 0 \rangle$ and $\langle 0, -1, -1 \rangle$.

At time 10.4 car #4 (*W*) enters the prioritization perimeter triggering a P-priority reassignment, also making zones *B* and *C* contested. The cars are estimated to arrive at the intersection in the following order: car #1 (*S*) then car #4 (*W*), followed by car #3 (*E*), resulting in P-priority assignments $\langle -1, -1, 0 \rangle$, $\langle -1, 2, 3 \rangle$, and $\langle 1, 2, -1 \rangle$ for zones *A*, *B*, and *C*. Notice that car #4 is farther away but has a greater velocity (steeper gradient) than car #3 (Fig. 3). At this point, the Q priorities are continued from earlier. Because the Q priorities do not

yet have any car with priority 1, merging simply results in the same R priorities as P priorities. Compacting produces the final S priorities $\langle -1, -1, 0 \rangle$, $\langle -1, 1, 2 \rangle$, and $\langle 1, 2, -1 \rangle$ for zones *A*, *B*, and *C*.

Note the revocation default priority 0 of car #3 (*E*) for zone *B* occurred by merit of car #3 still being able to come to a stop at the conflict zone. If car #3 would have been too close to come to a stop, its priority would have changed from 0 to 1 instead during the priority merging and car #3 would have retained the right of way.

Table 2: Priority merge at time 16.5.

Car	Zone *A*			Zone *B*			Zone *C*		
	S	*W*	*E*	*S*	*W*	*E*	*S*	*W*	*E*
P priority	3	−1	1	−1	2	1	3	2	−1
Q priority	−1	−1	0	−1	1	2	−1	2	−1
R priority	5	−1	2	−1	1	4	6	3	−1
S priority	2	−1	1	−1	1	2	2	1	−1

At time 16.5 car #1 exits zone *C* and its priority is reset to −1, depicted by an 'x' marker in Fig. 3. At this point, car #2 is the new head *S*. The Q priorities at time 16.5 correspond to the S priorities from time 10.4 carried forward, except that the new *S* car has a priority of −1 by virtue of not being the head of the queue up until this point. Based on estimated arrival order of car #3 followed by car #2 followed by car #4, P priorities are 1, 2, and 3 for *E*, (the new) *S*, and *W*, respectively, in Table 2.

Table 3: Merge priorities for Zone *A*.

Round	Car	Operation	Priorities		
			S	*W*	*E*
Initial	—	—	−1	−1	0
Default	car #3	Set default of highest priority	−1	−1	1
Locked	—	Lock irrevocable priorities	−1	−1	1
Existing	car #3	Set priority for existing queued heads	−1	−1	2
New	—	Set priority for new queued heads	−1	−1	2
Moving	car #2	Set priority for moving cars	5	−1	2

Since the merging of P and Q priorities at this time is nontrivial, it is summarized in Table 3 and explained next in the context of conflict zone *A*.

- Initial priorities are Q priorities from Table 2.
- First the 'Default' round changes the default priority 0 of car #3 (*E*) to the highest priority 1 because zone *A* has become contentious when car #2 became the new (head of road segment) *S*.
- In the 'Locked' round, priorities that cannot be revoked are determined (e.g., for cars in a conflict zone already) but no such cases exist.
- In the 'Existing' round, cars that were already the heads on a given road segment have their priority set such that they always have a value that exceeds any of the priority values assigned thus far. To this end, the value 1 is added to the highest priority of any processed car in the Locked round. Since there was no such car, the highest priority value is 0 and so 1 is added to any car that is processed in the Existing round. This results in car #4 (*E*) changing its priority from 1 to 2.
- In the 'New' round, cars are processed that had queued behind another car (i.e., cars that have stopped already) and that now become the head of a road segment. No such cars exist at this point.
- In the 'Moving' round, cars are processed that had not yet stopped near the conflict zone, which concerns car #2 (*S*). For these cars, the P priority determines their eventual priority ordering. To ensure priority values that are higher than any priority value assigned thus far, the P priority values are added to the highest currently assigned priority value, here priority 2 for car #3 (*E*) in the 'New'
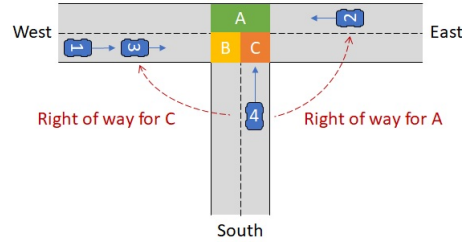
Figure 4: Trading the right of way to different conflict zones.

round. One such moving car #2 (*S*) is relevant. The estimated time of arrival priority value was 3 and adding 2 as the highest priority assigned so far results in merged `R` priority value 5.

The priorities at the end of the 'Moving' round are the new `R` priorities in Table 2. These are further adjusted to use smaller consecutive values while still maintaining the relative order as the `S` priorities. Thus, for zone *A* the resultant priority for car #3 is 1 and for car #2 is 2, shown as `S` priorities in Table 2.

The assignment of priorities for zones *B* and *C* is performed analogously. The scenario concludes with car #2 traversing the conflict zones (Fig. 3).

4   TRANSFERRING AND TRADING RIGHT-OF-WAY PRIVILEGES

Given that ownership has been determined, privileges can now be traded.

4.1 A Trading Protocol

To facilitate the transfer of traffic privileges such as the right of way, a protocol has been developed and layered on top of the priority assignment described previously. Once priorities have been assigned cars can transfer the right of way to other cars as long as they are able to safely come to a stop at the conflict zone. Moreover, in order to incentivize a car to transfer the right of way and accept a longer time to traverse the conflict zone, an amount of shared currency may be transferred in return. Since the transaction where the right of way and the amount of currency is agreed upon and transferred must be executed in a very short time frame an automated algorithm is employed. Because the temporal constraints do not allow a situational trading approach, a general *urgency* parameter is set for each car. This parameter indicates the cost that a car is willing to incur in order to acquire the right-of-way privilege.

If the urgency of a car is larger than 0, it is considered open to buying the right of way privilege for a conflict zone. To determine the value of a conflict zone to the Buyer, the number of conflict zones that it needs in order not to have to wait at the junction is determined. If more than one conflict zone is required, the value of the right of way privilege for each of the individual conflict zones decreases quadratically.

Next, an attempt is made to acquire the right of way for each of the conflict zones where it is needed. The first step is to find the current owner of the right of way (i.e., the car with priority 1). Once identified, the owner is only allowed to transfer the right of way if the owner is within a *negotiation perimeter*, can safely come to a stop before the conflict zone, and has a lower urgency than the Buyer. If these conditions hold, the value of the conflict zone to the Seller is determined. Analogous to the Buyer, the number of conflict zones where the Seller needs the right of way to traverse the junction are determined. Based on the number of conflict zones required and the urgency of the Seller, a value for the conflict zone in question is determined with a quadratic cost. Note that for this cost the conflict zone where the Seller currently owns the right of way is added to the number of required conflict zones. From the Seller value and the Buyer value, the cost of the conflict zone is determined as the mean and enforced to always be 1 or more. Finally, the priority values of the conflict zone are swapped between the Buyer and the Seller and the computed cost is transferred from the Buyer to the Seller.

4.2 A Trading Example

To explore this concept of priority transfer, consider a different scenario depicted in Fig. 4 with two cars approaching from the West (id: #1 and id: #3), one car from the East (id: #2) and one from the South (id: #4). The simulated trajectories for these cars are shown in Fig. 5. As can be seen in Fig. 5, car #4 approaches zone *A* faster than car #2. At time 6.1 car #4 moves within the prioritization perimeter, following the prioritization approach similar to the earlier example from Sec. 3.3, the final (S) priority assignment results in car #4 and car #2 receiving priorities 1 and 2 for zone *A*, respectively. Car #2 can now formally request car #4 to transfer the right of way. For example, car #2 may be in an urgent situation and may benefit from traversing the conflict zone quickly, more so than car #4. After a successful transfer of the right of way car #4 lets car #2 pass first. Similarly, for zone *C*, car #4 lets car #3 pass before itself.
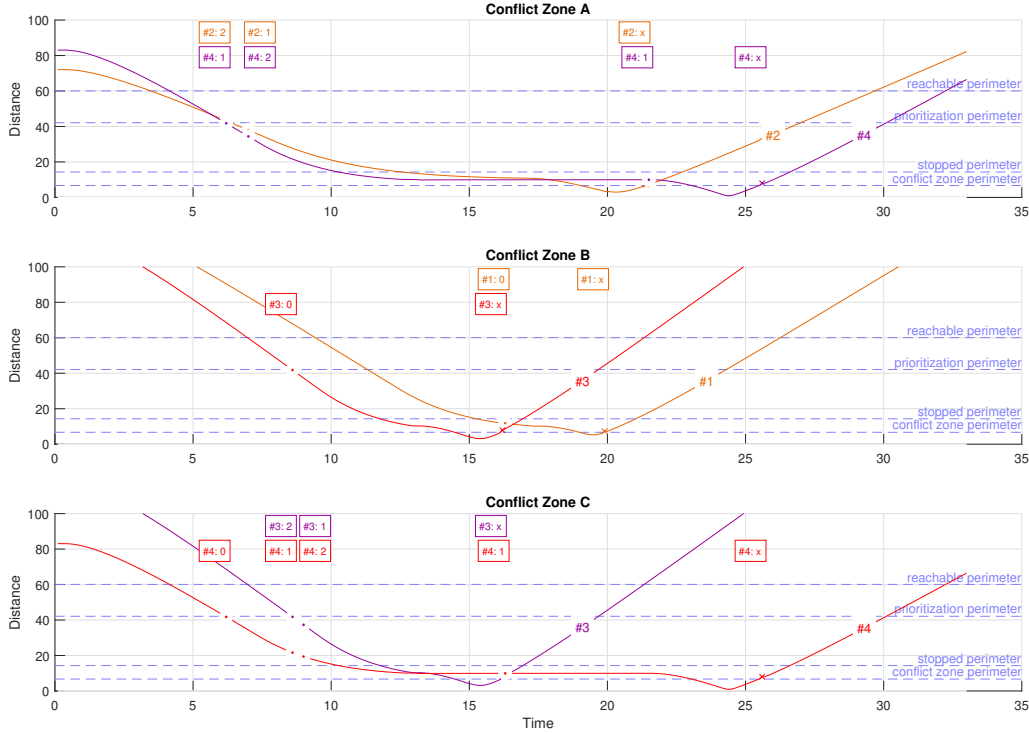

Figure 5: Transfer of priorities for zones A and C.

5   CONCLUSIONS

This work developed a computational structure for the automated assignment of priorities at a T junction to be traded as traffic privileges in return for a common currency. Appropriate abstraction for a computational implementation were developed as privileges, priorities, objectives, actor roles, and trade preferences. A protocol to assign priorities, transfer priorities, and to allow priority trades was developed based on these abstractions. The implementation was used to perform simulation studies of traffic scenarios at a T junction where privileges can be traded so as to enable the passengers of autonomously driving cars to select personal preferences (selfish behavior) based on their current disposition (e.g., whether being in a rush or not).

The T-junction framework from this paper can be generalized for other traffic intersections such as four-way intersections leading to more combinations of conflict zones and actors. The concept is also generally applicable to other driving scenarios such as traffic merging onto a highway, except that safe stopping expectations would need to be relaxed for cars already on the highway. Opportunities for future work include formal analyses such as proof of algorithmic soundness in the priority assignment (e.g., there is

always at most one car with priority 1) and analysis of trading approaches (e.g., whether certain urgency settings lead to better results based on the set objectives), especially in the face of more ambient traffic. The simulation framework presented in this work can be realized in practice using secure transactions, for example using a blockchain-based protocol (Leiding and Vorobev 2018).

REFERENCES

Antoniotti, M., A. Deshpande, and A. Girault. 1997. "Microsimulation analysis of automated vehicles on multiple merge junction highways". In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, Volume 1, pp. 839–844. IEEE.

Censi, A., S. Bolognani, J. G. Zilly, S. S. Mousavi, and E. Frazzoli. 2019. "Today Me, Tomorrow Thee: Efficient Resource Allocation in Competitive Settings using Karma Games". *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 686–693.

Ci, W. X., S. K. Ahmed, F. Zulkifli, and A. K. Ramasamy. 2009. "Traffic flow simulation at an unsignalized T-junction using Monte Carlo Markov Chains". In *2009 IEEE International Conference on Signal and Image Processing Applications*, pp. 346–351. IEEE.

Dai, P., K. Liu, Q. Zhuge, E. H.-M. Sha, V. C. S. Lee, and S. H. Son. 2016. "Quality-of-experience-oriented autonomous intersection control in vehicular networks". *IEEE Trans. on Intelligent Transportation Systems* vol. 17 (7), pp. 1956–1967.

Fayazi, S. A., and A. Vahidi. 2018. "Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection crossing". *IEEE Trans. on Intelligent Vehicles* vol. 3 (3), pp. 287–299.

Ganiyu, R., S. Olabiyisi, E. Omidiora, O. Okediran, and O. Alo. 2011. "Modelling and simulation of a multi-phase traffic light controlled T-type junction using timed coloured petri nets". *American Journal of Scientific and Industrial Research* vol. 2 (3), pp. 428–437.

Guney, M. A., and I. A. Raptis. 2018. "Scheduling-driven motion coordination of autonomous vehicles at a multi-lane traffic intersection". In *2018 Annual American Control Conference (ACC)*, pp. 4038–4043. IEEE.

Ito, Y., M. A. S. Kamal, T. Yoshimura, and S.-i. Azuma. 2018. "Coordination of connected vehicles on merging roads using pseudo-perturbation-based broadcast control". *IEEE Transactions on Intelligent Transportation Systems* vol. 20 (9), pp. 3496–3512.

Kamal, M. A. S., J.-i. Imura, T. Hayakawa, A. Ohata, and K. Aihara. 2014. "A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights". *IEEE Transactions on Intelligent Transportation Systems* vol. 16 (3), pp. 1136–1147.

Kamrani, M., S. M. H. E. Abadi, and S. R. Golroudbary. 2014. "Traffic simulation of two adjacent unsignalized T-junctions during rush hours using Arena software". *Simulation Modelling Practice and Theory* vol. 49, pp. 167–179.

Lee, J., and B. Park. 2012. "Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment". *IEEE Transactions on Intelligent Transportation Systems* vol. 13 (1), pp. 81–90.

Leiding, B., and W. V. Vorobev. 2018. "Enabling the V2X Economy Revolution Using a Blockchain-based Value Transaction Layer for Vehicular Ad-hoc Networks". In *Mediterranean Conference on Information Systems*.

Li, P. T., and X. Zhou. 2017. "Recasting and optimizing intersection automation as a connected-and-automated-vehicle (CAV) scheduling problem: A sequential branch-and-bound search approach in phase-time-traffic hypernetwork". *Transportation Research Part B: Methodological* vol. 105, pp. 479–506.

Liang, X., S. I. Guler, and V. V. Gayah. 2018. "Signal timing optimization with connected vehicle technology: Platooning to improve computational efficiency". *Transportation Research Record* vol. 2672 (18), pp. 81–92.

Lin, Y.-T., H. Hsu, S.-C. Lin, C.-W. Lin, I. H.-R. Jiang, and C. Liu. 2019, October. "Graph-Based Modeling, Scheduling, and Verification for Intersection Management of Intelligent Vehicles". *ACM Trans. Embed. Comput. Syst.* vol. 18 (5s), pp. 95:1–95:21.

MathWorks® 2019a, March. "MATLAB®". *Release R2019b*.

MathWorks® 2019b, March. "Vehicle Dynamics Blockset™". *Release R2019b*.

Paruchuri, P., A. R. Pullalarevu, and K. Karlapalem. 2002. "Multi agent simulation of unorganized traffic". In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pp. 176–183.

Qian, X., J. Gregoire, A. De La Fortelle, and F. Moutarde. 2015. "Decentralized model predictive control for smooth coordination of automated vehicles at intersection". In *2015 European Control Conference (ECC)*, pp. 3452–3458. IEEE.

Sayin, M. O., C. Lin, S. Shiraishi, J. Shen, and T. Başar. 2019, March. "Information-Driven Autonomous Intersection Control via Incentive Compatible Mechanisms". *IEEE Transactions on Intelligent Transportation Systems* vol. 20 (3), pp. 912–924.

Xu, B., S. E. Li, Y. Bian, S. Li, X. J. Ban, J. Wang, and K. Li. 2018. "Distributed conflict-free cooperation for multiple connected vehicles at unsignalized intersections". *Transportation Research Part C: Emerging Technologies* vol. 93, pp. 322–334.

Zhang, J., S. Pourazarm, C. G. Cassandras, and I. C. Paschalidis. 2018. "The price of anarchy in transportation networks: Data-driven evaluation and reduction strategies". *Proceedings of the IEEE* vol. 106 (4), pp. 538–553.

AUTHOR BIOGRAPHIES

Zhan Tu was a Ph.D. candidate at Purdue University. He contributed to this work during his internship in the Advanced Research & Technology Office at MathWorks. His email address is tu17@purdue.edu.

Anastasios Dimas is a Ph.D. candidate at Rutgers University, NJ, USA. He contributed to this work during his internship in the Advanced Research & Technology Office at MathWorks. His email address is tasos.dimas@rutgers.edu.

Mehmet Necip Kurt is a Ph.D. candidate at Columbia University, NY, USA. He contributed to this work during his internship in the Advanced Research & Technology Office at MathWorks. His email address is m.n.kurt@columbia.edu.

Anastasia Mavrommati is a research scientist in the Advanced Research & Technology Office at MathWorks. Her email address is amavromm@mathworks.com.

Pieter J. Mosterman is a Research Scientist in the Advanced Research & Technology Office at MathWorks. His email address is pmosterm@mathworks.com.

Akshay Rajhans is a Research Scientist in the Advanced Research & Technology Office at MathWorks. His email address is arajhans@mathworks.com.

Roberto G. Valenti is a Research Scientist in the Advanced Research & Technology Office at MathWorks. His email address is rvalenti@mathworks.com.