Socially-Optimal Auction-Theoretic Intersection Management System

Adam Morrissett[1], Patrick J. Martin[1], Sherif Abdelwahed[1]

***Abstract*—Unsignalized intersections are often sources of congestion and collisions. When human-driven vehicles arrive simultaneously, the drivers typically creep out into the intersection or wave each other through to break stalemates. While intuitive for human drivers, this approach would be challenging for autonomous vehicles (AVs). Current AVs typically operate in isolation without explicitly communicating their intentions to others. In this paper, we propose an auction-based intersection management system (IMS) to determine a crossing schedule. Vehicles bid for crossing time using a cost function over different possible crossing times, and the IMS assigns crossing times that maximize social utility. We evaluate our system with an ambiguous crossing scenario and demonstrate its usefulness in determining socially-optimal crossing schedules.**

I. Introduction

Road intersections are abundant and are known to cause congestion, and they accounted for over 28 % of collisions in 2019 [1]. Crossing unsignalized intersections, which are common on less-dense roads or in parking lots, can be particularly frustrating. Drivers seldom use turn signals, and the crossing order is ambiguous when multiple vehicles approach simultaneously.

For example, consider a four-way intersection in which four vehicles approach simultaneously, one on each road. Road laws require that the right-most vehicle proceeds first through the intersection [2]. However, all four vehicles approached simultaneously, so there is no right-most vehicle. In this scenario, drivers need to creep out into the intersection, wave each other through, or rely on other subtle cues to break stalemates.

Even in less ambiguous scenarios, conventional crossing orders may be suboptimal with respect to social utility. Some drivers may prefer to drive through the intersection quickly while other prefer moving slowly. Some drivers may want to cross as soon as possible while others are indifferent. The normal first-come-first-served and right-most-first schedules do not necessarily satisfy individuals' preferences.

These issues stem from a lack of communication among drivers. Without it, both human-driven and autonomous vehicles must assume others' intentions or preferences. Connected autonomous vehicles (CAVs) and cooperative driving automation (CDA) [3] help provide more ways to exchange information between vehicles and infrastructure. In this paper, we leverage CDA and vehicle-to-infrastructure (V2I) communication to develop an intersection management system (IMS) for CAVs.

IMSs for CAVs are divided into three groups. Direct control methods determine velocity profiles that vehicles must follow in order to cross the intersection when intended and without collisions [4], [5]. Scheduling-based approaches assign arrival or departure times to each vehicle so that the intersection is clear when they arrive [6]–[8]. As opposed to direct control methods, vehicles are responsible for determining a trajectory that satisfies their assigned time. In existing auction-based methods, vehicles bid for intersection access with real money or virtual tokens [9]–[11]. Vehicles with insufficient funds are often subsidized by the management system. The crossing order is determined by the bids, and typically the highest bidder proceeds first.

Each of these management methods has drawbacks. Direct control methods tightly couple the system with individual vehicle dynamics and may scale poorly. Scheduling approaches do not consider dynamics constraints when assigning arrival times. Currency-based auctions establish an economy in which intersection crossings are determined by economic status rather than physical abilities or utility. Finally, none of these approaches consider passenger preferences on how soon and how quickly the vehicle crosses the intersection. While traffic may be optimal with respect to throughput, it may be *suboptimal* with respect to passenger satisfaction.

In this paper, we propose an auction-based IMS that optimizes for passenger preferences while also considering vehicle dynamics and other constraints. Vehicles submit, as their bid, cost functions for how quickly they want to cross the intersection and how soon. The IMS allocates a crossing time interval to each vehicle. Each vehicle must cross the intersection within the amount of time they are assigned. For example, if the IMS assigns a five-second crossing time, the vehicle must cross the intersection in 5 seconds. The system also schedules vehicles according to their preferences on how soon they want to cross. Vehicles' cost functions abstract their dynamics, allowing for a loose coupling with the IMS and promoting scalability. To the best of our knowledge, this is the first IMS with these features.

The rest of this paper is as follows. Section II formulates

[1]Department of Electrical and Computer Engineering, Virginia Commonwealth University, Richmond, Virginia 23284-3068. Email: morrissettal2@vcu.edu, martinp@vcu.edu, sabdelwahed@vcu.edu

This work was supported (in part) by the Commonwealth Cyber Initiative (CCI), an investment in the advancement of cyber R&D, innovation, and workforce development in Virginia. For more information about CCI, visit www.cyberinitiative.org.

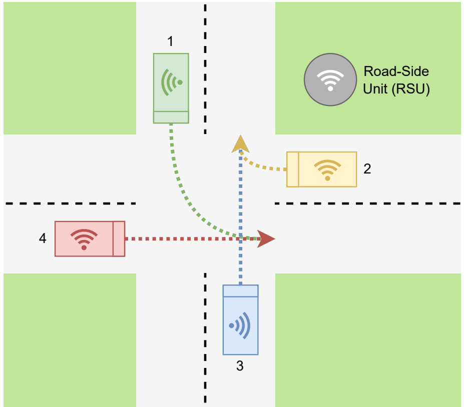

Fig. 1. Four vehicles approach an intersection. The dotted lines indicate each vehicle's desired path, but they do not indicate when the vehicles will cross. Only one vehicle can cross the intersection at a time.

the problem. Section III describes our proposed system. Section IV presents the evaluation results. Finally, Section V concludes the paper and discusses future work.

II. Problem Formulation

In our scenario, n vehicles drive on individual roads that lead to the same m-way intersection. Each road may have a variable number of vehicles. In this work, we are concerned with only the lead vehicle on each road. Let $\mathcal{I} \subseteq \mathbb{Z}_{>0}$ be the set of lead vehicles wanting to cross the intersection, where $\mathbb{Z}_{>0}$ is the set of positive integers. We define $I \triangleq |\mathcal{I}| \leq n$ to be the total number of lead vehicles. Fig. 1 illustrates an example four-way intersection scenario with one (lead) vehicle per road.

The intersection is unsignalized (*i.e.,* has no traffic signals) and may optionally have stop signs installed. Vehicles are equipped with a communication system. A road-side unit (RSU) is installed at the intersection to facilitate V2I communication.

We assume the vehicles approach the intersection simultaneously (or within some small finite time window $\epsilon \geq 0$). The crossing order initially is undetermined, so all of the vehicles plan to stop at the intersection. Like real-world intersections, vehicles may be perceived to have arrived simultaneously even if their exact arrival times differ. Let t_i and $t_{i'}$ be the arrival times of two different vehicles $i, i' \in \mathcal{I}$, respectively. We consider vehicles i and i' to arrive simultaneously if $|t_{i'} - t_i| \leq \epsilon$, where ϵ is determined by the system designer.

We assume that each vehicle has a preference on how quickly it crosses the intersection. A vehicle's preference represents the combined preferences of its passengers. Additionally, a vehicle's dynamics determine its ability to cross the intersection. We define vehicle i's *crossing time* t^i_{cross} as the time it takes for vehicle i to cross the intersection. So, if $t^i_{\text{cross}} = 5$ then the vehicle crosses the intersection in 5 seconds. We define vehicle i's *crossing cost* as the cost for crossing the intersection for a specific t^i_{cross} given its dynamics and passenger preferences. We define vehicle i's *waiting time* t^i_{wait} as the time a vehicle waits before starting to cross the intersection. For example, when $t^i_{\text{wait}} = 10$, the vehicle waits 10 seconds from the moment it received the time before attempting to cross the intersection. We define vehicle i's *waiting cost* as the cost for waiting at the intersection for a specific t^i_{wait} before crossing.

Our goal is to determine crossing times for each vehicle and a crossing order that maximizes the social welfare for all lead vehicles crossing the intersection. We use a utilitarian social cost function, which is the summed cost (negative utility) for each vehicle. Minimizing the social cost function given any constraints on crossing times or the schedule produces a socially-optimal solution [12, Sec. 22.C].

A. Vehicle Dynamics

Vehicle i moves according to a dynamics model $f^i \colon \mathcal{X}^i \times \mathcal{U}^i \to \mathcal{X}^i$ defined by

$$\dot{\boldsymbol{x}}^i = f^i(\boldsymbol{x}^i, \boldsymbol{u}^i) \quad , \tag{1}$$

where $\boldsymbol{x}^i \in \mathcal{X}^i \subseteq \mathbb{R}^\rho$ is the model's ρ-element state vector, and $\boldsymbol{u}^i \in \mathcal{U}^i \subseteq \mathbb{R}^\sigma$ is its σ-element input vector. The sets $\mathcal{X}^i$ and $\mathcal{U}^i$ are vehicle i's state and input spaces, respectively. The specific dynamics model may differ among vehicles.

We discretize the model over the planning horizon $t^i_{\text{cross}} \in \mathcal{T}^i_{\text{cross}} \subseteq \mathbb{R}_{>0}$, where $\mathbb{R}_{>0}$ is the set of positive real numbers. Set $\mathcal{T}^i_{\text{cross}}$ is the set of possible crossing times for vehicle i. Let $\delta^i \in \mathbb{R}_{>0}$ be the discretization sampling period. The number of samples in the planning horizon $N \colon \mathcal{T}^i_{\text{cross}} \to \mathbb{Z}_{>0}$ is a function, parameterized by δ^i:

$$N(t^i_{\text{cross}}) \triangleq \left\lceil \frac{t^i_{\text{cross}}}{\delta^i} \right\rceil \quad . \tag{2}$$

The intuition is that the planning horizon becomes longer as the vehicle has more time to cross the intersection (*i.e.*, a larger t^i_{cross}). Then for all $k = 0, 1, \ldots, N(t^i_{\text{cross}})$, we represent the discretized vehicle dynamics as

$$\boldsymbol{x}^i_{k+1} = f^i_d(\boldsymbol{x}^i_k, \boldsymbol{u}^i_k) \quad , \tag{3}$$

where $\boldsymbol{x}^i_k$ and $\boldsymbol{u}^i_k$ are the system's state and input vectors, respectively, at time k.

Each vehicle tracks a reference path, such as a turning arc or straight line, as they drive through the intersection. The reference path is the lane's center line, which is provided to the planner (*e.g.*, through a perception system). Let $\boldsymbol{r}^i \colon \mathbb{R}_{\geq 0} \to \mathcal{R}^i$ be a time-varying reference function, where $\mathbb{R}_{\geq 0}$ is the set of non-negative real numbers, and $\mathcal{R}^i \subseteq \mathbb{R}^\rho$ is the set of ρ-element reference state vectors for vehicle $i \in \mathcal{I}$.

B. Vehicle Costs

Vehicle i's crossing cost incorporates dynamics model (3) and passenger preferences. We define a twice continuously differentiable crossing objective function

$$J^i_{\text{cross}} \colon \mathcal{T}^i_{\text{cross}} \times \mathcal{X}^i \times \mathcal{U}^i \to \mathbb{R} \tag{4}$$

that is parameterized by passenger-selected preferences. Vehicles are free to select their own objective function.

For each crossing time $t^i_{\text{cross}} \in \mathcal{T}^i_{\text{cross}}$, vehicle i solves a nonlinear optimization problem to determine its crossing cost $c^{\star i}_{\text{cross}} \colon \mathcal{T}^i_{\text{cross}} \to \mathbb{R}$ for crossing the intersection in t^i_{cross} time units. The nonlinear optimal planning problem is defined by

$$\begin{aligned} c^{\star i}_{\text{cross}}(t^i_{\text{cross}}) \triangleq \min \quad & J^i_{\text{cross}}(\boldsymbol{x}^i, \boldsymbol{u}^i, t^i_{\text{cross}}) && \text{(5a)} \\ \text{subj. to} \quad & \boldsymbol{x}^i_{k+1} = f^i_d\left(\boldsymbol{x}^i_k, \boldsymbol{u}^i_k\right) && \text{(5b)} \\ & \underline{\boldsymbol{x}}^i_k \leq \boldsymbol{x}^i_k \leq \overline{\boldsymbol{x}}^i_k && \text{(5c)} \\ & \underline{\boldsymbol{u}}^i_k \leq \boldsymbol{u}^i_k \leq \overline{\boldsymbol{u}}^i_k \quad , && \text{(5d)} \end{aligned}$$

where $\underline{\boldsymbol{x}}^i_k$ and $\overline{\boldsymbol{x}}^i_k$ are the upper and lower bounds on $\boldsymbol{x}^i_k$, respectively; $\underline{\boldsymbol{u}}^i_k$ and $\overline{\boldsymbol{u}}^i_k$ are the upper and lower bounds on $\boldsymbol{u}^i_k$, respectively. Variables $\boldsymbol{x}^i$ and $\boldsymbol{u}^i$ (without the k subscript) represent states and inputs, respectively, over the entire planning horizon. We assume the vehicle model is controllable. Common motion models, such as the simple car, bicycle (under zero-slip assumptions), and unicycle models exhibit this property [13, Sec. 15.1.3], [14, Sec. 2.2]. We additionally assume the state space is connected, meaning at least one path exists connecting a vehicle's starting state to its goal state; otherwise, the road would be impassible. These two assumptions ensure (5) is feasible for at least one t^i_{cross}.

It should be noted that (5) may not have a solution for a given $\tilde{t}^i_{\text{cross}} \in \mathcal{T}^i_{\text{cross}}$, meaning $c^{\star i}_{\text{cross}}(\tilde{t}^i_{\text{cross}})$ is undefined. We define $\underline{t}^i_{\text{cross}}$ and $\overline{t}^i_{\text{cross}}$ as the lower and upper crossing times, respectively, for which (5) is defined. Let $\underline{t}^i_{\text{cross}}$ be the feasible t^i_{cross} mentioned above. Then there exists a larger feasible time $t^{i\prime}_{\text{cross}}$ that the vehicle can achieve by driving slower. This process can repeat up to $\underline{\boldsymbol{u}}^i$, thus establishing $\overline{t}^i_{\text{cross}}$. Therefore, (5) is defined for all $t^i_{\text{cross}} \in \left[\underline{t}^i_{\text{cross}}, \overline{t}^i_{\text{cross}}\right] \subseteq \mathcal{T}^i_{\text{cross}}$.

The waiting cost $c^i_{\text{wait}} \colon \mathbb{R}_{\geq 0} \to \mathbb{R}$ represents how long passengers are willing to wait at the intersection before crossing. Like the crossing cost function, vehicles are free to choose their own waiting cost function.

III. System Description

To solve the intersection crossing problem defined in Section II, we propose an IMS in which vehicles participate in an auction to determine the crossing schedule. In this work, the RSU serves as the auctioneer, but future work will investigate electing one of the vehicles to be the auctioneer, thus removing a dependency on road-side infrastructure. Vehicles use their crossing cost functions as bids, and the auctioneer assigns a crossing time to each vehicle. Additionally, the IMS determines a crossing order that is optimal with respect to each vehicle's waiting function. The goal of our system is to maximize the social welfare of all participating vehicles subject to any constraints imposed on the intersection.

Our system has three main components: vehicles, an auctioneer, and a scheduler. The auctioneer determines optimal crossing times with respect to its objective function and each vehicle's crossing cost. Afterwards, the scheduler uses those crossing times and each vehicle's waiting cost function to determine a crossing order. The IMS transmits vehicles' assigned crossing and start times. To simplify our initial system design, we separated the auctioneer and scheduler; however, future work will investigate combining them.

At this time, we assume only one vehicle crosses the intersection at a time. Additionally, only the lead vehicles

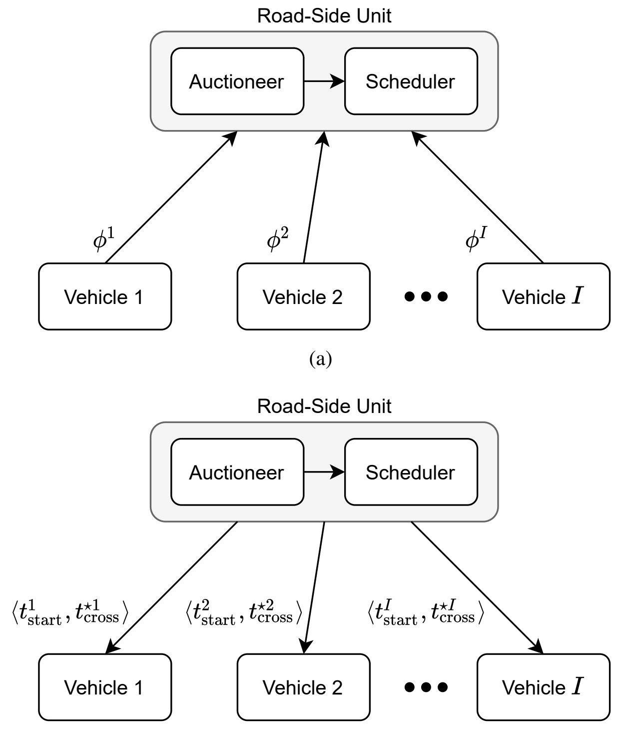

Fig. 2. (a) Vehicles transmit their bids as they near the intersection. (b) The IMS transmits each vehicle's optimal starting and crossing times.

for each road participate in the auction. Fig. 2 visualizes the data flow between the vehicles and IMS.

Fig. 3 depicts a scenarios with a continuous flow of vehicles approaching the intersection. The IMS sequences the crossings into several auctions. Once all lead vehicles in the current auction cross (green cars), the IMS begins another auction for the new set of lead vehicles (red cars). The blue cars will participate in a third auction once they are within the IMS's range, indicated by the black dashed circle. To minimize delays, an auction can begin as soon as the participants are within range. This way, vehicles can know their crossing and waiting times before reaching the intersection.

A. Auction Mechanism

We based our solution on a concept from economics and game theory called mechanism design [12, Ch. 23]. A mechanism is a collection of strategy sets and an outcome function; auctions are a type of mechanism. The auctioneer allocates goods to vehicles based on its outcome function, and vehicles place bids from their strategy sets to win those goods.

As opposed to other auction-based IMSs, where agents bid using money, the vehicles in our system bid using crossing and waiting cost functions. Vehicles calculate their cost curves for different crossing and waiting times then submit them as their bids. The IMS uses this information to determine an appropriate crossing time for each vehicle and a crossing schedule. Fig. 4 provides an overview of the algorithms used by the IMS and vehicles.

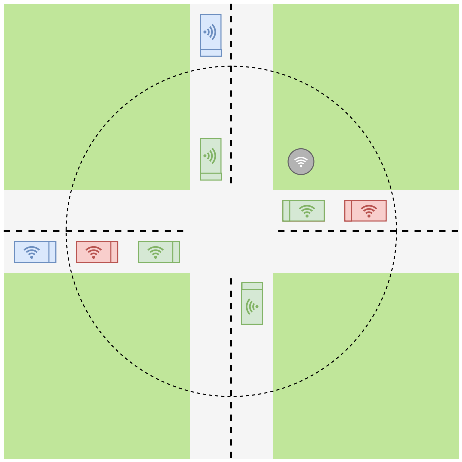

Fig. 3. A continuous flow of vehicles approach the intersection. The green vehicles participate in one auction, and the red ones participate in a separate one. The blue vehicles do not participate in any auction because they are not yet within range of the IMS (indicated by the dashed circle).

A desirable property for any auction is *ex post* (or Pareto) efficiency, meaning its outcome is Pareto optimal (efficient) given agents' utility functions. An outcome is Pareto optimal if no agent's utility can be improved without reducing another agent's utility.

Vehicles' bids represent and abstract private information that only they can observe, meaning auctions can be susceptible to strategic agents. To avoid this, auctions can be designed to incentivize agents to bid truthfully, a property call *incentive compatibility*. A sealed-bid, second-price auction (or Vickrey auction) is an example of an incentive compatible mechanism. For this initial work, we assume agents bid truthfully. Future work will analyze our auction formulation for incentive compatibility and modify it if necessary.

To demonstrate that our auction formulation is *ex post* efficient, we need to show that the crossing time allocations and crossing schedule are Pareto optimal. As briefly mentioned in Section II, optimizing a utilitarian social welfare (cost) function results in a Pareto optimal outcome.

Authors in [15] proposed a framework for comparing different market-based coordination methods for distributed energy systems. We use the same comparison framework to summarize our system:

- **Agent preference:** Vehicle and auctioneer preferences (objectives) are defined by (4) and (7), respectively.
- **Control decision:** For vehicle i, a trajectory $(\boldsymbol{x}^{\star i}, \boldsymbol{u}^{\star i})$ that minimizes c^i_{cross} for a given t^i_{cross}. For the IMS, a vector of optimal crossing times $\boldsymbol{t}^\star_{\text{cross}}$ and an optimal crossing schedule $\pi^\star$.
- **Information structure:** Type independence among agents; Decision dependence from auctioneer to vehicles.
- **Solution concept:** Auction-based optimization problem.

B. Vehicle Bids

Vehicles submit the crossing cost function $c^{\star i}_{\text{cross}}$ and feasibility bounds $\underline{t}^i_{\text{cross}}$ and $\overline{t}^i_{\text{cross}}$ (from Section II-B) that they

for all $i \in \mathcal{I}$ **do**
  {Vehicle i}
  $c^{i\star}_{\text{cross}}(t^i_{\text{cross}}) \leftarrow$ solve (5) $\forall t^i_{\text{cross}} \in \mathcal{T}^i_{\text{cross}}$
  $\underline{t}^i_{\text{cross}}, \overline{t}^i_{\text{cross}} \leftarrow$ feasibility bounds for $c^{i\star}_{\text{cross}}$
  $\phi^i \leftarrow \langle c^{i\star}_{\text{cross}}, \underline{t}^i_{\text{cross}}, \overline{t}^i_{\text{cross}}, c^i_{\text{wait}} \rangle$
  Submit ϕ^i as bid
end for

{Auctioneer}
Wait for all ϕ^i, $i \in \mathcal{I}$
$\boldsymbol{t}^\star_{\text{cross}} \leftarrow$ solve (8)

{Scheduler}
$\pi^\star \leftarrow$ solve (11) {See Fig. 5}
$t^{\pi^\star_0}_{\text{start}} \leftarrow 0$
$t^{\star\pi^\star_0}_{\text{cross}} \leftarrow 0$
for all $i = 1$ **to** I **do**
  $t^{\pi^\star_i}_{\text{start}} \leftarrow t^{\pi^\star_{i-1}}_{\text{start}} + t^{\star\pi^\star_{i-1}}_{\text{cross}}$
  Transmit $\langle t^{\pi^\star_i}_{\text{start}}, t^{\star\pi^\star_i}_{\text{cross}} \rangle$ to vehicle i
end for

Fig. 4. Proposed auction-based intersection management system (IMS).

calculated as their bids prior to nearing the intersection. They also submit their waiting cost functions c^i_{wait} so that a socially-optimal crossing schedule can be generated. We define vehicle i's bid ϕ^i by the tuple

$$\phi^i \triangleq \langle c^{\star i}_{\text{cross}}, \underline{t}^i_{\text{cross}}, \overline{t}^i_{\text{cross}}, c^i_{\text{wait}} \rangle \quad . \tag{6}$$

Vehicles submit their bids as shown in Fig. 2a.

C. Auctioneer

The auctioneer's role is to maximize social utility by assigning crossing times that best meet everyone's needs. After receiving each vehicle's bid, the auctioneer assigns crossing times according to vehicles' crossing cost functions. The auctioneer may consider other constraints or costs that affect crossing time assignment. For example, an upper limit on the intersection clearing time could be imposed to ensure all vehicles cross in a timely manner.

The auctioneer's *auction cost* function $c_{\text{auction}} \colon \mathbb{R}^I_{>0} \to \mathbb{R}$ is defined by

$$c_{\text{auction}}(\boldsymbol{t}_{\text{cross}}) \triangleq \sum_{i \in \mathcal{I}} c^{\star i}_{\text{cross}}(t^i_{\text{cross}}) + c_{\text{other}}(\boldsymbol{t}_{\text{cross}}) \quad , \tag{7}$$

where $\boldsymbol{t}_{\text{cross}} \in \mathcal{T}^1_{\text{cross}} \times \cdots \times \mathcal{T}^I_{\text{cross}}$ is an I-dimensional vector containing each vehicle's assigned crossing time. Function $c_{\text{other}} \colon \mathbb{R}^I_{>0} \to \mathbb{R}$ represents any other costs or soft constraints that can influence the assigned crossing times, and $c_{\text{other}}(\boldsymbol{t}_{\text{cross}}) = 0$ if there are no other costs. The auction cost function is a utilitarian social cost function. To determine an

$c_{\text{schedule}} : \Pi \rightarrow \mathbb{R}$ {Cost for each schedule π}
for all $\pi \in \Pi$ **do**
    $t_{\text{start}}^{\pi_0} \leftarrow 0$
    $t_{\text{cross}}^{\pi_0} \leftarrow 0$
    for all $i = 1$ **to** I **do**
        $t_{\text{start}}^{\pi_i} \leftarrow t_{\text{start}}^{\pi_{i-1}} + t_{\text{cross}}^{\star\pi_{i-1}}$
    end for
    $c_{\text{schedule}}(\pi) \leftarrow \sum_{i \in \mathcal{I}} c_{\text{wait}}^{\pi_i}(t_{\text{start}}^{\pi_i})$
end for
return $\arg\min_{\pi \in \Pi} \; c_{\text{schedule}}(\pi)$

Fig. 5. Socially-optimal (SO) scheduling algorithm.

optimal crossing time vector $\boldsymbol{t}_{\text{cross}}^{\star}$, the auctioneer solves the following optimization problem:

$$\boldsymbol{t}_{\text{cross}}^{\star} \triangleq \arg\min \quad c_{\text{auction}}(\boldsymbol{t}_{\text{cross}}) \tag{8a}$$

$$\text{subj. to} \quad \underline{t}_{\text{cross}}^{i} \leq t_{\text{cross}}^{i} \leq \overline{t}_{\text{cross}}^{i} \quad \forall i \in \mathcal{I} \tag{8b}$$

$$\boldsymbol{g}(\boldsymbol{t}_{\text{cross}}) = 0 \tag{8c}$$

$$\boldsymbol{h}(\boldsymbol{t}_{\text{cross}}) \leq 0 \quad , \tag{8d}$$

where $\underline{t}_{\text{cross}}^{i}$ and $\overline{t}_{\text{cross}}^{i}$ are the lower and upper feasibility bounds for vehicle i's crossing cost function (submitted in ϕ^i), and $\boldsymbol{g}$ and $\boldsymbol{h}$ are any equality or inequality constraints, respectively, imposed on the intersection. For example, the total time required for all leading vehicles to cross could be penalized to encourage vehicles to cross quickly. It is the system designer's responsibility to ensure constraints $\boldsymbol{g}$ and $\boldsymbol{h}$ do not make the problem infeasible.

The auctioneer determines a crossing time allocation that minimizes the utilitarian social cost function subject to any constraints, thus producing a Pareto optimal outcome. If the outcome was not Pareto optimal, there would exist another allocation benefiting all agents and further reducing the auction cost, thus creating a contradiction. Note that because (8) is a nonlinear optimization problem, our approach guarantees *local* Pareto efficiency.

D. Scheduler

After the auctioneer assigns optimal crossing times to each vehicle, the scheduler determines the crossing order. We schedule vehicles in an order that maximizes the social welfare with respect to each vehicle's waiting cost (*i.e.*, minimizes their costs). We call this a socially-optimal (SO) scheduling algorithm.

Let $\pi \in \Pi$ be a permutation on $\mathcal{I}$ representing a crossing order and defined using Cauchy's two-line notation as

$$\pi \triangleq \begin{pmatrix} 1 & 2 & \cdots & I \\ \pi_1 & \pi_2 & \cdots & \pi_I \end{pmatrix} \quad . \tag{9}$$

Using this notation, we interpret π_i as the i^{th} vehicle to cross the intersection under permutation (crossing schedule) π. An optimal permutation $\pi^\star$ (*i.e.*, a socially optimal crossing schedule) is one that minimizes the sum of vehicles' waiting cost functions, the *schedule cost*. We define the schedule cost $c_{\text{schedule}} : \Pi \rightarrow \mathbb{R}$ as

$$c_{\text{schedule}}(\pi) \triangleq \sum_{i \in \mathcal{I}} c_{\text{wait}}^{\pi_i}(t_{\text{start}}^{\pi_i}) \quad . \tag{10}$$

The schedule cost is a utilitarian social cost function. An optimal permutation can be determined from the following optimization problem:

$$\pi^\star \triangleq \arg\min \quad \sum_{i=1}^{I} c_{\text{wait}}^{\pi_i}(t_{\text{start}}^{\pi_i}) \tag{11a}$$

$$\text{subj. to} \quad t_{\text{start}}^{\pi_i} = t_{\text{start}}^{\pi_{i-1}} + t_{\text{cross}}^{\star\pi_{i-1}} \quad , \tag{11b}$$

where $t_{\text{start}}^{\pi_0} = t_{\text{cross}}^{\star\pi_0} = 0$. Constraint (11b) is used to ensure only one vehicle crosses the intersection at a time, guaranteeing collision-free crossings. Fig. 5 describes an implementation of the SO scheduling algorithm. Because the implementation exhaustively evaluates all possible permutations, the resulting solution is globally Pareto optimal. If there are several Pareto optimal solutions, the scheduler will choose the first one it encountered.

Note that while the implementation given in Fig. 5 has a runtime complexity of $\mathcal{O}(II!)$, the number of lead vehicles I per auction and scheduling iteration is low enough to maintain tractability. Additionally, the evaluations can be parallelized to reduce computation time.

After generating the crossing schedule, the scheduler assigns a start time $t_{\text{start}}^{\pi_i^\star}$ to each vehicle under the optimal permutation (crossing order) $\pi^\star$. The start time for vehicle $\pi_i^\star$ is the sum of the start and crossing times of the previous vehicle:

$$t_{\text{start}}^{\pi_i^\star} \triangleq t_{\text{start}}^{\pi_{i-1}^\star} + t_{\text{cross}}^{\star\pi_{i-1}^\star} \quad \forall i = 1, 2, \ldots, I \quad , \tag{12}$$

where $t_{\text{start}}^{\pi_0^\star} = t_{\text{cross}}^{\star\pi_0^\star} = 0$. The IMS transmits to each vehicle $\pi_i^\star$ the pair $\langle t_{\text{start}}^{\pi_i^\star}, t_{\text{cross}}^{\star\pi_i^\star} \rangle$ as illustrated in Fig. 2b.

IV. Evaluation

A. Experimental Setup

We evaluated our auction-based IMS using the four-way intersection depicted in Fig. 1. Vehicle 1 planed to turn left, vehicle 2 planed to turn right, and vehicles 3 and 4 planed to drive straight. Lanes were $10\,\mathrm{m}$ wide, making the intersection a $20\,\mathrm{m} \times 20\,\mathrm{m}$ square. We assumed all vehicles were equipped with an Automated Driving System (ADS) capable of driving them to the intersection.

Each vehicle moved according to the unicycle dynamics model:

$$\dot{x}^i = v^i \cos\theta^i \tag{13a}$$

$$\dot{y}^i = v^i \sin\theta^i \tag{13b}$$

$$\dot{\theta}^i = \omega^i \quad . \tag{13c}$$

The state vector $\boldsymbol{x}^i \triangleq \begin{bmatrix} x^i & y^i & \theta^i \end{bmatrix}^\intercal$ included the vehicle's position (x^i and y^i) and heading θ^i. The system inputs $\boldsymbol{u}^i \triangleq \begin{bmatrix} v^i & \omega^i \end{bmatrix}^\intercal$ were the linear velocity v^i and angular velocity ω^i. We discretized the dynamics model in (13) using the 4th-order Runge–Kutta (RK4) method.

Each vehicle used a quadratic objective function for calculating their crossing cost:

$$J^i_{\text{cross}}(\boldsymbol{x}^i, \boldsymbol{u}^i, t^i_{\text{cross}}) \triangleq \alpha^i t^i_{\text{cross}} + \sum_{k=1}^{N(t^i_{\text{cross}})} \left(\|\boldsymbol{x}^i_k - \boldsymbol{r}^i_k\|_{\boldsymbol{Q}^i} + \|\boldsymbol{u}^i_k\|_{\boldsymbol{R}^i} \right) \quad . \tag{14}$$

The matrices $\boldsymbol{Q}^i$ and $\boldsymbol{R}^i$ were passenger-defined preferences on vehicle i's states and inputs, respectively, while scalar α^i indicated passengers' value of time. Passengers conveyed how quickly they wanted to proceed through the intersection using α^i, $\boldsymbol{Q}^i$, and $\boldsymbol{R}^i$.

The intuition behind the quadratic cost is that passengers prefer to cross the intersection slower because it would result in a more comfortable ride, meaning the cost would decrease as t^i_{cross} increased. However, passengers do not want to spend an unreasonable amount of time crossing the intersection, so they add a cost based on the overall crossing time.

For the waiting cost, we chose a linear function defined by

$$c^i_{\text{wait}}(t^i_{\text{wait}}) \triangleq \beta^i t^i_{\text{wait}} \quad , \tag{15}$$

where $\beta^i \geq 0$ was a passenger-defined parameter. A linear function makes sense because most passengers would prefer to cross the intersection sooner rather than later. Other monotonically increasing functions, such as the logarithm, might also be applicable. As passengers wait longer to cross, they may become indifferent to small time increases.

Table I contains each vehicle's passenger preferences used in (14) and (15). The values were determined experimentally and chosen to showcase preferences for different types of passengers (see Fig. 6). The red line represents passengers that are behind in their trip schedule, so they want to cross faster. Passengers depicted by the green line are ahead of schedule, meaning they prefer to drive slowly. The orange and blue lines illustrate costs for everyday drivers.

For ease of notation, we denote specific state and input weights with their variable as a subscript on the associated matrix (*e.g.*, $\boldsymbol{Q}_x$ denotes the weight for the x position). We set $\boldsymbol{Q}_\theta = 0$ for all vehicles. Vehicles 1 and 2 had terminal heading constraints ensuring they were aligned with their target lanes. Vehicles 3 and 4 drove straight, so their heading error penalty was 0 regardless of $\boldsymbol{Q}_\theta$. We restricted the crossing time set to $\mathcal{T}^i_{\text{cross}} = \{t \in \mathbb{R}_{>0} \mid 5 \leq t \leq 25\}$ for all vehicles. The values for β_i were chosen randomly from a uniform distribution between 0 and 40.

We implemented the IMS in Python using CasADi [16] and the Interior Point OPTimizer (IPOPT) [17] solver. The code is available in our GitHub repository.[1]

The crossing cost function $c^{\star i}_{\text{cross}}$ as described in Section II is continuous within its feasibility bounds. However, it would be computationally infeasible to compute a crossing cost for all $t^i_{\text{cross}} \in \mathcal{T}^i_{\text{cross}}$. Instead, we sampled different times within $\mathcal{T}^i_{\text{cross}}$ and use a spline function to generate $c^{\star i}_{\text{cross}}$ within the feasibility bounds.

[1] https://github.com/the-hive-lab/autocross

TABLE I
PASSENGER PREFERENCES PER VEHICLE

	Vehicle			
Parameter	1	2	3	4
x-position ($\boldsymbol{Q}_x$)	1.5	3	2	4
y-position ($\boldsymbol{Q}_y$)	1.5	3	2	4
Linear velocity ($\boldsymbol{R}_v$)	1	4	3	1.2
Angular velocity ($\boldsymbol{R}_\omega$)	1	1	4	3
Max linear velocity ($\overline{u}_v$)	-	2	5	-
Crossing time weight (α^i)	15	12	3	10

To demonstrate the adaptability of our system, we imposed an upper limit on the total crossing time. All vehicles must cross the intersection within $T = 50$ time units. Note that other values for T could have also been used. The modified optimization problem for the auctioneer is

$$\begin{aligned} \boldsymbol{t}^\star_{\text{cross}} \triangleq \arg\min \quad & c_{\text{auction}}(\boldsymbol{t}_{\text{cross}}) + \gamma s && \text{(16a)} \\ \text{subj. to} \quad & \underline{t}^i_{\text{cross}} \leq t^i_{\text{cross}} \leq \overline{t}^i_{\text{cross}} \quad \forall i \in \mathcal{I} && \text{(16b)} \\ & \sum_{i \in \mathcal{I}} t^i_{\text{cross}} \leq T + s && \text{(16c)} \\ & s \geq 0 \quad , && \text{(16d)} \end{aligned}$$

where T was the upper limit on the intersection clearing time. To ensure the problem remained feasible, we used a slack variable s to make the total crossing time a soft constraint. However, we add a penalty parameter, $\gamma > 0$, to allow for more strict adherence to the desired upper limit, T. In this paper's experiments, we set $\gamma = 100$.

We compared our SO scheduling algorithm against a random crossing schedule. Because intersection stalemates are broken arbitrarily for human-driven vehicles, this provided a realistic baseline. The crossing schedule for human-driven vehicles normally would be determined by whichever driver creeps into the intersection first. Additionally, to the best of our knowledge, no other works provide passenger satisfaction metrics for their proposed solutions.

B. Results

Fig. 6 shows the crossing costs for each vehicle and assigned crossing times. When there were no intersection-level constraints, vehicles were assigned their optimal crossing times, indicated by the black circles. However, when the intersection was constrained by (16c), it assigned slightly faster crossing times so that all vehicles crossed within the required clearing time.

Fig. 7 shows the schedule costs for different scheduling iterations. As shown in the figure, the SO scheduling algorithm consistently produced a more socially-favorable schedule compared to the random schedule. It also significantly reduced the overall schedule cost in each iteration. In rare occurrences, the two schedules were equivalent; however, the random schedule was never more beneficial than the SO one.

V. Conclusion

In this paper, we presented a novel auction-based intersection management system (IMS) that optimized crossing

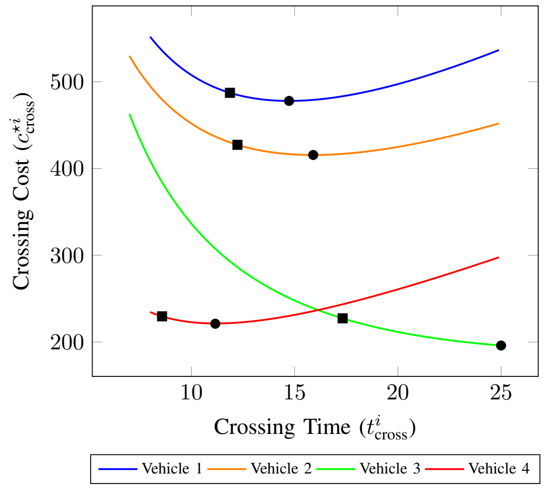

Fig. 6. Crossing costs for each vehicle. Black circles are assigned times without intersection constraints. Black squares are assigned times with intersection constraint (16c).

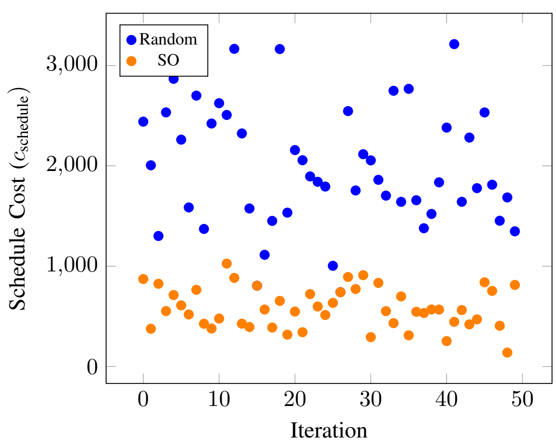

Fig. 7. Schedule costs for random and socially-optimal schedules.

schedules based on social welfare. Vehicles determined a cost curve for different crossing times based on their dynamics and passenger preferences. Those curves were used as bids, and the auctioneer assigned crossing times that maximized utility (*i.e.*, minimized cost) subject to any intersection-level constraints.

We demonstrated the usefulness of our system on a realistic intersection scenario in which multiple vehicles arrived simultaneously. Our IMS assigned cost-minimizing crossing times in both constrained and unconstrained scenarios. The scheduler similarly determined a crossing order that maximized social utility among the vehicles. The generated crossing schedules were consistently and significantly more favorable than the baseline random schedule that would be typical for human-driven vehicles. The simulations evaluated a single group of lead vehicles, but future work will evaluate our system against a continual flow of vehicles.

While this work presents a promising preliminary investigation, there are several extensions and enhancements. Two enhancements would be allowing multiple, non-conflicting vehicles to access the intersection simultaneously and evaluating against multi-lane roads. Throughout this work, we assumed vehicles submitted their crossing and waiting functions truthfully, leaving the IMS susceptible to strategic agents. A desirable property of mechanisms is that they are incentive compatible, meaning it is in the agents' best interests to report truthfully. An important next step will be to modify our mechanism (auction) to incentivize all vehicles to report their bids truthfully. Extensions include holding multiple auctions where vehicles need to compete only with others whose paths conflict, incorporating flow constraints to prioritize some roads over others, and coordinating with neighboring intersections.

References

[1] U.S. Department of Transportation, "About Intersection Safety," 3 2021. [Online]. Available: https://safety.fhwa.dot.gov/intersection/about/

[2] Virginia Department of Motor Vehicles, "The Virginia Driver's Manual," 12 2019.

[3] SAE International, "Taxonomy and Definitions for Terms Related to Cooperative Driving Automation for On-Road Motor Vehicles," SAE International, Tech. Rep., 2020.

[4] M. A. Guney and I. A. Raptis, "Scheduling-Based Optimization for Motion Coordination of Autonomous Vehicles at Multilane Intersections," *J. of Robot.*, vol. 2020, 2020.

[5] M. Stryszowski, S. Longo, E. Velenis, and G. Forostovsky, "A Framework for Self-Enforced Interaction Between Connected Vehicles: Intersection Negotiation," *IEEE Trans. on Intell. Transp. Syst.*, pp. 1–10, 2020.

[6] S. A. Fayazi and A. Vahidi, "Mixed-integer linear programming for optimal scheduling of autonomous vehicle intersection Crossing," *IEEE Trans. on Intell. Vehicles*, vol. 3, no. 3, pp. 287–299, 2018.

[7] Y. Xu, H. Zhou, T. Ma, J. Zhao, B. Qian, and X. Shen, "Leveraging Multiagent Learning for Automated Vehicles Scheduling at Nonsignalized Intersections," *IEEE Internet of Things J.*, vol. 8, no. 14, pp. 11 427–11 439, 7 2021.

[8] Q. Jin, G. Wu, K. Boriboonsomsin, and M. Barth, "Multi-agent intersection management for connected vehicles using an optimal scheduling approach," *2012 Int. Conf. on Connected Veh. and Expo (ICCVE)*, pp. 185–190, 5 2012.

[9] D. Carlino, S. D. Boyles, and P. Stone, "Auction-based autonomous intersection management," in *16th Int. IEEE Conf. on Intell. Transp. Syst. (ITSC 2013)*. IEEE, 2013, pp. 529–534.

[10] G. Cabri, L. Gherardini, and M. Montangero, "Auction-based Crossings Management," in *Proc. of the 5th EAI Int. Conf. on Smart Objects and Technol. for Social Good*, 2019, pp. 183–188.

[11] G. Cabri, L. Gherardini, M. Montangero, and F. Muzzini, "About auction strategies for intersection management when human-driven and autonomous vehicles coexist," *Multimedia Tools and Appl.*, vol. 80, no. 10, pp. 15 921–15 936, 2021.

[12] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic Theory*. New York, New York: Oxford University Press, 1995.

[13] S. M. LaValle, *Planning Algorithms*. New York, New York: Cambridge University Press, 2006.

[14] R. Rajamani, *Vehicle Dynamics and Control*, ser. Mechanical Engineering Series. Boston, MA: Springer US, 2012.

[15] S. Li, J. Lian, A. J. Conejo, and W. Zhang, "The Market-Based Coordination of Distributed Energy Resources," *IEEE Control Syst. Mag.*, vol. 40, no. 4, pp. 26–52, 2020.

[16] J. A. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: a software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, no. 1, pp. 1–36, 2019.

[17] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, 3 2006.