Transportation Research Part C 114 (2020) 463–483

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Stability-based analysis of autonomous intersection management with pedestrians

Rongsheng Chen[a,*], Jeffrey Hu[b], Michael W. Levin[a], David Rey[c]

[a] Department of Civil, Environmental, and Geo-Engineering, University of Minnesota, Minneapolis, MN 55455, United States
[b] Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, United States
[c] School of Civil and Environmental Engineering, The University of New South Wales, Sydney, NSW 2052, Australia

ARTICLE INFO

Keywords:
Autonomous intersection management (AIM)
Max-pressure control
Pedestrians

ABSTRACT

With the development of vehicle-to-infrastructure and vehicle-to-vehicle technologies, vehicles will be able to communicate with the controller at the intersection. Autonomous driving technology enables vehicles to follow the instructions sent from the controller precisely. Autonomous intersection management considers each vehicle as an agent and coordinates vehicle trajectories to resolve vehicle conflicts inside an intersection. This study proposes an autonomous intersection management algorithm called AIM-*ped* considering both vehicles and pedestrians which is able to produce the total optimal throughput when combined with max pressure control. This study analyzes the stability properties of the algorithm based on a simpler version of AIM-*ped*, which is a conflict region model of the autonomous intersection management. To implement the proposed algorithm in simulation, this study combines the max-pressure control with an existing trajectory optimization algorithm to calculate optimal vehicle trajectories. Simulations are conducted to test the effects of pedestrian demand on intersection efficiency. The simulation results show that delays of pedestrians and vehicles are negatively correlated and the proposed algorithm can adapt to the change in the pedestrian demand and activate vehicle movements with conflicting trajectories.

1. Introduction

Intersections are important components of urban traffic networks. They connect vehicle and pedestrian flows between network links and are also the main bottlenecks that contribute to most of the delays for vehicles and pedestrians. Consequently, intersection control plays an important role in improving traffic efficiency, enhancing the road safety level, and mitigating traffic congestion. Current intersection control is based on traffic signals. Vehicle movements in different directions are categorized into signal groups and vehicles whose trajectories are not conflicting or partially conflicting with other vehicles are allowed to move in the same time interval, which is called a phase. Fixed signal controls use different signal timing plans for multiple periods of a day based on historical traffic data. Typically, intersection control gives a phase that includes vehicle movements with larger volumes longer activation time. In contrast, adaptive signal controls use traffic data collected by detectors at the upstream or downstream of an intersection to flexibly adjust the duration of predefined phases aiming to reduce vehicle delay at the intersection. There are some widely used adaptive signal control system applied to a city-wide area, such as SCOOT (Bing and Carter, 1995), SCATS (Sims and Dobinson, 1980), RHODES (Mirchandani and Head, 2001), OPAC (Gartner, 1983), and etc.

* Corresponding author.
E-mail address: chen4416@umn.edu (R. Chen).

https://doi.org/10.1016/j.trc.2020.01.016
Received 9 July 2019; Received in revised form 13 December 2019; Accepted 21 January 2020

With the advances in vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) communications, it is easier to collect real-time traffic data for adjusting the signal or calculating the optimal control. There are studies that optimize phase-based intersection controls (Priemer and Friedrich, 2009; He et al., 2012; Goodall et al., 2013; Feng et al., 2015). The phase order or the phase time of the intersection signal is optimized with data collected by V2I and V2V devices and only vehicles that are not conflicting with each other are allowed to move in a specific phase. As autonomous vehicle technology develops, vehicles can be precisely controlled by computers. Some studies designed algorithms for vehicles to adjust their driving speeds or accelerations based on existing traffic signals so that vehicles can pass the intersection smoothly and avoid stopping for the red light (Kamalanathsharma and Rakha, 2013; Ma et al., 2017). Some other studies proposed signal-free intersection control algorithms to coordinate non-conflicting trajectories at the intersection. Once vehicle trajectories are determined, vehicles can follow these assigned trajectories and avoid collisions without the safety buffers of traffic signal phases. Autonomous intersection management (AIM), which was proposed by Dresner and Stone (2004), is an intersection control mechanism in which all vehicles that approach an intersection send their information to the controller at the intersection and follow its instructions.

Most AIM models do not consider pedestrian access for intersections. In the future, for a traffic network with autonomous vehicles, pedestrians will still require intersection access due to the costs of constructing separate right-of-way for pedestrians (e.g. tunnels or bridges). However, having pedestrians at an intersection controlled by AIM introduces a lot of unpredictable risks to the intersection. An AIM-controlled intersection calculates the trajectories of approaching vehicles based on their position and speed information and does not consider pedestrians so the calculated vehicle trajectories only service the condition when there is no pedestrian crossing the street. To minimize the vehicle delay at the intersection, AIM models often leave small gaps between vehicles. It is hard for pedestrians to find a safe gap for them to cross the street under the control of AIM as they typically cross with traffic signal phases. The detectors on autonomous vehicles enable vehicles to react to jaywalking pedestrians, but the resulting unplanned stop causes the temporary breakdown of the intersection traffic. Therefore, AIM needs significant modification to include pedestrian movements.

In traditional phase-based intersection control, crosswalks are activated with signal phases which significantly reduce the number of conflict points between the trajectories of pedestrians and vehicles. However, in AIM, phases that align with crosswalk activation do not exist. This study incorporates pedestrians into intersection control by adding crosswalk activation to a reservation-based algorithm. When one or multiple crosswalks are activated, vehicles whose trajectories do not intersect with the activated crosswalks are allowed to move.

Activating crosswalks in AIM reduces the throughput of vehicles at the intersection because it blocks vehicles with conflicting trajectories. However, using the conventional objective of AIM, which is to maximize the total throughput or reduce the total delay of vehicles, the efficiency of pedestrians will not be carefully considered. It is also hard to determine the weights assigned to vehicles and pedestrians respectively as there is a subtle trade-off between these two components. Therefore, we apply max-pressure control to the intersection control, which is able to maximize the network throughput of the total combined vehicle and pedestrian flow.

Max-pressure control was originally used as a scheduling policy in communication and power networks (Tassiulas, 1992). Max-pressure control defines weights and pressures of turning movements and uses a mathematical program to get the optimal control strategy which maximizes the total weight of the intersection. Most intersection control algorithms do not have network-level properties that max-pressure control algorithms have, including the stability of the total queue length. Besides, it is also a distributed algorithm with the controller at each intersection in a network calculating the control strategy by itself. The max-pressure control algorithm needs input data, normally the queue length, to calculate the weight of each turning movement, but pedestrians are not connected to the intersection controller. Therefore, a model is needed to estimate the queue length of pedestrians at the intersection.

The contributions of this study are: (1) proposing a max-pressure policy based on the conflict region model of AIM to address vehicles and the pedestrians at an intersection (2) designing a queue length estimation method for pedestrians. (3) proving the proposed algorithm can achieve optimal throughput of the network. (4) integrating the max-pressure control with an existing AIM algorithm. The new algorithm can calculate the optimal vehicle trajectory at the intersection under the max-pressure control. (5) testing the effects of the pedestrian demand on the efficiency of vehicles using the conflict-point model of AIM.

This paper is organized as follows: Section 2 summarizes the relevant studies about AIM and max-pressure control. Section 3 introduces the network model used to represent the flow propagation on the network. Section 3.3 proposes the modified max-pressure algorithm that integrates the pedestrian flow on the network. Section 3.4 formulates the stability region of the demand and proves the stability properties of the control algorithm. Section 4 formulates a mixed-integer program that integrates an existing trajectory planning algorithm with the max-pressure control. Section 5 presents the simulation experiment and simulation results.

2. Literature review

This section introduces the existing literature related to AIM and max-pressure control.

2.1. Autonomous intersection management

Autonomous intersection management (AIM) is a type of signal-free intersection control algorithm. Its application is based on V2I or V2V technologies to transmit messages between the intersection controller and the approaching vehicles. AIM algorithms coordinate vehicles with conflicting trajectories to avoid crashes. As AIM algorithms consider minimizing vehicle delays or vehicle gaps as their objectives and allow vehicles with conflicting trajectories to move in the same time interval, they are more efficient than traditional intersection controls, as highlighted by many studies in their simulations (Fajardo et al., 2011; Li et al., 2013; Kamal et al.,

2015; Wu et al., 2015; Levin and Boyles, 2016; Fayazi et al., 2017).

Dresner and Stone (2004) considered an intersection with autonomous vehicles as a multi-agent system and proposed a reservation-based approach to coordinate the reservation of tiles at the intersection for vehicles. Initial studies focused on conflict-free protocols for vehicle trajectory reservations (Dresner and Stone, 2006) with extensions to emergency vehicle and human vehicle access (Dresner and Stone, 2007a,b). AIM was also used to manage inter-connected intersections in a network (Hausknecht et al., 2011) or other types of intersections, such as roundabouts (Bento et al., 2012). The effects of AIM under the context of dynamic traffic assignment was also explored (Zhu and Ukkusuri, 2015).

AIM was formulated with multiple models, such as linear programming (Jin et al., 2012), mixed-integer linear programming (Zhu and Ukkusuri, 2015; Fayazi et al., 2017; Levin and Rey, 2017), mixed-integer nonlinear programming (Mirheli et al., 2019). There are also studies formulated this problem in model predictive control framework (Kamal et al., 2013, 2015) or as a dynamic optimization problem (De Campos et al., 2017; Wuthishuwong and Traechtler, 2013; Mirheli et al., 2018).

Different models of AIM provided different outputs. Some models calculated vehicle arrival times at conflict points or depart times and exit times at the intersection (Jin et al., 2012; Levin and Rey, 2017), while some models gave the number of vehicles allowed to move (Zhu and Ukkusuri, 2015). The objectives of these models included maximizing the total throughput (Fayazi et al., 2017; Levin and Rey, 2017), minimizing total travel times (Jin et al., 2012), minimizing fuel consumption (Zhang et al., 2016), and minimizing potential risk (Kamal et al., 2015).

AIM is originally designed for an intersection with all autonomous vehicles so that vehicles can follow the controller's instructions. There are studies that consider AIM with human vehicles that are not equipped or partially equipped with V2V and V2I communication facilities and do not have autonomous driving modules (Dresner and Stone, 2007b; Bento et al., 2013; Qian et al., 2014; Levin and Boyles, 2016). As human drivers are subject to high control uncertainty, these studies considered the vehicle dynamics of human vehicles and intersections used traffic lights to communicate with human vehicles. Due to the challenge of incorporating human vehicles in AIM with max-pressure control, this study focuses on the case where all vehicles are autonomous and can communicate with the intersection manager.

Existing studies of AIM optimized vehicle trajectories considering the traffic flow with all autonomous vehicles or a mixed-flow with both autonomous and legacy vehicles under various objectives, but none of them consider pedestrians in their model. This study is the first one that proposes an AIM algorithm with pedestrians. Furthermore, we go beyond adding a simple pedestrian phase to AIM by allowing simultaneous activation of crosswalks and non-conflicting vehicle movements. Adding the crosswalk activation may change the performance of the AIM, which is originally designed for improving vehicle efficiency. For example, the activation of crosswalks may prevent the controller from activating conflicting vehicles.

2.2. Max-pressure control

In existing studies about the max-pressure control, there are different ways of defining the weight and the pressure for a movement. The weight can be defined as the difference between queue lengths of an entry link and an exit link. The pressure of a phase is the sum of the product of movement weights and movement capacities (Wongpiromsarn et al., 2012). The weight can also be defined as the difference in its queue length and the average queue length of its downstream turning movements. The pressure of a control plan was the product of the weights and the saturation flow rates of activated turning movements (Varaiya, 2013). There are some variants of these two mainly used definitions. For example, Gregoire et al. (2014a) used aggregated link length to calculate the weight of a turning movement thus obviating the need for turning ratios. Normalized pressures (Gregoire et al., 2014b) and position-based weights (Li and Jabari, 2019) were also used to calculate the optimal control.

In most studies, a linear program is used to determine optimal control after the calculation of pressures. The optimal control is created by using the phase with the largest total pressure and only turning movements included in this phase are allowed to move (Wongpiromsarn et al., 2012; Varaiya, 2013; Gregoire et al., 2014a,b; Xiao et al., 2014; Lioris et al., 2016). To make sure all interested phases were activated, Pumir et al. (2015) added a constraint to their linear program to set the minimum activation duration. Another way to achieve this is by using fixed cycle time and assigned each of the predefined phases activation time proportionally based on their weights (Le et al., 2015). This method can prevent the condition in which a turning movement with a small queue length is forced to wait for a long time. In the calculation of the optimal control, Hsieh et al. (2017) incorporated the switch-over delay in a max-pressure model, which caused the capacity loss when the traffic light switched between red and green. To improve the practicality of the max-pressure control for mixed flow of autonomous vehicles and legacy vehicles, Rey and Levin (2019) designed an intersection controller that could address both autonomous vehicles and legacy vehicles at an intersection with dedicated lanes for autonomous vehicles. In Rey and Levin (2019)'s study, the autonomous and legacy vehicles were separated in time by using two exclusive phases so these two types of vehicle were not allowed to move at the time. In this study, the activation of crosswalks used by pedestrians and the activation of vehicle movements can be simultaneous.

Max-pressure control can achieve network-level queue length stability when the demand is in the stability region. Some studies proved the stability of the control algorithm analytically under the assumption of infinite queue capacity (Varaiya, 2013; Le et al., 2015; Hsieh et al., 2017) or finite queue capacity (Xiao et al., 2014). Some studies showed the queue length became stable using simulation (Lioris et al., 2016). Existing studies proved the stability of queue length considering only vehicle queues, but this study proves the network-level queue length stability considering both pedestrian and vehicle queues.

Max-pressure controls were proved to have higher efficiency than traditional signal control in many studies. Some studies showed that the max-pressure control had better performance than the adaptive traffic signal control (Wongpiromsarn et al., 2012; Sun and Yin, 2018), the fixed signal timing (Hsieh et al., 2017; Li and Jabari, 2019), the greedy policy by Lämmer and Helbing (2010) and the

proportional policy by Smith (1980) (Le et al., 2015). Some studies showed cycle-based max-pressure policies outperformed the traditional max-pressure policy (Le et al., 2015; Sun and Yin, 2018).

Both AIM algorithms and max-pressure control show better performances than the traditional signal control when solely used in intersection control. When these two algorithms are integrated in this study, the AIM algorithm is able to prevent collision between vehicles and coordinates the arrival times of vehicles while max-pressure control is able to achieve the optimal throughput for the combined vehicle-pedestrian flow.

3. Network model

Consider a traffic network consisting of a road network for vehicles $\mathcal{G}^{\text{veh}}(\mathcal{N}^{\text{veh}}, \mathcal{L}^{\text{veh}})$ and a sidewalk network for pedestrians $\mathcal{G}^{\text{ped}}(\mathcal{N}^{\text{ped}}, \mathcal{L}^{\text{ped}})$. These networks interact at intersections where crosswalks and vehicles can conflict. For both networks, $\mathcal{N}^{\text{veh}} \cup \mathcal{N}^{\text{ped}}$ denotes the node set and $\mathcal{L}^{\text{veh}} \cup \mathcal{L}^{\text{ped}}$ denotes the link set. The link set can be classified into three subsets $\mathcal{L}_{\text{entry}}$, $\mathcal{L}_{\text{int}}$, and $\mathcal{L}_{\text{exit}}$ representing the entry, interval, and exiting links respectively. The entry link set includes links that bring vehicles or pedestrians into the road network or the sidewalk network. The exiting link set includes links that take vehicles or pedestrians out of the road network or the sidewalk network. In this study, it is assumed that every link has a free flow travel time of 1 time step, which means the vehicle link and the pedestrian link may have different lengths if the vehicle and the pedestrians have different travel speeds. A long link can be divided into several shorter links whose travel time is the free flow travel time. Let Γ_i^- and Γ_i^+ be the sets of incoming and outgoing links of link i or sidewalk i respectively. If link i is connected with crosswalk j, it can be represented as $i \in \Gamma_j^-$ or $i \in \Gamma_j^+$.

In the vehicle network $\mathcal{G}^{\text{veh}}(\mathcal{N}^{\text{veh}}, \mathcal{L}^{\text{veh}})$, a turning movement can be denoted by a pair of links (i, j), which represents a turning movement leaving link i and entering link j. Let $\mathcal{M}$ be the set of all turning movements. The capacity of link i is denoted by Q_i. The capacity of turning movement (i, j) is calculated by $Q_{ij} = \min\{Q_i, Q_j\}$. We assume an intersection is divided into several conflict regions where trajectories of turning movements intersect with each other. Let C be the set of all conflict regions at an intersection. C_{ij} is the set of conflict regions on the trajectory of turning movement (i, j). Let δ_{ij}^c denote the relation between turning movement (i, j) and conflict region c. If turning movement (i, j) intersects with conflict region c, $\delta_{ij}^c = 1$. Otherwise, $\delta_{ij}^c = 0$.

The pedestrian network $\mathcal{G}^{\text{ped}}(\mathcal{N}^{\text{ped}}, \mathcal{L}^{\text{ped}})$ is consisted of the set of pedestrian nodes $\mathcal{N}^{\text{ped}}$ and the set of pedestrian links $\mathcal{L}^{\text{ped}}$. The pedestrian link set includes the sidewalk and the crosswalk. In Fig. 1, the pedestrian links in green are sidewalks, such as links a, b, c, d, e, f, g, and h. If two sidewalks are directly connected, pedestrians do not need to wait when they walk from one to another. For example, the sidewalk b is connected with sidewalk c and pedestrians at sidewalk b can walk to sidewalk c without stopping. The

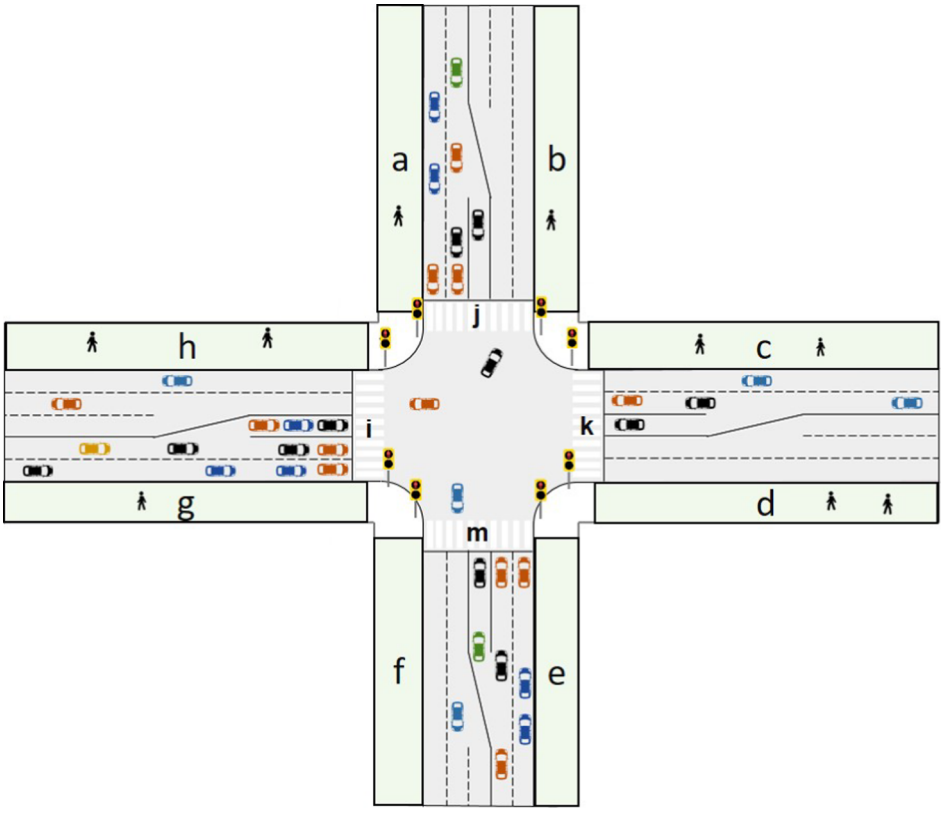


Fig. 1. Pedestrian network.

pedestrian links that intersect with vehicle links are crosswalks, such as links i, j, k, and m. Pedestrians need to wait for the activation of a crosswalk to cross the street. For example, pedestrians at sidewalk b should wait at crosswalk k if they wants to go to sidewalk e until the pedestrian signal turns green. Let $\mathcal{W}$ denote the set of all crosswalks. A pair of adjacent pedestrian links can define a pedestrian movement. A pedestrian movement whose direction is toward a crosswalk is restricted by the actuation of the crosswalk. For example, pedestrian movement (b, k) is restricted by the pedestrian signal on crosswalk k but (b, c) is not.

Considering the interaction between crosswalks and vehicles, δ_{ij}^{m} is used to indicate if the trajectory of turning movement (i, j) intersects with crosswalk m. If they have an intercept, $\delta_{ij}^{m} = 1$, otherwise, $\delta_{ij}^{m} = 0$.

3.1. Queue evolution

To calculate the length of vehicle queue and pedestrian queue at every time step, the point queue model based on the store-and-forward model in Varaiya (2013)'s paper study is used, as shown in Eqs. (1) and (2). z indicates the mode type is either vehicles or pedestrians. If the link is an internal or exit link, then the queue evolution follows Eq. (1). In Eqs. (1), $x_{ij}(t)$ is the queue length of vehicle movement (i, j) or pedestrian movement (i, j) at time t, $y_{ij}(t)$ is the number of vehicles or pedestrians that exit the queue at time t, and the last term is the total amount of flows that join the queue (i, j) from upstream links or upstream sidewalks. $p_{ij}(t)$ is a random variable which represents the proportion of vehicles or pedestrians on link i going to link j at time step t.

$$x_{ij}^{z}\left(t+1\right) = x_{ij}^{z}(t) - y_{ij}^{z}(t) + \sum_{h \in \Gamma_i^-} y_{hi}^{z}(t) p_{ij}(t) \qquad \forall\, i \in \mathcal{L}_{\text{int}}^{z} \cup \mathcal{L}_{\text{exit}}^{z},\, h \in \Gamma_i^-,\, j \in \Gamma_i^+,\, z \in \left\{\text{veh, ped}\right\} \tag{1}$$

If the link is an entry link, then the queue evolution follows Eq. (2). The last term in Eq. (1) is replaced by $d_i(t)$, which is the input demand at entry link i.

$$x_{ij}^{z}\left(t+1\right) = x_{ij}^{z}(t) - y_{ij}^{z}(t) + d_i^{z}(t) p_{ij}(t) \qquad \forall\, i \in \mathcal{L}_{\text{entry}}^{z},\, j \in \Gamma_i^+,\, z \in \left\{\text{veh, ped}\right\} \tag{2}$$

The vehicle flow $y_{ij}(t)$ is calculated at every time step based on the control algorithm. For vehicle turning movement (i, j), the flow is the minimum between the maximum number of vehicles allowed to move $Q_{ij} S_{ij}^{\text{veh}}(t)$ and the current queue length $x_{ij}^{\text{veh}}(t)$, where $S_{ij}^{\text{veh}}(t)$ is the proportion of time that turning movement (i, j) are allowed to move and Q_{ij}^{veh} is the capacity.

$$y_{ij}^{\text{veh}}(t) = \min\{Q_{ij}(t) S_{ij}^{\text{veh}}(t),\, x_{ij}(t)\} \qquad \forall\, i, j \in \mathcal{L}^{\text{veh}} \tag{3}$$

For pedestrian movement (i, j), if both link i and link j are sidewalks, S_{ij}^{ped} is always 1. If link j is a crosswalk, $S_{ij}^{\text{ped}}(t)$ depends on the intersection control. All pedestrian movements heading for the same crosswalk have the same pedestrian signal control, which is $S_{ij}^{\text{ped}}(t) = S_{hj}^{\text{ped}}(t) = S_{j}^{\text{ped}}(t)$, $\forall\, j \in \mathcal{W}$, $\forall\, h, i \in \Gamma_j^-$, $h \neq i$.

$$y_{ij}^{\text{ped}}(t) = \min\{Q_{ij}(t) S_{ij}^{\text{ped}}(t),\, x_{ij}(t)\} \qquad \forall\, i, j \in \mathcal{L}^{\text{ped}} \tag{4}$$

3.2. Estimating the pedestrian queue length

Unlike automated vehicles, which are assumed to communicate wirelessly with the intersection, pedestrians may only be able to indicate their presence through the crosswalk button. Consequently it is difficult to count the number of waiting pedestrians, which requires an estimation of the waiting queue. The estimated pedestrian queue length is used to calculate the weight for pedestrian turning movements in the max-pressure control and activate crosswalks.

The activation of crosswalk m is represented by $S_m(t)$. When the pedestrian signal is activated at time t, $S_m^{\text{ped}}(t) = 1$. Otherwise, $S_m^{\text{ped}}(t) = 0$. The activation of the pedestrian signal is related to the pedestrian queue, and we want the pedestrian queue length to be bounded. To estimate the pedestrian queue, the pedestrian waiting time should be recorded.

Let $\hat{\tau}_m(t)$ be the waiting time of pedestrian at crosswalk m since the last actuation of the pedestrian signal. $\hat{\tau}_m(t+1)$ can be updated with Eq. (5).

$$\hat{\tau}_m\left(t+1\right) = \begin{cases} \hat{\tau}_m(t) + 1, & \text{if } S_m^{\text{ped}}(t) = 0 \wedge \sum_{l \in \Gamma_m^-} x_{lm}(t) > 0 \\ 0, & \text{if } S_m^{\text{ped}}(t) = 1 \vee \sum_{l \in \Gamma_m^-} x_{lm}(t) = 0 \end{cases} \tag{5}$$

If the pedestrian signal is not activated at the last time step, and there are waiting pedestrians, then the waiting time will increase by one. If the pedestrian signal is activated at the previous time step or if there is no pedestrian going to cross the road, the waiting time is set to be 0. Based on this model, the waiting time of a pedestrian queue is determined by the pedestrian with the longest waiting time.

After the estimation of the pedestrian waiting time, the estimated pedestrian queue $\hat{x}_{ij}^{\text{ped}}$ can be calculated using Eq. (6).

$$\hat{x}_{ij}^{\text{ped}}\left(t+1\right) = \hat{x}_{ij}^{\text{ped}}(t) - \hat{y}_{ij}^{\text{out}}(t) + \hat{y}_{ij}^{\text{in}}(t) \tag{6}$$

Because of the difficulty to directly measure the queue length at crosswalk j, the estimated value instead of the actual value of the queue length is used. In Eq. (6), the estimated pedestrian queue $\hat{x}_{ij}^{\text{ped}}(t+1)$ is the estimated pedestrian queue $\hat{x}_{ij}^{\text{ped}}(t)$ added to the estimated pedestrian entering flow $\hat{y}_{ij}^{\text{in}}(t)$ and minus the estimated pedestrian exiting flow $\hat{y}_{ij}^{\text{out}}(t)$. The estimated entering flow is $\hat{y}_{ij}^{\text{in}}(t) = \hat{\tau}_j(t)\bar{u}_{ij}$. $\hat{\tau}_j$ is the waiting time since the last activation of the pedestrian signal at crosswalk j and $\bar{u}_{ij}$ is the mean arrival rate of pedestrians that are from pedestrian link i to crosswalk j and is assumed to be exogenous. The estimated exiting flow is $\hat{y}_{ij}^{\text{out}}(t) = \min\left(\hat{x}_{ij}^{\text{ped}}(t), Q_{ij}S_j^{\text{ped}}(t)\right)$, which is the minimum value between the estimated pedestrian queue length $\hat{x}_{ij}^{\text{ped}}(t)$ and the product of the capacity and the crosswalk control $S_j^{\text{ped}}(t)$. If the crosswalk is not activated, the estimated exiting flow is zero. If the crosswalk is activated and the estimated pedestrian queue length at time t does not exceed the capacity, all potential pedestrians can cross the street at the current time step. Otherwise, the number of supposed pedestrians that can pass the street is restricted by the capacity. We assume that the expected value of the difference between the estimated queue length and the actual queue length is bounded by $\omega \geqslant 0$, i.e. $\mathbb{E}\left[|x_{ij}^{\text{ped}}(t) - \hat{x}_{ij}^{\text{ped}}(t)|\right] \leqslant \omega$. In the perspective of queuing theory, the average queue length should be the product of the arrival rate and the waiting time. As the estimated queue length is calculated using the measured waiting time and the average arrival rate, the expectation of the estimated queue length and the actual queue length should be equal in the long term.

3.3. Max-pressure control policy

As max-pressure control has the property of maximizing the throughput at the network-level (Varaiya, 2013), this study uses a max-pressure algorithm to calculate how many vehicles at the intersection should be served at every time step and the actuation of the crosswalk. The weight of each vehicle turning movement (i, j) is defined by Eq. (7).

$$w_{ij}^{z}(t) = x_{ij}^{z}(t) - \sum_{k\in\Gamma_j^+} x_{jk}^{z}(t)p_{jk}(t) \qquad z \in \left\{\text{veh, ped}\right\} \tag{7}$$

The weight of movement (i, j) is the queue length of this movement minus the average queue length of movements on downstream links. For example, in Fig. 2, the downstream movements of movement (i, j) are movements (j, k_1), (j, k_2), and (j, k_2). The queue length of a vehicle turning movement is measured regularly. The calculation of the weights for pedestrian queues uses the estimated queue length in Eq. (6). The difference between the weights calculated by the actual and the estimated pedestrian queue length is also assumed to be bounded, which is $|w_{ij}^{\text{ped}}(t) - \hat{w}_{ij}^{\text{ped}}(t)| \leqslant \beta$.

After calculating the weight for each movement, a mixed-integer linear program is used to calculate the intersection control strategy, as shown in Eq. (8). In this program, the intersection is divided into several conflict regions and each of them has its capacity. Fig. 3 shows an intersection with four conflict regions (A, B, C, and D). The trajectory of a vehicle will pass one or several conflict regions. We use C_{ij} to denote the set of conflict regions passed by vehicle movement (i, j). For example, the northbound left-turn movement passes through conflict regions A, C, and D. The capacity of conflict region c is Q_c and is determined by the capacities of turning movements that pass through conflict region, which is $Q_c = \max_{\{(i,j)|c\in C_{ij}\}}\{Q_{ij}\}$. The total amount of vehicles passing through a conflict region per time step is restricted by the capacity of the conflict region.

$$\max \sum_{(i,j)\in\mathcal{M}} w_{ij}^{\text{veh}}(t)y_{ij}(t) + \sum_{n\in\mathcal{W},m\in\Gamma_n^-} w_{mn}^{\text{ped}}(t)S_{mn}^{\text{ped}}(t)Q_{mn}^{\text{ped}}(t) \tag{8a}$$

$$\text{s. t. } y_{ij}(t) \leqslant Q_{ij}(t)(1 - S_{mn}^{\text{ped}}(t)\delta_{ij}^{n}) \quad \forall\,(i,j)\in\mathcal{M}, \forall\, n\in\mathcal{W}, \forall\, m\in\Gamma_n^- \tag{8b}$$

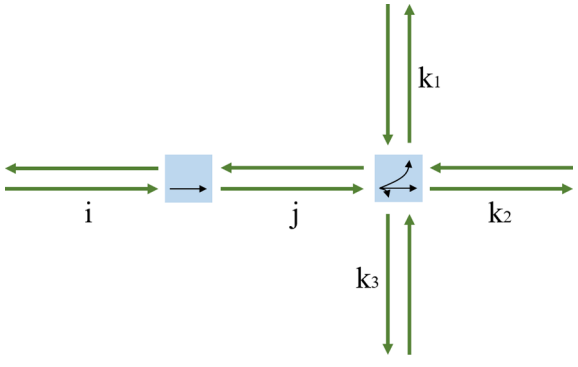


Fig. 2. Downstream movements.

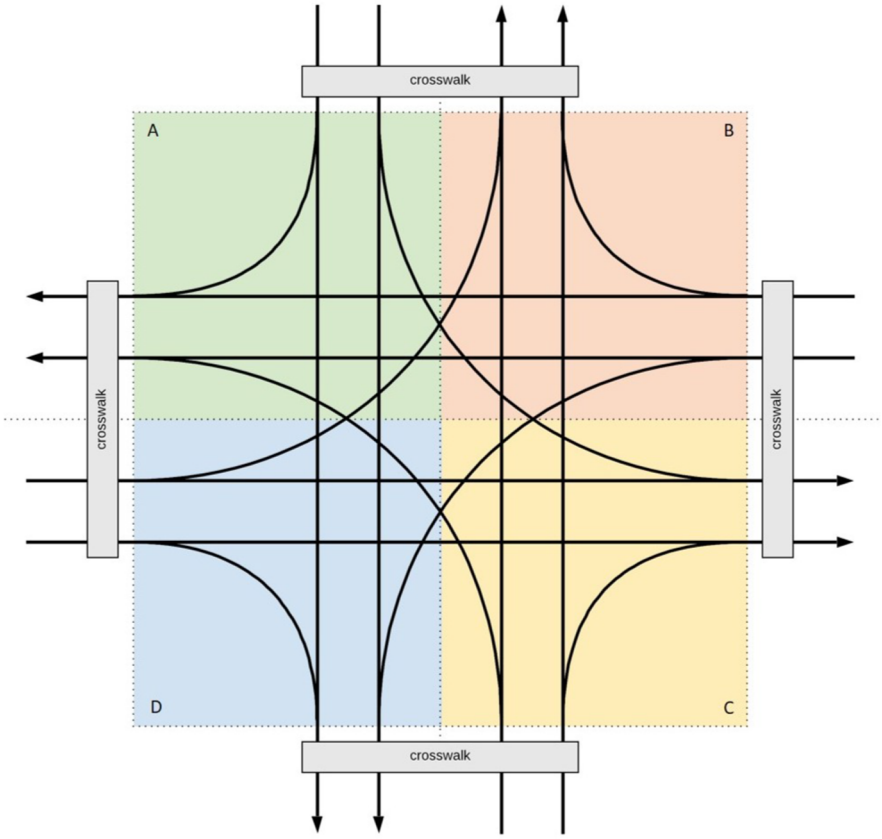


Fig. 3. Conflict region model of AIM.

$$\sum_{(i,j)\in\mathcal{M}} y_{ij}(t)\delta_{ij}^{c} \leqslant Q_c \quad \forall\, c \in C \tag{8c}$$

$$y_{ij}(t) \leqslant x_{ij}(t) \quad \forall\, (i,j) \in \mathcal{M} \tag{8d}$$

$$S_{mn}^{\text{ped}}(t) \in \{0,1\} \quad \forall\, n \in \mathcal{W}, \forall\, m \in \Gamma_n^- \tag{8e}$$

$$S_{mn}^{\text{ped}}(t) = S_{ln}^{\text{ped}}(t), \quad \forall\, m, l \in \Gamma_n^-,\ m \neq l \tag{8f}$$

$$y_{ij}(t) \geqslant 0 \quad \forall\, (i,j) \in \mathcal{M} \tag{8g}$$

The max-pressure control aims to optimize the total pressure. y_{ij} represents the number of vehicles in turning movement (i, j) that is allowed to move, which should be constrained by the capacity at the conflict region and the waiting vehicles at the entry approach. S_{mn}^{ped} controls the activation of pedestrian movement (m, n). The value for S_{ij} in Eq. (3) can be calculate using $S_{ij} = y_{ij}/Q_{ij}$. Let S_r^* denote the max-pressure control at intersection r in the network, which is $S_r^* = \arg\max_{y_{ij}, S_{mn}^{\text{ped}}} \left\{ \sum_{(i,j)\in\mathcal{M}} w_{ij}^{\text{veh}} y_{ij} + \sum_{(m,n)\in\mathcal{W}} w_{mn}^{\text{ped}} S_{mn}^{\text{ped}} Q_{mn}^{\text{ped}} \right\}$.

3.4. Stability region

Vehicles and pedestrians enter the network through all entry links $i \in \mathcal{L}_{\text{entry}}^{\text{veh}}$ and $\mathcal{L}_{\text{entry}}^{\text{ped}}$. Let demand vector $\bar{\mathbf{d}}$ represent the average demand of the traffic entering the network through each of the entry links or entry sidewalks. The dimension of $\bar{\mathbf{d}}$ is equal to the number of elements in the set $\mathcal{L}_{\text{entry}}$. The ith element of $\bar{\mathbf{d}}$ corresponds to the flow to link $i \in \mathcal{L}_{\text{entry}}^{\text{veh}}$ or sidewalk $i \in \mathcal{L}_{\text{entry}}^{\text{ped}}$. Let f_i denote the flow on link i or sidewalk i. $\bar{p}_{ij}$ is the average turning proportion from link i to link j or from sidewalk i to sidewalk j. Link flows and flows of vehicle movements (or pedestrians) follow rules in Eq. (9). With these constraints, a demand vector $\bar{\mathbf{d}}$ and average turning proportion vector $\bar{\mathbf{p}}$ can uniquely determine a flow vector $\mathbf{f}$ on the network if all turning proportions are fixed based on Varaiya (2013)'s assumption. When affected by the intersection control, the value of pedestrian arrival rate $\bar{u}_{ij}$ is smaller or equal to the flow rate $f_i \bar{p}_{ij}$ of the movement, which is calculated based on the input demand and the average proportions of turning movements.

$$f_i = \bar{d}_i \ \forall\, i \in \mathcal{L}_{\text{entry}}^{z},\ z \in \{\text{veh, ped}\} \tag{9a}$$

$$f_j = \sum_i f_i \bar{p}_{ij} \ \forall j \in \mathcal{L}_{\text{int}}^z \cup \mathcal{L}_{\text{exit}}^z, z \in \left\{\text{veh, ped}\right\} \tag{9b}$$

The activation of a movement (i, j) at time step t is represented by $S_{ij}(t)$. For vehicles, $S_{ij}(t)$ is a fraction between 0 and 1, which represents the percentage of time used for activating turning movement (i, j). For example, if $S_{ij}(t) = 0.5$ and the time step is 30 s, then turning movement (i, j) is allowed to move for 15 s in this time step. For pedestrians, it is a binary variable whose value is either 0 or 1 because it is assumed that the pedestrian can use the entire time interval if the pedestrian signal is activated. The turning flow or pedestrian flow $y_{ij}(t)$ is the minimum between the product of the capacity and the movement activation $Q_{ij}S_{ij}(t)$ and the current queue length $x_{ij}(t)$. Eq. (10) shows the constraint applying to the activation of movements at an intersection. The sum of S_{ij} should be less than or equal to 1, because the sum of percentages of time occupied by any turning movement should be less than or equal to 100%.

$$\sum_{(i,j)\in\mathcal{M}} S_{ij}(t)\delta_{ij}^c \leqslant 1, \ \forall \ c \in C \tag{10}$$

Let $S_r(t)$ be an intersection control matrix for intersection r that includes values of $S_{ij}^{\text{veh}}(t)$ for all vehicle turning movements and values of $S_n^{\text{ped}}(t)$ for all crosswalks at time step t. Using $S_r(t)$ at all time steps, we can define an intersection control sequence $S_r = \{S_r(t), t \in \mathbb{N}\}$ as a sequence of feasible controls for intersection r. We also have $\mathbf{S}(t) = \{S_r(t), r \in \mathcal{N}^{\text{veh}}\}$, which is a vector of $S_r(t)$ over all intersections. For any given intersection control sequence, the long-term average time used for serving turning movement (i, j) and pedestrian movement (i, j) can be calculated by Eq. (11).

$$\bar{S}_{ij} = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} S_{ij}(t) \tag{11}$$

An intersection control sequence $\mathbf{S}$ can accommodate demand $\mathbf{d}$ if the average serving time of this control sequence multiplied by the capacity is larger than or equal to the corresponding average flow for the turning movement or the pedestrian flow, as shown in (12).

$$f_i \bar{p}_{ij} \leqslant \bar{S}_{ij} Q_{ij} \tag{12}$$

Let $\mathcal{S}$ be the set for all intersection control matrices. We can define a convex hull $co(\mathcal{S})$ including all possible intersection control matrices, as shown in (13).

$$co(\mathcal{S}) = \left\{ \sum_{S\in\mathcal{S}} S\lambda_S \middle| \lambda_S \geqslant 0, \sum \lambda_S = 1 \right\} \tag{13}$$

At each time interval, an intersection control matrix is picked from $\mathcal{S}$ and an intersection control sequence $\mathbf{S}$ includes intersection control matrices for all time intervals. Then we can use a vector $\bar{\mathbf{S}}$ to include the average activation times for all turning movements or the pedestrian queues and we have $\bar{\mathbf{S}} \in co(\mathcal{S})$.

A demand vector $\mathbf{d}$ is feasible if there is an intersection control sequence $\mathbf{S}$ that can make the corresponding average flow for the turning movement or the pedestrian flow be always smaller than the average serving time multiplied by the capacity for every link (Varaiya, 2013). Let $\mathcal{D}$ denote the set of all feasible demand vectors.

Let $\mathcal{D}^o$ denote the interior of $\mathcal{D}$. When there is a demand vector $d \in \mathcal{D}^o$, the flow pattern $\mathbf{f}$ corresponding to it satisfies $f_i \bar{p}_{ij} < \bar{S}_{ij} Q_{ij}$, which is also equivalent to $f_i \bar{p}_{ij} + \epsilon = \bar{S}_{ij} Q_{ij}$ for some $\epsilon > 0$.

3.5. Stability of the control algorithm

Queuing stability is defined in Eq. (14). We prove the stability of the overall system when max-pressure control is used.

$$\text{limsup}_T \frac{1}{T} \sum_{t=0}^{T} \mathbb{E}\left[\sum_{(i,j)\in\mathcal{L}^{\text{veh}}\cup\mathcal{L}^{\text{ped}}} x_{ij}(t) \right] < \infty, \ \forall \ T \tag{14}$$

Lemma 3.1. *If* $\mathbb{E}[|x_{mn}^{\text{ped}}(t) - \hat{x}_{mn}^{\text{ped}}|] \leqslant \omega$, *then expectation of the the difference between the optimal solutions of two programs with objective functions* $\hat{O}$ *and* O *for an intersection is also bounded, where* $\hat{O} = \max \sum_{ij} w_{ij}^{\text{veh}} y_{ij} + \sum_{mn} \left(\hat{x}_{mn}^{\text{ped}} - \sum_o \hat{x}_{no}^{\text{ped}} p_{no} \right) y_{mn}$ *and* $O = \max \sum_{ij} w_{ij}^{\text{veh}} y_{ij} + \sum_{mn} \left(x_{mn}^{\text{ped}} - \sum_o x_{no}^{\text{ped}} p_{no} \right) y_{mn}$.

Proof. The general form of the mixed integer program in Eq. (8) can be expressed as:

$$\begin{aligned}
&\max h^{\mathrm{T}}\mathbf{y} + c^{\mathrm{T}}\mathbf{S}^{\mathrm{ped}} \\
&\text{s. t.} A\mathbf{y} + G\mathbf{S}^{\mathrm{ped}} \leqslant b \\
&\quad \mathbf{y} \geqslant 0 \\
&\quad \mathbf{S}^{\mathrm{ped}} \geqslant 0 \\
&\quad \mathbf{y} \in \mathbb{R}^{|\mathcal{M}|} \\
&\quad \mathbf{S}^{\mathrm{ped}} \in \{0, 1\}^{|\mathcal{W}|}
\end{aligned}$$

In the general form, $\mathbf{y}$ is a vector including decision variables y_{ij} for all vehicle turning movement. $\mathbf{S}^{\mathrm{ped}}$ is a vector including decision variables S_{mn}^{ped} for all crosswalks. Changing the weights of the pedestrian queue is actually changing the cost c in the objective function. The vector c is replaced by $c + \Delta c$ with $\Delta c_{ij} \leqslant \beta Q_{ij}^{\mathrm{ped}}$. If adding Δc does not change the optimal solution, then $O = \widehat{O}$. Even if adding Δc does change the optimal solution, $|O - \widehat{O}|$ is also bounded because all constraints build a feasible set which can be enclosed by a polyhedron. As the size of the polyhedron is limited, $|O - \widehat{O}|$ can always be bounded by a value, denoted by ξ.

Proposition 1. *If the demand vector* $\mathbf{d} \in \mathcal{D}^o$*, this max-pressure control is stabilizing.*

If the max-pressure control is stabilizing, the queue length for each turning movement and at each crosswalk will remain bounded in expectation. Equivalently, the control is throughput optimal.

Proof. The proof is in Appendix A. □

4. Intersection control: AIM-*ped*

In Section 3.3, the max-pressure algorithm is proposed based on the conflict region model of AIM (Levin and Boyles, 2015) and the proposed stability properties are also based on this model. When applied to microscopic simulation, the conflict region model of AIM has some limitations. This model only considers the capacity constraint at each conflict region but does not consider the order of arriving vehicles at the intersection. Therefore, the vehicle behavior in this model may violate the first-in-first-out assumption in some conflict regions. For example, if there are three through vehicles from northbound, westbound, and southbound approaches respectively. Assume the intersection has four conflict regions, so each pair of these three vehicles share a conflict region at the intersection. Using this model, it is hard to incorporate time-dependent conflict avoidance within each time step. To add pedestrians to the microscopic simulation with AIM, we create a new algorithm called AIM-*ped* by combining the max-pressure control algorithm proposed in Section 3.3 with the trajectory optimization of Levin and Rey (2017).

In the model of Levin and Rey (2017), two intersecting vehicle trajectories form a conflict point. Fig. 4 shows the conflict points on the trajectories of northbound left-turn, through, and right-turn movements at a standard intersection with four two-lane approaches. The original objective function used in their study is to minimize the exit time of the last vehicle at the intersection. The constraints are used to ensure vehicle trajectories are collision-free. In AIM-*ped*, we control vehicles individually rather than controlling the flow of turning movements and constraints relating vehicles to crosswalks are added to the existing constraints. The objective function is modified to maximize the total pressure:

$$\max \quad \sum_{v \in \mathcal{V}} w_v^{\mathrm{veh}} z_v^{\mathrm{veh}} + \sum_{n \in \mathcal{W}} w_n^{\mathrm{ped}} Q_n^{\mathrm{ped}} z_n^{\mathrm{ped}} \tag{16}$$

In Eq. (16), the decision variables are z_v^{veh} and z_n^{ped}, which control the movement of vehicle v, and the actuation of crosswalk n. $\mathcal{V}$ and $\mathcal{W}$ represent the sets of vehicles and crosswalks respectively.

We first show the constraints borrowed from the study of Levin and Rey (2017). Constraint (17) requires the calculated departure time of vehicle v to be larger than its earliest possible arrival time e_v at the intersection. γ_v^- is the first conflict point in the path of vehicle v, then $t_v(\gamma_v^-)$ represents the moment when vehicle v enters the intersection.

$$t_v(\gamma_v^-) \geqslant e_v \quad \forall\, v \in \mathcal{V} \tag{17}$$

Constraint (18) guarantees that if two vehicles v and v' share the same entry lane ($\gamma_v^- = \gamma_{v'}^-$), then the vehicle that reaches the intersection earlier should also enter the intersection earlier. e_v is the earliest time of vehicle v to reach the intersection and $t_v(\gamma_v^-)$ is the time when vehicle v enters the intersection. $\tau_v(\gamma_v^-)$ is the time that vehicle v occupies the entry point of the intersection on its path.

$$t_v(\gamma_v^-) + \tau_v(\gamma_v^-) \leqslant t_{v'}(\gamma_{v'}^-) \quad \forall\, v, v' \in \mathcal{V}\colon \gamma_v^- = \gamma_{v'}^-,\ e_v < e_{v'} \tag{18}$$

Constraint (19) calculates the time that vehicle v spends at conflict point c, denoted by $\tau_v(c)$. $L_v(c)$ is the distance that vehicle v travels around c and $d_v(\gamma_v^-, \gamma_v^+)$ is the travel distance of vehicle v along its path. w is the backward shock wave speed of the related fundamental diagram of this road, and ρ_v is the set of conflict points on the trajectory of vehicle v. The derivation of this constrained is included in the study of Levin and Rey (2017).

$$\tau_v(c) = \frac{L_v(c)}{w} + \frac{L_v(c)(t_v(\gamma_v^+) - t_v(\gamma_v^-))}{d_v(\gamma_v^-, \gamma_v^+)} \quad \forall\, v \in \mathcal{V}, \forall\, c \in \rho_v \tag{19}$$

Constraints (20) and (21) both control the vehicle travel times. Constraint (20) sets the upper bound and lower bound of vehicle travel time through the intersection. $t_v(\gamma_v^+) - t_v(\gamma_v^-)$ is the time that vehicle v uses to pass through the intersection. $\overline{U}_v$ and $\underline{U}_v$ are the

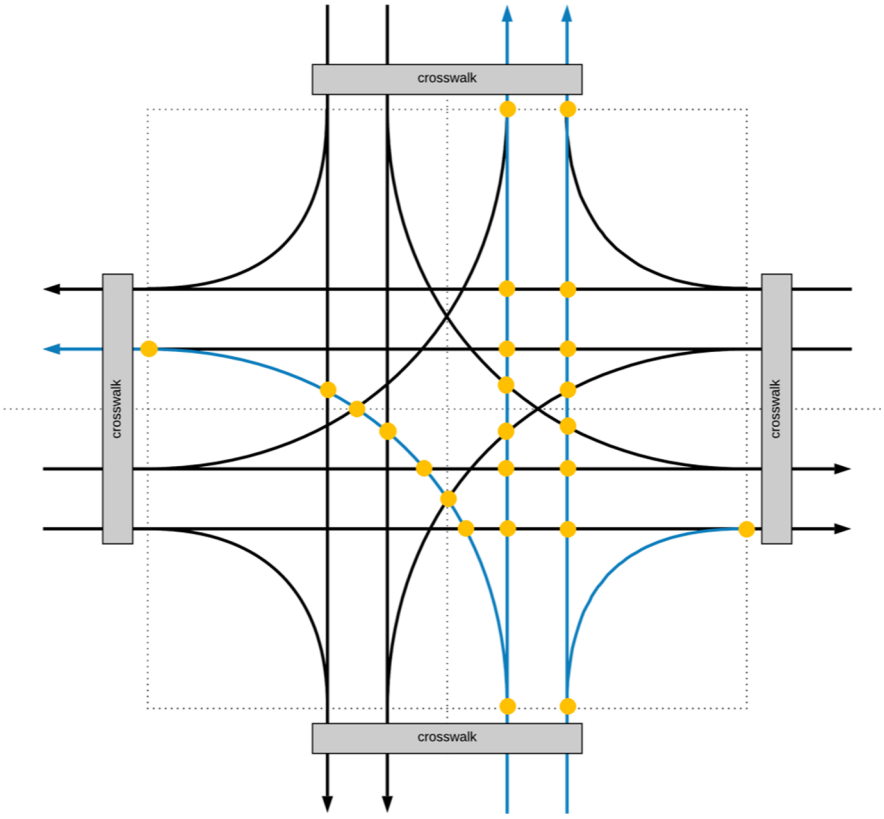


Fig. 4. Conflict point model of AIM.

maximum and the minimum speeds respectively. Constraint (21) calculates the travel time through each conflict point. $t_v(c)$ is the time that vehicle v arrives the conflict point c. $d_v(\gamma_v^-, c)$ is the travel distance from the entry point γ_v^- to conflict point c. This constraint requires that the average travel speed between the entry point and any conflict point on the path should equal the average travel speed between the entry point and the exit point, which pushes the vehicle to keep a constant speed.

$$\frac{d_v(\gamma_v^-, \gamma_v^+)}{\overline{U}_v} \leqslant t_v(\gamma_v^+) - t_v(\gamma_v^-) \leqslant \frac{d_v(\gamma_v^-, \gamma_v^+)}{\underline{U}_v} \quad \forall\, v \in \mathcal{V} \tag{20}$$

$$\frac{t_v(c) - t_v(\gamma_v^-)}{d_v(\gamma_v^-, c)} = \frac{t_v(\gamma_v^+) - t_v(\gamma_v^-)}{d_v(\gamma_v^-, \gamma_v^+)} \quad \forall\, v \in \mathcal{V}, \forall\, c \in \rho_v \tag{21}$$

In constraint (22), if vehicles v and v' use the same entry lane ($\gamma_v^- = \gamma_{v'}^-$) and vehicle v arrives earlier, then vehicle v' can only enter each shared conflict point c after vehicle v exits. In constraint (23), for two vehicles with conflicting trajectories, variables $\delta_{vv'}$ and $\delta_{v'v}$ are used to represent the order of vehicles entering conflict point c. If vehicle v enters first, $\delta_{vv'}(c) = 1$ and $\delta_{v'v}(c) = 0$. Otherwise, $\delta_{vv'}(c) = 0$ and $\delta_{v'v}(c) = 1$. Constraint (24) sets the range of arrival times of two vehicles at each conflict point with conflicting trajectories. $M_{vv'}$ is a large number. One vehicle can only enter the conflict point c when the other vehicle has passed and variable $\delta_{vv'}$ sets the passing order. If $\delta_{vv'}(c) = 1$, then vehicle v enters the conflict c earlier than vehicle v', then the right hand side equals to 0, and the constraint becomes $t_v(c) + \tau_v(c) \leqslant t_{v'}(c)$. If $\delta_{vv'}(c) = 0$, the constraint becomes $t_v(c) + \tau_v(c) \leqslant M_{vv'}$, which means there is no restriction on the arrival times of two vehicle at the conflict point.

$$t_v(c) + \tau_v(c) \leqslant t_{v'}(c) \quad \forall\, v, v' \in \mathcal{V}: \gamma_v^- = \gamma_{v'}^-,\ e_v < e_{v'}, \forall\, c \in \rho_v \cap \rho_{v'} \tag{22}$$

$$\delta_{vv'}(c) + \delta_{v'v}(c) = 1 \quad \forall\, v, v' \in \mathcal{V}: \gamma_v^- \neq \gamma_{v'}^-,\ v < v', \forall\, c \in \rho_v \cap \rho_{v'} \tag{23}$$

$$t_v(c) + \tau_v(c) - t_{v'}(c) \leqslant (1 - \delta_{vv'}(c))M_{vv'} \quad \forall\, v, v' \in \mathcal{V}: \gamma_v^- \neq \gamma_{v'}^-, \forall\, c \in \rho_v \cap \rho_{v'} \tag{24}$$

$$\delta_{vv'}(c) \in \{0, 1\} \quad \forall\, v, v' \in \mathcal{V}: \gamma_v^- \neq \gamma_{v'}^-, \forall\, c \in \rho_v \cap \rho_{v'} \tag{25}$$

To integrate the trajectory optimization model with max-pressure control, additional constraints are introduced to control the activation of vehicle movements and crosswalks. Constraint (26) controls the activation of crosswalks and vehicle movements. z_n^{ped} and z_v^{veh} are binary variables indicating the activation of crosswalk n and vehicle movement v. δ_v^n indicates whether the trajectory of vehicle v intersects with crosswalk n which is determined in advance. If crosswalk n is activated and the trajectory of vehicle v is

conflicting with crosswalk n, then this vehicle is not allowed to move at the current time step. Constraint (27) relates the activation of two vehicles on the same entry lane. If the preceding vehicle is not allowed to move, the following vehicle is not allowed to move either. Constraint (28) plays an important role in relating the max-pressure control with the trajectory optimization model because it builds the relationship between decision variables. t is the current time and Δt is the length of the time interval. When vehicle v is not allowed to move at the current time interval and $z_v^{\text{veh}} = 0$, then constraint becomes $t_v(\gamma_v^+) + \tau_v(\gamma_v^+) \leqslant t + \Delta t + M_v$, which means the exit time of vehicle v of the intersection is not restricted in the current time step and vehicles that are allowed to move at the current time step have priority over vehicle v. If $z_v^{\text{veh}} = 1$, then the constraint becomes $t_v(\gamma_v^+) + \tau_v(\gamma_v^+) \leqslant t + \Delta t$, which means the exit time of vehicle v should be in the current time interval.

$$z_v^{\text{veh}} \leqslant 1 - z_n^{\text{ped}} \delta_v^n \quad \forall\, v \in \mathcal{V}, \forall\, n \in \mathcal{W} \tag{26}$$

$$z_{v'}^{\text{veh}} \leqslant z_v^{\text{veh}} \quad \forall\, v, v' \in \mathcal{V}: \gamma_v^- = \gamma_{v'}^-, e_v < e_{v'} \tag{27}$$

$$t_v(\gamma_v^+) + \tau_v(\gamma_v^+) \leqslant t + \Delta t + (1 - z_v^{\text{veh}}) M_v \quad \forall\, v \in \mathcal{V} \tag{28}$$

$$z_v^{\text{veh}} \in \{0, 1\} \quad \forall\, v \in \mathcal{V} \tag{29}$$

$$z_n^{\text{ped}} \in \{0, 1\} \quad \forall\, n \in \mathcal{W} \tag{30}$$

Compared with model (8), AIM-*ped* has a smaller convex hull that contains all possible intersection control matrices because it has more constraints. If we generate a control policy based on the convex null mentioned in Eq. (13), this control policy may not be feasible in AIM-*ped* If we sum up the number of vehicles on the same turning movement to get y_{ij} and S_{ij} and create a feasible set that has the same form as the feasible set in model (8), feasible set of AIM-*ped* is smaller than model (8). Therefore, the largest demand that can be accommodated by model (8) and AIM-*ped* are different.

5. Numerical experiments

To test the effects of the pedestrian demand on the efficiency of the vehicle network, simulations with different pedestrian and vehicle demands are conducted. A multi-layer network is used in the simulation, as shown in Fig. 5.

5.1. Vehicle network

The first layer is the vehicle layer consisting of a 7-by-7 grid of intersections. Any two adjacent intersections are connected by a pair of directed links that are 1800 feet long. These directed links represent a two-way road that connects two adjacent intersections. What is not depicted is that each of these directed links is segmented into 4 intermediate links of equal distance (450 feet). These

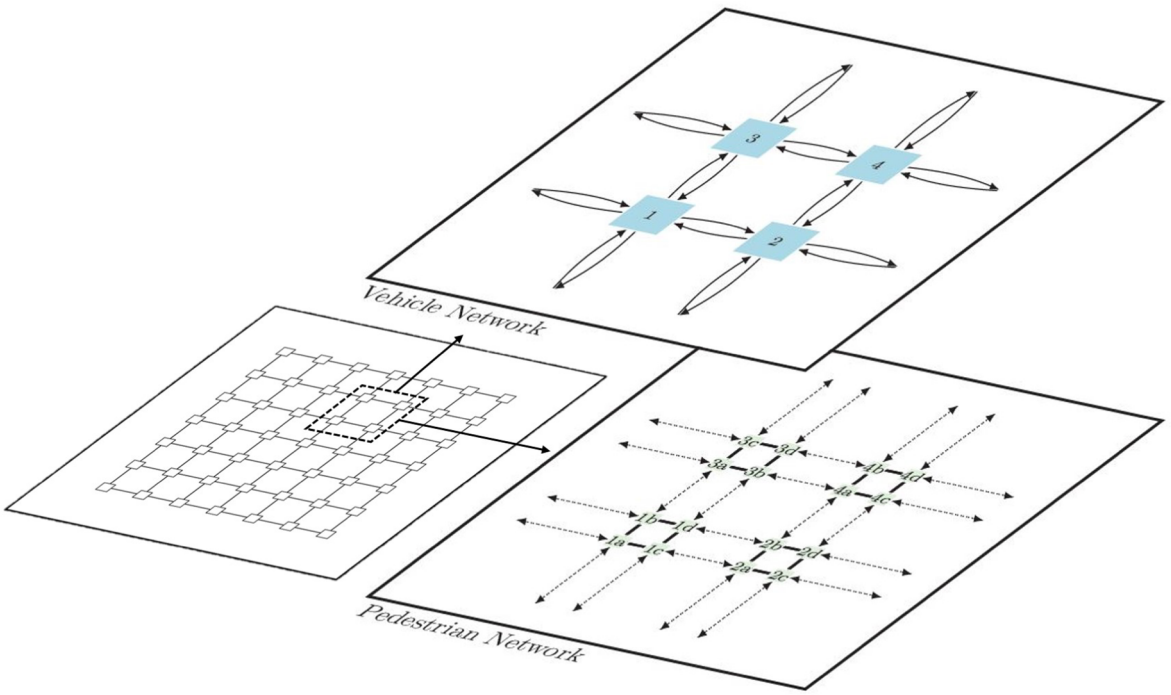


Fig. 5. 7-by-7 grid test network.

intermediate links are required for the assumption that each link has a free flow travel time of 1 time step. Throughout the edge of the grid reside vehicle centroids, denoting either a vehicle origin or destination, adjacent to every intersection on the edge of the grid.

5.2. Pedestrian network

The second layer of the network is the pedestrian layer consisting of sidewalks, crosswalks, and pedestrian nodes as depicted in Fig. 5. The second layer overlaps the vehicle layer. For every vehicle intersection there resides four pedestrian nodes, each of which is connected by a crosswalk. In Fig. 5, vehicle node 1 and pedestrian nodes $1a$, $1b$, $1c$, and $1d$ correspond to the same intersection. The solid lines connecting each pair of these four pedestrian nodes are crosswalks. Each crosswalk is controlled by a signal, which if activated, all pedestrians who are waiting on this crosswalk can move, and any vehicles who have turning movements that conflict with this crosswalk cannot move. Crosswalks are either active or inactive for an entire timestep of 15 s. Any two adjacent groups of four pedestrian nodes are connected by a pair of undirected links that are 1800 feet and overlay the directed vehicle links that connect adjacent intersections. These two undirected pedestrian links represent two sidewalks on both sides of a road. Pedestrians can move in both directions on either sidewalk, but cannot cross the road from one sidewalk to another. A pedestrian can only switch links, or sidewalks, by reaching the end of its current link and either traversing a crosswalk or entering another link directly that is not blocked by a crosswalk (jaywalking is not modeled). Like vehicle links, pedestrian links are also segmented into intermediate links. Pedestrian links are made up of 40 intermediate links of equal distance (45 feet). We assume that a pedestrian will be able to traverse 45 feet in a timestep of 15 s at a speed of 2 miles per hour. At the corner of every intersection is a pedestrian centroid, denoting either a pedestrian origin or destination.

5.3. Simulation parameters

The simulation time is 3 h with a timestep of 15 s. Vehicles and pedestrians continually enter the network at varying demand rates. The vehicle demand ranges from 2 to 18 vehicles per hour per origin-destination pair and the pedestrian demand ranges from 0 to 10 pedestrians per hour per origin-destination pair. Upon creation, each vehicle and pedestrian has a specified origin centroid and a specified destination centroid. For every vehicle, a random shortest path is generated and followed until the vehicle reaches its destination. For every pedestrian, a shortest path is selected from all predetermined paths and followed until the pedestrian reaches its destination. Every pedestrian origin-destination pair has 5 predetermined paths and when a pedestrian is generated, it will randomly pick one path from the 5 possible. Due to the large number of possible pedestrian paths, and the typical lack of congestion on sidewalks, we restrict the set of possible pedestrian paths to 5 per origin-destination.

Two sets of simulations are run. One set is run using estimated pedestrian queue lengths $\hat{x}_{ij}^{\text{ped}}$ as described earlier in the paper, and results are shown in Section 5.4. Another set is run using actual pedestrian queue lengths with x_{ij}^{ped} equal to the actual number of pedestrians waiting for the specific turning movement from i to j, and results are shown in Section 5.5. For every time step, mixed-integer linear programs are created and solved for each of the intersections using CPLEX. Computation times are analyzed in Section 5.6 and the conflict rate is analyzed in Section 5.7.

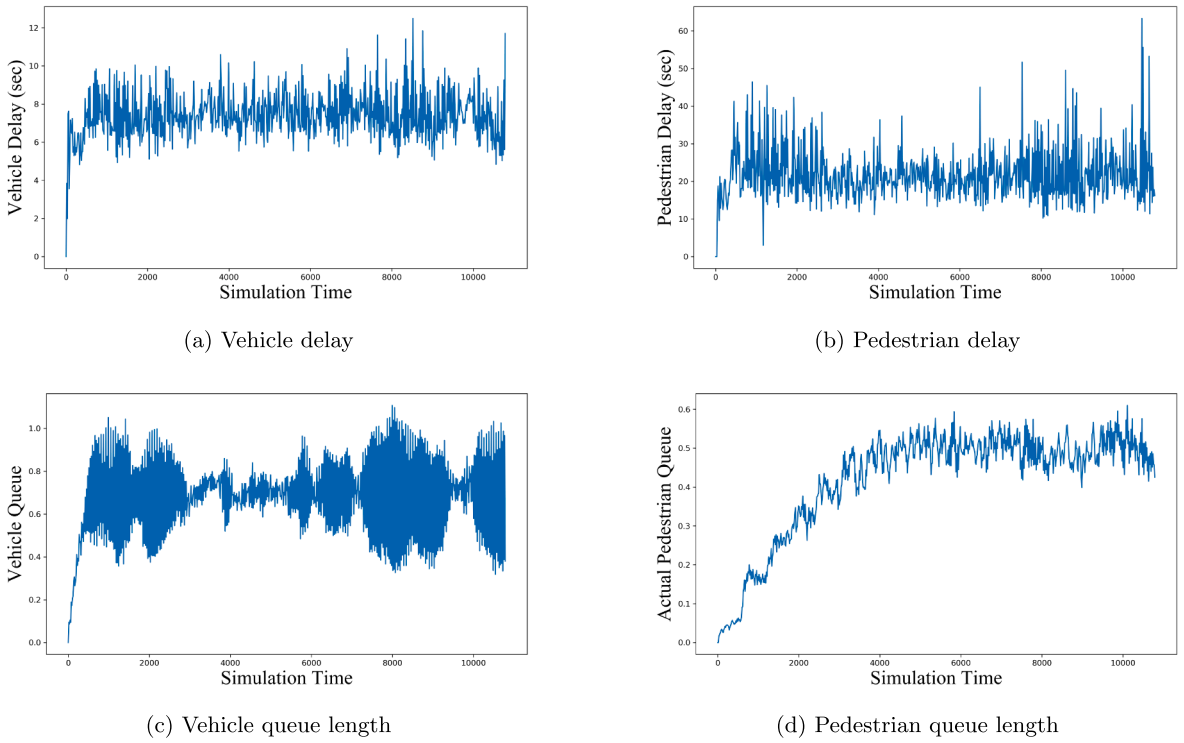
5.4. Results with estimated pedestrian queue lengths

The simulation is run over a time period of three hours and vehicle delay, pedestrian delay, vehicle queue length, and pedestrian queue length are all bounded. Vehicle delay, pedestrian delay, and vehicle queue length become stable after around 500 s, or about 8 min into the simulation, as shown in Fig. 6(a)–(c). Pedestrian queue length becomes stable after 4000 s, or 67 min into the simulation, as shown in Fig. 6(d).

In Fig. 7(a), we can see that an increase in pedestrian demand results in an increase in average vehicle delay. This behavior is expected because a high relative pedestrian demand would result in a higher priority for pedestrians to move at an intersection and thus a higher delay for vehicles. An increase in vehicle demand has more of a mixed effect on average vehicle delay. When vehicle demand is low, increasing vehicle demand decreases average vehicle delay, and when vehicle demand is high, increasing vehicle demand increases average vehicle delay. This could be because of a balancing act between two main factors that drive vehicle delay at an intersection: queue length and vehicle weights. When vehicle demand is low, queue lengths are low and as a result vehicle weights are low. Vehicles will be given low priority compared to pedestrians and vehicle delay will be higher. However, when vehicle demand is high, queue lengths will tend to be higher and so will vehicle weights but since only a certain number of vehicles can move in a timestep there could be situations where vehicles are given high priority to move but will still have to wait for multiple timesteps because queues are long, resulting in higher delay.

Fig. 7(b) describes the relationship between average pedestrian delay, vehicle demand, and pedestrian demand. As a trend average pedestrian delay increases as vehicle demand increases. This is explained by the fact that more vehicles at an intersection give lower priority to pedestrians, increasing the delay. Fig. 7(b) also implies that as pedestrian demand increases, pedestrian delay decreases, regardless of whether or not pedestrian demand is high or low. This is because if a crosswalk activates, all the pedestrians waiting on that crosswalk can move. As a result, there's no conflict between pedestrian queue lengths and pedestrian weights as there is with vehicles.

Fig. 7(c) shows the trend that the average vehicle queue length increases with the vehicle demand. The effect of pedestrian demand on vehicle queue length is not significant. We would expect that a higher pedestrian demand would increase vehicle queue



(a) Vehicle delay

(b) Pedestrian delay

(c) Vehicle queue length

(d) Pedestrian queue length

Fig. 6. Simulation results with the vehicle demand of 10 vehicles per hour per O-D pair and the pedestrian demand of 4 pedestrians per hour per O-D pair.

length, since more pedestrians would result in higher priority for pedestrians at intersections and thus a higher likelihood that vehicles will not move and a vehicle queue will accumulate. However, pedestrian demand seems to have a minimal effect on vehicle demand in Fig. 7(c).

Fig. 7(d) describes the relationship between average pedestrian queue length, vehicle demand, and pedestrian demand. Increasing pedestrian demand increases average pedestrian queue length, and increasing vehicle demand has a similar effect.

Fig. 7(f) describes the relation between the difference between actual and estimated pedestrian queue lengths, vehicle demand, and pedestrian demand. The difference between actual and estimated pedestrian queue lengths increases as pedestrian demand increases. This is because in our simulation using estimated pedestrian queue lengths, the estimated pedestrian queue length for every turning movement is constant. However, for higher pedestrian demand, once a crosswalk is activated, the flow of pedestrians to downstream links will be much higher than the constant estimated pedestrian queue length.

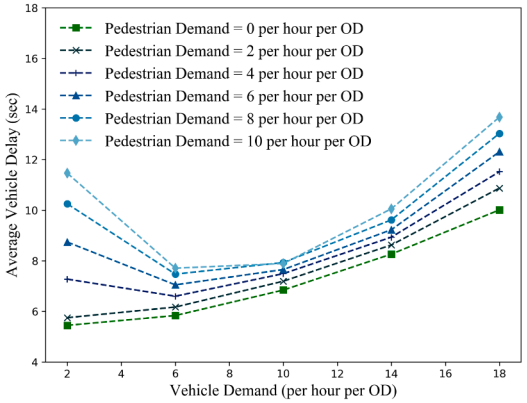
5.5. Results with actual pedestrian queue lengths

In Fig. 8(a), the effect of pedestrian demand and vehicle demand on the vehicle delay is similar to that in Section 5.4. An increase in the pedestrian demand results in larger vehicle delay. An increase in the vehicle demand reduces the vehicle delay when the vehicle demand is small but increases the vehicle delay when the vehicle demand is large. When the vehicle demand is between 10 and 14 vehicles per hour per OD pair, the change in the vehicle demand does not significantly affects the vehicle delay. Compared with the vehicle delay in Section 5.4, the average vehicle delay here is larger because the estimated pedestrian queue length is lower than the actual queue length which gives higher priority to pedestrians in the intersection control.

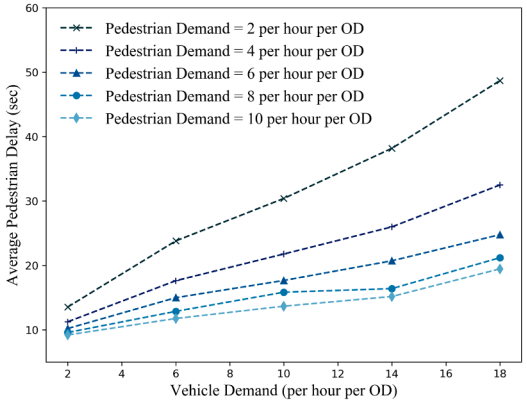
In Fig. 8(b), an increase in the pedestrian demand reduces the pedestrian delay and an increase in vehicle demand increases the pedestrian delay. Pedestrian delay is on average higher in Fig. 7 than in Fig. 8. This is because our estimated pedestrian queue length for an intersection is lower than the actual pedestrian queue length, as shown in Fig. 7(f).

In Fig. 8(c), increases in the pedestrian demand and the vehicle demand both increase the vehicle queue length. Compared with Fig. 7(c), the effects of pedestrian demand on the vehicle delay is more significant in Fig. 8(c).

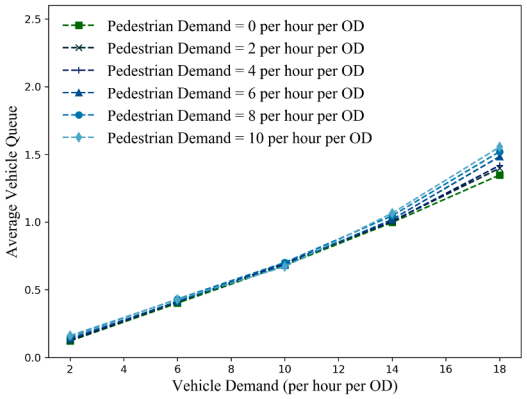
Fig. 8(d) describes the relationship between average pedestrian queue length, vehicle demand, and pedestrian demand. An increase in the pedestrian demand results in an increase in the pedestrian delay. An increase in the vehicle demand also increases the pedestrian delay but the effect is not significant.



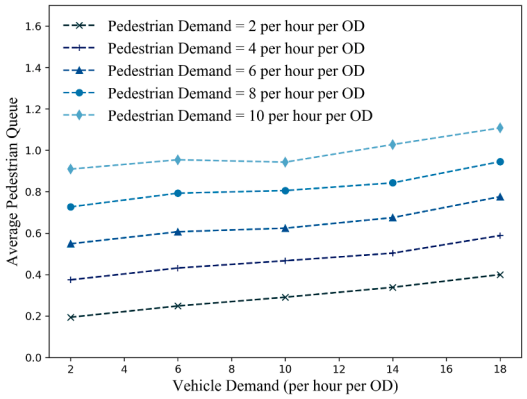
(a) Vehicle delay



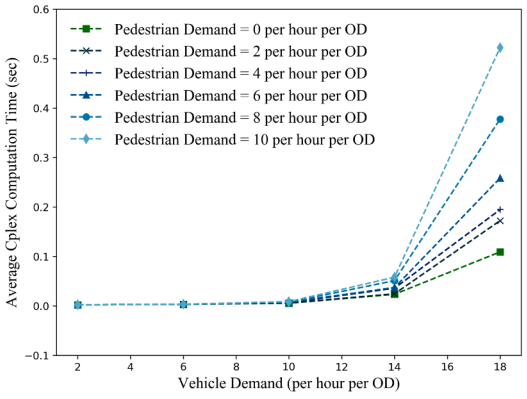
(b) Pedestrian delay



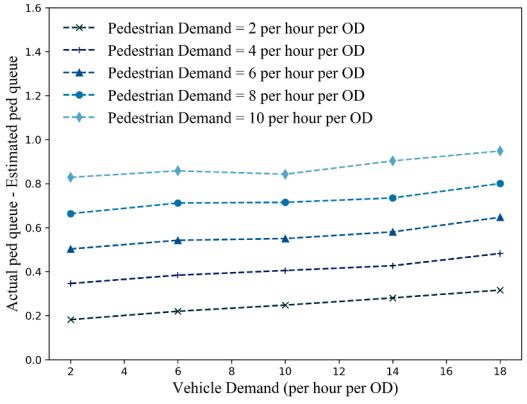
(c) Vehicle queue length



(d) Pedestrian queue length



(e) CPLEX calculation time

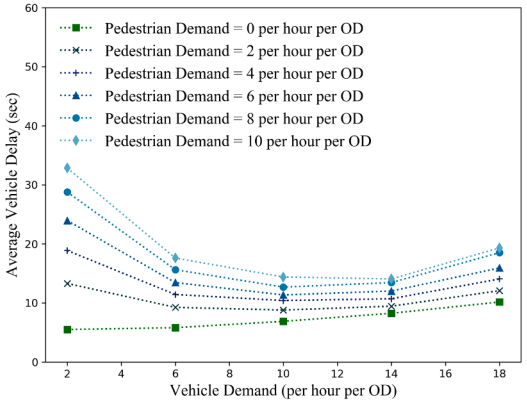


(f) The difference between the actual and the estimated pedestrian queue lengths

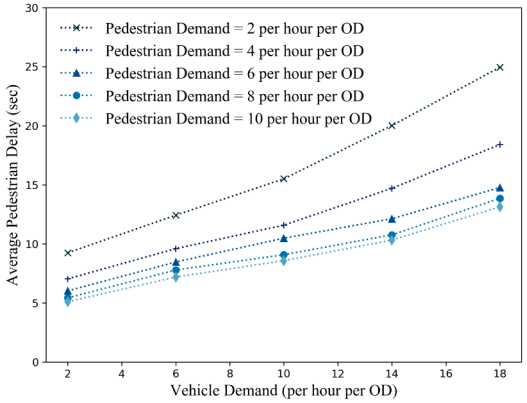
Fig. 7. Simulation result using estimated pedestrian queue length.

5.6. Discussion of computation times

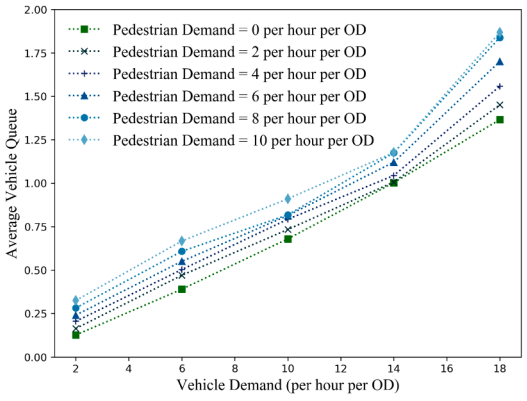
Figs. 7(e) and 8(e) describe the relation between CPLEX computation time, vehicle demand, and pedestrian demand. It is the average CPLEX computation time for one time step and one intersection in the simulation. As shown in Figs. 7(e) and 8(e), CPLEX computation time is close to zero when the vehicle demand is small and spikes upwards for high vehicle and pedestrian demand. This is because the high vehicle and pedestrian demands increase the vehicle queue lengths, and every vehicle creates several additional variables in the mixed-integer program that CPLEX needs to solve. Vehicles with different moving directions create different numbers



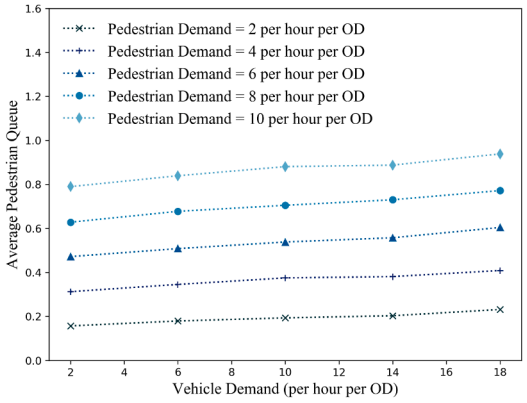
(a) Vehicle delay



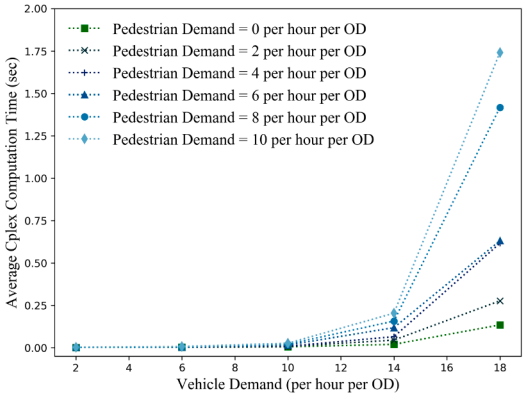
(b) Pedestrian delay



(c) Vehicle queue length



(d) Pedestrian queue length



(e) CPLEX calculation time

Fig. 8. Simulation result using actual pedestrian queue length.

of variables in the mixed-integer program which related with the number of conflict points they would pass through. As the proposed algorithm is a distributed algorithm, we only need the CPLEX computation time for an individual intersection to be smaller than the step size if we want to implement it in real-time. In the simulation, the average CPLEX computation time per intersection under all demands is far smaller than the time step size (15 s), which indicates that the algorithm can possibly be used in real-time.

5.7. The effect of pedestrian and vehicle demand on the conflict rate

Fig. 9 describes the relationship between pedestrian demand and the similarity of the simulation to a phase-based intersection

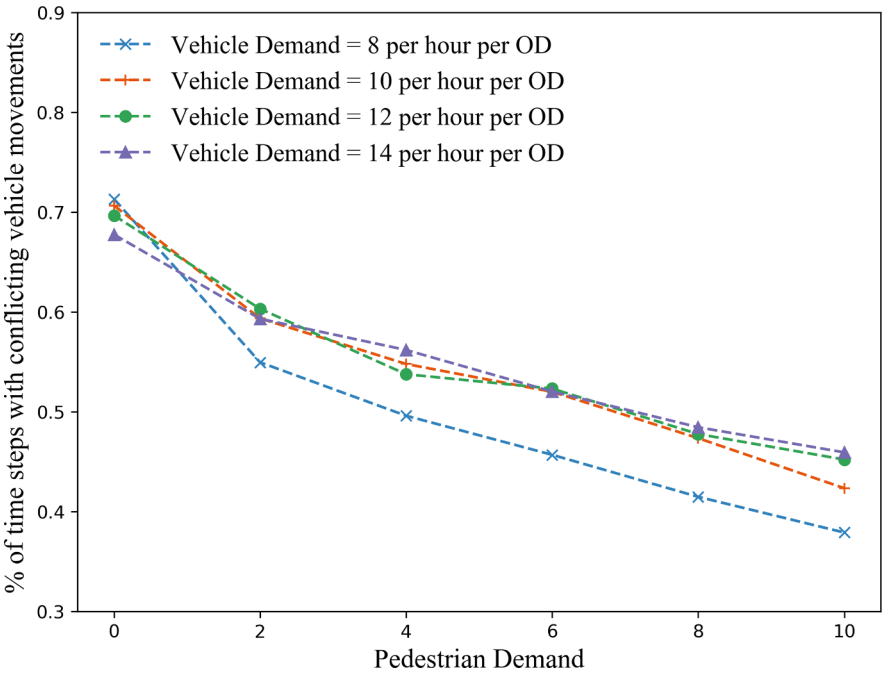


Fig. 9. Percent of time steps that allow conflicting vehicles to move.

control. As pedestrian demand increases, the percentage of time steps in which conflicting vehicle movements are simultaneously permitted to traverse an intersection decreases. The simulation becomes more and more like a traditional traffic light system where only non-conflicting vehicles can move in a certain time period. This is because of the high pedestrian demand and high pedestrian priority at intersections, there is a higher likelihood that crosswalks will be activated in any given timestep which restricts all vehicle movements that go through that crosswalk. Crosswalk activation limits the different possible vehicle movements in a time step making it more likely that moving vehicles have non-conflicting trajectories. Even when the pedestrian demand is high, the algorithm still allows about 40% of the conflicting vehicle movements, which means this algorithm can adapt to pedestrian demand and activate conflicting movements when optimal.

6. Conclusion

This study proposes an autonomous intersection management algorithm based on max-pressure control considering both vehicles and pedestrians. This study defines the stability region of the traffic demand and proves that this algorithm can produce throughput-optimal intersection control at network level. To apply this algorithm in simulation, this study combines an existing trajectory optimizing algorithm with max-pressure control and formulates a mixed-integer program model to calculate the optimal trajectories of vehicles and optimal control of pedestrian signals based on the max-pressure control. In the simulation, the proposed algorithm has a small computation time when the vehicle demand is small. Simulation results show that the pedestrian and vehicle delays become stable in a short time and the difference between the actual and the estimated pedestrian queue lengths are bounded. Simulation results with different demands of pedestrians and vehicles show a trade-off between the efficiency of vehicles and pedestrians in this algorithm. Delays of pedestrian and vehicles are negatively correlated. When demands of vehicles and pedestrian increase, the intersection control produced by this algorithm is more similar to phase-based intersection control, but this algorithm is still capable of adapting to the change of the pedestrian demand and allows vehicles with conflicting trajectories to pass the intersection.

In future work, different formulas will be used to calculate the pressure of turning movements. The stability properties of the algorithm will be explored under different assumptions, such as finite queue capacity. Other network components will be added to the model, such as cyclists or legacy vehicles. The network model used in this study is based on the point-queue model, the future work may use other network models, such as the spatial queue model and the cell transmission model, which results in more realistic results.

CRediT authorship contribution statement

Rongsheng Chen: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Jeffrey Hu:** Methodology, Software, Investigation, Writing - original draft, Visualization. **Michael W. Levin:** Conceptualization, Methodology, Software, Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition. **David Rey:** Conceptualization, Writing - review & editing.

Acknowledgements

The authors gratefully acknowledge the support of the National Science Foundation, Award No. 1935514.

Appendix A

Proof for Proposition 1. To calculate the queue length at time $t+1$, we apply the point queue model shown in Eqs. (1) and (2).

$$x_{ij}(t+1) = x_{ij}(t) - \min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right) + \sum_{\mathrm{h}\in\Gamma_{\mathrm{i}}^{-}} \min\left(\mathrm{Q_{hi}S_{hi}(t)}, \mathrm{x_{hi}(t)}\right)\mathrm{p_{ij}(t)} \quad \forall\, \mathrm{i}\in\mathcal{L}_{\mathrm{int}}, \mathrm{j}\in\Gamma_{\mathrm{i}}^{+} \tag{31}$$

$$x_{ij}(t+1) = x_{ij}(t) - \min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right) + d_i(t)p_{ij}(t) \quad \forall\, i\in\mathcal{L}_{\mathrm{entry}}, j\in\Gamma_i^{+} \tag{32}$$

Then we get the difference in the queue length between two consecutive time steps.

$$\Delta_{ij} = x_{ij}(t+1) - x_{ij}(t) = -min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right) + \sum_{h\in\Gamma_i^{-}} min\left(Q_{hi}S_{hi}(t), x_{hi}(t)\right)p_{ij}(t) \quad \forall\, i\in\mathcal{L}_{int}, j\in\Gamma_i^{+} \tag{33}$$

$$\Delta_{ij} = -\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right) + d_i(t)p_{ij}(t) \quad \forall\, i\in\mathcal{L}_{\mathrm{entry}}, j\in\Gamma_i^{+} \tag{34}$$

Let $X(t)$ be a matrix including all queue lengths of all vehicle movements and pedestrian movements.

Here we choose the Lyapunov function

$$\left|X(t)\right|^2 = \sum_{(i,j)\in\mathcal{L}} \left(x_{ij}(t)\right)^2 \tag{35}$$

$$|X(t+1)|^2 - |X(t)|^2 = |X(t)+\Delta|^2 - |X(t)|^2 = 2X(t)^{\mathrm{T}}\Delta + |\Delta|^2 \tag{36}$$

$$\begin{aligned} 2X(t)^{\mathrm{T}}\Delta = {} & -2\textstyle\sum_{i\in\mathcal{L}}\sum_{j\in\Gamma_i^{+}} x_{ij}(t)\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right) \;+\; 2\sum_{\mathrm{h}\in\Gamma_{\mathrm{i}}^{-}}\sum_{\mathrm{i}\in\mathcal{L}}\sum_{\mathrm{j}\in\Gamma_{\mathrm{i}}^{+}} \mathrm{x_{ij}(t)}\min\left(\mathrm{Q_{hi}S_{hi}}, \mathrm{x_{hi}(t)}\right)\mathrm{p_{ij}(t)} \\ & + 2\textstyle\sum_{i\in\mathcal{L}_{\mathrm{entry}}}\sum_{j\in\Gamma_i^{+}} x_{ij}(t)\left(-\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right) + d_i(t)p_{ij}(t)\right) \end{aligned} \tag{37}$$

$$= 2\sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}}\sum_{j\in\Gamma_i^{+}} \min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right)\left(-x_{ij}(t) + \sum_{k\in\Gamma_j^{+}} p_{jk}(t)x_{jk}(t)\right) + 2\sum_{\mathrm{i}\in\mathcal{L}_{\mathrm{entry}}}\sum_{\mathrm{j}\in\Gamma_{\mathrm{i}}^{+}} \mathrm{d_i(t)p_{ij}(t)x_{ij}(t)} \tag{38}$$

In Eq. (39), the random variable p is replaced with its mean value $\bar{p}$

$$\begin{aligned} \mathbb{E}\left[X(t)^{\mathrm{T}}\Delta|X(t)\right] = {} & \sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}}\sum_{j\in\Gamma_i^{+}} \mathbb{E}\left[\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right)\left(-x_{ij}(t)\right)|X(t)\right] \\ & + \sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}}\sum_{j\in\Gamma_i^{+}} \mathbb{E}\left[\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right)|X(t)\right]\left(\sum_{k\in\Gamma_j^{+}} \bar{p}_{jk}x_{jk}(t)\right) \\ & + \sum_{i\in\mathcal{L}_{\mathrm{entry}}}\sum_{j\in\Gamma_i^{+}} \mathbb{E}\left[d_i(t)\bar{p}_{ij}x_{ij}(t)|X(t)\right] \end{aligned} \tag{39}$$

$$\mathbb{E}\left[\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right)\bar{p}_{jk}(t)x_{jk}(t)|X(t)\right] = \mathbb{E}\left[\min\left(Q_{ij}S_{ij}(t), x_{ij}(t)\right)|X(t)\right]\bar{p}_{jk}x_{jk}(t) \tag{40}$$

By the definition of the pressure, $-x_{ij}(t) + \sum_{k\in\Gamma_j^{+}} \bar{p}_{jk}x_{jk}(t) = -w_{ij}(t)$, so we obtain

$$\mathbb{E}\left[\left.X(t)^{\mathrm{T}}\Delta\right|X(t)\right] = -\sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}} \mathbb{E}\left[\left.\min\left(Q_{ij}S_{ij}(t), x_{ij}\right)\right|X(t)\right]w_{ij}(t) + \sum_{i\in\mathcal{L}_{\mathrm{entry}}} \bar{d}_i\bar{p}_{ij}x_{ij}(t) \tag{41}$$

$$\sum_{i\in\mathcal{L}_{\mathrm{entry}}} \bar{d}_i\bar{p}_{ij}x_{ij}(t) = \sum_{i\in\mathcal{L}_{\mathrm{entry}}} f_{ij}x_{ij}(t) \tag{42}$$

$$= \sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}} f_i\bar{p}_{ij}x_{ij}(t) - \sum_{j\in\mathcal{L}_{\mathrm{int}}} f_j\bar{p}_{jk}x_{jk}(t) \tag{43}$$

$$= \sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}} f_i\bar{p}_{ij}x_{ij}(t) - \sum_{j\in\Gamma_i^{+}}\left[\sum_{i\in\mathcal{L}_{\mathrm{int}}\cup\mathcal{L}_{\mathrm{entry}}} f_i\bar{p}_{ij}\right]\sum_k \bar{p}_{jk}x_{jk}(t) \tag{44}$$

$$= \sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} f_i \bar{p}_{ij} \left[x_{ij}(t) - \sum_{k \in \Gamma_i^+} \bar{p}_{jk} x_{jk}(t) \right] \tag{45}$$

$$= \sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} f_i \bar{p}_{ij} w_{ij} \tag{46}$$

Therefore,

$$\mathbb{E}[X(t)^{\mathrm{T}} \Delta | X(t)] = \sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} (f_i \bar{p}_{ij} - \mathbb{E}[\min(Q_{ij} S_{ij}, x_{ij}(t)) | X(t)]) w_{ij}(t) \tag{47}$$

$$= \sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} (f_i \bar{p}_{ij} - \bar{Q}_{ij} S_{ij}(t)) w_{ij}(t) + \sum_{i \in \mathcal{L}_{int} \cup \mathcal{L}_{entry}} (\bar{Q}_{ij} S_{ij}(t) - \mathbb{E}[min(Q_{ij} S_{ij}(t), x_{ij}(t)) | X(t)]) w_{ij}(t) \tag{48}$$

$\bar{Q}$ is the average value of the random variable Q. Q is the maximum value of Q. For vehicle flow (i, j), $Q_{ij}^{\text{veh}} S_{ij}^{\text{veh}}(t) = y_{ij}^{\text{veh}}(t) \leqslant x_{ij}^{\text{veh}}(t)$, so $\min(Q_{ij}^{\text{veh}} S_{ij}^{\text{veh}}(t), x_{ij}^{\text{veh}}(t)) = Q_{ij}^{\text{veh}} S_{ij}^{\text{veh}}(t)$. Then,

$$(\bar{Q}_{ij}^{\text{veh}} S_{ij}^{\text{veh}}(t) - Q_{ij}^{\text{veh}} S_{ij}^{\text{veh}}(t)) w_{ij}(t) \leqslant \bar{Q}_{ij}^{\text{veh}} w_{ij}(t) \leqslant \bar{Q}_{ij}^{\text{veh}} Q_{ij} \tag{49}$$

For pedestrian flow (i, j), $S_{ij}^{\text{ped}}(t) \in \{0, 1\}$. to get the upper bound, $S_{ij}^{\text{ped}}(t)$ is pulled out.

$$(\bar{Q}_{ij}^{\text{ped}} S_{ij}^{\text{ped}}(t) - \mathbb{E}[\min(Q_{ij}^{\text{ped}} S_{ij}^{\text{ped}}(t), x_{ij}^{\text{ped}}(t)) | X(t)]) w_{ij}(t) = (\bar{Q}_{ij}^{ped} - \mathbb{E}[min(Q_{ij}^{ped}, x_{ij}^{ped}(t)) | X(t)]) S_{ij}^{ped}(t) w_{ij}^{ped}(t) \tag{50}$$

When $x_{ij}(t) \geqslant Q_{ij}^{\text{ped}}$, $(\bar{Q}_{ij}^{\text{ped}} - \mathbb{E}[\min(Q_{ij}^{\text{ped}}, x_{ij}^{\text{ped}}(t)) | X(t)]) = 0$. Otherwise,

$$(\bar{Q}_{ij}^{\text{ped}} - \mathbb{E}[\min(Q_{ij}^{\text{ped}}, x_{ij}^{\text{ped}}(t)) | X(t)]) = x_{ij}(t), \text{ and } x_{ij}(t) w_{ij}^{\text{ped}}(t) \leqslant \bar{Q}_{ij}^{\text{ped}} Q_{ij}^{\text{ped}} \tag{51}$$

Therefore,

$$\sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} (\bar{Q}_{ij} S_{ij}(t) - \mathbb{E}[\min(Q_{ij} S_{ij}(t), x_{ij}(t)) | X(t)]) w_{ij}(t) \leqslant \sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} \bar{Q}_{ij} Q_{ij} \tag{52}$$

At each time step, an intersection control matrix $S_r(t)$ is selected from the set of signal control matrices. The max-pressure algorithm can get an intersection control matrix with the maximum pressure

$$S^* = \underset{S_{ij}, S_n}{\operatorname{argmax}} \left\{ \sum_{(i,j) \in \mathcal{M}} S_{ij} Q_{ij} w_{ij}^{\text{veh}} + \sum_{n \in \mathcal{W}} S_n Q_n w_n^{\text{ped}} \right\} \tag{53}$$

As demand vector $\mathbf{d}$ is in the stability region, the relation between the arrival rate and the intersection control has the relation $f_i \bar{p}_{ij} + \epsilon = \bar{S}_{ij} \bar{Q}_{ij}$ for both vehicle turning movements and pedestrian flows, where $\epsilon > 0$. Inequality 55 holds because of Lemma 3.1.

$$\begin{aligned} &\mathbb{E}\left[\sum_{i \in \mathcal{L}_{\text{int}}^{\text{veh}} \cup \mathcal{L}_{\text{entry}}^{\text{veh}}} \left[f_i \bar{p}_{ij} - Q_{ij} S_{ij}^*(t) \right] w_{ij}^{\text{veh}}(t) \right] + \mathbb{E}\left[\sum_{m \in \mathcal{L}_{\text{int}}^{\text{ped}} \cup \mathcal{L}_{\text{entry}}^{\text{ped}}} \left[f_m \bar{p}_{mn} - Q_{mn} S_n^*(t) \right] w_{mn}^{\text{ped}}(t) \right] \\ &\leqslant \mathbb{E}\left[\sum_{i \in \mathcal{L}_{\text{int}}^{\text{veh}} \cup \mathcal{L}_{\text{entry}}^{\text{veh}}} \left[f_i \bar{p}_{ij} - Q_{ij} \bar{S}_{ij}^{\text{veh}} \right] w_{ij}^{\text{veh}}(t) \right] + \mathbb{E}\left[\sum_{m \in \mathcal{L}_{\text{int}}^{\text{ped}} \cup \mathcal{L}_{\text{entry}}^{\text{ped}}} \left[f_m \bar{p}_{mn} - Q_{mn} \bar{S}_{mn}^{\text{ped}} \right] w_{mn}^{\text{ped}}(t) \right] \end{aligned} \tag{54}$$

$$\leqslant \mathbb{E}\left[\sum_{i \in \mathcal{L}_{\text{int}}^{\text{veh}} \cup \mathcal{L}_{\text{entry}}^{\text{veh}}} \left[f_i \bar{p}_{ij} - Q_{ij} \bar{S}_{ij}^{\text{veh}} \right] w_{ij}^{\text{veh}}(t) \right] + \mathbb{E}\left[\sum_{m \in \mathcal{L}_{\text{int}}^{\text{ped}} \cup \mathcal{L}_{\text{entry}}^{\text{ped}}} \left[f_m \bar{p}_{mn} - Q_{mn} \bar{S}_{mn}^{\text{ped}} \right] \hat{w}_{mn}^{\text{ped}}(t) \right] + \left| \mathcal{N}^{\text{ped}} \right| \xi \tag{55}$$

$\bar{S}_{ij}^{\text{veh}}$ and $\bar{S}_{mn}^{\text{ped}}$ correspond to the average actuation rates of the turning movement (i, j) and the crosswalk n. If $w_{ij}^{\text{veh}}(t) > 0$, then turning movement (i, j) will be activated. So the average actuation rate for the period when turning movement (i, j) is activated times the turning capacity has the relation $Q_{ij} \bar{S}_{ij}^{\text{veh}} > f_i \bar{p}_{ij} + \epsilon$ because $\bar{S}_{ij}^{\text{veh}}$ should satisfy that $Q_{ij} \bar{S}_{ij}^{\text{veh}} > f_i \bar{p}_{ij}$. Otherwise, when $w_{ij}^{\text{veh}} \leqslant 0$, $Q_{ij} \bar{S}_{ij}^{\text{veh}} = 0$ because turning movement (i, j) is not actuated based on max-pressure algorithm. Similarly, for crosswalk n, when $w_{mn}^{\text{ped}} > 0$, the average actuation rate times the capacity have the relation $Q_{mn} \bar{S}_{ij}^{\text{ped}} = f_i \bar{p}_{ij} + \epsilon$. Otherwise, $w_{ij}^{\text{ped}} \leqslant 0$, $Q_{ij} \bar{S}_{ij}^{\text{ped}} = 0$.

$$\begin{aligned} &\textstyle\sum_{i \in \mathcal{L}_{\text{int}} \cup \mathcal{L}_{\text{entry}}} [f_i \bar{p}_{ij} - Q_{ij} \bar{S}_{ij}(t)] w_{ij}(t) \leqslant -\epsilon \sum_{ij} \max\{w_{ij}^{\text{veh}}, 0\} + \sum_{ij} f_i \bar{p}_{ij} \max\{-w_{ij}^{\text{veh}}, 0\} \\ &\qquad - \epsilon \textstyle\sum_{mn} \max\{w_{mn}^{\text{ped}}, 0\} + \sum_{mn} f_m \bar{p}_{mn} \max\{-\hat{w}_{mn}^{\text{ped}}, 0\} \end{aligned} \tag{56}$$

$$\leqslant -\epsilon |w_{ij}^{\text{veh}}| - \epsilon |w_{mn}^{\text{ped}}| \tag{57}$$

As we assume that $|x_{mn}^{\text{ped}}(t) - \hat{x}_{mn}^{\text{ped}}(t)| \leqslant \omega$, and $|w_{mn}^{\text{ped}} - \hat{w}_{mn}^{\text{ped}}| \leqslant \beta$ and based on Lemma 3.1, we have

$$\sum_{i\in\mathcal{L}_{\text{int}}\cup\mathcal{L}_{\text{entry}}} \left[f_i \bar{p}_{ij} - Q_{ij} S_{ij}(t) \right] w_{ij}(t) \leqslant -\epsilon \left| w_{ij}^{\text{veh}} \right| - \epsilon \left| \hat{w}_{ij}^{\text{ped}} \right| + \left| \mathcal{N}^{\text{ped}} \right| \xi \tag{58}$$

The formula used to get the pressure w is a linear function of the matrix X that represent the queue length, so we can find $\eta_1, \eta_2 > 0$ such that $\sum_{ij\in\mathcal{M}} w_{ij}^{\text{veh}} \geqslant \eta_1 \left| X^{\text{veh}}(t) \right|$ and $\sum_{mn\in\mathcal{W}} w_{mn}^{\text{ped}} \geqslant \eta_2 |X^{\text{ped}}(t)|$. Therefore, we have

$$-\epsilon |w_{ij}^{\text{veh}}| - \epsilon |\hat{w}_{ij}^{\text{ped}}| + |\mathcal{N}^{\text{ped}}|\xi \leqslant -\epsilon\, \eta_1 |X^{\text{veh}}(t)| - \epsilon\, \eta_2 |X^{\text{ped}}(t)| + |\mathcal{N}^{\text{ped}}|\xi \tag{59}$$

$$\leqslant \sum_{i\in\mathcal{L}_{\text{int}}\cup\mathcal{L}_{\text{entry}}} \overline{Q}_{ij} Q_{ij} - \epsilon\, \eta_1 \left| X^{\text{veh}}(t) \right| - \epsilon\, \eta_2 \left| X^{\text{ped}}(t) \right| + \left| \mathcal{N}^{\text{ped}} \right| \xi \tag{60}$$

Let $\eta = \min(\eta_1, \eta_2)$ and let $\psi = |\mathcal{N}^{\text{ped}}|\xi$, we have:

$$\mathbb{E}\left\{ X(t)^{\mathrm{T}} \Delta \middle| X(t) \right\} \leqslant \sum_{i\in\mathcal{L}_{\text{int}}\cup\mathcal{L}_{\text{entry}}} \overline{Q}_{ij} Q_{ij} - \epsilon\, \eta \left| X(t) \right| + \psi \tag{61}$$

For $|\Delta|^2$, we have:

$$\left| \Delta_{ij} \right| = \left| -\min\left\{ Q_{ij} S_{ij}(t), x_{ij}(t) \right\} + \sum_{h\in\Gamma_i^-} \min\left\{ Q_{hi} S_{hi}(t), x_{hi}(t) \right\} p_{ij}(t) \right| \tag{62}$$

$$\leqslant \max\left\{ Q_{ij}, \sum_{h\in\Gamma_i^-} Q_{hi} \right\} \quad \forall\, i \in \mathcal{L}_{\text{int}}, j \in \Gamma_i^+ \tag{63}$$

$$|\Delta_{ij}| = |-\min\{Q_{ij} S_{ij}(t), x_{ij}(t)\} + f_{ij}(t)| \leqslant \max\{Q_{ij}, \hat{d}_{ij}\} \quad \forall\, i \in \mathcal{L}_{\text{entry}}, j \in \Gamma_i^+ \tag{64}$$

where $\hat{d}_{ij}$ represent the maximum value of the flow rate

$$\text{Let } \gamma_1 = \max\left\{ Q_{ij}^{\text{veh}}, \sum_{h\in\Gamma_i^-} Q_{hi}^{\text{veh}}, \hat{d}_{ij}^{\text{veh}} \right\}, \gamma_2 = \max\left\{ Q_{ij}^{\text{ped}}, \sum_{h\in\Gamma_i^-} Q_{hi}^{\text{ped}}, \hat{d}_{ij}^{\text{ped}} \right\} \tag{65}$$

Therefore, $\sum_{ij} \left| \Delta_{ij} \right|^2 \leqslant \mathcal{N}_1 \gamma_1^2 + \mathcal{N}_2 \gamma_2^2$, where $\mathcal{N}_1$ is the total number of vehicle movements and $\mathcal{N}_2$ is the total number of pedestrian movements.

$$|X(t+1)|^2 - |X(t)|^2 = 2X(t)^{\mathrm{T}} \Delta + |\Delta|^2 \tag{66}$$

$$\leqslant 2\left(\sum_{i\in\mathcal{L}_{\text{int}}\cup\mathcal{L}_{\text{entry}}} \overline{Q}_{ij} Q_{ij} - \epsilon\, \eta \left| X(t) \right| + \psi \right) + \mathcal{N}_1 \gamma_1^2 + \mathcal{N}_2 \gamma_2^2 \tag{67}$$

$$= \kappa - \lambda |X(t)| \tag{68}$$

$$\text{where } \kappa = 2 \sum_{i\in\mathcal{L}_{\text{int}}\cup\mathcal{L}_{\text{entry}}} \overline{Q}_{ij} Q_{ij} + \mathcal{N}_1 \gamma_1^2 + \mathcal{N}_2 \gamma_2^2 + \psi,\ \epsilon\, \eta = \lambda \tag{69}$$

Now we have:

$$\mathbb{E}\left[|X(t+1)|^2 - |X(t)|^2 | X(t) \right] \leqslant \kappa - \lambda |X(t)| \tag{70}$$

If we sum from $t = 1$ to T,

$$\sum_{t=1}^{t=T} \left(\mathbb{E}\left[|X(t+1)|^2 - |X(t)|^2 | X(t) \right] \right) \leqslant \sum_{t=0}^{t=T} \left(\kappa - \lambda |X(t)| \right) \tag{71}$$

$$\mathbb{E}\left[|X(T+1)|^2 - |X(1)|^2 | X(t) \right] \leqslant T\kappa - \lambda \mathbb{E}\left[\sum_{t=0}^{t=T} \left| X(t) \right| \right] \tag{72}$$

$$\lambda \mathbb{E}\left[\sum_{t=0}^{t=T} \left| X(t) \right| \right] \leqslant T\kappa + \mathbb{E}\left[|X(1)|^2 \right] - \mathbb{E}\left[|X(T+1)|^2 \right] \leqslant T\kappa + \mathbb{E}\left[|X(1)|^2 \right] \tag{73}$$

$$\frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{t=T}\left|X(t)\right|\right] \leqslant \frac{1}{\lambda}(\varkappa + \mathbb{E}\left[|X(1)|^2\right]) \tag{74}$$

References

Bento, L.C., Parafita, R., Nunes, U., 2012. Intelligent traffic management at intersections supported by V2V and V2I communications. IEEE Conference on Intelligent Transportation Systems, Proceedings ITSC, 1495–1502. https://doi.org/10.1109/ITSC.2012.6338766.

Bento, L.C., Parafita, R., Santos, S., Nunes, U., 2013. In: Intelligent traffic management at intersections: Legacy mode for vehicles not equipped with v2v and v2i communications 2013). IEEE, pp. 726–731.

Bing, B., Carter, A., 1995. Scoot: The world's foremost adaptive traffic control system. In: Traffic Technology International'95.

De Campos, GR., Falcone, P., Hult, R., Wymeersch, H., Sjöberg, J., 2017. Traffic coordination at road intersections: autonomous decision-making algorithms using model-based heuristics. IEEE Intell. Transp. Syst. Mag. 9 (1), 8–21, 2017. ISSN 19391390. doi:https://doi.org/10.1109/MITS.2016.2630585.

Dresner, K., Stone, P., 2004. Multiagent traffic management: a reservation-based intersection control mechanism. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, vol. 2. IEEE Computer Society, pp. 530–537.

Dresner, K., Stone, P., 2007. Learning policy selection for autonomous intersection management. In: The AAMAS 2007 Workshop on Adaptive and Learning Agents (ALAg 2007), pp. 34–39.

Dresner, K., Stone, P., 2006. Traffic intersections of the future. In: Proceedings of the National Conference on Artificial Intelligence. 21. AAAI Press; MIT Press; 1999, CA; Cambridge, MA; London, pp. 1593–1596.

Dresner, K.M., Stone, P., 2007b. Sharing the road: autonomous vehicles meet human drivers. IJCAI 7, 1263–1268.

Fajardo, D., Au, T.-C., Waller, S., Stone, P., Yang, D., 2011. Automated intersection control: performance of future innovation versus current traffic signal control. Transp. Res. Rec.: J. Transp. Res. Board 2259, 223–232.

Fayazi, S.A., Vahidi, A., Luckow, A., 2017. Optimal scheduling of autonomous vehicle arrivals at intelligent intersections via MILP. In: Proceedings of the American Control Conference. ISSN 07431619, pp. 4920–4925. doi:https://doi.org/10.23919/ACC.2017.7963717.

Feng, Y., Head, K.L., Khoshmagham, S., Zamanipour, M., 2015. A real-time adaptive signal control in a connected vehicle environment. Transp. Res. Part C: Emerg. Technol. 55, 460–473.

Gartner, N.H., 1983. OPAC: A demand-responsive strategy for traffic signal control. Number 906.

Goodall, N., Smith, B., Park, B., 2013. Traffic signal control with connected vehicles. Transp. Res. Rec.: J. Transp. Res. Board 2381, 65–72.

Gregoire, J., Frazzoli, E., De La Fortelle, A., Wongpiromsarn, T., 2014. Back-pressure traffic signal control with unknown routing rates, vol. 19. IFAC. ISBN 9783902823625. doi:https://doi.org/10.3182/20140824-6-ZA-1003.01585.

Gregoire, J., Qian, X., Frazzoli, E., De La Fortelle, A., Wongpiromsarn, T., 2014b. Capacity-aware backpressure traffic signal control. IEEE Trans. Control Network Syst. 2 (2), 164–173.

Hausknecht, M., Au, T.-C., Stone, P., 2011. Autonomous intersection management: Multi-intersection optimization. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 4581–4586.

He, Q., Head, K.L., Ding, J., 2012. Pamscod: Platoon-based arterial multi-modal signal control with online data. Transp. Res. Part C: Emerg. Technol. 20 (1), 164–184.

Hsieh, P.-C., Liu, X., Jiao, J., Hou, I.-H., Zhang, Y., Kumar, P.R., 2017. Throughput-Optimal Scheduling for Multi-Hop Networked Transportation. Syst. Switch-Over Delay.

Jin, Q., Wu, G., Boriboonsomsin, K., Barth, M., 2012. Multi-agent intersection management for connected vehicles using an optimal scheduling approach. In: Proceedings – 2012 International Conference on Connected Vehicles and Expo, ICCVE, pp. 185–190. https://doi.org/10.1109/ICCVE.2012.41.

Kamal, M.A., Imura, J., Ohata, A., Hayakawa, T., Aihara, K., 2013. Coordination of automated vehicles at a traffic-lightless intersection. In: IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, (Itsc), pp. 922–927. doi:https://doi.org/10.1109/ITSC.2013.6728350.

Kamal, M.A.S., Imura, J.I., Hayakawa, T., Ohata, A., Aihara, K., 2015. A vehicle-intersection coordination scheme for smooth flows of traffic without using traffic lights. IEEE Trans. Intell. Transp. Syst. 16 (3), 1136–1147. https://doi.org/10.1109/TITS.2014.2354380. ISSN 15249050.

Kamalanathsharma, R.K., Rakha, H.A., 2013. Multi-stage dynamic programming algorithm for eco-speed control at traffic signalized intersections. In: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). IEEE, pp. 2094–2099.

Lämmer, S., Helbing, D., 2010. Self-stabilizing decentralized signal control of realistic, saturated network traffic. Santa Fe Institute.

Le, T., Kovács, P., Walton, N., Vu, H.L., Andrew, L.L., Hoogendoorn, S.S., 2015. Decentralized signal control for urban road networks. Transp. Res. Part C: Emerg. Technol. 58, 431–450.

Levin, M.W., Boyles, S.D., 2015. Intersection auctions and reservation-based control in dynamic traffic assignment. Transp. Res. Rec.: J. Transp. Res. Board 2497, 35–44.

Levin, M.W., Boyles, S.D., 2016. A multiclass cell transmission model for shared human and autonomous vehicle roads. Transp. Res. Part C: Emerg. Technol. 62, 103–116.

Levin, M.W., Rey, D., 2017. Conflict-point formulation of intersection control for autonomous vehicles. Transp. Res. Part C: Emerg. Technol. 85 (September), 528–547. https://doi.org/10.1016/j.trc.2017.09.025. ISSN 0968090X.

Li, Z., Chitturi, M., Zheng, D., Bill, A., Noyce, D., 2013. Modeling reservation-based autonomous intersection control in Vissim. Transp. Res. Rec.: J. Transp. Res. Board 2381, 81–90.

Lioris, J., Kurzhanskiy, A., Varaiya, P., 2016. Adaptive Max Pressure Control of Network of Signalized Intersections. IFAC-PapersOnLine 49 (22), 19–24. ISSN 24058963. doi:https://doi.org/10.1016/j.ifacol.2016.10.366.

Li, L., Jabari, S.E., 2019. Position weighted backpressure intersection control for urban networks. Transp. Res. Part B: Methodol. 128, 435–461.

Ma, J., Li, X., Zhou, F., Hu, J., Park, B.B., 2017. Parsimonious shooting heuristic for trajectory design of connected automated traffic part ii: computational issues and optimization. Transp. Res. Part B: Methodol. 95, 421–441.

Mirchandani, P., Head, L., 2001. A real-time traffic signal control system: architecture, algorithms, and analysis. Transp. Res. Part C: Emerg. Technol. 9 (6), 415–432.

Mirheli, A., Hajibabai, L., Hajbabaie, A., 2018. Development of a signal-head-free intersection control logic in a fully connected and autonomous vehicle environment. Transp. Res. Part C: Emerg. Technol. 92, 412–425.

Mirheli, A., Tajalli, M., Hajibabai, L., Hajbabaie, A., 2019. A consensus-based distributed trajectory control in a signal-free intersection. Transp. Res. Part C: Emerg. Technol. 100, 161–176.

Priemer, C., Friedrich, B., 2009. A decentralized adaptive traffic signal control using v2i communication data. In: 12th International IEEE Conference on Intelligent Transportation Systems, 2009. ITSC'09. IEEE, pp. 1–6.

Pumir, T., Anderson, L., Triantafyllos, D., Bayen, A.M., 2015. Stability of modified max pressure controller with application to signalized traffic networks. In: Proceedings of the American Control Conference, 2015-July: 1879–1886, 2015. ISSN 07431619. doi:https://doi.org/10.1109/ACC.2015.7171007.

Qian, X., Gregoire, J., F. Moutarde, and A. De La Fortelle. Priority-based coordination of autonomous and legacy vehicles at intersection. In 17th international IEEE conference on intelligent transportation systems (ITSC), pages 1166–1171. IEEE, 2014.

Rey, D., Levin, M.W., 2019. Blue phase: Optimal network traffic control for legacy and autonomous vehicles. Transp. Res. Part B: Methodol. 130, 105–129.

Sims, A.G., Dobinson, K.W., 1980. The sydney coordinated adaptive traffic (scat) system philosophy and benefits. IEEE Trans. Veh. Technol. 29 (2), 130–137.
Smith, M., 1980. A local traffic control policy which automatically maximises the overall travel capacity of an urban road network. Traffic Eng. Control 21 (HS-030 129).
Sun, X., Yin, Y., 2018. A simulation study on max pressure control of signalized intersections. Transp. Res. Rec. 2672 (18), 117–127.
Tassiulas, L., 1992. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. IEEE Trans. Automatic Control 31 (12).
Varaiya, P., 2013. Max pressure control of a network of signalized intersections. Transp. Res. Part C: Emerg. Technol. 36, 177–195.
Wongpiromsarn, T., Uthaicharoenpong, T., Wang, Y., Frazzoli, E., Wang, D., 2012. Distributed traffic signal control for maximum network throughput. In: 2012 15th International IEEE Conference on Intelligent Transportation Systems. IEEE, pp. 588–595.
Wu, W., Zhang, J., Luo, A., Cao, J., 2015. Distributed mutual exclusion algorithms for intersection traffic control. IEEE Trans. Parallel Distrib. Syst. 26 (1), 65–74. https://doi.org/10.1109/TPDS.2013.2297097. ISSN 10459219.
Wuthishuwong, C., Traechtler, A., 2013. Vehicle to infrastructure based safe trajectory planning for autonomous intersection management. In: 2013 13th International Conference on ITS Telecommunications, ITST 2013, pp. 175–180. doi:https://doi.org/10.1109/ITST.2013.6685541.
Xiao, N., Frazzoli, E., Li, Y., Wang, Y., Wang, D., 2014. Pressure releasing policy in traffic signal control with finite queue capacities. In: 2014 IEEE 53rd Annual Conference on Decision and Control (CDC). IEEE, pp. 6492–6497.
Zhang, Y.J., Malikopoulos, A.A., Cassandras, C.G., 2016. Optimal control and coordination of connected and automated vehicles at urban traffic intersections BT – 2016 American Control Conference, ACC 2016, July 6, 2016 - July 8, 2016. 2016-July: 6227–6232, 2016. doi:https://doi.org/10.1109/ACC.2016.7526648.
Zhu, F., Ukkusuri, S.V., 2015. A linear programming formulation for autonomous intersection control within a dynamic traffic assignment and connected vehicle environment. Transp. Res. Part C: Emerg. Technol. 55, 363–378. ISSN 0968090X. doi:https://doi.org/10.1016/j.trc.2015.01.006.