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 less than 500 PCUs/hr/lane.**

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]. Similarly, in the EU, 20% of fatalities are intersection-related, and in the USA, 21.5% of intersection crashes lead to fatalities out of a total of 40% [2]. The data suggests intersections account for the majority of road accidents, highlighting the urgent need for innovative intersection management solutions like Autonomous Intersection Management (AIM) [3].

In many towns and cities, uncontrolled intersections lacking proper infrastructure are common, particularly in areas with low traffic density. The absence of infrastructure can cause traffic flow disruptions leading to safety issues. 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).

In this paper, we propose a novel framework specifically tailored for intersections where installing traffic signals or any other infrastructure is redundant due to low traffic density ($\leq 500 PCUs/hr/lane$ refer section V). The algorithm is designed to enable fully autonomous vehicles to navigate through intersections without relying on any infrastructure or communication protocols. This is achieved by building a graph using a harmony matrix, and the maximal clique of the graph provides the best combination of vehicles to enter the intersection.

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

II. Related work

The development of 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 by integrating advanced sensors, algorithms, and communication technologies into connected vehicles and autonomous vehicles.

Several surveys in the past decade have shed light on various aspects of intersection management. Qureshi and Abdullah [4] discuss intelligent intersection management technologies and their practical applications. Li et al. [5] explore the relationship between traffic control systems and vehicular communication, while Chen and Englund [2] provide a comprehensive review of signalized and unsignalized intersections, emphasizing trajectory planning, virtual traffic light, and space-time reservation methods. Ross-Torres and Malikopoulos [6] delve into heuristic and optimization-based scheduling policies, while Guo et al. [7] conduct an extensive examination of signalized intersection management using connected autonomous vehicles. Namazi et al. [8] undertake a systematic review focused on signalized and unsignalized four-way intersection management, and Khayatian et al. [9] analyze the potential drawbacks of different intersection management strategies. Zhong et al. [10] survey autonomous intersection management, addressing challenges such as computation, collision avoidance, and priority policies. Finally, Gholamhosseinian et al. [11] conducted a thorough survey covering signalized, unsignalized, and hybrid intersections.

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 [12]. However, despite their widespread use, traffic lights have been shown to be inefficient in minimizing waiting times at intersections. Prolonged waiting times lead to increased fuel consumption, resulting in higher levels of harmful emissions.

On the contrary, Adaptive traffic lights utilize 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 [7]. However, a major obstacle to its widespread adaptation is cost. On average, the cost of installing an adaptive traffic control system is approximately 65,000 USD and won't even be justified for low-traffic intersections [13].

Another class of intelligent traffic control systems leverages the communication infrastructure. Connected autonomous vehicles (CAV) use communication protocols viz vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) for decision-making on intersections. In vehicle-to-infrastructure, CAVs communicate with a roadside infrastructure requesting the right of way by providing the necessary parameters. For instance, Li and Liu [14] communicate CAV routes and departure timestamps. The infrastructure provides maximum time after which the CAVs can cross the intersections safely. Dresner and Stone [15] use a space-reservation method. CAVs make reservation requests with arrival time, velocity, and vehicle characteristics. Then, the request is accepted or rejected based on a First-Come-First-Serve policy (FCFS) and other conflict resolutions. Yang et al. [16] use position information to find the optimal departure sequence and trajectory to minimize time delay. However, a failure in the road infrastructure could lead to system breakdown and safety hazards.

In Vehicle-to-Vehicle architecture, CAVs broadcast their parameters to establish a connection with other vehicles. CAVs then perform a distributed computation, or one of the CAVs assumes a leadership position and guides them through the intersection. For instance, Li and Wang [17] use V2V to exchange current and desired lane, speed profile, driving plan, position, speed, and emergency signals to build a movement spanning tree, then a trajectory with the least execution time is selected. Katriniok et al. [18] share vehicle trajectories to obtain an optimal trajectory through a non-linear model predictive controller. However, they rely on limited computational resources at the vehicle end and require high communication and computational bandwidth to reach a consensus. These approaches are similar to ours, where vehicles individually perform the calculations, except with our approach, we eliminate the need for broadcasting and heavy computation.

Other intersection management techniques rely only on onboard perception for their decision-making. To make up for uncertainty in perceived sensor data, scholars have used partially observed Markov decision processes (POMDPs) for decision-making. For instance, Zhang et al. [19] use the concept of pseudo-vehicles for occluded vehicles by solving a POMDP formulation. The motion planner then uses these pseudo-vehicles to generate safe trajectories at crosswalks and intersections. Similarly, Xia et al. [20] use a belief updater along POMDPs for trajectory planning. Nan et al. [21] uses a combination of the Hidden Markov Model, Gaussian Mixture Model, and Support Vector Machines for intent prediction. Following this, a simple bezier curve is fitted on the predicted trajectory, and Mixed Strategy, Nash Equilibrium theory, is used to decide whether to yield or cross. However, POMPDs and game-theoretic methods suffer through the curse of dimensionality, and their computational complexity hinders real-time implementation. Another popular method for trajectory prediction is using Recurrent Neural Networks [22], [23]

Aksjonov et al. [24] uses a rule-based technique to address a similar problem to ours. They use the rule-based technique for the accommodation of human-driven vehicles at intersections. The open nature of our algorithm allows easy integration of a rule-based system to integrate human-driven vehicles at lesser computational and infrastructural costs.

In this paper, we propose an algorithm for autonomous intersection management without using infrastructure or communication protocols. The algorithm uses a deterministic harmony matrix to build a graph among the participating vehicles. Further solving for a maximal clique on the graph gives a unanimous solution that maximizes the throughput of the intersection. The algorithm is characterized by the following

- The proposed algorithm uses a cost-effective method for decision-making at unsignaled intersections.
- The method does not rely on V2I or V2V communication protocol and is based on available onboard sensors (e.g., camera)
- The algorithm is free from deadlocks and can provide real-time computations

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 algorithm's effectiveness for 3-way and 5-way intersections is 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). For simplicity, we have considered that the intent of the incoming vehicle is available. However, this assumption can be easily relaxed through the use of PODMPS [19] or Neural Networks [22], [25].

To make strategic and unanimous decisions at each incoming lane, there are three distinct zones: the *red zone*, the *yellow zone*, and the *green zone* (refer to Figure 1). The *red zone* is defined to decelerate the vehicles to a safer speed (assumed 20km/hr) to reduce the fatality in case of an accident. The vehicles moving at high speed might be unable to stop if not given a right of way. The length of *the red zone* depends on the average speed of approaching vehicles and the average deceleration rate. Assuming the average rate of deceleration of the passenger car to be *g* (9.81) and approaching the speed of 70 km/hr, the length of the *red zone* comes around 20 m using the third equation of motion.

To have halted until the right of way is not permitted, we have the *yellow zone*. While in the *yellow zone*, vehicles can observe other vehicles present at the intersection and continuously check for the right of way using the algorithm. The length of the yellow zone has to be at least equal to the length of the car so that the vision algorithm perceiving intent doesn't confuse it with other vehicles. Considering different vehicle types, the length of the *yellow zone* is 1.5 times of a passenger car.

The *green zone* acts as a safety buffer when crossing an intersection. Since the algorithm operates decentralized at 100Hz, there may be instances where a new vehicle enters the intersection after a decision has been made but before the previous vehicles can enter the intersection. Consequently, the new decision might not allow the right of way to vehicles that were permitted in the previous iteration. The *green zone* functions as a safety buffer in such cases, preventing vehicles from entering the intersection in an undesired scenario. Once a vehicle has crossed over half of the *green zone*, it is granted the right of way by others. The length of the *green zone* is determined based on the frequency of such interventions. In low-traffic areas, where interventions are less frequent, the *green zone* is set to be half the length of a passenger car.

To enable autonomy, each lane is identified by a lane id $a, b, c \dots$ 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, as illustrated in 1. V_j^i denotes a vehicle in lane $i \in a, b, c, \dots$ with future maneuver $j \in 1, 2, 3$.

Note: In this particular work, we haven't considered human-driven vehicles and have assumed all vehicles are autonomous vehicles.

IV. Decentralized Intersection Management

The problem in section III 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. The harmony matrix defines the connections between the vehicles in their respective lanes and their associated maneuver. If any two vehicle maneuvers are non-conflicting, then the value in the harmony matrix is 1 and 0 otherwise. For instance, table I demonstrates a harmony matrix for a 4-way intersection similar to one in figure 1. The m^{th} column of the harmony matrix shows the co-existence of the V_j^i maneuver with all other maneuvers.

Consider an n-way intersection with n incoming lanes. In this scenario, up to n vehicles will be present in the

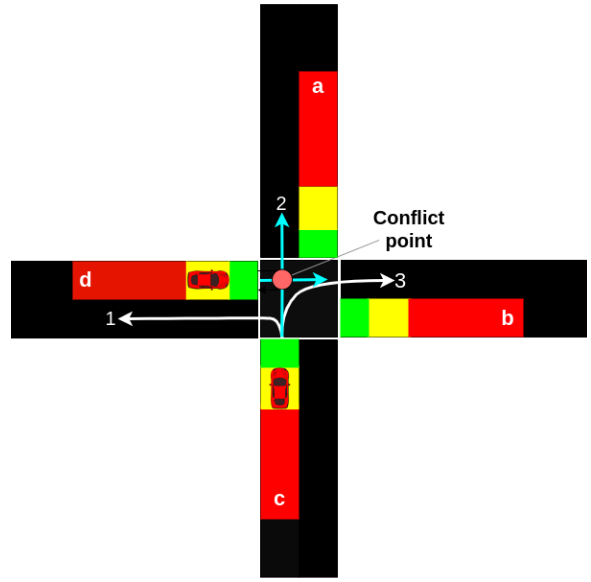


Figure 1: A 4-way intersection: White arrows represent the possible maneuver; Cyan arrow represents the intended maneuver; Pink dot represents the conflict point of the vehicles in lanes "c" and "d"

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, i.e., an edge will exist if there is harmony between two maneuvers. 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* [26]. 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 [27], [28].

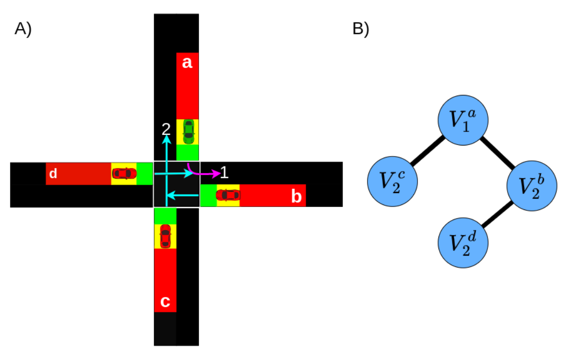


Figure 2: A) A scenario of 4 crossing vehicles with conflicting movements B) A graph generated using a harmony matrix based on vehicle movements

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

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

reduces its speed to 20km/hr (line 3). Upon entering the *Yellow Zone*, the vehicle utilizes its onboard sensors to observe the surrounding vehicles and find 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:         $IVehs \leftarrow$ Vehicles and their intent
6:         $Graph \leftarrow Create\_graph(IVehs, \mathcal{M})$
7:         $RoW \leftarrow Solve\_max\_clique\_prob(Graph)$
8:         **if** $Vehicle\ in\ RoW\ and\ No\ vehicles\ crossing$ **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 the same size, then the clique containing the higher 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.

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. For instance, Figure 2 (A) shows a scenario where multiple vehicles have conflicting movements. As 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; (V_2^c, V_1^a), (V_1^a, V_2^b), (V_2^b, V_2^d). In this case, the clique with higher lane priority is selected, i.e., (V_1^a, V_2^b).

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 algorithm's robustness has been validated through various real-world simulations, ensuring an efficient traversal of vehicles through the intersection while guaranteeing deadlock-free operation.

Note The algorithm runs at 100 Hz, and the high-frequency run eliminates the synchronization requirement. Along with lane priorities and harmony matrix, the vehicles make unanimous decisions without communicating.

Remark 1: 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. [29] demonstrate the successful application of such a monitor for merging on roundabouts, which can be extended in this case.

V. Simulation Setting

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) [30]. 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 [31], 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 various 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 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 intersection management algorithm's performance is assessed using two key metrics: travel time and average waiting time. These metrics offer a comprehensive evaluation of the algorithm's impact on traffic flow, congestion, and overall intersection efficiency.

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.

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.

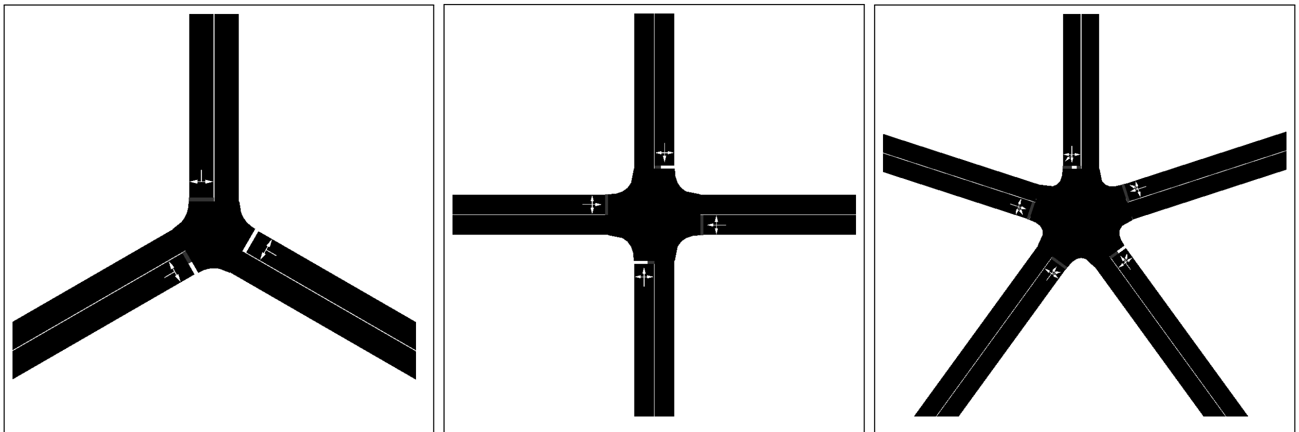
1) Fixed-Time Traffic Signal (FTS): A fixed-time traffic signal is the most common method to control junction traffic. In a fixed-time traffic signal, the green time is fixed and does not change with respect to time. Webster's formula [32] is often used to determine the optimal cycle length and effective 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. [33]. 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.

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[34] 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.



(a) Three-way ('Y') Intersection

(b) Four-way Intersection

(c) Five-way Intersection

Figure 3: Various intersections for considered for analysis

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

3) Intersection Management using V2I protocols (V2I-C): We also compare our strategy and an algorithm that employs communication for intersection management. Li and Liu [14] utilize vehicle-to-infrastructure communication to gather data. Using this collected data, conflicts are identified through a predefined conflict matrix, and the infrastructure provides vehicle arrival times. Vehicles then adapt their speeds based on these arrival times, ensuring a seamless passage through the intersection.

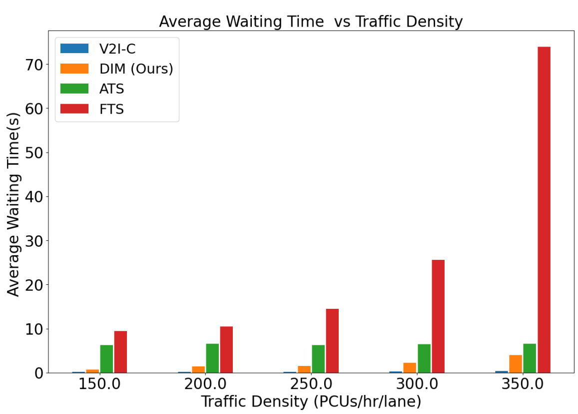


Figure 4: Comparative study on average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for balanced traffic

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.

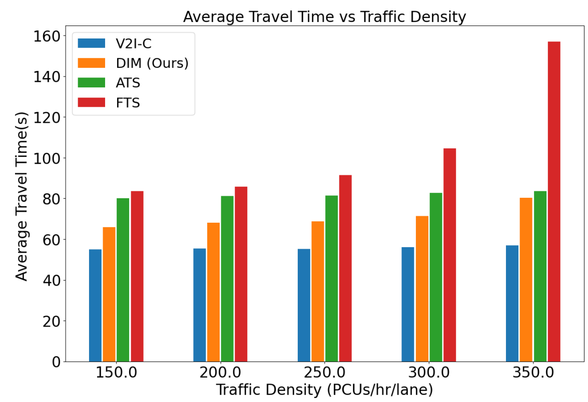


Figure 5: Comparative study on average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for balanced traffic

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

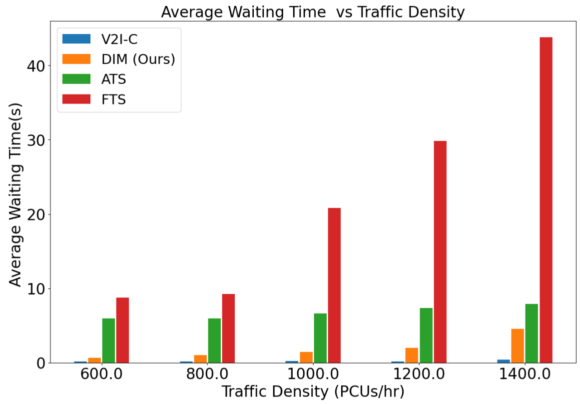


Figure 6: Comparative study on average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:3:2:1

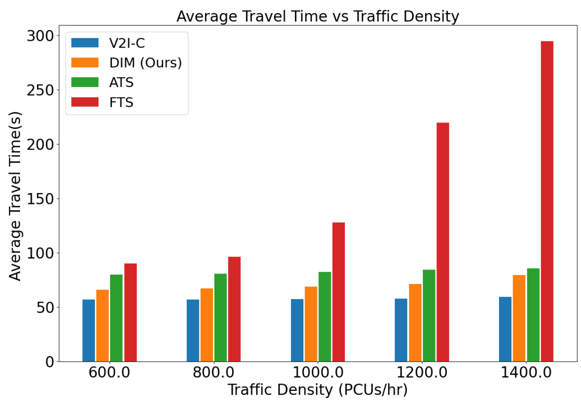


Figure 7: Comparative study on average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:3:2:1

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 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.

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

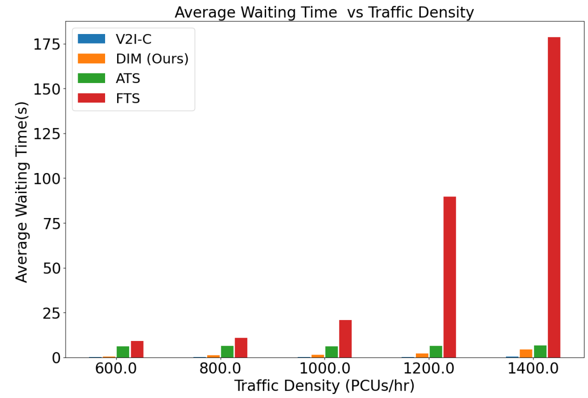


Figure 8: Comparative study on average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:1:4:1

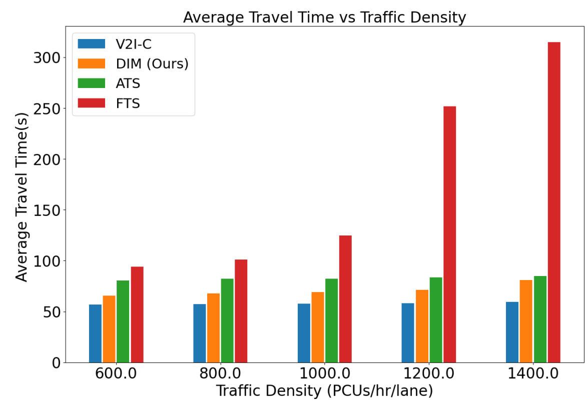


Figure 9: Comparative study on average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for unbalanced traffic ratio of 4:1:4:1

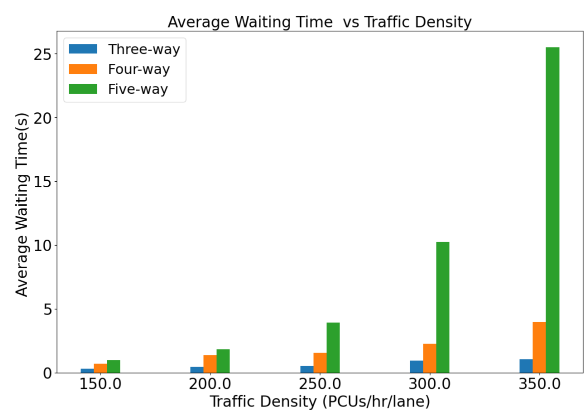


Figure 10: Average Waiting Time(s) vs Traffic Density (PCUs/hour/lane) for different intersections

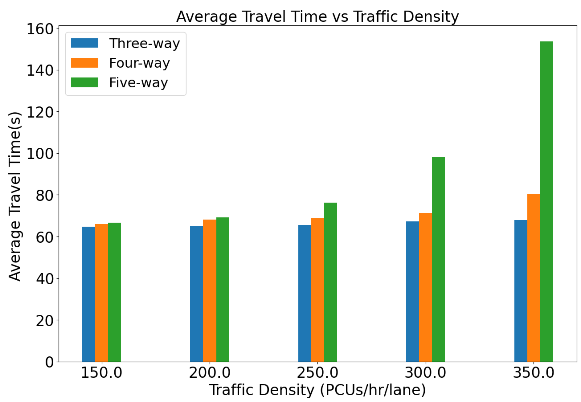


Figure 11: Average Travel Time(s) vs Traffic Density (PCUs/hour/lane) for different intersections

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

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 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. 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 [14]). The algorithm is more suitable for unbalanced traffic environments due to its prioritizing nature. This approach can solve the uncontrolled intersections where basic infrastructure installation is also wasteful. In this work, we have made a strong assumption about perceiving the intent of other vehicles, which might not hold in the real world. In our future work, we would be relaxing this assumption by accommodating the perceptions' uncertainties using a belief model or through Convolution Neural Networks. Based on the belief of the intent, the vehicle can make decisions with a certain confidence. Another direction we want to extend this would be accommodating human-driven vehicles through a rule-based method. This would be necessary as we transition from human-driven to fully autonomous vehicles.

References

[1] Transport Research Wing. Road Accidents In India,. Technical report, Ministry Of Road Transport and Highways, 2021.
[2] Lei Chen and Cristofer Englund. Cooperative Intersection Management: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 17(2):570–586, 2 2016.
[3] Crash Factors in Intersection-Related Crashes: An On-Scene Perspective. Technical report, 2010.
[4] Kashif Naseer Qureshi and Abdul Hanan Abdullah. A survey on intelligent transportation systems. *Middle East Journal of Scientific Research*, 15(5):629–642, 2013.
[5] 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.
[6] 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.
[7] 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.
[8] Elnaz Namazi, Jingyue Li, and Chaoru Lu. Intelligent Intersection Management Systems Considering Autonomous Vehicles: A Systematic Literature Review, 2019.
[9] Mohammad Khayatian, Mohammadreza Mehrabian, Edward Andert, Rachel Dedinsky, Sarthake Choudhary, Yingyan Lou, and Aviral Shrivastava. A Survey on Intersection Management of Connected Autonomous Vehicles. *ACM Transactions on Cyber-Physical Systems*, 4(4):1–27, 10 2020.
[10] Zijia Zhong, Mark Nejad, and Earl E. Lee. Autonomous and Semiautonomous Intersection Management: A Survey, 6 2021.
[11] Ashkan Gholamhosseinian and Jochen Seitz. A Comprehensive Survey on Cooperative Intersection Management for Heterogeneous Connected Vehicles. *IEEE Access*, 10:7937–7972, 2022.
[12] Tom Urbanik, Alison Tanaka, Bailey Lozner, Eric Lindstrom, Kevin Lee, Shaun Quayle, Scott Beaird, Shing Tsoi, Paul Ryus, Doug Gettman, Srinivasa Sunkari, Kevin Balke, and Darcy Bullock. *Signal Timing Manual - Second Edition*. Transportation Research Board, Washington, D.C., 9 2015.
[13] Aleksandar Stevanovic. *Adaptive Traffic Control Systems: Domestic and Foreign State of Practice*. National Academies Press, Washington, D.C., 4 2010.
[14] Yuying Li and Qipeng Liu. Intersection management for autonomous vehicles with vehicle-to-infrastructure communication. *PLOS ONE*, 15(7):e0235644, 7 2020.
[15] K. Dresner and P. Stone. A Multiagent Approach to Autonomous Intersection Management. *Journal of Artificial Intelligence Research*, 31:591–656, 3 2008.
[16] 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.
[17] Li Li and Fei-Yue Wang. Cooperative Driving at Blind Crossings Using Intervehicle Communication. *IEEE Transactions on Vehicular Technology*, 55(6):1712–1724, 11 2006.
[18] Alexander Katriniok, Pantelis Sopasakis, Mathijs Schuurmans, and Panagiotis Patrinos. Nonlinear Model Predictive Control for Distributed Motion Planning in Road Intersections Using PANOC. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 5272–5278. IEEE, 12 2019.

[19] Chi Zhang, Florian Steinhauser, Gereon Hinz, and Alois Knoll. Improved Occlusion Scenario Coverage with a POMDP-based Behavior Planner for Autonomous Urban Driving. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 593–600. IEEE, 9 2021.

[20] Canming Xia, Mali Xing, and Shenghuang He. Interactive Planning for Autonomous Driving in Intersection Scenarios Without Traffic Signs. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):24818–24828, 12 2022.

[21] Jiangfeng Nan, Weiwen Deng, and Bowen Zheng. Intention Prediction and Mixed Strategy Nash Equilibrium-Based Decision-Making Framework for Autonomous Driving in Uncontrolled Intersection. *IEEE Transactions on Vehicular Technology*, 71(10):10316–10326, 10 2022.

[22] Alex Zyner, Stewart Worrall, and Eduardo Nebot. Naturalistic Driver Intention and Path Prediction Using Recurrent Neural Networks. *IEEE Transactions on Intelligent Transportation Systems*, 21(4):1584–1594, 4 2020.

[23] Nachiket Deo and Mohan M. Trivedi. Multi-Modal Trajectory Prediction of Surrounding Vehicles with Maneuver based LSTMs. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1179–1184. IEEE, 6 2018.

[24] Andrei Aksjonov and Ville Kyrki. Rule-Based Decision-Making System for Autonomous Vehicles at Intersections with Mixed Traffic Environment. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 660–666. IEEE, 9 2021.

[25] Subhash Chand Agrawal, Rajesh Kumar Tripathi, Ashutosh Gupta, and Manish Sahu. Traffic Sign Recognition System using CNN. In *2023 World Conference on Communication & Computing (WCONF)*, pages 1–5. IEEE, 7 2023.

[26] Wikipedia. Clique Problem, 2023.

[27] Coen Bron and Joep Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16(9):575–577, 9 1973.

[28] 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.

[29] 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.

[30] 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.

[31] Manual on Uniform Traffic Control Devices for Streets and Highways. *U.S. Department of Transportation FHWA*, 2009.

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

[33] 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.

[34] 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.