Chapter 8
Digital Twin for the Internet of Vehicles

Abstract As a working combination of smart vehicles, advanced communication infrastructures, and intelligent transportation units, the Internet of Vehicles (IoV) has emerged as a new paradigm for safe and efficient urban life in the future. However, various types of smart vehicles with distinct capacities, diverse IoV applications with different resource demands, and unpredictive vehicular topology pose significant challenges to fully realize IoV systems. To cope with these challenges, we leverage digital twin (DT) technology to model complex physical IoV systems in virtual space, to identify the relation between application characteristics and IoV services, which facilitates effective service scheduling and resource management. In this chapter, we discuss the motivation, benefits, and key issues of applying DT in IoV systems. Then, we use vehicular edge computing and caching as two typical IoV application scenarios to present DT-empowered task offloading and content caching scheduling schemes and their performance.

8.1 Introduction

Vehicles are undergoing a fundamental shift, from simple transportation units to smart ones empowered with environmental sensing, autonomous driving, and information interaction capabilities. Integrating such smart vehicles with pedestrians and the infrastructures around them gives rise to Internet of Vehicles (IoV) systems, which provide a range of powerful vehicular applications and lead to pioneering advances in safety and the efficiency of intelligent transportation. For instance, IoV helps to deliver information gathered from the urban traffic environment to adjacent vehicles for safe navigation and traffic management. In addition, IoV can provide real-time information and interactive entertainment for vehicle occupants.

The development of IoV technology has received much attention in recent years, and high expectations have been raised about the benefits that its application will bring, prompting researchers and engineers to engage in in-depth discussions on possible obstacles in the IoV evolutionary graph.

Y. Zhang, *Digital Twin*, Simula SpringerBriefs on Computing 16,
https://doi.org/10.1007/978-3-031-51819-5_8

The key feature of IoV is its massive connections and dynamic topology. As we mentioned, IoV is a network consisting of vehicles, drivers, pedestrians, roadside units (RSUs), and other intelligent units participating in traffic applications, communicated in vehicle-to-vehicle (V2V), vehicle-to-RSU (V2R), vehicle-to-person, and vehicle-to-sensor modes. The mobility of vehicles and pedestrians can cause drastic changes in data transmission performance and even the interruption of communication links. The large scales of connected units and time-varying communication associations make IoV characteristic modelling and operation management seriously complex and difficult.

Another issue worth considering is the ultra-low latency constraints of some IoV applications. For example, in vehicle driving, when the vehicle in front brakes in an emergency, the following autonomous vehicle needs to complete the braking action within a few milliseconds according to the detected vehicle distance or the warning notification sent by the front vehicle. To meet such a strict delay constraint, the vehicle's control, environmental perception, vehicular communication, and information processing must be comprehensively coordinated.

The last issue to be addressed is closely related to the previous one. Different types of IoV applications rely on different forms of cooperative services from heterogeneous resources. For example, vehicular augmented reality needs to consume a great deal of computing and sensing resources, while onboard interactive entertainment mainly relies on communication and storage resources. Furthermore, synergy and competition exist between heterogeneous resource services. For instance, the premise of data processing is that the data can be transmitted to the corresponding processor node by communication resources, which may be in contention due to multiple vehicle communication pairs. The complex relation between these resources makes it challenging to efficiently implement IoV applications.

Several technical approaches to the above challenges have emerged, with Digital twin (DT), in particular, showing promise. By mapping physical IoV networks to virtual space, DT helps improve IoV application performance and resource efficiency. Some of the main benefits provided by DT to IoV are shown in Fig. 8.1 and are listed below.

Accurate mapping and unified modelling: In DT-empowered IoV networks, DT servers collect road traffic status and application service characteristics from sensors installed on smart vehicles and through communication facilities spread throughout the vehicular network, to construct a real-time and accurate reflection of physical IoV networks. Since a reflection model in virtual space is represented by multidimensional digital parameters, irrelevant physical difference between various types of vehicles can be shielded by normalizing the feature parameters, to build a unified model that enables modelling interaction and migration.

Feature digging and trend prediction: In the process of autonomous driving and on-board application services, vehicles can consume various resources, such as urban roads, vehicular communications, and edge computing. Therefore, collaboration, competition, and even social associations among multiple vehicles are generated. DT reflection helps to explore such potential features and relations in IoV systems.

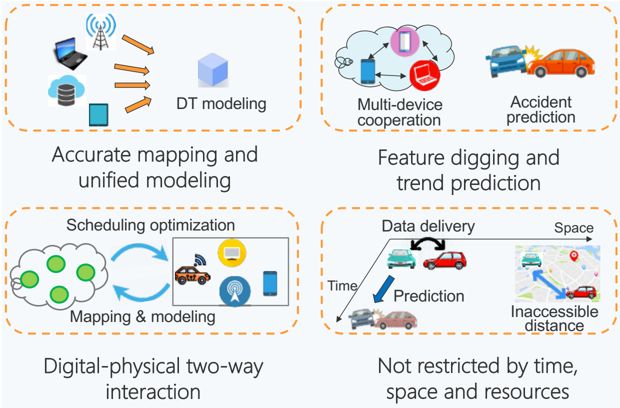

Fig. 8.1 Benefits provided by DT to IoV

Going a step further, based on these relations, DT can predict future physical actions, states, and events in IoV systems, such as possible traffic congestion or collisions.

Digital–physical two-way interaction: There is a two-way interaction between the DT model and the real physical entities of the IoV system. On the one hand, physical entities determine the digital mirroring. On the other hand, digital models logically guide physical action strategies. Both model accuracy and physical strategy performance can be improved during this iterative evolution process.

Not restricted by time, space, or resources: In physical IoV networks, the safety predictions of vehicle driving behaviour, inter-vehicle communications, and resource cooperation between vehicles are restricted by event sequences, wireless transmission distances, and vehicle resource capacities, respectively. However, in the DT image of IoV, these constraints can be broken. For example, by dynamically changing the timeline, retrospective determinations and predictions of traffic events are convenient to make. In addition, in virtual space, communications between inaccessible vehicles can be realized by data sharing between vehicle model processes in a DT server.

Motivated by the potential benefits of DT technology, a few works have addressed the incorporation of DT into IoV systems. In [106], the authors leveraged DT to facilitate collaborative and distributed autonomous driving. Based on vehicle DT models, driving decisions can be obtained at low cost. In [107], two DT models of vehicle driving states based on a Gaussian process and deep convolutional neural networks were respectively established that provide a scheme for the optimization of vehicle driving states and the realization of DT entity interactions. The authors in [108] introduced a DT-enabled edge intelligent cooperation scheme that guides optimal edge resource allocation and edge intelligent cooperation. Combining DT with vehicle-to-cloud communications, the authors in [109] presented a cooperative ramp merging system for connected vehicles that allows merging vehicles to cooperate with others prior to arriving at a merging zone. In [110], the authors focused on the security issues of cooperative intelligent transportation systems and constructed a DT model based on convolutional neural networks and support vector regression. Aided by the DT model, system security prediction accuracy was improved.

Despite much promising recent work in the area of DT-empowered IoV, several questions remain open for further investigation, and are discussed below.

Delays in DT modelling: Traffic safety is an important application scenario of DT-enabled IoV, in which some functions, such as early warnings of upcoming traffic accidents and adjustments of vehicle driving behaviour, have strict delay constraints. Meeting these constraints requires the DT model for the traffic environment and vehicle state to be constructed in a short time and to remain updated in real time. Considering the highly dynamic IoV topology and massive amounts of connected IoV nodes, the maintenance and tracking of such a complex system in real time are a challenge.

Efficiency in DT modelling: Following the previous challenge, to reduce DT modelling delays, many resources need to be allocated for vehicular environment sensing, state information delivery, and modelling processing. However, in addition to serving in the construction and update of DT models, constrained IoV resources are also used to support vehicular communication, autonomous driving, and onboard multimedia applications. How to reduce the resource costs of DT modelling and improve DT efficiency has become an important issue to be investigated.

Fault tolerance in DT modelling: The last but not least question concerns fault tolerance in DT modelling. Due to a limited sensing range, vulnerable wireless transmission parameters, and poor modelling processing power, established IoV DT models can have errors. These errors can seriously affect the control of vehicles' driving action and mislead the prediction of road traffic trends, thereby undermining the safety and efficiency of road traffic. In a harsh IoV environment, how to construct a DT model with high fault tolerance is still an unexplored problem.

8.2 DT for Vehicular Edge Computing

Driven by advances in vehicular communication and sensing and processing capabilities, many powerful IoV applications have emerged, such as autonomous driving, smart logistics, and driving augmented reality. However, the implementation of these applications requires intensive computation for environmental information processing and obtaining traffic behaviour under strict delay constraints, posing great challenges for vehicles with limited onboard computing resources.

VEC, which enables computing resource sharing at the edge of vehicular networks, is an appealing paradigm for meeting the intensive computation demands. In VEC, resource-hungry vehicles can offload their computing tasks to other smart vehicles or an RSU with spare computing power. However, to achieve efficient task offloading in such a dynamic and complex IoV environment, key issues still need to be addressed. For example, the communication scheduling for task data delivery is closely related to the computing resource management for task processing, which makes task offloading complicated. Moreover, resource competition between different offloading vehicle pairs, as well as the time-varying topology of vehicular networks, introduces further unprecedented challenges in managing VEC.

Recent advancements in machine learning provide significant capabilities to an aware dynamic IoV environment, determine action strategies, and tackle complex problems that rely in VEC applications. However, the effective implementation of the learning approach always relies on accurate and real-time system information gathered by learning agents. In vehicular networks characterized by massive amounts of connected smart vehicles, a highly dynamic topology, and a limited wireless spectrum, it is impractical to form a centralized artificial intelligence (AI) manager that schedules edge services for the entire network. To address this problem, we turn to multi-agent distributed learning empowered vehicular edge management. However, efficient collaboration and joint decision optimization among these multiple agents still face critical challenges.

DT is a promising technology to address these challenges. DT's state mapping between real and virtual dimensions provides users with comprehensive insights into the investigated system and dramatically reshapes the design and engineering process. Merging DT with machine learning will generate great benefits. On the one hand, DT provides AI with comprehensive and accurate system state information, which is exactly what learning processes require. On the other hand, AI provides much intelligence to DT, making its information collection and system description smart and efficient.

In this section, we propose a new VEC network based on DT and multi-agent learning that improves agent collaboration and optimizes task offloading efficiency [72]. In this network, DT is leveraged to reveal the potential cooperation between different vehicles and adaptively form multi-agent learning groups, which reduces learning complexity. Moreover, we design a distributed multi-agent learning scheme that minimizes vehicular task offloading costs under strict delay constraints in complex vehicular networks and dynamically adjusts the state-mapping mode of the DT network (DTN).

8.2.1 System Model

Figure 8.2 shows the framework of a DT-empowered VEC network. There are N smart vehicles on the road. These vehicles are equipped with computing power to process tasks and perform learning functions. The computing capability of vehicle i, $i \in \mathcal{N}$, is denoted as f_i CPU cycles per second. To enable powerful vehicular applications, such as autonomous driving and onboard entertainment, vehicles generate various types of tasks to be processed. Without loss of generality, we consider vehicle i to have J_i types of tasks, and task $w_{i,j}$ is described in the form of three elements, as $w_{i,j} = \{C_{i,j}, D_{i,j}, T_{i,j}^{\max}\}$. Here, $C_{i,j}$ is the amount of computing resources required to execute the task, $D_{i,j}$ presents the size of the task input data, and $T_{i,j}^{\max}$ is the maximum delay that task $w_{i,j}$ can tolerate.

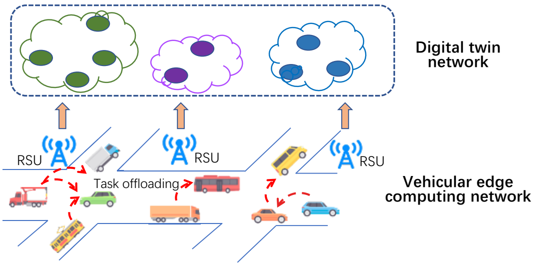

Fig. 8.2 A DT-empowered VEC network

Since different vehicles have diverse computing capabilities and task processing requirements, parts of the vehicles can have sufficient computing resources, whereas others are lacking. Through V2V communication, one vehicle can offload its tasks to others. We call the target vehicles "vehicular edge servers". Let $\beta_{i,j,k} = 1$ denote vehicle i, which offloads its task j to vehicular server k, and $\beta_{i,j,k} = 0$ denotes when the vehicle does not offload task j to server k. The time consumed to complete task $w_{i,j}$ is divided into two parts, namely, the offloading task transmission time and the task execution time. The transmission time of task $w_{i,j}$ from vehicles i to k through channel l is shown as $T^{\mathrm{tran}}_{i,j,k,l} = D_{i,j}/R_{i,k,l}$, where $R_{i,k,l}$ is the transmission rate.

A target vehicular server can receive multiple tasks from the other vehicles, and it puts these tasks in a queue. Taking into account task delay constraints, the target server executes the tasks in order according to the length of remaining time, from shortest to longest. Consequently, a task's execution time consists of the waiting time in the queue and the time processed in the CPU. The execution time of $w_{i,j}$ can be presented as

$$T^{\mathrm{exe}}_{i,j,k} = \sum_{i'=1}^{N} \sum_{j'=1}^{J_{i'}} \mathbf{1}\{T^{\mathrm{rem}}_{i',j'} \leq T^{\mathrm{rem}}_{i,j}\} \beta_{i',j',k} C_{i',j'}/f_{i'}, \tag{8.1}$$

where $\mathbf{1}\{\hat{x}\}$ is an indicator function that equals one if $\hat{x}$ is true, and zero otherwise, and $T^{\mathrm{rem}}_{i,j}$ is the remaining time of task $w_{i,j}$ before the deadline.

To improve vehicular computing resource utilization, a price-based incentive mechanism is incorporated into the resource scheduling. For a vehicular server, the weaker its computing power, the greater the resource demands of its queuing tasks, the tighter the tasks' delay constraints, and the higher the price of resources providing for guest tasks. We denote the price of a unit of computing resource of vehicle i as z_i.

In the vehicular edge system, a DTN continually maps the vehicles' physical states, such as the communication topology and computing resource demands, to virtual digital space. With the help of the DTN, edge service optimization and resource allocation strategies can be efficiently obtained.

8.2.2 DT and Multi-Agent Deep Reinforcement Learning for VEC

Merged with DT, AI learning gains comprehensive state information and effective guidance for agent learning, while helping DT to accurately model the physical system. We investigate the incorporation of DT and multi-agent learning in VEC networks and propose optimal edge service scheduling schemes. The main framework of these schemes is shown in Fig. 8.3.

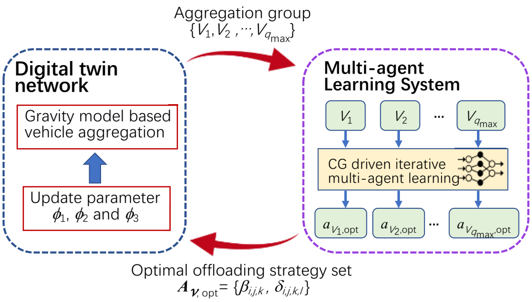

Fig. 8.3 Incorporation of DTN and multi-agent learning for VEC

Owing to the large-scale distribution of massive numbers of vehicles, it is costly and impractical to globally schedule the task offloading of the whole edge network. To address this issue, we leverage a DTN and gravity model to design an edge service aggregation scheme that efficiently aggregates vehicles based on the potential matching relations between the supply and demand of computing resources and greatly reduces the complexity of task offloading scheduling.

To guide the edge service aggregation, DTNs of the vehicular edge network are constructed in the RSUs. A DTN can be regarded as a combination of logical models and parameters recorded in digital space to characterize the states of the objects in physical space. We define the element of a DTN as $D_s = \{\mathcal{M}, \Phi, \varpi\}$. Here, $\mathcal{M}$ denotes the digital model of the vehicles in the physical system, which is described by a vehicle task set $\{w_{i,j}\}$, a computing capability set $\{f_i\}$, a resource price set $\{z_i\}$, and an available transmission rate set $\{R_{i,j}\}$. The modelling parameters are $\Phi = \{\phi_1, \phi_2, \phi_3\}$, which reflect the importance of the three factors of resources, pricing, and communication in the DTN modelling, respectively. The values of the parameters update periodically, and ϖ is the sequence number of the mapping periods.

With the aid of the DTN, we develop a gravity model–based vehicle aggregation scheme. Here we reform the gravity model and make it suitable to characterize the supply and demand relations of the vehicular edge service. The gravitation in the service association between vehicles i and i' is calculated as

$$F_{i,i'} = \frac{\phi_1 \max(m_i/m_{i'}, m_{i'}/m_i)}{\left(\phi_2(z_i/m_i + z_{i'}/m_{i'}) + \phi_3/R_{i,i'}\right)^2}. \tag{8.2}$$

According to the gravitation obtained in (8.2), we split the vehicles into multiple aggregation groups, which are denoted as $\{\mathcal{V}\}$. Based on this aggregation, we leverage a multi-agent learning approach to optimize edge resource allocation. Since the vehicles in the edge network have computing and communication capabilities, they can act as agents to learn the optimal edge scheduling strategies. To minimize the task offloading costs under delay constraints, the optimization problem is given in the following form:

$$\begin{aligned}
&\min_{\{\beta_{i,j,k},\delta_{i,j,k,l}\}} \sum_{V_q \in \mathcal{V}} \sum_{i=1}^{|V_q|} \sum_{j=1}^{J_i} \sum_{k=1}^{|V_q|} \beta_{i,j,k} \sum_{l=1}^{L} \delta_{i,j,k,l} C_{i,j} z_i \\
&\text{C1}: \sum_{k=1}^{V_q} \beta_{i,j,k} = 1, \quad \forall V_q \in \mathcal{V}, i,k \in V_q, j \in J_i \\
&\text{C2}: \beta_{i,j,k} = 0, \quad \forall V_q \in \mathcal{V}, i \in V_q, j \in J, k \notin V_q \\
&\text{C3}: \sum_{k=1}^{V_q} \beta_{i,j,k} (T_{i,j,k}^{\text{tran}} + T_{i,j,k}^{\text{exe}}) \le T_{i,j}^{\max}, \quad \forall V_q \in \mathcal{V}, i,k \in V_q, j \in J_i
\end{aligned}, \tag{8.3}$$

where constraint C1 ensures that a task can be offloaded at most to only one vehicle for processing, C2 indicates that task offloading occurs only between vehicles belonging to the same aggregation group, and C3 shows that the time consumption, including the transmission and execution time, should be within the delay constraints of the tasks. Problem (8.3) is an integer programming problem and has been proved to be NP complete.

Let $U_{V_q} = \sum_{i=1}^{|V_q|} \sum_{j=1}^{J_i} \sum_{k=1}^{|V_q|} \beta_{i,j,k} \sum_{l=1}^{L} \delta_{i,j,k,l} C_{i,j} z_i$. The target function of (8.3) can be written as $\min \sum_{V_q \in \mathcal{V}} U_{V_q}$. According to C2, there is no offloading correlation between the different aggregation groups. Thus, to address problem (8.3), we turn to minimize U_{V_q} by adopting a multi-agent deep deterministic policy gradient (MADDPG) learning approach, where $V_q \in \mathcal{V}$. The number of learning iterations is represented by the time slot t. For vehicle i belonging to aggregation group V_q, its action taken at time slot t is $a_i^t = \{\beta_{i,j,k}^t, \delta_{i,j,k,l}^t\}$, where $i,k \in V_q$, $j \in J_i$ and $l \in L$. Then, the action set of the multiple agents is given as $A^t = \{a_i^t\}$. The state at time slot t can be presented as $S^t = \{T_{i,j}^{\text{rem,t}}, \Gamma_k^t\}$, where $T_{i,j}^{\text{rem,t}}$ and Γ_k^t are the remaining completion time of the task $w_{i,j}$ and the set of tasks that have been queued for processing in vehicle k in time slot t, respectively. Taking action A^t in state S^t, the learning system of V_q gains the reward

$$Q_q^t(S^t, A^t) = \sum_{i=1}^{|V_q|} \sum_{j=1}^{J_i} \sum_{k=1}^{|V_q|} \beta_{i,j,k}^t \sum_{l=1}^{L} \delta_{i,j,k,l}^t C_{i,j} z_i. \tag{8.4}$$

The main goal of multi-agent learning in group V_q is to find the optimal action strategy for the agents to minimize the group's task offloading costs, presented as

$$Q_q(S^0, A) = \mathbb{E}\left[\sum_{t=0}^{\infty} \xi Q_q^t(S^t, A^t)|S^0\right], \tag{8.5}$$

where ξ is a discount coefficient that indicates the effect of a future reward on the current actions, and $0 < \xi < 1$.

The DTN and the multi-agent learning system operate cooperatively in scheduling the vehicular edge service. On the one hand, the DTN determines the distributed learning environments of the multiple agents by aggregating vehicular groups under the guidance of the parameters $\Phi = \{\phi_1, \phi_2, \phi_3\}$. This aggregation improves the supply and demand matching of edge resources and reduces the multi-agent learning complexity. On the other hand, the multi-agent learning results, that is, the task offloading target selection and edge resource allocation, affect the vehicular edge service performance and the performance indicators can be used in turn to evaluate the pros and cons of the aggregation mechanism, to adjust the aggregation parameter set Φ. These two parts iteratively interact and update to adapt to the changes in application scenarios.

8.2.3 Illustrative Results

We evaluate the performance of our proposed vehicular edge task offloading schemes based on real traffic data sets, which are extracted from the historical mobility traces of taxi cabs in the San Francisco Bay area. There are approximately 500 cabs, and the average time interval for their GPS coordinate updates is less than 10 seconds [112]. To investigate the influence of traffic environment characteristics on the offloading scheme performance, we further divide the Bay Area into six square areas. We consider a scenario in which the computation capacities of the vehicles are randomly taken from (10, 20) units. The computation resource requirements, data size, and maximum tolerable latency of the tasks are randomly chosen from (30, 50) units, (5, 10) MB, and (0.5, 2) seconds, respectively. In addition, there are five orthogonal channels for offloading transmissions, and the bandwidth of each channel is 0.3 MHz.

Figure 8.4 shows the offloading costs under different scheduling schemes. Compared with the other two schemes, our proposed MADDPG obtains the lowest cost. In the independent learning scheme, each vehicle works as an agent to aware edge service environments and makes self-interested offloading actions without interactions among the agents. This independent decision-making approach can create a resource surplus or shortage between some vehicular service pairs, thereby undermining the offloading efficiency of the whole system. In the MADDPG without aggregation, all the agents in the same area adopt joint decision making. Due to the complexity of the vehicle topology and potential service relations, in this scheme, it is difficult to reach the optimal offloading strategy under constrained learning iterations. In contrast to the previous two schemes, the MADDPG scheme aggregates vehicular agents based on DTN-aided edge service matching, which helps the scheme to realize low-complexity multi-agent collaborative learning under the premise of efficient resource utilization and obtains the lowest cost.

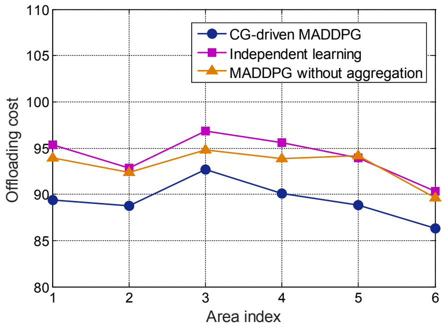

Fig. 8.4 Comparison of offloading costs with different schemes

Figure 8.5 presents the convergence of the MADDPG learning scheme. We randomly select two agents from areas 3 and 5, respectively. All the agents' learning converges around 3,300 iterations. Furthermore, this figure demonstrates that the difference in edge network characteristics and aggregation groupings between the two areas has little effect on the convergence performance.

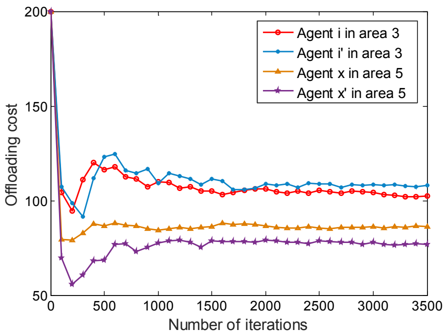

Fig. 8.5 Convergence of MADDPG learning

8.3 DT for Vehicular Edge Caching

Along with the proliferation of smart vehicles and powerful IoV applications, the huge amounts and high diversity of content need to be disseminated and shared between interactive vehicles under stringent delay constraints. However, due to limited spectrum resources, it is challenging for current wireless systems to deliver content while meeting such requirements, especially in heavy traffic scenarios with high vehicle density.

Vehicular edge caching is a promising paradigm for addressing this challenging issue. Edge caching technology locates popular content close to end users via distributed cache vehicles and RSUs and considerably accelerates the responsiveness of content acquisition from the edge, compared to fetching them from remote content providers. However, unstable communications and the highly dynamic topology between smart vehicles and RSUs still pose critical challenges in designing optimal caching schemes for vehicular edge networks. In practice, an individual edge cache server always has constrained storage space, which makes it impossible for a single server to hold multiple large files at the same time. Moreover, when the cache servers are equipped on several RSUs, the limited coverage range of an individual RSU can lead to short communication durations and small amounts of data delivered.

To effectively utilize the constrained cache and communication resources with dynamic topology, cooperative caching needs to be leveraged, where content subscribers can be served by multiple caching servers. Moreover, to make full use of the caching capabilities of smart vehicles, social interactions among the vehicles can be utilized to improve content dispatch efficiency. The social characteristics of the vehicles are basically related to their drivers, who determine their content preferences and daily driving routines and affect the other vehicles that may be encountered on the road or in parking lots.

Integrating socially aware smart vehicles and the mobile edge computing framework also requires addressing the challenges brought about by socially aware smart vehicles. For instance, vehicular social characteristics are time varying and can change dynamically according to content popularity, traffic density, and vehicle speeds. Furthermore, owing to the mobility of vehicles, highly intermittent connectivity between vehicular content providers and subscribers can seriously undermine the efficiency of socially aware content transmission. In addition, the cooperation between vehicular cache resources needs to cater to road traffic distribution, channel quality, and content popularity. Thus, supporting delay-bounded content delivery over vehicular social networks with multiple cache-enabled smart vehicles is a challenge.

DT technology can be used to address the above challenges. In socially aware vehicular edge caching networks, the DT approach can enable cache controllers to grasp the social relations between vehicles, understand the vehicle flow distribution, and effectively allocate communication and storage resources for content delivery. In this section, we propose a DT-empowered content caching mechanism for socially aware vehicular edge networks [111]. We present a DT-based vehicular edge caching framework that comprehensively captures vehicular social features and improves caching scheduling in highly dynamic vehicular networks. Moreover, by applying a deep deterministic policy gradient (DDPG) learning approach, we propose an optimal vehicular caching cloud formulation and edge caching resource arrangement that maximize the system's utility in diverse traffic environments.

8.3.1 System Model

Figure 8.6 shows a DT-empowered vehicular social edge network. We consider an intelligent transport system in urban areas, where smart vehicles provide various powerful applications, such as smart navigation, online video, and interactive gaming. The implementation of these applications always requires content generated by the data centre, which is located in the core network. The required content is classified into G types. Each type of content is described in three terms, as $T_g = \{f_g, t_g^{\max}, \mu_g\}$, and $g \in \mathcal{G}$, where f_g is the size of content type g, $t_g^{\max}$ is its maximum delay tolerance, and μ_g is the delay sensitivity coefficient that can be taken as the utility gained from a unit time reduction compared to $t_g^{\max}$ during the content delivery process.

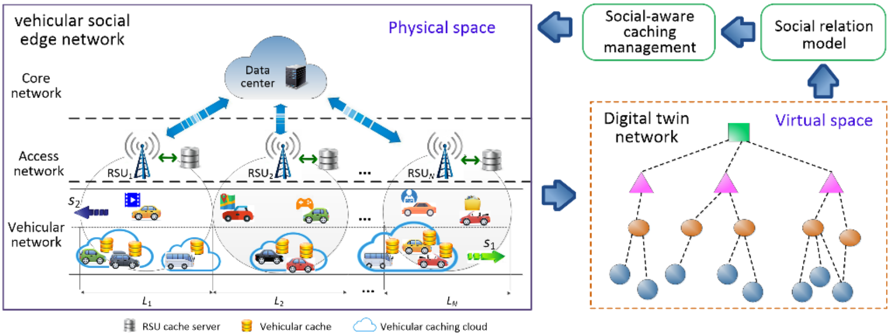

Fig. 8.6 A DT-empowered vehicular social edge network

To form access networks and provide data to vehicular content subscribers, N RSUs are located along bidirectional roads that can receive content from the data centre and then relay it to the vehicles. The diameters of the regions covered by these RSUs are $\{L_1, L_2, \ldots, L_N\}$, respectively. Each RSU is equipped with an edge caching server. The caching capabilities of these servers are $\{C_1, C_2, \ldots, C_N\}$, respectively. To avoid long transmission latencies between the data centre and the vehicles, the servers can retrieve popular content from the centre and store them in their cache for later use.

Besides being cached in RSUs, content can also be pre-stored in smart vehicles. Cache-enabled smart vehicles on the road act as content carriers and forward cached data to vehicles they encounter through V2V communication. To fully exploit V2V content delivery, vehicular social relations are leveraged in edge cache management. When the supply and demand content between vehicles is consistent and the communication link for data delivery can be established, we say that the vehicles are socially related. From this viewpoint, vehicular social relations are characterized by two elements. One element is the content matching between the supply and demand sides, and the other is the communication contact rate of the vehicles. We consider that the vehicles in this system demand G types of content with probability

$\beta = \{\beta_1, \beta_2, ..., \beta_G\}$, respectively, where $\sum_{g \in \mathcal{G}} \beta_g \leq 1$. When a vehicle with type g content in its cache is on the road, the probability of encountering a vehicle that needs exactly this type of content is β_g. Thus, the content-matching element can be described by the probability β. The communication contact rate is defined as the number of vehicles with which a given vehicle can be associated in a unit time while it is driving.

8.3.2 DT-Empowered Content Caching

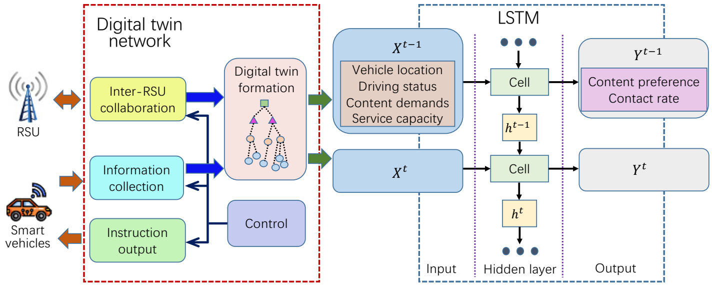

Fig. 8.7 DT and LSTM-based social model construction

Figure 8.7 illustrates the main framework of the proposed DT and a social model construction approach based on long short-term memory (LSTM). The DTN consists of five modules, where the information collection module obtains vehicular network states from smart vehicles through V2R communication. The control module determines the update cycle and adjusts the data type and interactive frequency in information collection. The adjustment will be issued to the smart vehicles through the instruction output module, thereby changing the vehicles' state sampling and reporting mode. After establishing the DT, which offers a virtual representation of the physical vehicular network, we use an LSTM recurrent network to extract the social features from the received data sets.

We use $\psi_g(\xi_g)$ to denote the accuracy of the social model that reflects the relations between the supply and demand of vehicles for type g content, where ξ_g is the amount of system information gathered by DT to train the LSTM network and obtain the social model $\{\beta, s_1, s_2\}$. The value of $\psi_g(\xi_g)$ is the modulus ratio of the estimated social model parameters to those of the true model, and $0 \leq \psi_g(\xi_g) \leq 1$. Since more information would help improve the model's accuracy, $\psi_g(\xi_g)$ is a monotonically increasing function in terms of ξ_g.

In the proposed vehicular edge caching network, to improve the delivery time efficiency while reducing transmission costs, the content needs to be efficiently prestored in appropriate cache nodes. Moreover, as the caching arrangement depends on the vehicular social model obtained from the DT-empowered LSTM system, the more information gathered by the DTN, the higher the accuracy of the model. However, the information collection process incurs a V2R communication cost. Thus, the trade-off between the V2R communication cost and model accuracy and its impact on the caching system utility also need to be considered in the cache scheduling.

Let x_g and $\mathcal{Y}_g = \{y_{g,1}, y_{g,2}, ..., y_{g,N}\}$ denote the probability of pre-storing type g content in the vehicular caching cloud and in the caching servers equipped on RSUs, respectively. The size of the content segment cached in a vehicle is Q_g. The proposed optimal edge caching problem, which maximizes the utility of the caching system under the constraints of node cache capacity and content delivery delay, can therefore be formulated as

$$\begin{aligned}
\max_{\{x_g, \mathcal{Y}_g, Q_g, \xi_g\}} \quad & U = \sum_{g \in \mathcal{G}} \{ \sum_{v' \in \mathcal{V}_2} \sum_{n \in \mathcal{N}} \Psi_g(\xi_g) \beta_g [x_g (\mu_g (t_g^{\max} \\
& -t_{g,v'}^{V}) - f_g \varsigma_v) + (1 - x_g) w_{g,v',n} (\mu_g (t_g^{\max} \\
& -t_{g,v',n}^{R}) - f_g (\varsigma_r + (1 - y_{g,n}) \varsigma_c))] - \xi_g \varsigma_r \} \\
\text{s.t.} \quad \mathrm{C1}: \quad & 0 \leqslant x_g \leqslant 1, \quad g \in \mathcal{G}, \\
\mathrm{C2}: \quad & 0 \leqslant \mathcal{Y}_g \leqslant 1, \quad g \in \mathcal{G}, \\
\mathrm{C3}: \quad & e_v \leqslant C_v, \quad v \in \mathcal{V}_1, \\
\mathrm{C4}: \quad & \textstyle\sum_{g \in \mathcal{G}} y_{g,n} f_g \leqslant C_n, \quad n \in \mathcal{N}, \\
\mathrm{C5}: \quad & \mathbf{1}\{x_g > 0\} t_{g,v'}^{V} \leqslant t_g^{\max}, g \in \mathcal{G}, v' \in \mathcal{V}_2, n \in \mathcal{N}, \\
\mathrm{C6}: \quad & \mathbf{1}\{y_{g,n} > 0\} t_{g,v',n}^{R} \leqslant t_g^{\max}, g \in \mathcal{G}, v' \in \mathcal{V}_2, n \in \mathcal{N}, \\
\mathrm{C7}: \quad & Q_g \leqslant Q_v^{\max}, \quad g \in \mathcal{G}, \\
\mathrm{C8}: \quad & \textstyle\sum_{g \in G} \xi_g \leq \xi^{\max}, \quad \xi_g > 0, \quad g \in \mathcal{G},
\end{aligned} \tag{8.6}$$

where $\mathcal{V}_1$ and $\mathcal{V}_2$ denote the sets of the content provider and subscriber vehicles in an area, respectively; $\Psi_g(\xi_g)$ is an influence function that presents the impact of social model deviation caused by different amounts of gathered information on the system's utility; and $w_{g,v',n}$ is the probability that vehicle v' is located within the coverage of RSU n and obtains type g content from the cache server equipped on this RSU in V2R mode.

In (8.6), the first two constraints show the range of the caching probability. Constraints C3 and C4 guarantee that the amount of content on a vehicle and on an RSU server should not exceed the maximum storage capacity of the respective caching node. Constraints C5 and C6 ensure the time cost for type g content remains within its delay constraint. Constraint C7 indicates that the size of the content segment cached in a vehicle should not exceed the upper limit. The last constraint ensures that the amount of information related to type g content is positive and the total amount of gathered information should not exceed the maximum threshold $\xi^{\max}$.

In the proposed optimal caching problem, the edge cache scheduling relies on the social model built, while, in the model construction, the adjustment of information collection depends on its effect on the system's utility. Moreover, due to possible content segmentation and cache resource sharing, there exists strong correlation between the various types of content cached in heterogeneous edge caching nodes. These features make solving problem (8.6) a critical challenge. To address this issue, we propose a DDPG learning–based iterative approach. In each iteration, we first obtain the cache scheduling strategies according to a given social model and then modify the amount of information gathered in model construction based on the determined caching strategies. The iteration continues until the system's utility converges.

8.3.3 Illustrative Results

We evaluate the performance of the proposed DT-empowered and socially aware edge caching schemes based on vehicular traffic data sets gathered in different areas. We consider a scenario in which one to three RSUs are randomly located in each area. The data storage capacity of the cache server equipped on each RSU is randomly set within the interval (300, 700) MB. There are 10 types of content requirements, of which the content size, maximum delay tolerance, and delay sensitivity coefficient are randomly chosen from (10,100) MB, (0.5, 3) seconds, and (0.1, 0.3), respectively.

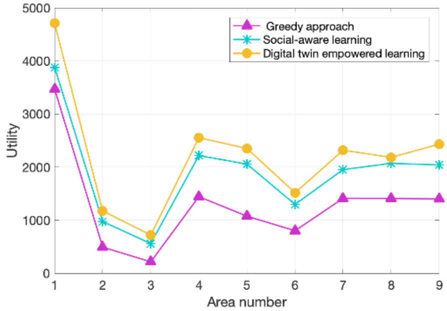

Fig. 8.8 Comparison of the caching utilities of multiple areas under different schemes

Figure 8.8 compares the utilities of multiple areas with different edge caching scheduling schemes. Our proposed DT-empowered learning approach gains the highest utilities in all the urban areas compared to the others. Here, the greedy approach, which obtains the lowest utility, arranges the content storage in the edge cache nodes only according to content popularity and ignores the social relations between smart vehicles and thus fails to make full use of the communication contacts between vehicles to implement V2V data delivery. In contrast to this approach, the socially aware learning scheme takes the content delivery among vehicles directly into account and dynamically allocates cache and communication resources based on the content requirements and known environmental characteristics, thus achieving higher utility. However, its social feature perception mode is fixed, which can increase detection costs or reduce perception accuracy. Unlike the two previous schemes, the one we proposed leverages DT to reflect the vehicular network states while adaptively adjusting social model construction strategies with balanced accuracy and costs, thus resulting in the highest caching utility.

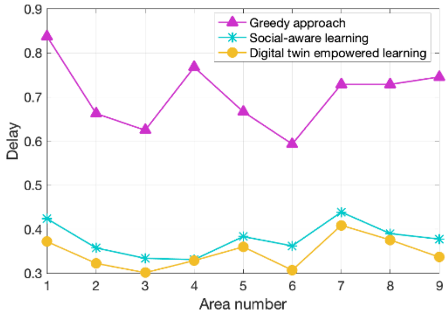

Fig. 8.9 Comparison of the content acquisition delays of multiple areas under different schemes

Figure 8.9 compares the content acquisition delays under different schemes in multiple areas. Our proposed DT-empowered learning scheme outperforms the other two approaches. Since this scheme smartly utilizes vehicular social relations and caching capacity in enabling direct data delivery between vehicles, the content acquisition delay is reduced. It is worth noting that, although in a few areas, such as area 3 in Fig. 8.8 and area 4 in Fig. 8.9, the performance of the DT-empowered learning scheme is close to that of the socially aware learning approach, in all areas as a whole, the utility (delay) of the DT-empowered scheme is increased (decreased) by 17% (10%), on average, over the simple socially aware scheme. Since both these schemes leverage vehicular social relations to schedule cache resources, the difference in their performance is smaller than the performance gap between the socially aware schemes and the greedy approach, which ignores vehicular social relation effects. Moreover, the performance gain provided by the DT mechanism is affected by the different vehicle distributions, driving states, and caching capacities in various areas. Therefore, there are differences in the gain effects of DT in these areas.