Autonomous Vehicle Management at Unsignalized Intersections without any Communication

Rugved katole[1], Arpita Sinha[1]

Abstract—**This paper addresses the traffic management problem for autonomous vehicles at intersections without traffic signals. In the current system, a road junction has no traffic signals when the traffic volume is low to medium. Installing infrastructure at each unsignalled crossing to coordinate autonomous cars can be formidable. We propose a novel decentralized strategy where the vehicles use a harmony matrix to find the best possible combination of the cars to cross the intersection without any crashes. This algorithm does not require communication between the vehicles. We compared our work with a strategy using communication between vehicles and infrastructure, and through extensive simulation, we showed that our algorithm is comparable when the traffic volume is low to medium.**

I. Introduction

Efficient management of unsignalized intersections is crucial for smooth undisrupted traffic flow. An inadequately managed unsignalized intersection can affect the network's signalized intersections as well as intelligent transportation systems (ITS) and may eventually lead to congestion and road accidents. In accordance with the data provided by the Ministry of Road Transport and Highways, India (MORTH), in 2021, a staggering 98,571 accidents were related to intersections, with a vast majority of 74.21% (i.e., 73,155) occurring at uncontrolled intersections [1].

In many towns and cities, uncontrolled intersections lacking proper infrastructure are common, particularly in areas with low traffic density. The absence of infrastructure can lead to safety issues and traffic flow disruptions. To mitigate these problems, the installation of the required infrastructure is imperative. As autonomous vehicles become more accessible, it is possible that intersections will be managed autonomously through a command center in the near future. Nevertheless, even with technological advancements, some infrastructure, such as a traditional traffic signal or a command center, will still be necessary for uncontrolled intersections. However, from an economic standpoint, building such infrastructure may come with exorbitant costs that may not justify its utility.

Given the potential risks associated with uncontrolled intersections, it is crucial to find effective ways of managing them. This question of intersection management becomes even more pressing with the increasing prevalence of autonomous vehicles. However, it remains unclear how such vehicles can navigate uncontrolled intersections without the aid of infrastructure or communication protocols like V2X/V2I/V2V (vehicle-to-everything/vehicle-to-infrastructure/vehicle-to-vehicle).

[1]The authors belong to the Systems and Control Engineering department, Indian Institute Of Technology, Bombay

In this paper, we propose a novel framework for fully autonomous vehicles to navigate intersections. The algorithm is designed to provide real-time output without relying on any infrastructure or communication protocols. The framework is specifically tailored for intersections where installing traffic signals or any other infrastructure is redundant due to low traffic density ($\leq 500PCUs/hr/lane$ refer section V). A preliminary ideation is formulated in [2]

II. Related work

Over the last decade, the development of Autonomous Intersection Management (AIM) has gained significant attention due to the pressing issues of traffic growth and on-road safety at intersections. With the increasing number of vehicles on the road and rising concerns over accidents and delays, AIM offers a promising solution to optimize traffic flow and minimize collisions through the integration of advanced sensors, algorithms, and communication technologies into connected vehicles and autonomous vehicles. According to recent data from the USA and India, intersections account for the majority of road accidents, highlighting the urgent need for innovative intersection management solutions like AIM [1], [3].

Traffic lights have been a commonly used tool for regulating traffic flow at intersections for many years. In most cases, the signal phase timings are determined using various parameters such as traffic density (PCUs/hr) and queue length. For instance, the Traffic Signal Timing manual by NCHRP provides guidelines for optimizing signal timings based on these factors [4]. However, despite their widespread use, traffic lights have been shown to be inefficient when it comes to minimizing waiting times at intersections. Prolonged waiting times lead to increased fuel consumption, resulting in higher levels of harmful emissions. Over the last decade, researchers have utilized data collection technologies (viz. connected autonomous vehicles (CAVs), mobile sensing, etc.) to gather real-time traffic data and optimize traffic phase timings based on parameters such as flow volume, travel time, queue length, and shockwave boundary [5].

The literature on intersection management classifies approaches into two main segments: V2I (vehicle-to-infrastructure communication), known as the Centralized approach, and V2V (vehicle-to-vehicle communication), referred to as the Distributed approach [6]. In the Centralized approach, a roadside infrastructure communicates with vehicles, performs necessary computations, and guides them safely through intersections. This method is particularly effective for high-traffic density intersections, as the infrastructure can handle heavy computational loads and periodically store traffic information, eliminating the need for continuous information broadcasting by vehicles [7], [8], [9]. However, a failure in the roadside infrastructure can lead to a breakdown of the entire system, which is not the case in the Distributed approach. In the Distributed approach, computations are either performed on the vehicles themselves or a designated vehicle assumes a leadership position and carries out the required computations[10], [11], [12]. This approach creates a more robust and scalable system, making it well-suited for low to medium-traffic density intersections. However, it relies on limited computational resources at the vehicle end and requires high communication and computational bandwidth to reach a consensus.

Two prominent approaches for intersection management are space-time reservation and trajectory planning. In a space-time reservation system, the intersection is divided into an occupancy grid, ensuring that no two autonomous vehicles (AVs) occupy the same grid cell at the same timestamp. Resources are allocated based on scheduling or priority policies. The First Come First Serve (FCFS) policy has proven to be effective, where vehicles communicate their arrival time to the infrastructure or other vehicles, and priority is given to the one with the shortest travel time [8]. Various heuristic and optimization-based algorithms exist for scheduling and reservation [13], [14].

On the other hand, in trajectory planning, vehicles follow predetermined paths to navigate the intersection safely. The trajectory planning layer considers parameters such as acceleration/deceleration, travel time, and the arrival of other vehicles to plan the optimal trajectory for each vehicle[15].

In the past decade, several surveys have been published, shedding light on various aspects of intersection management. Qureshi and Abdullah [16] delve into intelligent intersection management technologies and their practical applications. Li et al. [17] explore the relationship between traffic control systems and vehicular communication, comparing three pairs of control strategies: Big-Data-Based Versus Concise-Data-Based Controls, Model-Based Versus Simulation-Based Predictive Controls, and Planning-Based Versus Self-Organization-Based Controls. Turning to signalized and unsignalized intersections, Chen and Englund [18] provide a comprehensive review, emphasizing trajectory planning, virtual traffic light, and space-time reservation methods. On a related note, Ross-Torres and Malikopoulos [19] delve into heuristic and optimization-based scheduling policies from both centralized and decentralized control perspectives. Furthermore, Guo et al. [5] conduct an extensive examination of signalized intersection management using connected autonomous vehicles, addressing aspects such as flow estimation and the optimization of traffic signal timings. Namazi et al. [20] undertake a systematic review focused on signalized and unsignalized four-way intersection management, including a comparative analysis of goal satisfaction among different methods. Khayatian et al. [6] adopt a different perspective, discussing intersection management from multiple key angles and shedding light on the limitations associated with each approach. Their analysis provides valuable insights into the challenges and potential drawbacks of different intersection management strategies. Zhong et al. [21] survey autonomous intersection management, categorizing prior research into three hierarchical layers: corridor coordination, intersection management, and vehicle control. They also delve into the transition from signalized intersection management to autonomous intersection management, addressing challenges such as computation, collision avoidance, and priority policies. Finally, Gholamhosseinian et al. [22] conducted a thorough survey on various intersection management architectures, covering signalized, unsignalized, and hybrid intersections. Their analysis is based on the intersection of management goals like efficiency, safety, infotainment, and environmental considerations.

Contributions: To the best of our knowledge there are no works for intersection management without using communication protocols (V2V or V2I). This being the first work in this area our contributions are as follows:

- A autonomous intersection management algorithm capable of functioning without any dedicated infrastructure or communication between vehicles at low to medium traffic junctions.

III. Problem Formulation

Consider an unsignaled intersection with medium to low traffic density. The objective is to autonomously navigate the intersection without utilizing any vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), or vehicle-to-everything (V2X) communication. The intersection can have multiple connected roads, but to formulate the problem, we focus on a 4-way intersection as depicted in Figure 1; the effectiveness of the algorithm for 3-way and 5-way intersection are shown in the simulation. Since the intersection has low traffic density, it is reasonable to assume that each road at the intersection consists of one incoming lane and one outgoing lane. The vehicles approaching the intersection indicate their intentions using indicator lights (left, right, or none for going straight).

At each incoming lane, there are three distinct zones: *the red zone*, *the yellow zone*, and *the green zone* (refer to Figure 1). To ensure that vehicles can stop if necessary, we define *the red zone*, where vehicles decelerate to a speed of 20 km/hr. The length of *the red zone* depends on the average speed of approaching vehicles. Given that this intersection scenario is designed for low to medium-traffic situations, the length of *the red zone* is set to five times the length of a vehicle (i.e., 20 meters). In *the yellow zone*, vehicles are able to observe other vehicles present at the intersection. It is assumed that the length of the *yellow zone* is 1.5 times that of a passenger car, while the *green zone* is 0.7 times the length of a passenger car. If more than half of a vehicle has crossed into *the green zone*, it is not required to stop and can proceed through the intersection.

To enable autonomy each lane is identified by a lane id $a, b, c \ldots$ in a clockwise direction starting from absolute north. The letter also denotes the lane priorities, i.e., lane a has higher priority than lane b, and so on. The requirement of lane priority will be discussed later. A vehicle maneuver is denoted using 1,2,3 for left, straight, and right maneuvers, respectively. Refer

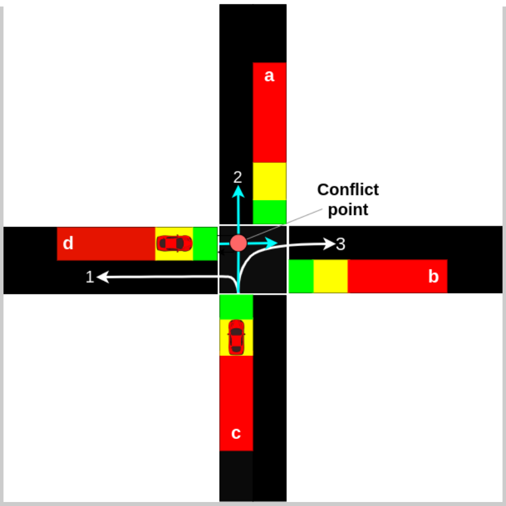


Figure 1: A 4-way intersection: White arrows represent the possible maneuver; Cyan arrow represents the intended maneuver; Pink dot represents the conflict point of the vehicles in lane "c" and "d".

to Figure 1 for illustration. V_j^i denotes a vehicle in lane $i \in a, b, c, \ldots$ with future maneuver $j \in 1, 2, 3$.

The problem is formalized with the following assumptions:

- The intersection does not have a traffic signal or roundabout. The traffic volume is low/medium.
- All vehicles at the intersection indicate their intended motion through the indicator light.
- The autonomous cars are able to perceive the intent of other vehicles once inside *the yellow region*.
- Vehicles do not take U-turns at the intersection
- Pedestrians are not present.

The special cases of tackling road or giving way to emergency vehicles like ambulances are not considered.

Problem III.1 (Autonomous Intersection Management)**.** *Given an un-signaled intersection I_n with n approaching lanes and a set of vehicles intending to cross the intersection, find a strategy to allow the vehicles to pass the intersection as soon as possible without any crashes.*

IV. Decentralized Intersection Management

The problem III.1 is addressed using a harmony matrix for an intersection I_n with n incoming lanes. A harmony matrix encompasses all feasible combinations of two maneuvers that can be executed simultaneously, ensuring coexistence. An example of a harmony matrix for a four-way intersection is illustrated in Table I.

Consider an n-way intersection with n incoming lanes. In this scenario, up to n vehicles will be present in the yellow zones of the intersection, and a unanimous decision is required to determine the right of way. To identify the optimal combination of vehicles that can pass simultaneously, we construct a graph with n nodes, where the harmony matrix defines the connections between these nodes. The search for the best possible combination corresponds to finding the largest fully connected sub-graph, a well-known problem in graph theory referred to as the *Maximal Clique Problem*[23]. A clique in an undirected graph refers to the complete subgraph. A maximal clique is a complete subgraph to which no more vertices can be added. The problem of finding the maximal clique is a fundamental problem in graph theory. The solution to this is achieved using the branch and bound method [24], [25].

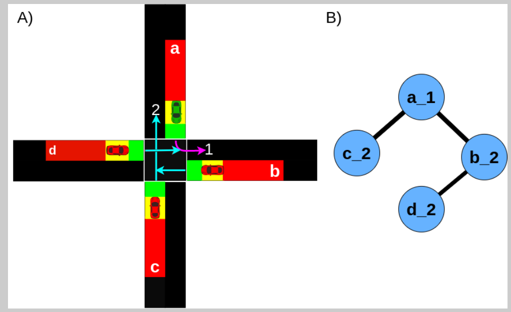


Figure 2: A) A scenario of 4 crossing vehicles with conflicting movements B) A graph generated using harmony matrix based on vehicle movements

The proposed solution is given in Algorithm 1. Whenever a vehicle approaches the intersection, the algorithm 1 is invoked. Initially, the vehicle enters the *Red Zone* of the intersection and reduces its speed to 20km/hr (line 3). Upon entering the *Yellow Zone*, the vehicle utilizes its camera to observe the surrounding vehicles and their intended maneuvers (line 5). Based on this information, a graph is constructed using the harmony matrix (line 6). The best possible combination of vehicles is determined by solving the maximal clique problem on the created graph (line 7). Various algorithms exist to solve this problem, and in our implementation, we utilize the *networkx* package. The vehicles identified in the solution are granted the right of way. This process is repeated until the vehicle successfully crosses the intersection.

Algorithm 1 Intersection management

Input: $\mathcal{M} \leftarrow$ *Harmony Matrix*

1: **while** $vehicle\ at\ intersection$ **do**
2:     **if** $vehicle\ in$ *Red Zone* **then**
3:         $Reduce\ Speed\ 20km/h$
4:     **else if** $vehicle\ in$ *Yellow Zone* **then**
5:         $IVehs \leftarrow Get\_Vehicles\_and\_their\_intent()$
6:         $Graph \leftarrow Create\_graph(IVehs, \mathcal{M})$
7:         $ROD \leftarrow Solve\_max\_clique\_prob(Graph)$
8:         **if** $Vehicle\ in\ ROD\ and\ No\ vehicles\ crossing$ **then**
9:             $cross\ the\ intersection$
10:         **end if**
11:     **else if** $vehicle\ in$ *Green Zone* **then**
12:         $cross\ the\ intersection$
13:     **end if**
14: **end while**

If there exist multiple cliques of the same length i.e. multiple sub-graphs of same size, then the clique containing the higher

Table I: Harmony matrix for a four-way intersection. In the matrix, 0 stands for conflict and 1 for harmony/coexistence

	V_1^a	V_2^a	V_3^a	V_1^b	V_2^b	V_3^b	V_1^c	V_2^c	V_3^c	V_1^d	V_2^d	V_3^d
V_1^a	0	0	0	1	1	1	1	1	0	1	0	1
V_2^a	0	0	0	0	0	0	1	1	0	1	0	0
V_3^a	0	0	0	1	0	0	0	0	1	1	0	0
V_1^b	1	0	1	0	0	0	1	1	1	1	1	0
V_2^b	1	0	0	0	0	0	0	0	0	1	1	0
V_3^b	1	0	0	0	0	0	1	0	0	0	0	1
V_1^c	1	1	0	1	0	1	0	0	0	1	1	1
V_2^c	1	1	0	1	0	0	0	0	0	0	0	0
V_3^c	0	0	1	1	0	0	0	0	0	1	0	0
V_1^d	1	1	1	1	1	0	1	0	1	0	0	0
V_2^d	0	0	0	1	1	0	1	0	0	0	0	0
V_3^d	1	0	0	0	0	1	1	0	0	0	0	0

priority lane is given the preference. The absolute lane priorities are defined using the lane IDs $a, b, c \ldots$ as given in section III. Figure 2 (A) shows a scenario where multiple vehicles have conflicting movements. Per the algorithm, we construct a graph using harmony matrix and vehicle movements as shown in Figure 2 (B). There are multiple cliques of the same length possible; (c_2, a_1), (a_1, b_2), (b_2, d_2). In this case, the clique with higher lane priority is selected, i.e., (a_1, b_2).

A. Deadlock Analysis

The proposed algorithm is designed to ensure deadlock-free operation during vehicle navigation through the intersection. Several key measures are incorporated to prevent the occurrence of deadlocks.

Firstly, the algorithm is activated only when a vehicle enters the intersection, and it terminates as soon as the vehicle obtains the right of way and successfully crosses the intersection. By limiting the algorithm's execution to the precise moment when the vehicle is actively engaged in the intersection, the likelihood of deadlocks occurring is greatly reduced. Furthermore, in situations where there are multiple cliques of the same length that can potentially pass, the algorithm takes into account lane priorities to select the optimal combination. This intelligent selection process helps avoid any eternal wait for a vehicle and eliminates the possibility of deadlocks arising from conflicting priorities.

Through extensive simulations and evaluations, it has been demonstrated that the algorithm effectively manages traffic flow without encountering deadlocks. Although vehicles with lower lane priorities may experience slightly longer waiting times, these delays are finite on a low traffic volume road. The robustness of the algorithm has been validated through various real-world simulations, ensuring an efficient traversal of vehicles through the intersection while guaranteeing deadlock-free operation.

Remark: As all the vehicles follow pre-determined non-conflicting paths to cross intersection, safety is assured. In case of discrepancies, a safety monitor can be used to enhance safety. Tian et al. [26] demonstrate the successful application of such a monitor for merging on roundabouts which can be extended in this case.

B. Synchronization

The algorithm runs at 100hz and the high frequency run eliminates the the requirement of synchronization. Along with lane priorities and harmony matrix the vehicles take unanimous decisions without communicating.

V. Simulation Setting

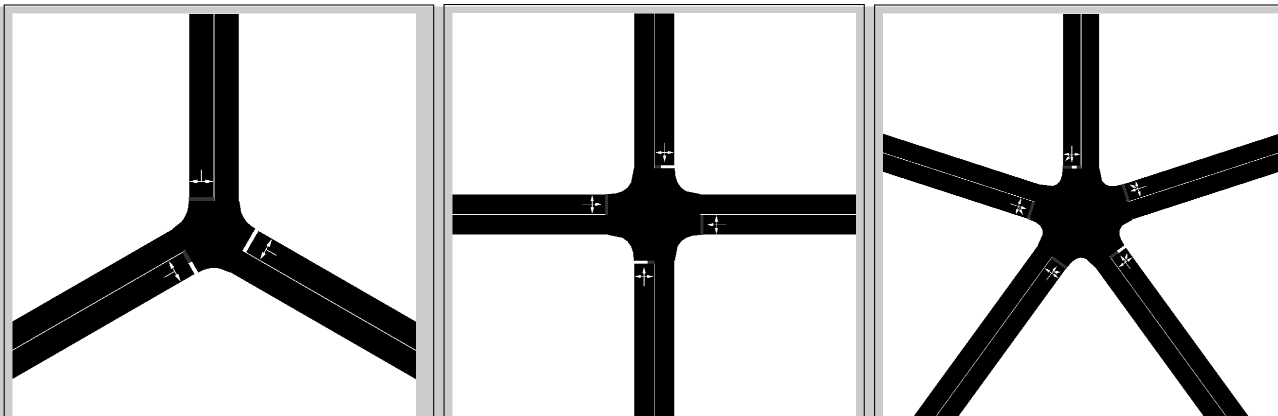
48 pt
0.667 in
16.9 mm

The efficacy of the developed algorithm is assessed by conducting comprehensive simulations that replicate real-world scenarios. In order to simulate realistic traffic conditions, we employ the Simulation of Urban MObility (SUMO) [27]. SUMO is an open-source traffic simulator renowned for its ability to handle large-scale traffic simulations. Within SUMO, the built-in functions are leveraged to facilitate motion planning, enabling seamless evaluation of intersection management algorithms. To facilitate real-time control of vehicles, the Traffic Control Interface (TraCI), a Python API specifically designed for SUMO, is utilized.

The developed algorithm is subjected to a simulation duration of 1 hour under conditions of low to medium traffic density. To determine the appropriate range for low to medium traffic density, we refer to the Manual on Uniform Traffic Control Devices (MUTCD), specifically Chapter 4 [28], which states that a traffic signal is warranted if the traffic volume on a single-lane road reaches 500 PCUs per hour or above for both directions. Thus, we conduct simulations across a spectrum of values: 150, 200, 250, 300, and 350 PCUs/hour/lane.

Intersection Types: The algorithm's robustness is evaluated on various types of intersections, including a three-way intersection with a 'Y' shape, a four-way junction, and a five-way junction. These intersections are characterized by roads extending up to 500m, each with a single incoming and outgoing lane. Figure 3 provides a visual representation of the three junctions that are considered for testing the algorithm's performance.

Real-world traffic modeling/distribution: The assumption of a uniform distribution of incoming vehicles is not realistic



(a) Three-way ('Y') Intersection (b) Four-way Intersection (c) Five-way Intersection

Figure 3: Various intersections for considered for analysis

in real-world scenarios. To model the arrival of vehicles more accurately, we employ a *Poisson Distribution*. The Poisson Distribution, represented by Equation 1, describes the probability of an event occurring a certain number of times within a fixed time interval.

$$P(k) = \frac{\lambda^k e^- \lambda}{k!} \tag{1}$$

In the equation, $P(k)$ denotes the probability of the event occurring k times, while λ represents the average number of events in the fixed time interval t. The Poisson distribution is particularly useful for modeling independent and random events, given the knowledge of the average occurrences within a specific time interval. In our context, we utilize the Poisson distribution to estimate the probability of incoming vehicles. The λ is the function of volume density per lane, ranging from 150 to 350 PCUs/hr/lane. These values serve as the average rate of events in the Poisson distribution, allowing us to calculate the probability of different numbers of vehicles arriving in a given time period.

VI. Results and Discussion

A. Evaluation Metrics

The performance of the intersection management algorithm is assessed based on two primary metrics: travel time and average waiting time.

1) Travel Time: Travel time is defined as the duration it takes for a vehicle to traverse a road segment extending 500 meters from the center of the intersection on each side. It encompasses the time from when a vehicle arrives at the intersection until it completely passes through the specified road segment.

2) Average Waiting Time: Average waiting time measures the duration that vehicles spend in a queue at the intersection before being able to proceed. It represents the average time a vehicle has to wait before entering the intersection and beginning its traversal through the road segment. The waiting time is calculated by summing up the waiting times of all vehicles in the queue and dividing it by the total number of vehicles.

By considering travel time and average waiting time, the evaluation provides a comprehensive understanding of the algorithm's impact on traffic flow, congestion, and overall intersection efficiency.

B. Comparison Models Reasoning

To assess the algorithm's effectiveness, we perform a comparative study using non-communicative and communicative methods defined below. The comparative study consists of two types of traffic distribution: 1. Balanced traffic and 2. Unbalance traffic. In balanced traffic, the traffic density is equal in all lanes, and in unbalanced traffic, the traffic densities are distributed non-uniformly, and the lane priorities are set accordingly.

1) Fixed-Time Traffic Signal (FTS): A fixed-time traffic signal is the most common method to control junction traffic. In a fixed-time traffic signal, the green time is fixed and does not change with respect to time. Webster's formula [29] is often used to determine the optimal cycle length and effective green time.

$$\text{Optimal cycle length}(Co) = \frac{1.5 * L + 5}{1 - y} \tag{2}$$

$$\text{Effective Green Time}(Ga) = \frac{ya/y}{Co - L} \tag{3}$$

where L represents the total lost time, including all red time, we set $L = 2n$, where n is the number of incoming lanes. y is the sum of critical ratios for each lane, which is the ratio of observed volume to saturation flow. Saturation flow refers to the maximum number of PCUs that can pass per hour. To establish the saturation flow, we refer to an empirical study by Kumar et al. [30]. In their study, they observed a busy three-legged junction in Vellore, India, and noted a peak traffic of 7573 PCUs per hour at the junction. Since the three-legged junction has 5 incoming lanes, we consider a saturation flow of approximately $7573/5 \approx 1500$ in our case. Table II displays phasing timings for the respective observed volumes. During the simulation, we utilize a 4-phase traffic signaling approach since there is a single incoming lane, and all movements are required during the green time.

Table II: Signal phase timing for fixed-time signal

Traffic Density	Green time(s)	Amber time(s)
150	5	2
200	7	2
250	10.75	2
300	19.75	2
350	61.75	2

2) Adaptive Traffic Signal (ATS): Adaptive traffic signals dynamically adjust to traffic demands, enhancing traffic flow. Real-time data is gathered through traffic sensors or cameras, and algorithms then adapt signal timings according to this data. We employ a delay-based algorithm developed by Oertel and Wagner[31] for comparison. This algorithm modifies green times based on queue sizes, adhering to maximum and minimum green time bounds. In practice, the algorithm is given fixed-time signals as per Table II, and then the algorithm modifies them.

3) Intersection Management using V2I protocols (V2I-C): We also conduct a comparison between our strategy and an algorithm that employs communication for intersection management. Li and Liu [7] utilize vehicle-to-infrastructure communication to gather data. Using this collected data, conflicts are identified through a predefined conflict matrix, and the infrastructure provides arrival times to vehicles. Vehicles then adapt their speeds based on these arrival times, ensuring a seamless passage through the intersection.

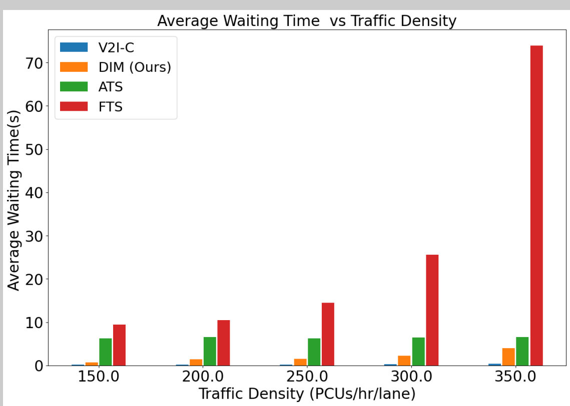


Figure 4: Comparative study on average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for balanced traffic

Figure 4 and 5 shows the comparison study between the 4 algorithms for Balanced traffic. The waiting time delay of our approach is lower than fixed-time traffic signals (FTS) and adaptive traffic signals (ATS) and comparable to V2I-C's at lower traffic densities. A similar trend is observed in travel time. At a traffic density of 350 PCUs/hr/lane, V2I-C's waiting time and travel time are 11.8 and 1.4 times lower than ours respectively, but considering the fact that they have used V2I communication for management, it is expected. The huge difference in the waiting time is due to the fact that in V2I-C the vehicles slow down before approaching the intersection, so that the vehicles do not stop at the intersection. On the contrary, in our Decentralized Intersection Management(DIM) algorithm the vehicles make a stop at intersection leading to reasonable waiting time. In this context, comparing travel times offers more accurate comparisions and our algorithm yeilds comparable results at low traffic densities without using any infrastructure.

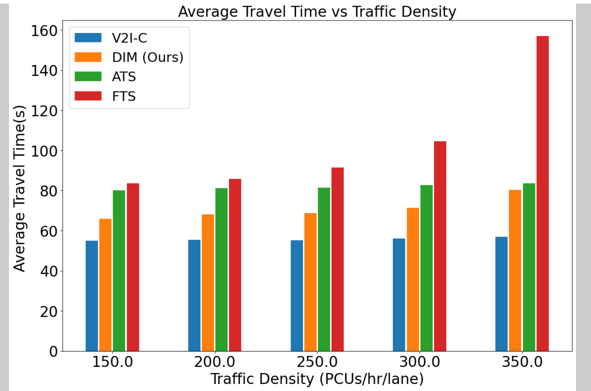


Figure 5: Comparative study on average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for balanced traffic

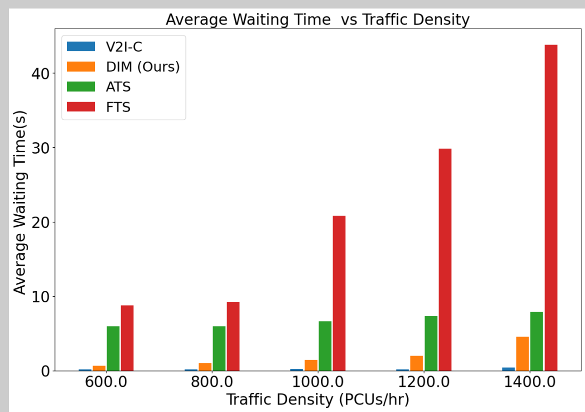


Figure 6: Comparative study on average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:3:2:1

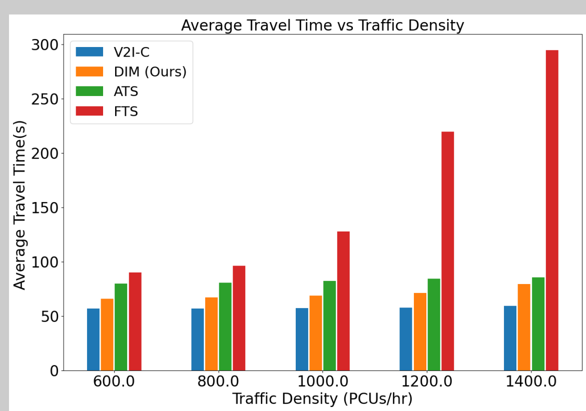


Figure 7: Comparative study on average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:3:2:1

In unbalanced traffic density, the total traffic density is divided into the ratio of 4:3:2:1 and 4:1:4:1, and the ratios denote the volume of traffic incoming from North:East:South:West. The lane priorities are set according to the incoming traffic, the highest being the one with the maximum incoming traffic. Figure 6 and 7 show the average waiting time and average travel time for unbalanced traffic. The performance of fixed-time and adaptive traffic signals declines, whereas our algorithm's performance is improved compared to balanced traffic. This is due to the fact that we use absolute lane priorities, which results in queuing in the least priority lanes for balanced traffic, whereas the queue length in unbalanced traffic is lower and a large amount of traffic is resolved faster by assigning higher priority to lane. The highest average travel time is within the bound of 50% of V2I-C algorithm with communication infrastructure.

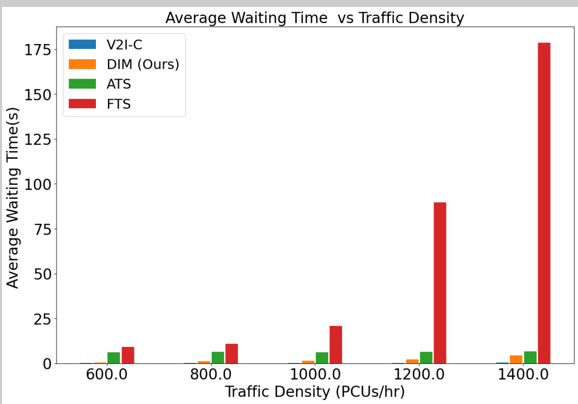


Figure 8: Comparative study on average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:1:4:1

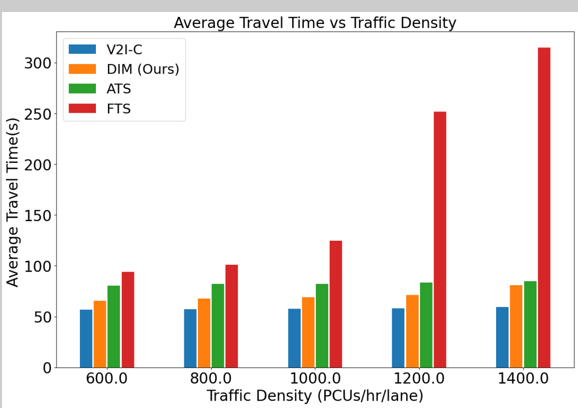


Figure 9: Comparative study on average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:1:4:1

Figure 8 and 9 depict the results for unbalanced traffic with a ratio of 4:1:4:1. The results show similar trend to unbalanced traffic with ration 4:3:2:1.

C. *Intersections types*

The versatility of the algorithm allows for seamless extension to intersections with n number of incoming lanes by adjusting the conflict matrix. Here, we showcase the algorithm's performance across 3-way, 4-way, and 5-way intersections. Figures 10 and 11 illustrate the average waiting time and average travel time for varying traffic densities, respectively.

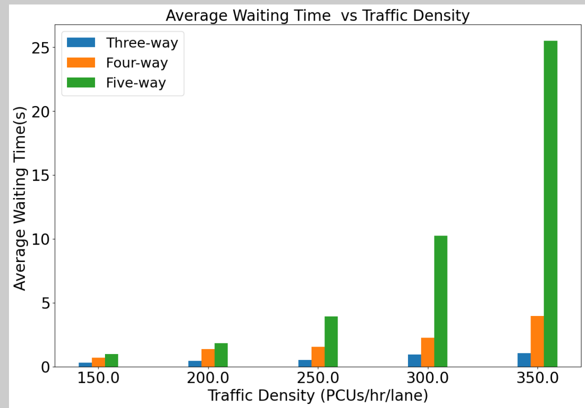


Figure 10: Average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for different intersections

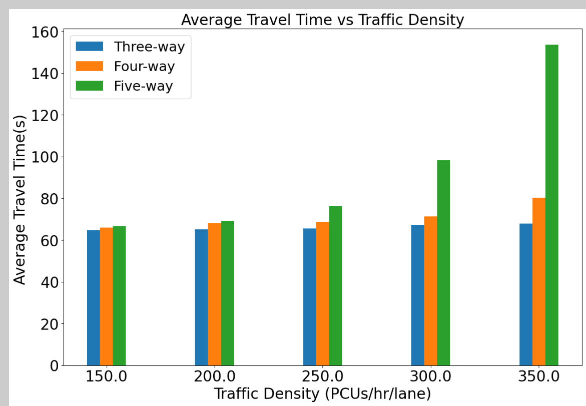


Figure 11: Average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for different intersections

The average waiting time increases with the number of incoming lanes as the traffic density is directly proportional to incoming lanes. The average time delay does not vary as much as the average waiting time but shows a similar trend. This is because the queue length in the least priority lane increases, leading to a higher waiting time for vehicles in the lane. Table III shows the average waiting time for each lane at a traffic density of 350 PCUs/hr/lane. It can be clearly observed that the values for the least priority lane (alphabetical priority is used as defined in problem formulation) values increase significantly.

VII. Conclusion and Future works

In this paper, we presented a novel framework for fully autonomous cars to navigate unsignalized intersections with low traffic density. The algorithm refrains from using any

Table III: Lane-wise waiting time (s) for 3-way, 4-way, 5-way intersections at a traffic density of 350 PCUs/hr/lane

Traffic Density	3-way	4-way	5-way
lane ’a‘	0.4606	0.8660	1.0970
lane ’b‘	0.9275	1.7263	2.0858
lane ’c‘	1.6415	2.8863	4.2837
lane ’d‘	N/A	10.2094	13.3636
lane ’e‘	N/A	N/A	146.1092

dedicated infrastructure or communication protocols for intersection management and relies only on sensory inputs to navigate the intersection safely. We have made assumptions about lane color codes and lane priorities that all vehicles must acknowledge. The expense of implementing color codes at the intersection is not substantial, constituting a one-time expenditure. Lane priorities can be incorporated into road markings alongside other relevant indicators. These solutions are more robust than the sophisticated V2I communication infrastructure. The efficacy of the algorithm was evaluated through a number of traffic simulations in SUMO against existing methodologies. It was observed that our algorithm performs better than current non-communicative methods (FTS and ATS) and provides comparable results to the communicative method (Li and Liu [7]). The algorithm is more suitable for unbalanced traffic environments due to its prioritizing nature. This approach can provide the solution to the uncontrolled intersections where basic infrastructure installation is also wasteful.

Future work can be pursued in a multitude of directions. We aim to take it forward by integrating the algorithm with a safety monitor[26] and testing with the Carla simulator[32]. With safety monitors in place, experiments can be carried out with mobile robots along with sensory inputs from cameras to identify other vehicle intents.

References

[1] ROAD ACCIDENTS IN INDIA ROAD ACCIDENTS IN INDIA ROAD ACCIDENTS IN INDIA. Technical report, 2021.

[2] Sushant Suryawanshi. Autonomous Driving at an Unsignalised Intersection: A Decision-making Scheme. Technical report, Indian Institute of Technology, Bombay, 2022.

[3] Crash Factors in Intersection-Related Crashes: An On-Scene Perspective. Technical report, 2010.

[4] T. (Thomas) Urbanik, National Research Council (U.S.). Transportation Research Board, National Cooperative Highway Research Program, American Association of State Highway and Transportation Officials, United States. Federal Highway Administration, Kittelson & Associates., Kimley-Horn and Associates., Texas A & M Transportation Institute., and Purdue University. *Signal timing manual*.

[5] Qiangqiang Guo, Li Li, and Xuegang (Jeff) Ban. Urban traffic signal control with connected and automated vehicles: A survey. *Transportation Research Part C: Emerging Technologies*, 101:313–334, 4 2019.

[6] Mohammad Khayatian, Mohammadreza Mehrabian, Edward Andert, Rachel Dedinsky, Sarthake Choudhary, Yingyan Lou, and Aviral Shrivastava. A Survey on Intersection Management of Connected Autonomous Vehicles, 8 2020.

[7] Yuying Li and Qipeng Liu. Intersection management for autonomous vehicles with vehicle-to-infrastructure communication. *PLOS ONE*, 15(7):e0235644, 7 2020.

[8] K. Dresner and P. Stone. A Multiagent Approach to Autonomous Intersection Management. *Journal of Artificial Intelligence Research*, 31:591–656, 3 2008.

[9] Kaidi Yang, S. Ilgin Guler, and Monica Menendez. Isolated intersection control for various levels of vehicle technology: Conventional, connected, and automated vehicles. *Transportation Research Part C: Emerging Technologies*, 72:109–129, 11 2016.

[10] Li Li and Fei-Yue Wang. Cooperative Driving at Blind Crossings Using Intervehicle Communication. *IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY*, 55(6), 2006.

[11] Alexander Katriniok, Pantelis Sopasakis, Mathijs Schuurmans, and Panagiotis Patrinos. Nonlinear Model Predictive Control for Distributed Motion Planning in Road Intersections Using PANOC.

[12] Alexander Katriniok, Stefan Kojchev, Erjen Lefeber, and Henk Nijmeijer. Distributed Scenario Model Predictive Control for Driver Aided Intersection Crossing. In *2018 European Control Conference (ECC)*, pages 1746–1752. IEEE, 6 2018.

[13] Masoud Bashiri and Cody H Fleming. A Platoon-Based Intersection Management System for Autonomous Vehicles.

[14] Dustin Carlino, Stephen D Boyles, and Peter Stone. Auction-based autonomous intersection management.

[15] Bing Liu, Qing Shi, Zhuoyue Song, and El Kamel. Trajectory planning for autonomous intersection management of connected vehicles. 2018.

[16] Kashif Naseer Qureshi and Abdul Hanan Abdullah. A survey on intelligent transportation systems. *Middle East Journal of Scientific Research*, 15(5):629–642, 2013.

[17] Li Li, Ding Wen, and Danya Yao. A survey of traffic control with vehicular communications. *IEEE Transactions on Intelligent Transportation Systems*, 15(1):425–432, 2 2014.

[18] Lei Chen and Cristofer Englund. Cooperative Intersection Management: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 17(2):570–586, 2 2016.

[19] Jackeline Rios-Torres and Andreas A. Malikopoulos. A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps. *IEEE Transactions on Intelligent Transportation Systems*, 18(5):1066–1077, 5 2017.

[20] Elnaz Namazi, Jingyue Li, and Chaoru Lu. Intelligent Intersection Management Systems Considering Autonomous Vehicles: A Systematic Literature Review, 2019.

[21] Zijia Zhong, Mark Nejad, and Earl E. Lee. Autonomous and Semiautonomous Intersection Management: A Survey, 6 2021.

[22] Ashkan Gholamhosseinian and Jochen Seitz. A Comprehensive Survey on Cooperative Intersection Management for Heterogeneous Connected Vehicles. *IEEE Access*, 10:7937–7972, 2022.

[23] Wikipedia. Clique Problem, 2023.

[24] Coen Bron and Joep Kerbosch. Algorithm 457 Finding All Cliques of an Undirected Graph [H] The following properties obviously hold.

[25] Etsuji Tomita, Akira Tanaka, and Haruhisa Takahashi. The worst-case time complexity for generating all maximal cliques and computational experiments. *Theoretical Computer Science*, 363:28–42, 2006.

[26] Ran Tian, Liting Sun, Andrea Bajcsy, Masayoshi Tomizuka, and Anca D. Dragan. Safety Assurances for Human-Robot Interaction via Confidence-aware Game-theoretic Human Models. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 11229–11235. IEEE, 5 2022.

[27] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic Traffic Simulation using SUMO. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 2575–2582. IEEE, 11 2018.

[28] U.S. Department of Transportation FHWA. Manual on Uniform Traffic Control Devices for Streets and Highways.

[29] WEBSTER F V. Traffic signal setting. *Road Research Technical Paper*, 39, 1958.

[30] S. Vasantha Kumar, P. Priyadharshini, B. Dhanalakshmi, and K. Settu. Study of Traffic Flow at a Three Legged Busy Intersection in Vellore, India. In *IOP Conference Series: Materials Science and Engineering*, volume 396. Institute of Physics Publishing, 8 2018.

[31] Robert Oertel and Peter Wagner. Delay-Time Actuated Traffic Signal Control for an Isolated Intersection. In *Proceedings 90st Annual Meeting Transportation Research Board (TRB)*, 8 2011.

[32] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An Open Urban Driving Simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*, pages 1–16, 2017.