

Article

Digital Twin-Enhanced Adaptive Traffic Signal Framework under Limited Synchronization Conditions

Hong Zhu ¹, Fengmei Sun ^{1,*}, Keshuang Tang ^{1,*}, Hao Wu ¹, Jialong Feng ¹ and Zhixian Tang ²

¹ The Key Laboratory of Road and Traffic Engineering, Ministry of Education, College of Transportation Engineering, Tongji University, Shanghai 201804, China; hongzhu1990@tongji.edu.cn (H.Z.); mas@tongji.edu.cn (H.W.); 2331743@tongji.edu.cn (J.F.)

² Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, Hong Kong 999077, China; zhi-xian.tang@connect.polyu.hk

* Correspondence: fengmeisun@tongji.edu.cn (F.S.); tang@tongji.edu.cn (K.T.)

Abstract: Unmanned traffic signal control is regarded as a sustainable intelligent management methodology. However, it faces the challenge of unpredictable traffic flow due to stochastic arrivals. The digital twin (DT) has emerged as a promising approach to address the challenges of time-varying traffic demand in urban transportation. Previous studies of DT-based adaptive traffic signal control (ATSC) methods all assume ideal synchronization conditions between the DT and the physical twin (PT). It means that the DT can immediately figure out the next neglecting limitation of realistic conditions, i.e., discrepancies between the DT and PT and computational ability. This paper proposes a DT-ATSC framework aimed at reducing the total delay at isolated intersections under limited synchronization conditions. The framework contains two parts: (1) a cell transmission model-based intersection simulation model featuring less computational consumption and the parameter self-calibration mechanism; and (2) scheme searching algorithms that can guide the DT to create optimization-oriented signal timing scheme candidates. Three options are provided for the scheme searching algorithms, i.e., grid search (GS), the genetic algorithm (GA), and Bayesian optimization (BO). A testing platform is created to validate the effectiveness of the proposed DT-ATSC. Experimental results indicate that the proposed DT-ATSC-BO outperforms the DT-ATSC-GA and DT-ATSC-GS. Meanwhile, the average vehicle delay of the DT-ATSC-BO is up to 53% lower than that of the current adaptive signal control method, which indicates that the proposed DT-ATSC has achieved the expected effect. Moreover, compared to the previous related work, the proposed DT-ATSC framework is more likely to be able to be applied in realistic situations because synchronization issues are incorporated in the design of the DT-ATSC by assuming a limited margin time for a decision.

Keywords: sustainable smart traffic control; isolated signalized intersection; adaptive signal control; digital twin; synchronization problem



Citation: Zhu, H.; Sun, F.; Tang, K.; Wu, H.; Feng, J.; Tang, Z. Digital Twin-Enhanced Adaptive Traffic Signal Framework under Limited Synchronization Conditions.

Sustainability **2024**, *16*, 5502. <https://doi.org/10.3390/su16135502>

Academic Editor: Matjaž Šraml

Received: 13 May 2024

Revised: 19 June 2024

Accepted: 23 June 2024

Published: 27 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Evolving demand is a major challenge for traffic signal control. Adaptive traffic signal control (ATSC) has emerged as a promising solution to address this issue by dynamically adjusting signal timings based on real-time traffic conditions. Recently, the technology of real-time and reliable communication between onboard units (OBUs) and RSUs has been realized. Connected vehicles (CVs) within a certain distance to RSUs of an intersection can update their driving state through wireless communication (speed, position, lane index, and turning direction) frequently and accurately. Existing methods process data in an aggregated manner, by which the great potential of such fine-grained and real-time CV data cannot be fully utilized. With the availability of individual continuous trajectory data from CVs, control strategies of ATSC are expected to be improved.

In recent years, the development of digital twin technology has unveiled new possibilities across various domains, particularly in the realm of ATSC [1]. A digital twin is

a virtual representation of a physical object or process, enabling two-way data exchange between the digital twin (DT) and physical twin (PT) in real-time. By establishing a digital counterpart of the actual traffic environment, this approach enables dynamic simulation, deduction, and selection of signal control at signalized intersections. Real-time data are continuously utilized to update the DT so that closed-loop control can be achieved.

Assuming the signalized intersection as the PT, some studies have tried to combine the DT with ATSC and achieved good results [2]. In such an ATSC method, the DT system includes a simulator created based on traditional traffic flow models or directly applies commercial simulation software such as simulation of urban mobility (SUMO 1.5.0). At any time when the controller is working on the next step signal phase plan, a DT will be generated referring to the real-time traffic data obtained from sensors, CVs, and traffic signals. Then, the future performance of the intersection will be deduced repeatedly assuming that the signalized intersection in the DT applies different signal phase timing schemes. Such deduction work terminates until the performance of the generated scheme converges into a satisfying level. In this way, the algorithm could optimize traffic signal operations adaptively.

However, the current DT-based ATSC methods are tested under ideal synchronization conditions between the DT and PT. They do not consider discrepancies between the DT and PT and computational resource constraints. Therefore, the current DT-based ATSC methods may not achieve the expected performance once they are applied in real-world systems. The specific problem is dissected in the following two aspects:

- Discrepancies between the DT and PT: The DT serves as an approximation and imitation of the PT; there should be inevitable differences between them because their operating mechanisms are different. Minimizing the discrepancies between the DT and PT using real-world data is necessary because it directly determines the effectiveness of the solution validated in the DT;
- Computational limitations: Current DT-based ATSC methods assume unlimited computational resources can be called upon for the deduction work of the DT. In their methods, the calculation will terminate only when the performance indicator in the DT converges, then an optimal solution will be found. However, in the real world, the computation ability of RSUs is limited, which means that the total number of signal timing schemes to be tested is not enough. Therefore, how to find a better solution through testing a limited number of schemes will be a critical problem to be solved.

Considering the above two issues, this paper proposes a new DT-ATSC framework that is able to accommodate the synchronization problem. In particular, the framework includes a cell transmission model (CTM)-based intersection simulation model featuring less computational consumption and a scheme searching algorithm that guides the DT to create optimization-oriented signal timing schemes to be tested. Overall, the key contribution, which distinguishes this study from previous works, can be summarized as follows:

- A DT based on a modified CTM with a parameter self-calibration mechanism is introduced based on real-time CV data, aiming to mitigate the discrepancies between the DT and PT;
- Three scheme searching algorithms for the DT-ATSC are created and compared, including grid search and two other heuristics algorithms, i.e., the genetic algorithm and Bayesian optimization.

The rest of this paper is organized as follows. Related works are summarized in Section 2. Section 3 mainly introduces the whole structure of the framework and the algorithms, i.e., the CTM-based DT and scheme searching algorithm. The performance of the proposed DT-ATSC is interpreted in Section 4. Finally, Section 5 summarizes the conclusion, current limitations, and future works.

2. Related Work

Addressing inadequate control at signalized intersections resulting from evolving demands has long been a priority for researchers and practitioners. ATSC can dynamically modify the signal timing in response to real-time traffic flow, thereby improving traffic operation efficiency. Over the past decades, a large number of scholars have dedicated their endeavors to advancing ATSC methods [3].

Conventional ATSC methods primarily rely on basic traffic data from inductive loop detectors and connected vehicles (CVs). Inductive loop detectors are usually positioned near the stop line [4], upstream of the stop line [5,6], or at upstream intersections [7,8]. Due to its mature technological development, it is widely applied in existing adaptive signal control methods. Classic systems such as SCATS [4] and SCOOT [8] employ pattern matching or real-time prediction of intersection traffic conditions, followed by dynamic plan selection or real-time plan generation to optimize signal control parameters. Larry predicted downstream vehicle arrivals based on signal timings, queues, and free-flow speeds from upstream intersections [7]. However, these systems have limitations, as the fundamental traffic data derived from traditional detectors may not provide the required accuracy and completeness to achieve theoretical performance, and low-dimensional analytical models are required to make assumptions and simplifications, resulting in a loss of precision [9].

In recent years, vehicle trajectory data from connected vehicle (CV) environments have been utilized to refine ATSC performance [9–11]. Predictive traffic control algorithms leveraging robust trajectory data have been developed, optimizing signal timings based on future traffic conditions [10]. Aljaafreh et al. explored multi-agent-based control methods for an integrated network of ATSCs within a CV communication environment [11]. Additionally, Bhawe et al. developed real-time adaptive signal phase allocation algorithms using CV data to optimize phase sequence and duration [12].

Digital twin (DT) technology has emerged as a promising approach to enhance traffic signal control, with its potential in modeling road traffic and enabling autonomous vehicles as a crucial component of smart mobility [13–15]. Based on the evolution of the concept of the DT in transportation, Aslani et al. developed DT simulation models capable of providing traffic performance measures in real-time [16]. DT-assisted real-time traffic data prediction methods have been explored for analyzing traffic flow and velocity data transmitted via 5G networks [17]. Successful implementations of DTs on local roads have accurately measured road characteristics and assisted in road planning [18]. Additionally, principles of developing DTs for active vehicle safety systems, such as the braking system [19] and automated driving system [20], have been studied. Furthermore, several studies have proposed techniques and methods for evaluating the reliability and safety of digital twin technology based on its application at various stages of the vehicle development lifecycle [21,22].

However, despite the transformative potential of DT technology, there remains a lack of sufficient evidence supporting its direct application in traffic signal control. Thus, this study aims to address this gap by proposing a comprehensive DT-ATSC framework.

Meanwhile, it is also necessary to explain the relationship between simulation and digital twins before the main part of this paper for distinguishing these two terms. According to the definition given in Wikipedia, this study defines traffic simulation as a tool to approximate and imitate traffic evolution, whereas the digital twin is defined as the technical process including collecting real-world data, calibrating the simulation platform, obtaining control scheme suggestions through running the simulation, and implementing the scheme in the real world. Therefore, to some extent, the simulation can be regarded as the core part of the digital twin.

3. Methods

3.1. Framework of the Digital Twin-Based Adaptive Traffic Signal Control (DT-ATSC)

As in Figure 1, a framework of the DT-ATSC has been built for isolated intersections. The whole framework includes two parts, i.e., the modified CTM-based DT and the scheme

searching algorithm. Unlike in previous studies, the framework in our study has been modified to consider computation time and scheme limitations. As illustrated in Figure 2, the deduction work in the DT of each scheme takes a certain amount of time. After experiments based on the average computational ability of the current RSU, the time length for each scheme is given as 0.01 s. It is obvious that the more the scheme is tested, the more computational time is needed in the DT, and the more margin time should be reserved before the next cycle. However, longer margin time leads to greater discrepancies between the PT and DT due to the difference in external demand input and the internal traffic flow evolution mechanism. Therefore, the maximal number of schemes to be tested is limited. A “2 s margin time” is assumed to be acceptable considering the synchronization problem discussed above, which indicates the maximal number of schemes to be tested is 200 for each decision.

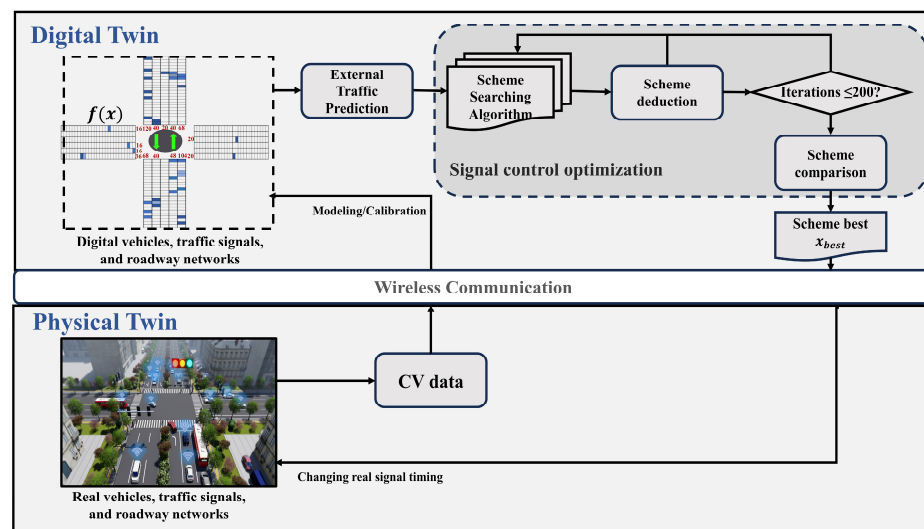


Figure 1. Framework of the DT-ATSC.

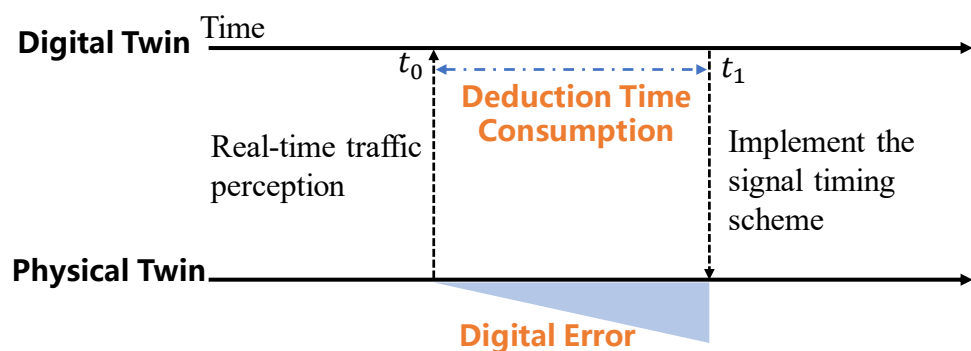


Figure 2. Discrepancies between the DT and PT.

Therefore, the workflow of the DT-ATSC was consequently modified. At 2 s prior to the end of every cycle, the DT will create a simulation platform according to the obtained CV data. Based on the simulation platform, the algorithm optimizes the control performance of the signalized intersection by repeatedly testing different signal timing schemes generated by the scheme searching algorithm. Within the 200 schemes tested in the DT, the scheme with the best future performance among the ones tested will be selected and implemented for the next cycle.

3.2. DT Based on Modified CTM

3.2.1. Simulation Model for Isolated Intersection Based on Modified CTM

In most of the previous studies related to DT-ATSC [2], a commercial simulation software SUMO 1.5.0 is introduced to play the role of the simulation platform in the DT. To balance the simulation accuracy and computational burden of the DT, this study applies CTM as the core simulation model for the DT.

The CTM is a meso-level simulation tool that effectively replicates the hydrodynamic characteristics of interrupted traffic flow on each lane at intersections while maintaining lower computational demands. This model aligns well with the synchronization needs of our framework due to these advantages. CTM predicts the spread of traffic by calculating the boundary fluxes and the density across a defined number of cells at various time intervals. These fluxes between cells are determined through specific sending (upstream cell) and receiving (downstream cell) functions outlined in Equations (1)–(4) of the modified CTM. However, the discharge flow as modeled by the standard CTM does not realistically depict actual traffic conditions. To address this, Srivastava et al. refined the sending function, as detailed in Equation (5) [19]. This enhancement enables the model to more accurately reflect the discharge characteristics of queues. As a result, the revised CTM is particularly utilized for cells situated immediately upstream of the stop lines.

$$k_j(t+\Delta t) = k_j(t) + \frac{\Delta t}{L} (\Phi_j(t) - \Phi_{j+1}(t)) \quad (1)$$

$$\Phi_j(t) = \min\{S_{j-1}(t), R_j(t)\} \quad (2)$$

$$S_i(t) = S(k_i(t)) = \min\{v_f k_i(t), q_c\} \quad (3)$$

$$R_j(t) = R(k_j(t)) = \min\{q_c, w(k_{jam} - k_j(t))\} \quad (4)$$

$$S'_j(t) = S'(k_j(t)) = \min\{v_f k_j(t), c^*(k_{jam}^* - k_j(t))\} \quad (5)$$

where j is the cell number, $k_j(t)$ is the density of cell j at time t (veh/m), Δt is the time step, $\Phi_j(t)$ is the flux from cell $j-1$ to cell j (veh/h), v_f is the free speed (m/s), $S_j(t)$ is the sending value of cell j at time t (veh/h), $R_j(t)$ is the receiving value of cell j at time t (veh/h), $S'_j(t)$ is the modified sending value of cell j at time t (veh/h), q_c is the cell capacity (veh/h), w is the shockwave speed (m/s), c^* is the slope of modified sending curve (m/s), k_{jam}^* is the projected jam density for the modified sending curve (veh/m), and k_c is the critical density, k_{jam} is the jam density (veh/m).

In this study, the conventional and modified CTM are expanded by analyzing the movement of vehicles from cell j to cell $j-1$ (denoted as y_j) and the vehicle occupancy (n_j) at multiple intermediate points over varying time steps, as illustrated in Figure 3. Two constants are defined for each cell: Vehicle Holding Capacity (N_j): This represents the maximum number of vehicles that can occupy cell j at any given time t . N_j is determined by multiplying the cell's length by its jam density; Cell Capacity (Q_j): This defines the maximum number of vehicles that can enter cell j during the time interval from t to $t + \Delta t$. These constants change over time, enabling the model to accurately simulate temporary traffic incidents, as described in Equations (6)–(13).

$$N_j = k_{jam} L \quad (6)$$

$$n_j = k_j(t) L \quad (7)$$

$$Q_j = q_c / 3600 \quad (8)$$

$$y_{j+1}(t) = \min\{n_j(t), Q_{j+1}, w/v_f [N_{j+1} - n_{j+1}(t)]\} \quad (9)$$

$$n_{j+1}(t + \Delta t) = n_j(t) + y_j(t) - y_{j+1}(t) \quad (10)$$

$$N_j^* = k_{jam}^* n_j L \quad (11)$$

$$y_{j+1}(t) = \min\{n_j(t), Q_{j+1}, w/v_f[N_{j+1} - n_{j+1}(t)], c^*/v_f[N_{j+1}^* - n_{j+1}(t)]\} \quad (12)$$

$$n_{j+1}(t + \Delta t) = n_j(t) + y_j(t) - y_{j+1}(t) \quad (13)$$

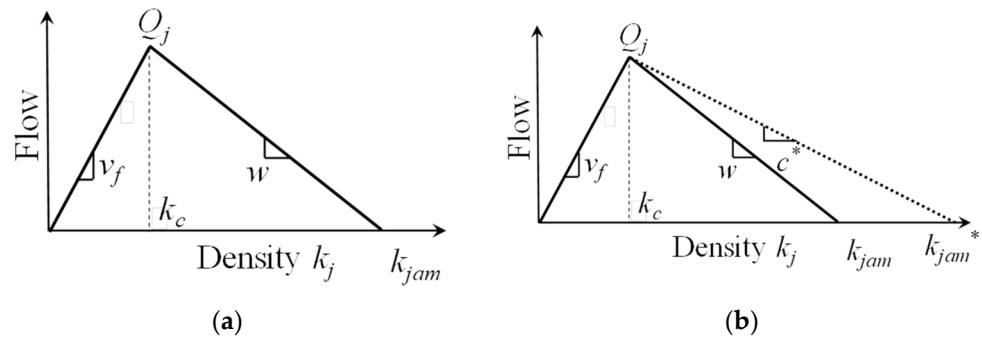


Figure 3. The generalized CTM; (a) traditional CTM; (b) modified CTM.

Five types of functional cells are created to compose the simulation platform, i.e., normal cell, source cell, ending cell, merging cell, and diverging cell. The iterative formulae for each type are shown in Equations (14)–(18). Any types of intersection (different lane number and lane function) can be modelled by simply stacking cells as shown by the example illustrated in Figure 4.

$$\begin{cases} y_j(t) = \min\{S_{j-1}(t), R_j(t)\} \\ y_{j+1}(t) = \min\{S_j(t), R_{j+1}(t)\} \end{cases} \quad j \in \text{Normal} \quad (14)$$

$$\begin{cases} y_j(t) = D_j(t) \\ y_{j+1}(t) = \min\{S_j(t), R_{j+1}(t)\} \end{cases} \quad j \in \text{Source} \quad (15)$$

$$\begin{cases} y_j(t) = \min\{S_{j-1}(t), R_j(t)\} \\ y_{j+1}(t) = S_j(t) \end{cases} \quad j \in \text{Ending} \quad (16)$$

$$\begin{cases} y_{i,j}(t) = \min\left\{S_i(t), \frac{S_i(t)}{\sum_{M(j)} S_i(t)} R_j(t)\right\} \\ y_{j+1}(t) = \min\{S_j(t), R_{j+1}(t)\} \end{cases} \quad i \in M(j), j \in \text{Merging} \quad (17)$$

$$\begin{cases} y_j(t) = \min\{S_{j-1}(t), R_j(t)\} \\ y_{j,i}(t) = \min\{\beta_{j,i} S_j(t), R_i(t)\} \end{cases} \quad i \in D(j), j \in \text{Diverging} \quad (18)$$

where $D_j(t)$ is the traffic demand generated at the source cell at time t , $M(j)$ is the set of upstream cells merging into the cell j , $\beta_{j,i}$ denotes the diverging proportion of traffic from j to i , and $D(j)$ is the set of downstream cells that cell j diverges into.

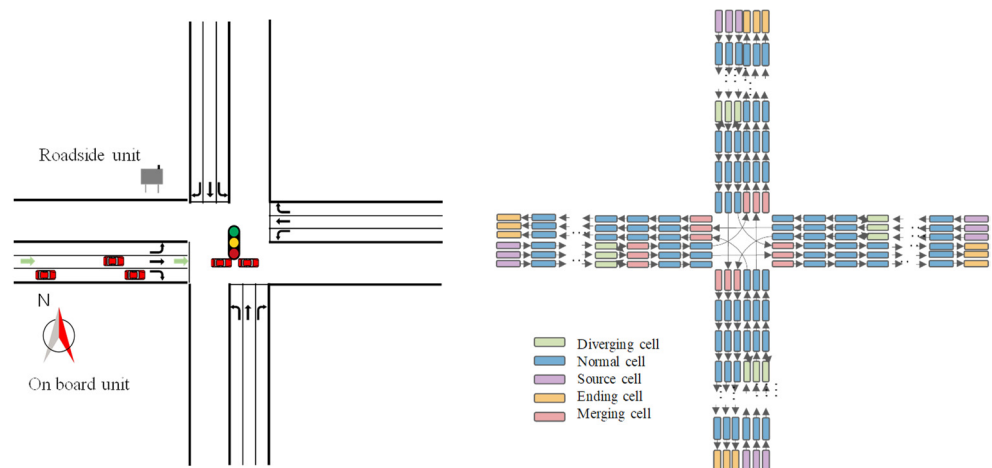


Figure 4. Cell types in cell transmission models.

Any existing traffic in a cell that cannot leave in the next steps incurs a delay. In this way, the total delay (TD) is introduced to evaluate the performance of each signal scheme as in Equation (19).

$$TD = \sum_t \sum_j ((n_j(t) - \Phi_{j+1}(t) \times \Delta t) \times \Delta t) \quad (19)$$

3.2.2. Updating Internal Traffic State and Predicting the External Traffic Input

Firstly, approaches within the communication distance (200 m radius area) are sliced into the virtual grid form referring to the layout of the CTM model in the DT as in Figure 4. At every decision time, CVs in this area will report their position (geometric center point) and turning direction to the RSU. Then, the state parameters of each cell in CTM model are updated:

- (1) the DT counts the number of vehicles in each grid as n_j ;
- (2) the $\beta_{j,i}$ of each diverging cell will be updated on the basis of turning direction information.

There is no doubt that traffic input from the outside should be considered for the testing of traffic signal schemes. Therefore, many traffic input prediction approaches are worth being considered. For instance, by accessing the departure data from upstream intersections, the arrival flow can be accurately mastered. However, this part is not the highlight of this article. In this study, we simply assume that the traffic input will not drastically change over two consecutive cycles. Therefore, the RSU records the profile of arrival flow in each cycle. The assumed traffic input profile for the next period will simply copy the one recorded during the last cycle.

3.2.3. Parameter Self-Calibration

When the DT creates a simulation platform, the parameters of each cell will be updated before the deduction work starts according to the data recorded during the last cycle. In this study, two types of CTM are applied, i.e., the traditional CTM and the modified CTM. Their parameters are calibrated by following procedures: (1) the v_f and k_{jam} of the traditional CTM and modified CTM are kept invariant; (2) the SFR_l and SLT_l of each lane l should be calculated based on the CVs' passing time at each stop line through Equations (20)–(22); (3) q_c , w , and c^* of cells applying the modified CTM should be adjusted through Equations (23)–(25). The mechanism of the self-calibration is illustrated in Figure 5.

$$h_s = \frac{T_q - T_4}{n_q - 4} \quad (20)$$

$$SFR = \frac{3600}{h_s} \quad (21)$$

$$SLT = \sum_{i=1}^4 (h_i - h_s) \quad (22)$$

$$q_c = SFR \quad (23)$$

$$w = \frac{v_f q_c}{v_f k_{jam} - q_c} \quad (24)$$

$$c^* = \frac{SLT \times w^2}{SLT \times w + L} \quad (25)$$

where h_s is the average saturation headway (s), T_q is the discharge time of the last queued vehicle in the platoon (s), T_4 is the discharge time of the 4th queued vehicle (s), n_q is the total number of queued vehicles observed, and h_n defines the queue discharge headway of the i^{th} vehicle (s).

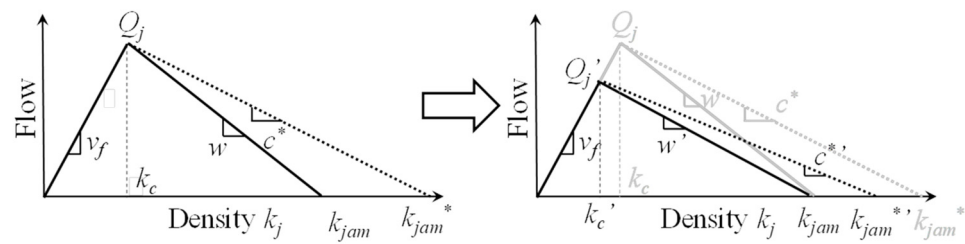


Figure 5. Calibration of modified CTM.

3.2.4. Scheme Searching Algorithms

As in Figure 6, when the structure and order of the phase is fixed, different signal control schemes are generated by adjusting the length of the green time (g_1, g_2, g_3, g_4) in each phase respectively. ATSC is time-sensitive and requires the decision-maker to find a better control plan from the given scheme. In this study, the maximal number of schemes to be tested for each decision is assumed as 200. We conducted extensive research on several MEC devices available in the Chinese market and found significant variability in their computational capabilities. Consequently, the assumed simulation time of 0.01 s per signal control scheme in this paper cannot be confirmed as realistic. However, it is anticipated that in the near future, a guiding standard will be established, specifying the configuration requirements for roadside MEC devices. At that point, it will be possible to provide a more concrete answer to this question.

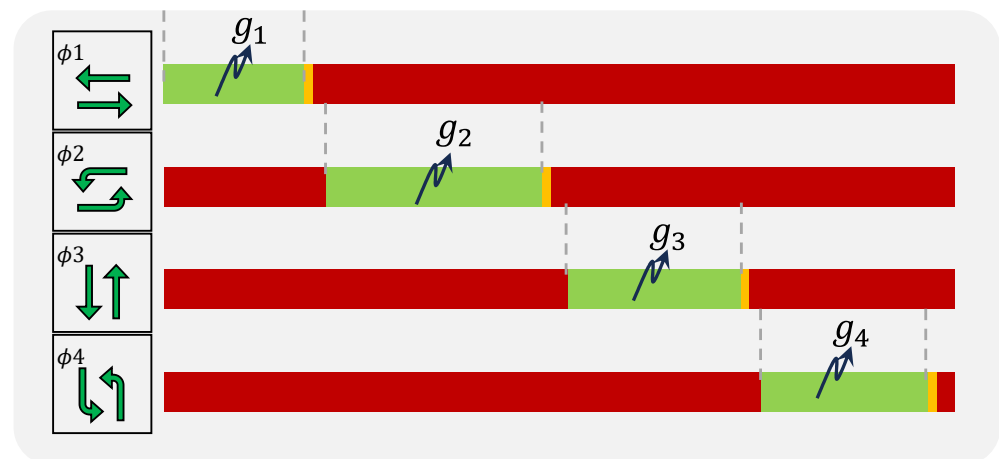


Figure 6. Signal phase diagram.

Under limited conditions, tests of signal control schemes should not be carried out at high dimensions, so (g_1, g_2, g_3, g_4) is taken as the decision variable of signal control optimization in this paper. The goodness of the schemes provided for the DT is a necessary and controllable issue in this framework. Such a problem can be represented in Equation (26). For letting the provided X be close to optimal x , three scheme searching algorithms for the DT-ATSC are created in this study, i.e., the grid search (GS), and the other two heuristics algorithms, i.e., the genetic algorithm (GA) and Bayesian optimization (BO). The overall procedure of those algorithms can be summarized into the following two steps: (1) Generate an initial scheme; (2) Search the traffic signal scheme that is close to the optimal solution according to a given rule starting from the initial scheme. Their details are explained by pseudo-code as Appendix A.

$$\begin{aligned}
 & \min TD(x) \\
 & s.t. \\
 & x_i \in X = \{x_1, x_2, \dots, x_i, \dots, x_n\} \\
 & \sum_{i=1}^n time(x_i) \leq T_{margin}
 \end{aligned} \tag{26}$$

4. Experiment

4.1. Experimental Platform

In this study, SUMO 1.11.0, C++ 20, and Python 3.11 were integrated to build the DT-ATSC experimental platform as in Figure 7. The function of each component includes: (1) SUMO 1.11.0 was used as the data source and operation object. In every time step, the information of all vehicles was obtained and relayed via the traffic control interface function (TraCI) from SUMO to Python; (2) The CTM was programmed in C++. Pybind11 library was used to package it as a Python extension package; (3) Python is a middle layer that implements data conversion, preprocessing, and scheme searching algorithms. In every time step, the signal control command for the intersection was calculated and sent back to the SUMO through TraCI.

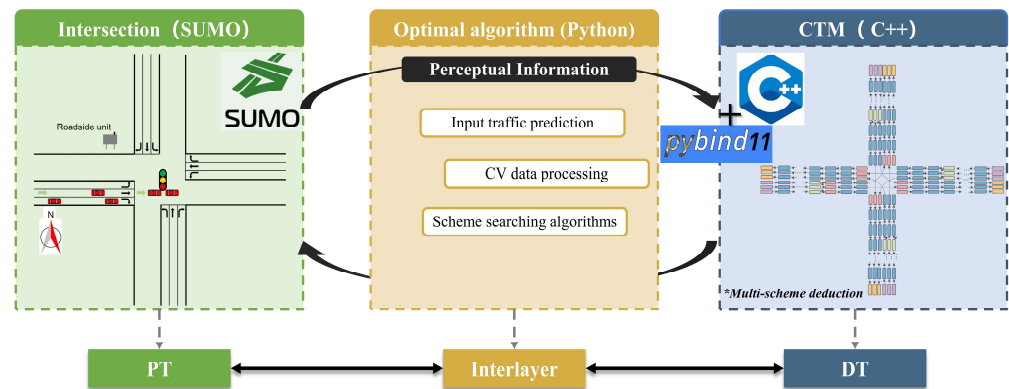


Figure 7. DT-ATSC experimental platform.

To promote academic exchanges, the code of the platform can be accessed from the following links: <https://github.com/Eightina/Digital-Twin-CTM> (accessed on 23 January 2024). A study site of an isolated intersection was created, as illustrated in Figure 8. The traffic volume and vehicular parameters were directly set as the default one in SUMO.

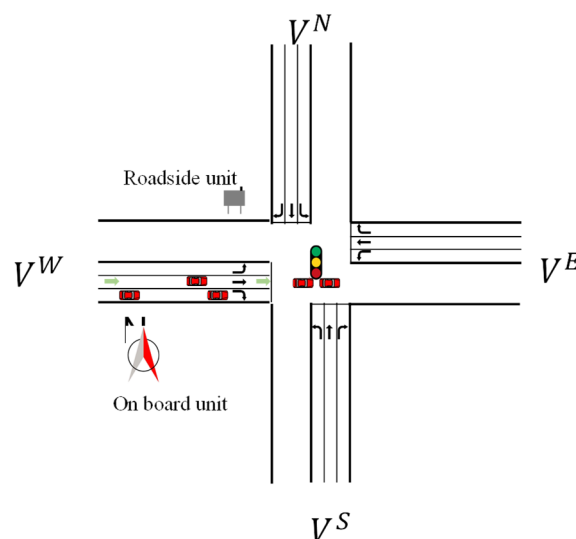


Figure 8. Study site of an individual intersection.

4.2. Discussion

4.2.1. Comparison of Three Scheme Searching Algorithms

The performance of DT-ATSC using three scheme searching algorithms, i.e., grid search (GS), the genetic algorithm (GA), and Bayesian optimization (BO) was tested in two different demand scenarios. For the two scenarios, the traffic volume in the opposite direction was assumed to be at the same level, i.e., $V^N = V^S$ and $V^W = V^E$. As in Figure 9a,b, scenario 1 was designed for testing the adaptive ability of control algorithms to changes in demand level, while scenario 2 was created for comparing their performances under changing demand splits. It is noteworthy that in both scenarios the ratio of left-turn/straight/right-turn was always equal to 1:7:2.

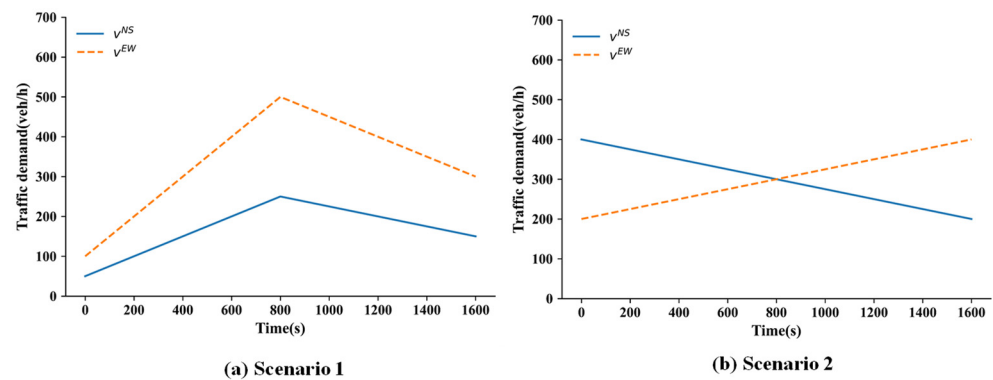


Figure 9. Tested scenarios with different demands.

This paper introduces the cumulative total waiting time (CTWT) to measure the control performance as in Equation (27).

$$CTWT_t = \sum_{i=0}^t \sum_{j=1}^n WT_{ij} \quad (27)$$

where n is the total number of vehicles that are within the intersection at time t , WT_{ij} is a Boolean value indicating whether vehicle i is waiting at time j , and WT_{ij} equals 1 if the speed of the vehicle V_{ij} is less than the threshold, else it equals 0.

Figure 10 illustrates the result of scenario 1 during a 30 min period (including 24 cycles). From Figure 10a, it can be observed that among the three scheme searching algorithms, the trend line of DT-ATSC-BO is consistently lower than those of DT-ATSC-GA and DT-ATSC-GS. This indicates that the DT-ATSC-BO algorithm achieves superior control performance compared to the other two algorithms when the demand level at the intersection changes. Notably, the cumulative total weighted tardiness (CTWT) difference between DTs-ATSC-BO and DT-ATSC-GS reaches approximately 7000 s at the 850 s mark. The same trend is evident in Figure 10b, which further demonstrates that DT-ATSC-BO has a better adaptive ability to demand split changes than both DT-ATSC-GA and DT-ATSC-GS. As the simulation time progresses, the performance gap between DT-ATSC-BO and DT-ATSC-GS becomes more pronounced. By the end of the simulation at 1700 s, the CTWT difference between DT-ATSC-BO and DT-ATSC-GS increases to 10,000 s. These results suggest that, with a limited number of searching schemes (up to 200 schemes assumed in this study), applying Bayesian optimization (BO) enables the decision tree (DT) to identify more effective signal control schemes compared to using genetic algorithms (GAs) and grid search (GS). The superior performance of DT-ATSC-BO highlights its potential for optimizing traffic signal control under varying demand conditions, ensuring better adaptability and efficiency in managing intersection traffic.

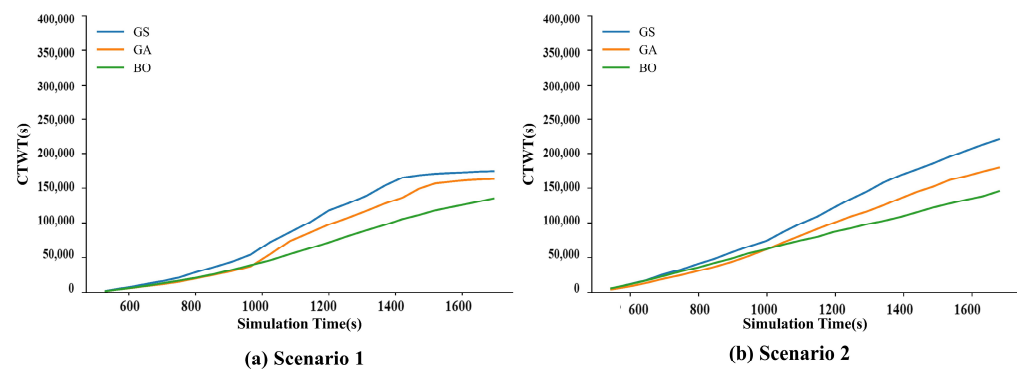


Figure 10. Results of three scheme searching algorithms.

In more detail, the observed trends in Figure 10a,b reflect the ability of the DT-ATSC-BO algorithm to respond dynamically to real-time changes in traffic demand, minimizing delays more effectively than the other methods. The DT-ATSC-GA, while better than DT-ATSC-GS, still lags behind DT-ATSC-BO, indicating that the evolutionary approach of GA, although powerful, does not match the precision of the BO method within the same timeframe. The increasing divergence between the DT-ATSC-BO and the other methods as simulation time advances suggests that BO is particularly advantageous in longer operational periods, where its optimization capability compounds benefits over time. This is crucial for real-world applications where traffic conditions can fluctuate widely and unpredictably, demanding robust and flexible optimization techniques. Furthermore, the results underscore the practicality of Bayesian optimization in scenarios with limited computational resources, as it efficiently narrows down the optimal solutions from a vast search space. This efficiency is vital for real-time traffic management systems that require swift and accurate decision-making to enhance overall traffic flow and reduce congestion.

In conclusion, the empirical evidence from Figure 10 clearly demonstrates that DT-ATSC-BO outperforms DT-ATSC-GA and DT-ATSC-GS in managing traffic signal control, especially under varying demand levels. The Bayesian optimization method not only provides better control performance but also exhibits superior adaptability to changing traffic conditions, making it a valuable tool for modern intelligent transportation systems.

4.2.2. Comparison with Other Adaptive Signal Control Methods

With respect to the performance of DT-ATSC-BO (better than the other two) and DT-ATSC-GS (worse than the other two), they were selected to be further tested in more scenarios compared to the Adaptive Webster proposed by [23]. The signal control mode of Adaptive Webster will generate a signal timing scheme for the next cycle through Webster theory at the end of every cycle according to the aggregated demand information gathered during the current cycle.

Notably, it was originally intended to compare the proposed method with existing DT-ATSC methods; however, there are very few such methods available. To the best of our current knowledge, only one paper accurately falls into the DT-ATSC category. Unfortunately, that study does not consider the computational cost of the approach. In their methodology, tens of thousands of scenarios are simulated instantaneously before generating a solution, which is clearly unrealistic under real-world conditions. Additionally, their method employs SUMO as the platform for both the DT and PT, implying no differentiation between the DT and PT, which is also unrealistic. Consequently, this paper does not use DT-ATSC methods for comparison but instead selects the Adaptive Webster method as the benchmark.

As in Figure 11, it is important to clarify that the experimental scenarios in this paper were designed based on data collected from real-world situations. The scale of traffic flow rate and the range of variation in distribution ratios were referenced from actual data (intersections in Shanghai). Additionally, to highlight the technical characteristics

of the method, two scenarios were specifically considered: traffic flow fluctuation and distribution ratio fluctuation. Furthermore, four different levels of arrival traffic fluctuation were assumed. Through this sensitivity analysis experiment, it can be demonstrated that the proposed method is robust and superior to other methods under different traffic settings. Group 1 refers to scenarios with changing demand levels (four different peak volumes) and group 2 refers to scenarios with changing demand splits (four different changing rates).

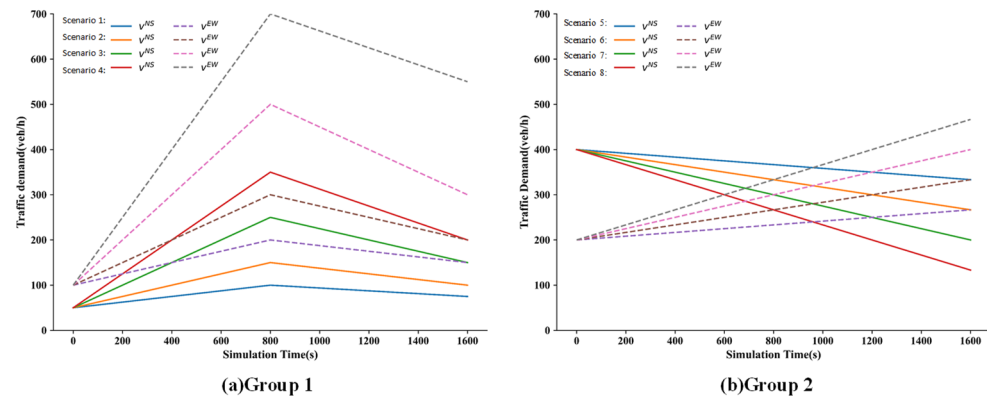


Figure 11. Groups of tested scenarios.

The CTWT was introduced to measure the control performance as well. From Figures 12 and 13, it can be observed that the trend line of the proposed DT-ATSC-BO is consistently lower than those of DT-ATSC-GS and the Adaptive Webster. This indicates that both DT-ATSC methods maintain superior control performance compared to the Adaptive Webster. The results further reveal that by using the solution generated by the Webster method as the initial solution, the controller can find better solutions through iterative searching and optimization, utilizing grid search or Bayesian optimization.

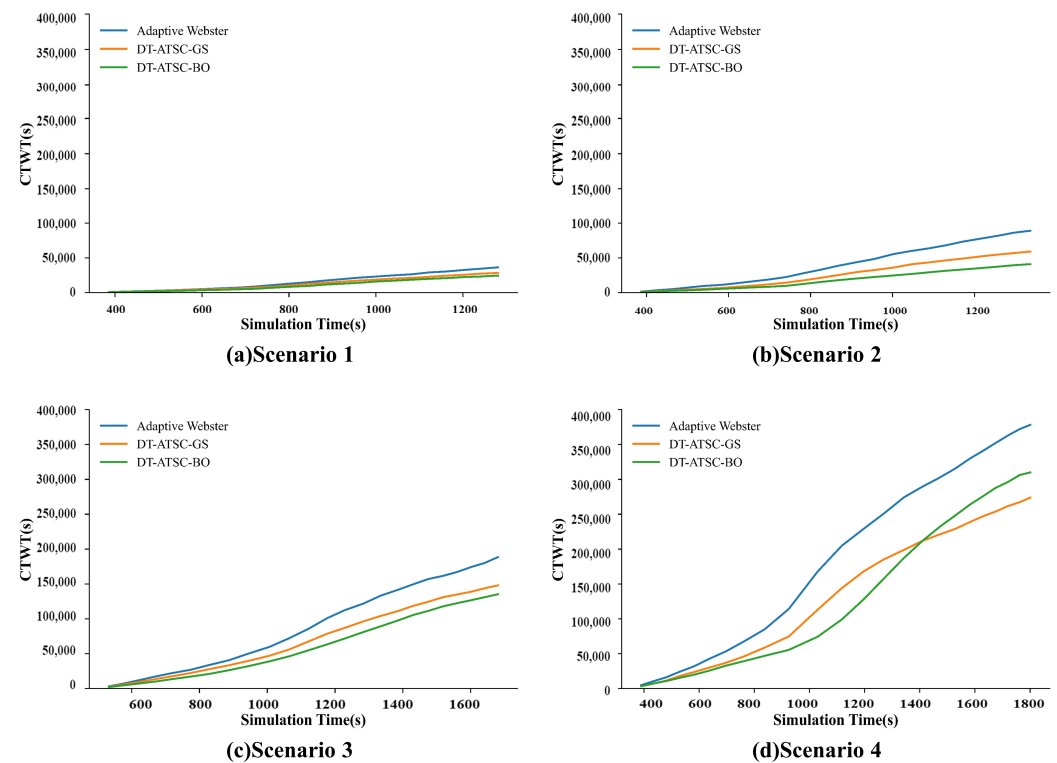


Figure 12. Results of group 1.

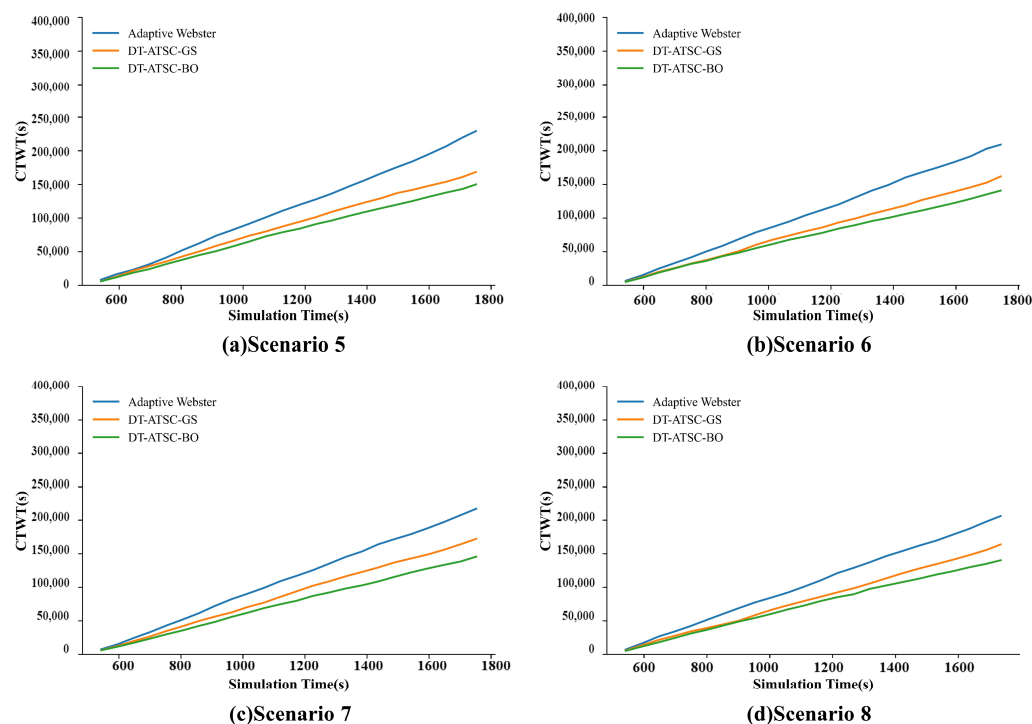


Figure 13. Results of group 2.

A critical observation is made when the peak volume of the group becomes $V_N = V_S = 350$ veh/h and $V_W = V_E = 700$ veh/h, where the trend lines of DT-ATSC-BO and DT-ATSC-GS intersect at around 850 s. Upon examining the simulation records, it was found that DT-ATSC-BO may generate inappropriate signal control schemes when traffic congestion, such as secondary queuing, occurs in certain approaches. This highlights a potential area for further refinement in the DT-ATSC-BO method to enhance its robustness under varying traffic conditions.

Additionally, the average delay and control effectiveness of the three ATSC methods are compared across different scenarios in Table 1. In Scenario 1, DT-ATSC-BO achieves an average delay of 8.9 s, which is 16.6% lower than the Adaptive Webster method at 10.6 s. DT-ATSC-GS shows a slight improvement with an average delay of 9.0 s, 1.2% lower than Adaptive Webster. In Scenario 2, DT-ATSC-BO significantly outperforms, with an average delay of 9.2 s, 53.7% lower than Adaptive Webster's 19.8 s. DT-ATSC-GS also shows substantial improvement with an average delay of 13.2 s, 30.4% lower than Adaptive Webster. Scenario 3 results indicate DT-ATSC-BO with an average delay of 32.9 s, 22.8% lower than Adaptive Webster's 42.6 s, while DT-ATSC-GS achieves 33.7 s, 2.5% lower than Adaptive Webster. In Scenario 4, DT-ATSC-GS performs the best with an average delay of 44.8 s, 23.4% lower than Adaptive Webster's 58.4 s. However, DT-ATSC-BO shows a delay of 49.8 s, 10.1% higher than DT-ATSC-GS.

The detailed analysis for Group 2, illustrated in Figure 13a–d, reinforces the superior performance of DT-ATSC-BO and DT-ATSC-GS compared to the Adaptive Webster. Interestingly, as the changing rates of V_N (VS) and V_W (VE) increase, the performance gaps between DT-ATSC-BO, DT-ATSC-GS, and the Adaptive Webster tend to diminish. However, this trend is relatively minor, suggesting that the changing rate of demand splits does not significantly impact the performance of adaptive control methods.

In conclusion, the proposed DT-ATSC methods, especially DT-ATSC-BO, exhibit remarkable performance in reducing average delay and enhancing control effectiveness compared to the traditional Adaptive Webster method. This is evident across various traffic scenarios, validating the robustness and efficiency of the DT-ATSC approach. These findings underscore the potential of DT-ATSC methods in achieving more efficient and responsive traffic signal control, contributing to smoother traffic flow and reduced congestion in urban

environments. Further research and development could focus on refining these methods to handle more complex and unpredictable traffic conditions, thereby enhancing their practical applicability and reliability.

Table 1. Average delay evaluation results.

Scenarios	ATSC Method	Average Delay(s)	Ratio Compare to the Lowest One(-)
Scenario 1	Adaptive Webster	10.6	16.6%
	DT-ATSC-GS	9.0	1.2%
	DT-ATSC-BO	8.9	-
Scenario 2	Adaptive Webster	19.8	53.7%
	DT-ATSC-GS	13.2	30.4%
	DT-ATSC-BO	9.2	-
Scenario 3	Adaptive Webster	42.6	22.8%
	DT-ATSC-GS	33.7	2.5%
	DT-ATSC-BO	32.9	-
Scenario 4	Adaptive Webster	58.4	23.4%
	DT-ATSC-GS	44.8	-
	DT-ATSC-BO	49.8	10.1%

5. Conclusions

This paper proposes a DT-ATSC framework to address the signal control problem at signalized intersections under changing traffic demand. This study assumed that the traffic flow consists of 100% CVs and vehicular driving state data within the control area of the intersection will be transmitted to RSU through wireless communication between the CV and RSU every second. At 2 s prior to the end of every cycle, the algorithm optimized the control performance of the signalized intersection by repeatedly generating and testing different scenarios with different signal timing schemes in the DT. Then, the scheme with the best future performance among the ones tested are selected for the next cycle. The proposed method aims to optimize by reducing delays, and compared to traditional methods, it can reduce delays by up to 53.7%. Lower vehicle delays mean fewer stops and starts, which in turn contributes to energy savings and emission reductions, supporting the sustainable development of cities.

Unlike previous works, the DT-ATSC framework consists of two key components, i.e., the modified CTM-based DT which highlights the self-calibration mechanism and the scheme searching algorithm. The DT is constructed on the basis of the modified CTM which is a meso-level simulation model that reproduces hydrodynamic properties of interrupted traffic flow at signalized intersections resulting in low computational consumption. To ensure synchronization between the DT and the PT of the signalized intersection, we introduced a parameter self-correction mechanism that calibrates the parameters of the modified CTM model based on real-world CV data. For facilitating the DT-ATSC to find a better signal timing scheme, three scheme searching algorithms were employed and tested to approximate the global optimal solution considering computational limitations (the number of schemes is assumed up to 200).

Through numerical studies, the performance of scheme searching algorithms (GS, GA, and BO) were compared. The DT-ATSC-BO consistently demonstrated superior performance and the average vehicle delay of DT-ATSC-BO under 750 veh/h demand level was 32.9 s lower than the other two (DT-ATSC-GS and DT-ATSC-GA). Furthermore, experiments for comparing the performance of three adaptive signal control methods (DT-ATSC-GS, DT-ATSC-BO, and Adaptive Webster) across various scenarios were conducted. The DT-ATSC-BO and DT-ATSC-GS consistently outperform the Adaptive Webster. The delay of

DT-ATSC-BO is up to 53% lower than Adaptive Webster. Overall, it can be concluded that the proposed DT-ATSC has achieved the expected effect. Moreover, compared to the previous related work, the proposed DTATSC framework is more likely to be able to be applied in realistic situations since synchronization issues are considered.

Although this article addresses the technical challenges of applying digital twin technology to traffic systems, applying the DT-ATSC method to real-world scenarios presents the following issues that need to be addressed: (1) The uncertainties present in real-world scenarios are key factors limiting the effectiveness of solutions derived using the CTM model, which is a deterministic model; (2) In real-world scenarios, there is a certain delay in the collection of traffic operation data at intersections by sensing devices, which significantly constrains the real-time effectiveness of the signal control schemes generated by DT-ATSC. Moreover, the system's behavior and characteristics are described through rigorous mathematical models and logical reasoning, a process known as formal methods [20]. By using formal methods, errors in the system can be significantly reduced, thereby enhancing the system's sustainability.

The proposed framework can be further improved considering the following two aspects. First, the DT-ATSC can be further improved by incorporating an advanced external traffic input prediction. For instance, by accessing the departure data from upstream intersection [24–26], the arrival flow can be accurately mastered. Second, this study assumes traffic flow with 100% CV, the method should be further modified to accommodate situations of traffic flow with different CV penetration rates. Third, the proposed DT-ATSC is designed and verified at a single intersection. We also plan to further extend the idea to multi-intersection scenarios. Meanwhile, future work will also focus on incorporating feedback mechanisms and adaptive learning algorithms to enhance the adaptive capabilities of the DT-ATSC framework. For instance, the parameter self-calibration function of the digital twin model can adjust model parameters in real-time based on perceived parameters, ensuring that the digital twin platform consistently aligns with the operational dynamics of the actual intersection. This will improve system performance and ensure long-term sustainability and resilience across various operational scenarios.

Author Contributions: Conceptualization, H.Z. and K.T.; methodology, F.S. and J.F.; software, H.W. and F.S.; validation, F.S., H.Z. and K.T.; formal analysis, H.W.; investigation, H.Z. and Z.T.; data curation, H.Z. and Z.T.; writing—original draft preparation, H.Z. and J.F.; writing—review and editing, K.T.; visualization, K.T.; supervision, K.T.; project administration, K.T. and H.Z.; funding acquisition, K.T. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This study was greatly supported by the National Key Research & Development Program Grant No. 2023YFB4301900), the National Natural Science Foundation of China Project (Grant No. 52302414 and No. 52372319), the Belt and Road Cooperation Program under the 2023 Shanghai Action Plan for Science, Technology and Innovation (No. 23210750500), and the Fundamental Research Funds for the Central Universities.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Algorithm A1: Genetic Algorithm

Input: parameter space

Output: scheme x_i

1 **Initialize Population** x_0 :

Algorithm A1: Cont.

```

2      Create an initial population of individuals according to the solution by Webster method.
3  Define Fitness Function  $f(x)$ :
4      Evaluate the fitness of each individual in the population.
5      Assign a fitness score to each individual based on their performance.
6  Repeat for a Fixed Number of Generations:
7      For generation = 1 to max_generations do
8          //Step1: Selection
9              Select individuals from the current population to create a mating pool.
10             The selection probability of an individual is proportional to its fitness score.
11          //Step2: Crossover
12             Perform crossover between pairs of parents in the mating pool.
13             Create new individuals (offspring) by exchanging genetic information from the
14             parents.
15          //Step3: Mutation
16             Apply mutation to some of the offspring.
17             Randomly change some of the genes in the offspring to introduce diversity.
18          //Step4: Evaluate Offspring
19             Calculate the fitness of the newly generated offspring.
20          //Step5: Survival Selection
21             Combine the offspring and the current population to form a new
22             population for the next.
23             Use some strategy to determine which individuals will survive and form
24             the next generation.
25             Possible strategies include elitism, or a combination of offspring and
26             parents.
27          END FOR
28  Select Best Individual:
29      Select the individual with the highest fitness score as the best solution to the optimization
30      problem.
31  Output:
32      Return the best individual.

```

Algorithm A2: Bayesian Optimization**Input:** X : parameter space**Output:** optimal x

```

1  Initialize:
2      Define the search space of the parameters.
3      Set the initial set of observations  $x_0$  by the Webster timing method and their
4      corresponding evaluations.
5  Define Acquisition Function  $f(x)$ :
6      Choose an acquisition function to guide the search.
7      Evaluates the utility of different points in the search space based on previous
8      observations.
9  Repeat for a Fixed Number of Iterations:
10     FOR iteration = 1 to max_iterations
11         //Step1: Optimize Acquisition Function:
12             Find the next point in the search space that maximizes the acquisition function.
13             This step balances the exploration and exploitation trade-off.
14         //Step2: Evaluate Objective Function:
15             Evaluate the objective function at the selected point in the search space.
16             Add the new observation and its evaluation to the set of observations.
17         //Step3: Update Gaussian Process Model:
18             Update the surrogate model (e.g., Gaussian Process) using the new
19             observations.

```

Algorithm A2: Cont.

```

17         Estimates the underlying function and its uncertainty based on the
18         observations.
18     END FOR
19 Select Best Parameter Configuration:
20     Return the parameter configuration with the best evaluation value as the optimized
21     solution.
21 Output:
22     Return the optimized parameter configuration and its corresponding evaluation value.

```

Algorithm A3: Grid Search**Input:** X: parameter space**Output:** optimal x **1 Define Parameter Grid:****2** Define a grid of hyperparameter values for each parameter of the model.**3** Specify the range or discrete values for each hyperparameter.**5 Initialize Best Score and Best Parameters:****6** Set best_score to a small value (for minimization tasks).**7** Set best_parameters to None.**9 Repeat for Each Combination of Hyperparameters:****10 FOR** hyperparameter_combination **in** parameter_grid**11** Set Model Hyperparameters:**12** Set the model's hyperparameters to the values specified in hyperparameter_combination.**13** Fit and Evaluate Model:**14** Train the model with the current hyperparameters on the training data.**15** Evaluate the model's performance on the validation or cross-validation set.**16****17** Check for the Best Score:**18** IF the model's performance is better than best_score**19** Update best_score with the new score.**20** Update best parameters with the current hyperparameter combination.**21** END FOR**22** Output:**23** Return best_parameters as the optimal hyperparameter combination.**References**

1. Wágner, T.; Ormándi, T.; Tettamanti, T.; Varga, I. SPaT/MAP V2X communication between traffic light and vehicles and a realization with digital twin. *Comput. Electr. Eng.* **2023**, *106*, 108560. [\[CrossRef\]](#)
2. Dasgupta, S.; Rahman, M.; Lidbe, A.D.; Lu, W.; Jones, S. A Transportation Digital-Twin Approach for Adaptive Traffic Control Systems. *arXiv* **2023**, arXiv:2109.10863. [\[CrossRef\]](#)
3. Miletić, M.; Ivanjko, E.; Gregurić, M.; Kušić, K. A review of reinforcement learning applications in adaptive traffic signal control. *IET Intell. Transp. Syst.* **2022**, *16*, 1269–1285. [\[CrossRef\]](#)
4. Sims, A.G.; Dobinson, K.W. The Sydney coordinated adaptive traffic (SCAT) system philosophy and benefits. *IEEE Trans. Veh. Technol.* **1980**, *29*, 130–137. [\[CrossRef\]](#)
5. Cai, C.; Wong, C.K.; Heydecker, B.G. Adaptive traffic signal control using approximate dynamic programming. *Transp. Res. Part C Emerg. Technol.* **2009**, *17*, 456–474. [\[CrossRef\]](#)
6. Chen, S.; Sun, D.J. An Improved Adaptive Signal Control Method for Isolated Signalized Intersection Based on Dynamic Programming. *IEEE Intell. Transp. Syst. Mag.* **2016**, *8*, 4–14. [\[CrossRef\]](#)
7. Head, L. Event-Based Short-Term Traffic Flow Prediction Model. *Transp. Res. Rec.* **1995**, *1510*, 45–52.
8. Robertson, D.I.; Bretherton, R.D. Optimizing networks of traffic signals in real time—the SCOOT method. *IEEE Trans. Veh. Technol.* **1991**, *40*, 11–15. [\[CrossRef\]](#)
9. Wang, Y.; Yang, X.; Liang, H.; Liu, Y. A Review of the Self-Adaptive Traffic Signal Control System Based on Future Traffic Environment. *J. Adv. Transp.* **2018**, *2018*, 1096123. [\[CrossRef\]](#)
10. Lidbe, A.; Tedla, E.; Hainen, A.; Jones, S. Comparative assessment of arterial operations under conventional time-of-day and adaptive traffic signal control. *Adv. Transp. Stud.* **2017**, *2017*, 5–22. [\[CrossRef\]](#)

11. Aljaafreh, A.; Al Oudat, N. Optimized Timing Parameters for Real-Time Adaptive Traffic Signal Controller. In Proceedings of the 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, 26–28 March 2014; pp. 244–247.
12. Bhave, N.; Dhagavkar, A.; Dhande, K.; Bana, M.; Joshi, J. Smart Signal—Adaptive Traffic Signal Control using Reinforcement Learning and Object Detection. In Proceedings of the 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 12–14 December 2019; pp. 624–628.
13. Gao, J.; Shen, Y.; Liu, J.; Ito, M.; Shiratori, N. Adaptive Traffic Signal Control: Deep Reinforcement Learning Algorithm with Experience Replay and Target Network. *arXiv* **2017**, arXiv:1705.02755. [[CrossRef](#)]
14. Zeng, J.; Hu, J.; Zhang, Y. Adaptive Traffic Signal Control with Deep Recurrent Q-learning. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Suzhou, China, 26–30 June 2018; pp. 1215–1220.
15. Rachmadi, M.F.; Al Afif, F.; Jatmiko, W.; Mursanto, P.; Manggala, E.A.; Ma'sum, M.A.; Wibowo, A. Adaptive traffic signal control system using camera sensor and embedded system. In Proceedings of the TENCON 2011—2011 IEEE Region 10 Conference, Bali, Indonesia, 21–24 November 2011; pp. 1261–1265.
16. Aslani, M.; Mesgari, M.S.; Wiering, M. Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. *Transp. Res. Part C Emerg. Technol.* **2017**, *85*, 732–752. [[CrossRef](#)]
17. Goodall, N.J.; Smith, B.L.; Park, B. (Brian) Traffic Signal Control with Connected Vehicles. *Transp. Res. Rec.* **2013**, *2381*, 65–72. [[CrossRef](#)]
18. Abdulhai, B.; Pringle, R.; Karakoulas, G.J. Reinforcement Learning for True Adaptive Traffic Signal Control. *J. Transp. Eng.* **2003**, *129*, 278–285. [[CrossRef](#)]
19. Pandit, K.; Ghosal, D.; Zhang, H.M.; Chuah, C.-N. Adaptive Traffic Signal Control with Vehicular Ad hoc Networks. *IEEE Trans. Veh. Technol.* **2013**, *62*, 1459–1471. [[CrossRef](#)]
20. Krichen, M. Formal Methods and Validation Techniques for Ensuring Automotive Systems Security. *Information* **2023**, *14*, 666. [[CrossRef](#)]
21. Hähnle, R.; van der Aalst, W. (Eds.) *Fundamental Approaches to Software Engineering, Proceedings of the 22nd International Conference, FASE 2019, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2019, Prague, Czech Republic, 6–11 April 2019*; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2019; Volume 11424, ISBN 978-3-030-16721-9. [[CrossRef](#)]
22. Song, L.-K.; Li, X.-Q.; Zhu, S.-P.; Choy, Y.-S. Cascade ensemble learning for multi-level reliability evaluation. *Aerosp. Sci. Technol.* **2024**, *148*, 109101. [[CrossRef](#)]
23. Genders, W.; Razavi, S. An Open-Source Framework for Adaptive Traffic Signal Control. *arXiv* **2019**, arXiv:1909.00395. [[CrossRef](#)]
24. Zhang, C.; Tian, Y.; Zhang, Z.; Xue, W.; Xie, X.; Yang, T.; Ge, X.; Chen, R. Neighborhood Cooperative Multiagent Reinforcement Learning for Adaptive Traffic Signal Control in Epidemic Regions. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 25157–25168. [[CrossRef](#)]
25. Zhao, M.; Li, W.; Fu, Y.; Ruan, K.; Di, X. CVLight: Decentralized Learning for Adaptive Traffic Signal Control with Connected Vehicles. *arXiv* **2022**, arXiv:2104.10340. [[CrossRef](#)]
26. Hong, W.; Tao, G.; Wang, H.; Wang, C. Traffic Signal Control with Adaptive Online-Learning Scheme Using Multiple-Model Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 7838–7850. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.