Cost-effective Vehicular Data Offloading in ISTNs: A Reinforcement Learning Approach

Shen Wu*, Nan Cheng*, Zhisheng Yin†, Jingchao He* and Haibo Zhou‡
*School of Telecommunications Engineering, Xidian University, Xi'an, China
†School of Cyber Engineering, Xidian University, Xi'an, China
‡School of Electronic Science and Engineering, Nanjing University, Nanjing, China
Email: {swu_2, jchhe}@stu.xidian.edu.cn, dr.nan.cheng@ieee.org, zsyin@xidian.edu.cn, haibozhou@nju.edu.cn

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10000949

***Abstract*—Integrated satellite-terrestrial network (ISTN) can provide a continuous service for vehicular users in remote areas with a seamless network coverage. However, considering the difference in the usage costs between satellite and terrestrial networks and the variability of services for latency requirements, it is of great significance to design a cost-effective data offloading decision for reducing network overhead and ensuring task delay requirements. In this paper, we design a cost-effective data offloading mechanism for vehicles in ISTN. The default transmission for remote areas is via the satellite, where the terrestrial networks can offload the data with intermittent coverage in an opportunistic manner due to the vehicle mobility. To model the diversity in service delay requirements, a virtual queue is exploited to capture the residual maximum delay tolerance of each service as time elapses. We formulate the satellite-terrestrial collaborative transmission as a non-linear programming (NLP) problem. To solve the problem, we propose a reinforcement learning (RL)-based data offloading algorithm for real-time decision making. Simulation results show that the RL-based data offloading algorithm reduces the network overhead and outperforms other baseline schemes we proposed.**

***Index Terms*—Data offloading, integrated satellite terrestrial networks, remote areas, reinforcement learning**

I. Introduction

With the rapid development of the Internet of Vehicle (IoV) and the improvement of vehicle intelligence, vehicles are no longer only a means of transportation to meet people's travel needs but can provide in-vehicle infotainment (IVI) services to meet various needs [1]. IoV primarily offer three services, which are information service, vehicle intelligence service, and the intelligent transportation service [2]. Among them, the provisioning of information services is of great challenge since the requirements of such services vary significantly [3]. Online video conferencing services, for example, are more sensitive to latency, whereas playback and downloading of online music and videos on the Internet typically prefer data rate than the latency.

Currently, there are two main vehicle communication and networking technologies, i.e., dedicated short-range communication (DSRC) based on the IEEE 802.11p standard and cellular vehicle-to-everything (C-V2X) communications [4]. Although these two technologies have complementary benefits in terms of communication rate, latency, flexibility, and reliability, the high dependency on network infrastructure such as roadside units (RSUs) and base stations (BSs) makes them unable to deliver services to vehicles within remote areas [5], [6]. Combining the seamless connectivity of the satellite network with the flexibility of the air network and the high data rate access of the ground network, space-air-ground integrated network (SAGIN) can provide users with ubiquitous, high-quality network services [7]. Integrated satellite-terrestrial network (ISTN) is a typical SAGIN network architecture that combines the benefits of satellite networks and terrestrial networks to enlarge network coverage, promote network reliability, ensure service continuity, and provide efficient transmission [8]. With the deployment of the mobile edge computing (MEC) server, SAGIN is able to provide ubiquitous, high-quality MEC services [9]. Software-defined networking (SDN) and artificial intelligence (AI) are also applied in space-air-ground integrated vehicular network to manage SAGIN flexibly and efficiently, thus providing better quality and more economical vehicle services for vehicular networking [10], [11].

To use the satellite and terrestrial networks of SAGIN flexibly and on-demand, an efficient data offloading strategy is needed. A mobility-aware task offloading and migration problem is proposed in [12], which aims to reduce the migration cost due to the mobility of users. However, the diversity of service requirements is not considered. In [9], satellites and unmanned aerial vehicles (UAVs) are used to deliver IoV services to vehicles within remote areas, and a deep imitation learning approach is proposed to achieve real-time decision making. Nonetheless, the cost for deploying and maintaining the UAV is usually prohibitive, and is thus not suitable for covering large remote areas. On the other hand, the terrestrial networks, although may be deployed sparsely, can also be used for data transmission. In [13], a data offloading scheme for ISTN is proposed, which satisfies the different demands of URLLC and eMBB and improves the network availability. However, the mobility of users and the difference in service costs between satellite and terrestrial networks are not considered. In order to comprehensively consider the mobility of users, the diversity of service requirements, and the different costs between satellite and terrestrial networks, a more cost-effective data offloading strategy is further required.

In this paper, the cost-effective collaborative task transmission scheme is proposed for vehicular users with ISTN. The remote areas are considered, where the cellular network provides intermittent coverage and the satellite network provides full coverage for vehicular users. Specifically, we use an on-off process to model the mobility of vehicles. By modeling the task arrivals, channel state, and task queues, the final problem is formulated as a mathematical optimization problem. Due to the mobility of vehicles and the dynamic arrival of tasks, we propose a reinforcement learning (RL) based data offloading method to obtain the optimal real-time decision for the problem. The main contributions of this paper are as follows:

- We use an on-off process to model the mobility of vehicles in remote areas. Vehicle on-off states are calculated by estimating the sojourn time of on and off states using memoryless exponential distributions with parameters λ and μ. By changing λ and μ, vehicles with different mobility features can be simulated.
- For the convenience of the DDPG algorithm in observing the queue state, a queue model is proposed to describe the queue state of tasks with different maximum waiting time. There are different task queues with different maximum waiting time. Task arrivals will be appended to the end of the corresponding task queue at each time slot t. These task queues are merged into a single total queue in the order of the maximum wait time.
- A cost-effective data offloading strategy for vehicles within remote areas is proposed, and in order to achieve real-time decision making, it is transformed into a RL problem. We used the DDPG algorithm to solve this problem. In addition, simulation results show that our proposed data offloading strategy can achieve lower network overhead than other baseline algorithms. We also verify the effect of different mobility features on network overhead, which reveals that the more frequently the vehicle moves in and out of the coverage of terrestrial networks, the less network overhead can be achieved by the proposed RL-based algorithm.

The remainder of this paper is organized as follows. Section II introduces the system model and formulates the cost-effective data offloading problem for vehicles within remote areas. In Section III, a data offloading strategy based on DDPG is proposed to make real-time decision. Simulation results are conducted in Section IV, where the effectiveness of the data offloading strategy is verified. Section V concludes this paper.

II. System Model

We consider an ISTN-based collaborative data offloading model for vehicles within remote areas, as shown in Fig. 1, where the LEO satellite network covers the whole area while the cellular BSs are deployed sparsely. We assume vehicles continuously send service requests to the LEO satellites, and the LEO satellites can collaborate with the cellular BSs to transmit the service data, where the data offloading decision is made at the LEO satellite. Since transmitting data using LEO satellites may lead to extensive costs, considering the differences in delay requirements of tasks, the delay-tolerant tasks can until the vehicle enters the coverage area of a cellular BS and then use the BS for data transmission, which reduces the network overhead.

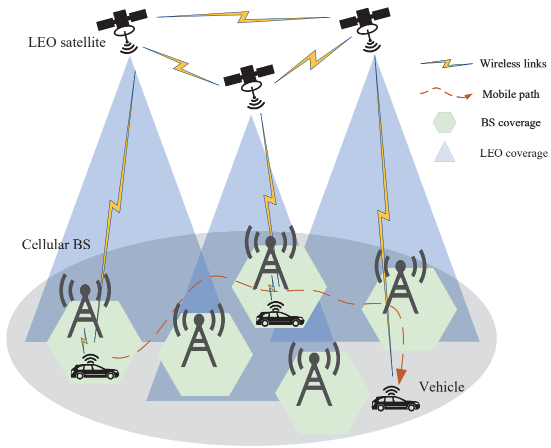


Fig. 1. ISTN-based collaborative data offloading model for vehicles within remote areas.

Specifically, in a time slot $t \in \{1, 2, ..., T\}$, vehicles will randomly generate a variety of service requests. First, the vehicle will send the service request to the LEO satellite, and then the LEO satellite will make a data offloading decision denoted by $a_t \in [0, 1]$, which determines the maximum amount of tasks that can be transmitted by BS and LEO satellite. The service request will be collaboratively transmitted by the BS and the LEO satellite, with the BS having a maximum task amount of a_t times the current total task and the LEO satellite having a maximum task amount of $(1 - a_t)$ times the current total task. It is important to note that the tasks can be fine-grained, so the data transmitted by BS and LEO satellite is different.

A. Communication Model

In the ISTN-based collaborative data offloading model for vehicles within remote areas architecture, only downlink for communication between vehicles and BSs and LEO satellites is considered, and the frequency bands used by BSs and LEO satellites are different, so they do not interfere with each other. From the perspective of a single vehicle, the downlink rates of the BS and the LEO satellite to the vehicle in time slot t are denoted by R_t^B and R_t^S, respectively. The downlink rate of the BS to the vehicle can be calculated by

$$R_t^B = B_B \log_2 \left(1 + \frac{p_B h_t^B}{\sigma^2}\right), \tag{1}$$

where B_B represents the channel bandwidth between the vehicle and BS. p_B represents the transmit power of the BS. h_t^B are the channel fading coefficients of the BS to vehicle link. σ^2 is the Gaussian white noise power.

Considering the effect of rain attenuation, the downlink rate of the LEO satellite to the vehicle is expressed as

$$R_t^S = \Lambda B_S \log_2 \left(1 + \frac{p_S (h_t^S)^2}{\sigma^2}\right), \tag{2}$$

where B_S represents the channel bandwidth between the vehicle and the LEO satellite. p_S represents the transmit power of the LEO satellite. h_t^S is the channel fading coefficient of the LEO satellite to vehicle link. Λ is the rain attenuation.

B. Mobility Model

To describe the process of vehicles intermittently accessing BSs, we proposed a vehicle mobility model in ISTN based on the on-off process proposed in [14]. Fig. 2 shows the vehicle mobility model in ISTN, where the vehicle can access LEO satellites seamlessly while intermittently accessing BSs as it moves. The vehicle can receive the BS signal in the on-state while it cannot receive the BS signal in the off-state. The on-state and off-state sojourn time follow an unexpected, memoryless exponential distribution with parameters λ and μ, where λ and μ are the average sojourn times of the on-state and off-state, respectively. The moving state of the vehicle can be obtained by predicting the sojourn time of the vehicle in on and off states using the exponential function and putting them together sequentially.

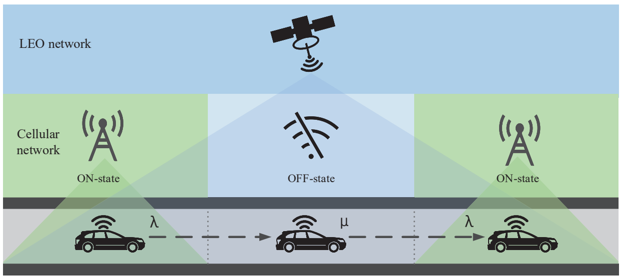


Fig. 2. Vehicle mobility model in ISTN.

C. Task Model

Vehicles can request services with different maximum waiting time. The arrivals of tasks with different maximum waiting time at time slot t are denoted by A_t^i, $i \in \{0, 1, ..., N\}$. i indicates the number of task types, and it implies that the task can wait at most i time slots before transmitting. We assume the task arrivals of different types of tasks follow a normal distribution and the parameters increase as the maximum waiting time of the service increases [15]. Thus, tasks with lower latency requirements will have more task arrivals per time slot. Total task arrivals at time slot t can be calculated by $A_t = \sum_{i=0}^{n} A_t^i$.

D. Queue Model and Data Offloading

To describe the task transmitting process, we exploit a queue model, as shown in Fig. 3. Corresponding to the types of tasks, there are also $j \in 1, 2, ..., J$ task queues. These task queues are arranged in order according to the maximum waiting time to form a total queue. The queueing model is denoted as $L(t) \triangleq [L_0(t), L_1(t), ..., L_J(t)]$, where $L(t)$ is the total queue and $[L_0(t), L_1(t), ..., L_J(t)]$ are task queues of different delay requirements. A_t^i will join the end of the corresponding task queue $L_j(t)$ at the beginning of time slot t, where $i = j$. The LEO satellite will then make an offloading decision a_t, and the tasks will be collaboratively transmitted by the BS and the LEO satellite.

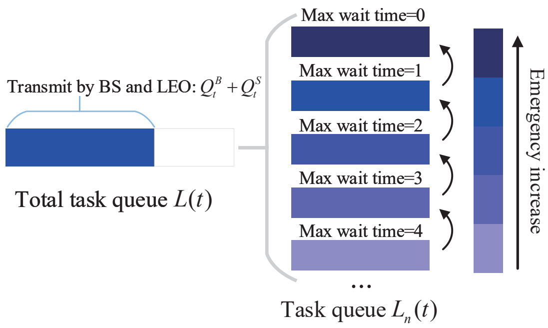


Fig. 3. Queueing model.

The maximum amount of data that the BS can transmit in time slot t is τR_t^B, where τ is the length of the time slot. The actual amount of data that the BS can transmit according to the offloading decision a_t is denoted by

$$Q_t^B = \min\{a_t L(t), \tau R_t^B\}. \tag{3}$$

Therefore, the network overhead of the BS in time slot t can be calculated by

$$C_t^B = \beta_1 Q_t^B. \tag{4}$$

Similarly, the maximum amount of data that can be transmitted by the LEO satellite in time slot t is τR_t^S , and the actual amount of data that can be transmitted according to the offloading decision a_t is denoted as

$$Q_t^S = \min\{(1 - a_t)L(t), \tau R_t^S\}. \tag{5}$$

The network overhead of the LEO satellite in time slot t can be calculated by

$$C_t^S = \beta_2 Q_t^S, \tag{6}$$

where β_1 and β_2 are the unit costs of using BSs and LEO satellites, respectively. The total cost to be paid by the vehicle at time slot t is denoted as

$$C_t = C_t^B + C_t^S. \tag{7}$$

The total amount of data that can be transmitted in time slot t is

$$Q_t = Q_t^B + Q_t^S, Q_0 = 0. \tag{8}$$

The amount of data that can be transmitted in different task queues at time slot t can be denoted by the following formula

$$Q_t^j = \begin{cases} \min\{Q_t - \sum_{i=0}^{j-1} Q_t^i, L_j(t)\}, & j \neq J \\ Q_t - \sum_{i=0}^{j-1} Q_t^i, & j = J \end{cases} \tag{9}$$

The total queue length at time slot t can be denoted by

$$L(t) = \max\{L(t-1) + A_t - Q_{t-1}, 0\}, L(0) = 0, Q_{-1} = 0. \tag{10}$$

The update formula of task queues at time slot t is

$$L_j(t) = \begin{cases} A_t^j + L_{j+1}(t-1) - Q_{t-1}^{j+1} + L_j^{t-1} - Q_{t-1}^j, & j = 0 \\ A_t^j + L_{j+1}(t-1) - Q_{t-1}^{j+1}, & else \\ A_t^j, & j = J \end{cases} \tag{11}$$

E. Problem Formulation

Different tasks have different needs for latency. Some need to be transmitted immediately, while others can wait for a while before being transmitted. Furthermore, the network overhead of LEO satellite is many times that of cellular BS. Waiting until the BS signal is received might save a lot of network overhead for tasks that do not require immediate transmission. Therefore, we must carefully analyze the above situation to achieve the cost-effective goal.

Through the above modeling, we can express the long-term cost optimization problem of vehicles in ISTN over a period of time as the following mathematical model.

$$\min_{a} \sum_{t=1}^{T} C_t \tag{12}$$

$$s.t. \quad 0 \le a_t \le 1, \qquad \forall t \in T \tag{12a}$$

$$Q_t \ge L_0(t), \qquad \forall t \in T \tag{12b}$$

$$R_t^B = B^B \log_2\left(1 + \frac{p_B h_t^B}{\sigma^2}\right), \qquad \forall t \in T \tag{12c}$$

$$R_t^S = \Lambda B^S \log_2\left(1 + \frac{p_S (h_t^S)^2}{\sigma^2}\right). \qquad \forall t \in T \tag{12d}$$

Constraint (12a) restricts the range of the offloading decision. Constraint (12b) restricts the offloading decision of each time slot to at least completing the transmission of tasks in $L_0(t)$. Constraints (12c) and (12d) are generation rules for the transmission rates of cellular BS and LEO satellite.

III. Data Offloading Strategy Based on DDPG

In this section, we propose a RL-based algorithm to solve the optimization problem (12). Considering the continuous action space of the problem, DDPG is exploited in our proposed scheme. State space, action space, and reward function are the three key elements of DDPG. According to the system model, we define these elements as follows.

A. State Space

State space contains all the information about the system that the environment can observe at time slot t, including queue state, BS channel state, and LEO satellite channel state. We denoted the state space at time slot t is defined as

$$s_t = \{L(t), R_t^B, R_t^S\}, \tag{13}$$

where $L(t) \triangleq [L_0(t), L_1(t), ..., L_J(t)]$, and it contains information of all task queues. R_t^B and R_t^S represent the channel rates of the vehicle to the BS and the LEO satellite, respectively.

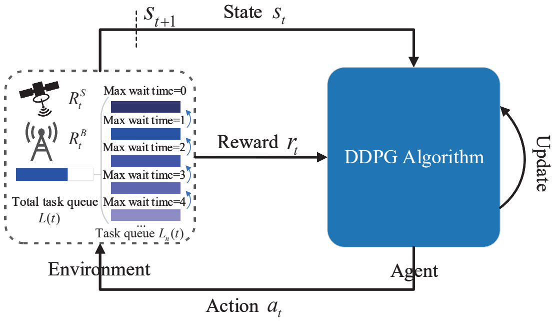


Fig. 4. DDPG algorithm flow for collaboratively data offloading.

B. Action Space

Based on the currently observed state s_t, the LEO satellite needs to make an offloading decision $a_t \in [0, 1]$ at each time slot t to determine how much data should be transmitted through the BS and the LEO satellite, respectively, in the current time slot.

C. Reward Function

For state s_t and action a_t at time slot t, there is a corresponding reward r_t. Our goal is to reduce the network overhead of the vehicle as much as possible while ensuring service quality, so we define the reward function as the following formula

$$r_t = -(\beta_1 R_t^B + \beta_2 R_t^S) + \beta_3 (Q_t - L_0(t)), \tag{14}$$

where the first item of the formula is the network overhead of the vehicle. Because the DDPG algorithm will maximize the reward function R_t as much as possible, we turn it into a negative number to minimize the overhead. The second is a penalty for not completing a task on time, where β_3 is the penalty factor.

D. Data Offloading Algorithm Based on DDPG

Fig. 4 shows the flow of the DDPG algorithm for collaborative data offloading. It can be seen that the LEO satellite acts as an agent to execute the DDPG algorithm. The LEO satellite must first obtain state information s_t from the environment, which includes the channel rates and queue information of the vehicle, and then give an action a_t by calculation. The next state s_{t+1} is obtained by applying the action a_t to the current environment s_t. At the same time, the environment also gives a reward r_t for the update of the DDPG algorithm.

We designed a data offloading algorithm based on DDPG as shown in Algorithm 1.

In steps 1-3. The DDPG network parameters are initialized. In steps 4-8. The LEO satellite selects an action a_t according to the current state s_t for each time slot t in each episode M. The offloading decision a_t acting on the environment will get the state of the next time slot s_{t+1} and the reward of the current action r_t . In steps 9-14. The transformation (s_t, a_t, r_t, s_{t+1}) is stored in the replay buffer $\mathcal{R}$, and the DDPG algorithm randomly samples a mini-batch of B transitions for the update of itself.

Algorithm 1 DDPG-based data offloading algorithm

Input: N, λ, μ, T and number of episodes M.

Initialize the critic network $Q(s, a \mid \theta^Q)$ and actor $\mu(s \mid \theta^\mu)$ with weights θ^Q and θ^μ.

Initialize the target network Q' and μ' with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$.

Initialize replay buffer $\mathcal{R}$ and mini-batch size B.

for $episode = 1, 2, ..., M$ **do**

  Generate a random initial state s_1.

  for $t = 1, ..., T$ **do**

    LEO satellite selects an action $a_t = \mu(s_t \mid \theta^\mu) + N_t$ according to the current policy and exploration noise as offloading decision.

    Execute action a_t and observe reward r_t according to (14) and the new state s_{t+1}.

    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{R}$.

    Randomly sample B transitions (s_t, a_t, r_t, s_{t+1}) from $\mathcal{R}$.

    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1} \mid \theta^{\mu'} \mid \theta^{Q'}))$.

    Update critic network by minimizing the loss of critic network $L(\theta^Q)$,

$$L(\theta^Q) = \frac{1}{B} \sum_{i=1}^{B} (y_i - Q(s_i, a_i \mid \theta^Q))^2.$$

    Update the actor network using the sampled policy gradient ascent by

$$\nabla_{\theta^\mu} J \approx \frac{1}{B} \sum_{i=1}^{B} \nabla_a Q(s_i, a_i \mid \theta^Q) \nabla_{\theta^\mu \mu(s_i|\theta^\mu)} .$$

    Update the target networks as:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}.$$

  end for

end for

Output: Optimal policy μ^* and offloading decisions a_t.

IV. Simulation Results

In this section, we evaluate the performance of the proposed DDPG-based data offloading algorithm. Values for each parameter are summarized in Table I.

We consider a remote area where a vehicle is driving. LEO satellites can provide ubiquitous connectivity for vehicles. There are several cellular BSs sparsely deployed on the ground. The sojourn time of the on and off state of the vehicle follows an exponential distribution with parameters $\lambda = 5$ and $\mu = 15$. The average task arrivals of A_t^i, $i \in \{0, 1, 2, 3, 4, 5\}$ are [0.1, 0.2, 0.3, 0.4, 0.5, 0.6], respectively. We set the time slot length as $\tau = 1$.

To verify the effectiveness of the proposed DDPG-based data offloading algorithm, we employ four baseline schemes for comparison.

TABLE I
Parameter Setting

Parameter	B^B	B^S	Λ	p^B	p^S	λ	μ
Value	5Mhz	3Mhz	0.8	1dbm	20dbm	5s	15s

1. Random proportional offloading: This offloading scheme randomly makes offloading decisions $a_t \in [0, 1]$ in each time slot, which is labeled as "Rand".

2. Fixed offloading: This offloading scheme makes a fixed offloading decision $a_t = 0.5$ each time slot, which is labeled as "Fixed".

3. Dynamic proportional offloading: This offloading scheme dynamically adjusts the offloading ratio according to the channel conditions of the BS and the LEO satellite, which is labeled as "Dynamic". And the data offloading decision is denoted as $a_t = R_t^B/(R_t^B + R_t^S)$.

4. Base station first: This offloading scheme will give priority to the BS to transmit tasks in the queue, and then use the LEO satellite to transmit when the BS cannot transmit all the tasks in the queue, which is labeled as "BS-first". This scheme can achieve the maximum throughput of ISTN.

Fig. 5 reveals the impact of the number of task types on cost saved of the DDPG algorithm. The definition of "cost saved" is the difference in network overhead between BS-first and the DDPG algorithm. The DDPG algorithm can save more network overhead when the number of task types increases. That is because the DDPG algorithm reduces the satellite usage for tasks that don't require immediate transmission. Fig. 6 shows that regardless of the types of tasks, the decision failure rate of the DDPG algorithm is close to 0, which shows the reliability of the proposed DDPG algorithm. The definition of the "fail rate" in this paper is the number of failed decisions divided by the total number of time slots. We consider it a failure if the offloading decision made in time slot t fails to transmit the data in $L_0(t)$. It's worth noting that we didn't consider the scenarios when the task types were 1 and more than 6. Because when the task types are 1, the offloading decision becomes deterministic, and the DDPG algorithm can't make cost-saving. And when the task types are greater than 6, task arrivals will exceed the network load, and the DDPG algorithm can't converge.

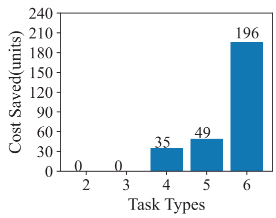


Fig. 5. Cost saved with task types.

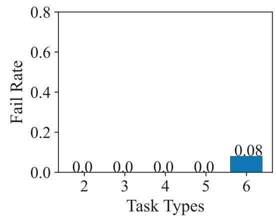
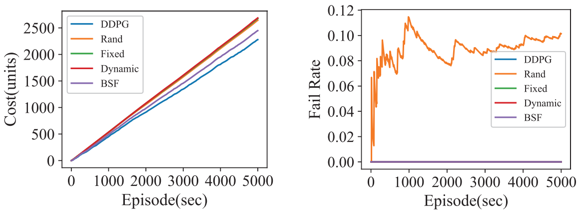


Fig. 6. Fail rate with task types.

Fig. 7 shows the changes in cost and fail rate with episode when task types are 5. In Fig. 7(b), all the algorithms can reach zero fail rate except "Rand". Combing Fig. 7(a) and Fig. 7(b), the DDPG-based data offloading algorithm can get the lowest network overhead with a zero fail rate compared with other data offloading schemes. This is because the DDPG algorithm can fully observe and benefit from the mobile state of the vehicle and the channel state. The BSF scheme prefers to transmit data via cellular BS and only uses the LEO satellite when the total tasks in the queue exceed the transmission capability of BS. As a result, BSF can also achieve a lower cost. Nonetheless, the network overhead of the BSF scheme is higher than that of the DDPG algorithm because it does not consider the difference in the delay requirements of tasks.



(a) Cost (b) Fail Rate

Fig. 7. Cost and fail rate with episode when task types are 5.

We also study the impact of user mobility on the DDPG-based data offloading algorithm. Fig. 8 shows the changes in the cost saved with λ and μ when the task types are 5. It can be seen that when λ and μ decrease, the DDPG algorithm can get more cost-saving. Because the frequency of vehicles moving in and out of the terrestrial network increases as λ and μ decrease, more delay-tolerant tasks can be transmitted via cellular BSs rather than the LEO satellites. Fig. 9 shows the changes in fail rate with λ and μ when the task types are 5. It can be seen that the fail rate decreases as λ increases and decreases as μ decreases.

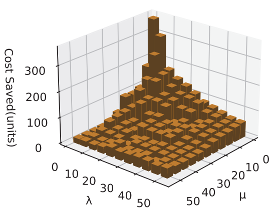


Fig. 8. Cost saved with λ and μ.

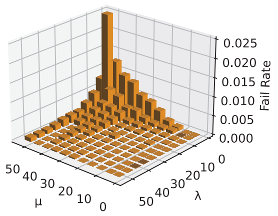


Fig. 9. Fail rate with λ and μ.

V. Conclusion

In this paper, we have investigated the vehicular data offloading in ISTN to reach the cost-effective decision addressing the trade-off between the network overhead and the task queuing. Based on the on-off mobility model of vehicles, the queuing model of task arrivals with different maximum waiting time is first established. To trade off the network overhead of vehicles and the requirement for task latency, a non-linear programming (NLP) problem is formulated to rationalize the use of LEO satellites. RL-based algorithm is carried out to solve this problem, and a DDPG-based data offloading decision approach with cost-effectiveness is reached.

Acknowledgement

This work was supported by the National Key Research and Development Program of China (2020YFB1807700), the National Natural Science Foundation of China (NSFC) under Grant No. 62071356, the Fundamental Research Funds for the Central Universities under Grant No. JB210113, the Fundamental Research Funds for the Central Universities of Ministry of Education of China under Grant XJS221501, and the National Natural Science Foundation of Shaanxi Province under Grant 2022JQ-602.

References

[1] D.-K. Choi, J.-H. Jung, S.-J. koh, J.-I. Kim, and J. Park, "In-vehicle infotainment management system in Internet-of-Things networks," *Proc. IEEE ICOIN*, pp. 88–92, Kuala Lumpur, Malaysia, 2019.

[2] T. Liu, L. Tang, W. Wang, X. He, Q. Chen, X. Zeng, and H. Jiang, "Resource Allocation in DT-assisted Internet of Vehicles via Edge Intelligent Cooperation," *IEEE Internet Things J.*, early access, doi: 10.1109/JIOT.2022.3156100.

[3] X. Zhu, S. Yuan, and P. Zhao, "Research and application on key technologies of 5G and C-V2X intelligent converged network based on MEC," *Proc. IEEE ICPECA*, pp. 175–179, Shenyang, China, 2021.

[4] W. Qi, B. Landfeldt, Q. Song, L. Guo, and A. Jamalipour, "Traffic Differentiated Clustering Routing in DSRC and C-V2X Hybrid Vehicular Networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7723–7734, 2020.

[5] G. Cui, Y. Long, L. Xu, and W. Wang, "Joint Offloading and Resource Allocation for Satellite Assisted Vehicle-to-Vehicle Communication," *IEEE Syst. J.*, vol. 15, no. 3, pp. 3958–3969, 2021.

[6] N. Cheng, F. Lyu, W. Quan, C. Zhou, H. He, W. Shi, and X. Shen, "Space/Aerial-Assisted Computing Offloading for IoT Applications: A Learning-Based Approach," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 5, pp. 1117–1129, 2019.

[7] Z. Yin, M. Jia, N. Cheng, W. Wang, F. Lyu, Q. Guo, and X. Shen, "UAV-Assisted Physical Layer Security in Multi-Beam Satellite-Enabled Vehicle Communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2739–2751, 2022, doi:10.1109/TITS.2021.3090017.

[8] X. Zhu and C. Jiang, "Integrated Satellite-Terrestrial Networks Toward 6G: Architectures, Applications, and Challenges," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 437–461, 2022.

[9] S. Yu, X. Gong, Q. Shi, X. Wang, and X. Chen, "EC-SAGINs: Edge-Computing-Enhanced Space–Air–Ground-Integrated Networks for Internet of Vehicles," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5742–5754, 2022.

[10] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. Zhuang, and X. S. Shen, "Software Defined Space-Air-Ground Integrated Vehicular Networks: Challenges and Solutions," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 101–109, 2017.

[11] H. Wu, J. Chen, C. Zhou, W. Shi, N. Cheng, W. Xu, W. Zhuang, and X. S. Shen, "Resource Management in Space-Air-Ground Integrated Vehicular Networks: SDN Control and AI Algorithm Design," *IEEE Wirel. Commun.*, vol. 27, no. 6, pp. 52–60, 2020.

[12] D. Wang, X. Tian, H. Cui, and Z. Liu, "Reinforcement Learning-based Joint Jask Offloading and Migration Schemes Optimization in Mobility-aware MEC Network," *CHINA Commun.*, vol. 17, no. 8, pp. 31–44, 2020.

[13] W. Abderrahim, O. Amin, M.-S. Alouini, and B. Shihada, "Latency-Aware Offloading in Integrated Satellite Terrestrial Networks," *IEEE OJ-COMS*, vol. 1, pp. 490–500, 2020.

[14] N. Cheng, N. Lu, N. Zhang, X. S. Shen, and J. W. Mark, "Opportunistic wifi offloading in vehicular environment: A queueing analysis," *Proc. IEEE GLOBECOM*, pp. 211–216, Austin, TX, USA, 2014.

[15] X. Chen, J. Hu, Z. Chen, B. Lin, N. Xiong, and G. Min, "A Reinforcement Learning-Empowered Feedback Control System for Industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 18, no. 4, pp. 2724–2733, 2022.