2024-01-22

Performance analytical modelling of mobile edge computing for mobile vehicular applications: a worst-case perspective

Miao, W

10.1109/TMC.2024.3356443
IEEE Transactions on Mobile Computing
Institute of Electrical and Electronics Engineers

Performance analytical modelling of mobile edge computing for mobile vehicular applications: a worst-case perspective

Wang Miao, Geyong Min, Zhengxin Yu and Xu Zhang

***Abstract*—Quantitative performance analysis plays a pivotal role in theoretically investigating the performance of Vehicular Edge Computing (VEC) systems. Although considerable research efforts have been devoted to VEC performance analysis, all of the existing analytical models were designed to derive the average system performance, paying insufficient attention to the worst-case performance analysis, which hinders the practical deployment of VEC systems to support mission-critical vehicular applications, such as collision avoidance. To bridge this gap, we develop an original performance analytical model by virtue of Stochastic Network Calculus (SNC) to investigate the worst-case end-to-end performance of VEC systems. Specifically, to capture the bursty feature of task generation, an innovative bivariate Markov Chain is firstly established and rigorously analysed to derive the stochastic task envelope. Then, an effective service curve is created to investigate the severe resource competition among vehicular applications. Driven by the stochastic task envelope and effective service curve, a closed-form end-to-end analytical model is derived to obtain the latency bound for VEC systems. Extensive simulation experiments are conducted to validate the accuracy of the proposed analytical model under different system configurations. Furthermore, we exploit the proposed analytical model as a cost-effective tool to investigate the resource allocation strategies in VEC systems.**

***Index Terms*—Vehicular Edge Computing, Performance Analysis, Markov Modulated Poisson Process, Stochastic Network Calculus**

I. Introduction

AUTONOMOUS driving vehicles rely on an array of advanced sensors, *e.g.,* cameras, sonars, radars, to accurately perceive the surrounding environment. These sensors generate vast quantities of content-rich data, essential for making critical and time-sensitive decisions in vehicular applications. As reported by IDC Global DataSphere Forecast [1], an autonomous driving vehicle can generate 5-20 TB data daily, depending upon the quality of videos captured and the accuracy of the sensors equipped. However, due to space and energy constraints, it is not feasible or cost-effective to install high-performance servers on vehicles to process this deluge of data. In this context, cloud computing has been widely adopted as a pivotal technology to liberate vehicles from the burden of computationally intensive data analysis [2] [3]. However, the inevitable high latency during network transmission hinders its practical adoption to underpin latency-sensitive vehicular applications. As an extension of cloud computing, Vehicular Edge Computing (VEC) is gradually attracting attention from both academia and industrial communities to provide ubiquitous computing resources for vehicular applications [4].

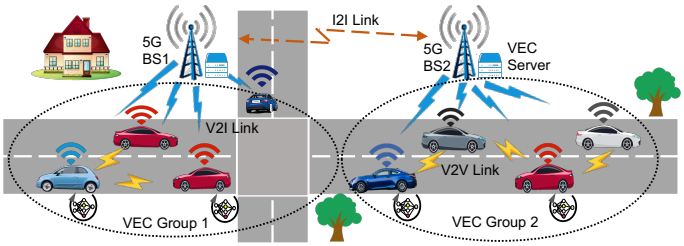


Fig. 1: Working scenario of VEC systems

As shown in Fig. 1, VEC migrates powerful computational resources from cloud data centres to the edge of networks in the vicinity of vehicles. By exploiting the Vehicle to Infrastructure (V2I) and Vehicle to Vehicle (V2V) communications, one on hand, VEC enables vehicles to offload the computation-intensive tasks to edge servers for execution. On the other hand, VEC provides the opportunity for vehicles to share computational and storage resources with their neighbouring vehicles to conduct activities that a single vehicle is not capable of. Through complementarily using the computation resources available across all vehicles and network edges, VEC brings unprecedented benefits for vehicular applications such as real-time data analytics, powerful computation capability, and agile service provisioning.

However, compared with the cloud computing architecture, VEC exhibits substantially different features, *e.g.,* limited resource deployment, time-varying channel conditions and highly dynamic vehicle mobility. Such features make it extremely challenging to achieve Quality-of-Service (QoS) guaranteed service provisioning for vehicular applications. For realising QoS-guaranteed service provisioning, one of the key obstacles in the way is how to accurately quantify the QoS metrics of VEC system operations. In this regard, considerable research efforts have been made to develop accurate analytical models to investigate the performance limits of VEC systems [5]–[11]. Although some interesting research results have been reported, all of the existing works mainly focused on deriving the average performance metrics of VEC systems, *e.g.,* average latency and average throughput, paying insufficient attention to the worst-case performance analysis. While, for mission-critical and delay-sensitive vehicular applications, any abnormal QoS provisioning, such as high latency incurred by unstable wireless transmission or constrained computation resources, would lead to potentially catastrophic results, such as incidents and casualty on public roads [12]. Therefore,

average latency analysis is far from sufficient for VEC systems to guarantee reliable service provisioning. It is imperative to develop new analytical models and methods to investigate worst-case performance for VEC systems. In this context, there is a growing interest in exploiting Stochastic Network Calculus (SNC) theory to investigate the cumulative queueing behaviour and derive the worst-case performance of networking and communication systems, such as Satellite Data Networks [13] and Cellular Networks [14] [15]. The key idea of SNC is to quantitatively reveal the relationship between the performance metric and the probability that this metric can be satisfied by the underlying systems. Despite being such a useful tool, exploiting the SNC to investigate the performance of VEC systems comes with a plethora of special challenges, including:

- Recent studies revealed that the vehicular tasks in VEC systems exhibit a high degree of burstiness. For instance, the work in [16] modelled the payload traffic exchanged among vehicles as bursty traffic. The work in [17] [18] claimed that the safety tasks, *e.g.,* road accidents, urban zone warning and collision avoidance, are highly bursty, which would affect the QoS of wireless channel transmission. Additionally, the work in [19] pointed out that the traffic for entertainment and comfort applications, *e.g.,* video, networking games and Internet access, has a significant degree of burstiness. However, to the best of our knowledge, the impact of bursty task generation on worst-case performance is still a mystery in theory for VEC systems.
- For a practical VEC system, the server deployed at vehicles should simultaneously provide the computation services for multiple vehicular applications and edge services will be shared by the tasks coming from multiple vehicles [9]. Although this kind of co-existence deployment strategy brings the benefits of higher resource utilisation and potential cost reduction, it poses a serious challenge to guarantee the QoS during service provisioning. In this regard, the impact of the co-existence of the service deployment on the worst-case performance guarantee of VEC service provisioning still remains unclear.
- Most of the theoretical findings in SNC were developed with the assumption that the buffer size of the queueing system is infinite [20] [21], and would fail to provide an effective solution for VEC systems with limited processing capabilities. Given the fact that it is impractical to deploy unlimited storage resources in VEC nodes, how to advance the theoretical foundation of SNC with limited buffer configuration has become one of the key roadblocks to developing a practical and useful performance analytical model in VEC systems.

To bridge these gaps, the novelty of this paper lies in making the first effort to develop a cost-effective analytical model using SNC to quantify the worst-case end-to-end latency bound for VEC systems. Specifically, we consider the stochastic arrival of bursty computation tasks generated by vehicular applications and develop a stochastic envelope function to stochastically model the cumulative behaviour of bursty task arrivals under limited buffer configurations. For the dynamic VEC service provisioning, we create a statistical service curve to model the dynamic service process of individual VEC nodes and derive their leftover service curve to characterise the amount of effective service resources assigned to the interested application in the presence of multiple competing ones. Driven by the developed stochastic envelope function and leftover service curves, a closed-form end-to-end latency bound is derived for VEC systems, where multi-vehicle cooperation and two-level computation service provisioning are jointly considered. The main contributions of this paper are summarised as follows,

- To characterise the stochastic feature of the vehicular task arrival, we developed a stochastic envelope function that obtains the upper bound of cumulative task arrivals in VEC systems. We adopt the Markov Modulated Poisson Process (MMPP) to model the bursty correlation among time-varying task arrivals.
- To facilitate the proposed analytical model applicable to the practical VEC systems with the limited buffer size, we established an innovative bivariate Markov Chain. By conducting a rigorous theoretical analysis of this model, we derived the steady-state distribution probability and effective stochastic envelope of task arrivals.
- To capture the dynamic service provisioning of VEC nodes, we developed a stochastic service curve that probabilistically provides the lower bound of service capability at VEC nodes, where the amount of the effective service provided to the interesting vehicular application is derived in the presence of multiple competing applications, paving the way for investigating the serious resource competition among vehicular applications.
- To analyse the worst-case performance analysis, we developed a new closed-form analytical model based on the upper bound of the task arrival and the lower bound of service provisioning. This model captures key features of VEC systems, including bursty task arrivals, multi-vehicle cooperation, two-level computation service provisioning, and limited buffer configurations.
- To assess the precision of the proposed analytical model, we conducted a series of simulation experiments with different system configurations. The simulation results demonstrate that the proposed analytical model can reliably predict the stochastic end-to-end latency bound of task offloading in VEC systems.

The rest of this paper is organised as follows, Section II introduces the related work of VEC analytical model development. Section III presents the overall VEC system model. Section IV derives the worst-case latency bound with bursty task arrival, dynamic service provisioning and limited buffer configurations. Section V evaluates the accuracy of the proposed analytical model, followed by Section VI, which concludes this study and provides future research directions.

II. Related Work

VEC has been regarded as a promising technology to support emerging delay-sensitive applications in autonomous driving, such as environment awareness and collision avoidance

[22]. To unleash the power of the VEC system, a tremendous amount of effort has been made to design accurate analytical models to investigate the performance of VEC systems. For instance, the authors in [5] created an analytical model to study the task offloading performance for mobile applications, which reveals that a partial offloading strategy could potentially outperform the traditional offloading algorithms to support delay-sensitive mobile services. Following [5], the authors in [6] employed the technology of stochastic geometry to examine the probability of service interruptions in mobile cloud systems, where tasks are offloaded to the radio access network for execution. Similar to [6], the authors in [7] also exploited the stochastic geometry modelling approach to analyse the success task offloading probabilities and approximated MEC computing load. The authors in [8] looked at another aspect of VEC performance modelling related to cooperation strategies among edge servers. Herein, the authors exploited queueing theory to design a new analytical model to study the efficiency of load sharing in MEC systems with respect to average processing latency and packet dropping probability. Meanwhile, the authors in [9] designed a multi-server queueing model based on Markov chains to investigate the performance limitation of MEC systems with constrained computation capabilities. Built upon this model, the minimum number of processors is determined to satisfy the pre-defined QoS requirements. Furthermore, the authors in [10] proved that the queueing behaviour of service-providing vehicles in VEC systems can be modelled as a $M/G/N/K$ system, which is utilised to derive the average end-to-end processing latency during task offloading and execution. In addition, the authors in [11] developed an analytical model to investigate the end-to-end processing latency of VEC systems, where a new priority-based task processing scheme is created to improve the overall processing performance of vehicular task offloading. Although some promising progress has been made with respect to VEC quantitative performance analysis, all of the existing research works mainly focus on investigating the average system performance and no efforts have been devoted to quantifying system performance from the perspective of the worst-case performance investigation. To fill this gap, the aim of this work is to develop a cost-effective analytical model with the aim of shedding light on how to theoretically quantify the worst-case performance of VEC systems, where the unique features of system operation, such as bursty task arrivals, limited processing capabilities and dynamic service provisioning, are taken into consideration.

III. VEC System Model

A VEC system, as presented in Fig. 2, is comprised of an edge server and a set of N_v vehicles, indexed by $\mathrm{N} = \{1, 2, ..., \mathrm{N_v}\}$. Each vehicle can host up to N_a intelligent applications. During ask offloading, the VEC system employs both horizontal and vertical offloading strategies. On one hand, vehicles create Vehicle-to-Vehicle (V2V) communication links with nearby vehicles and take advantage of their computational resources to enable horizontal task offloading. On the other hand, each vehicle establishes a cellular-based

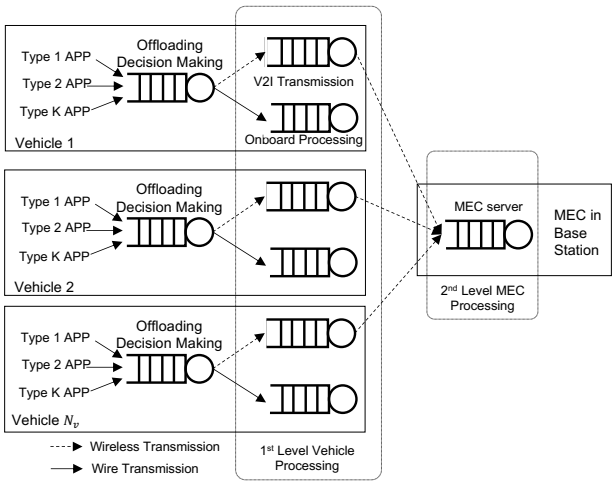


Fig. 2: Model abstraction for VEC systems

Vehicle-to-Infrastructure (V2I) communication link with the VEC server to perform vertical task offloading. Inspired by the work in [11], we exploit an abstracted VEC system model to support both the horizontal and vertical task offloading in the process of the analytical model development, which is formed by five component modules depicted in Fig. 2, Offloading Decision Making (ODM), Vehicle Processing (VP) server, V2I Transmission, V2V Transmission and Edge Server (ES). Specifically, the ODM module is responsible for making task offloading decisions for vehicular applications. VP and ES offer the computation resources for task execution at the vehicle and network edge servers. V2I and V2V modules are responsible for building the communication links and transmitting the computation tasks to VP and ES modules. The service rates of the five modules are represented by $\mu_{d,n}$, $\mu_{vp,n}$, $\mu_{ve,n}$, $\mu_{vv,n}$ and μ_{es}. To capture the limited-service capability of practical VEC systems, the buffer sizes of the ODM, VP, and ES modules are set to be limited, denoted as $K_{d,n}$, $K_{vp,n}$, and K_{es}, respectively. To facilitate the designed analytical model capable of investigating VEC performance with different offloading algorithms, we harness a parameter-based approach [23] [24] to employ the variables η, ϑ_i, and ξ to represent the probabilities of tasks being executed at the local server, the ith neighbour vehicle and the edge server. Herein, if the ith neighbour vehicle is not selected by the offloading algorithm, potentially because of hardware issues or unstable V2V link, the value of ϑ_i would be set to be zero in the analytical model and no tasks will be scheduled to the ith vehicle for processing. Otherwise, the ith vehicle will participate in the processes of the task offloading and will be in charge of processing the amount of ϑ_i tasks. Let Nc_n represent the number of the neighbouring vehicles involved in the horizontal task offloading. Then, ϑ_i could be calculated by a V2V channel-aware scheduling method presented in [11], expressed as $\vartheta_i = \vartheta \frac{1}{T_{vv,i}} / \sum_{j=0}^{Nc_n} \frac{1}{T_{vv,j}}$, where $T_{vv,i}$ denotes the transmission time of task offloading from the local vehicle to its ith neighbouring vehicle. Furthermore, it is worth noting that this work mainly concentrates on VEC performance analysis in 5G communication systems. By

tailoring parameter values to specific use cases and carrying out performance analysis, the proposed analytical model can be potentially applied to other communication protocols, including 4G and IEEE 802.11p. Meanwhile, keeping in line with the works in [2] [4] [25] [26] [27], this work focuses on interference-free channel modelling that exploits Orthogonal Frequency-Division Multiplexing Access (OFDMA) to assign transmission data into different frequency resource units for vehicular task transmissions. The choice of OFDMA is driven by its superior ability to combat multi-path interference with higher spectral efficiency and greater service robustness, which is critically important for VEC systems to support mission-critical vehicular applications. Then, the transmission rates of V2V and V2I for the vehicle n at the time t, are calculated as follows [26],

$$R_n^{vv/ve}(t) = B_n \log_2\left(1 + \frac{p l_n^{vv/ve}(t)^{-\varsigma} |h_n|^2}{\sigma^2}\right), \tag{1}$$

where B_n presents the bandwidth allocated to the vehicle n; p is the transmission power of the vehicle n; $l_n^{vv/ve}(t)$ represents the distance between the vehicle n and its neighbouring vehicle or edge server at the time t; ς is the path loss factor of V2V or V2I communication links; h_n denotes the antenna gain; and σ^2 presents the received noise power. Herein, because $l_n(t)$ is determined by the positions of vehicles and the MEC server, we exploit the vehicle mobility model of the work in [28] to update the position of the vehicle n at the time t, which is described as follows,

$$\begin{aligned} X_n(t+\tau) &= X_n(t) + \tau V_n(t) \\ V_n(t+\tau) &= V_n(t) + \tau \Omega_n(t), \end{aligned} \tag{2}$$

where $X_n(t+\tau)$, $V_n(t+\tau)$ and $\Omega_n(t)$ denote the position, velocity and acceleration of the vehicle n, respectively. Given the position of the MEC service as X_{es}, the distance of the vehicle n and the edge server can be directly obtained by,

$$l_n^{ve}(t+\tau) = \left|X_{es} - X_n(t+\tau)\right|. \tag{3}$$

Similarly, we can obtain the distance between the vehicle n and its neighbouring vehicles, $l_n^{vv}(t+\tau)$. Therefore, driven by the vehicle mobility model, we can evaluate the transmission capacities of both V2V and V2I communication links and analyse their worst-case transmission latency for task offloading.

Furthermore, to enhance the efficiency of task offloading, we employ a multicast transmission strategy to establish V2V communication for horizontal task offloading and an unicast strategy for V2I transmission to implement vertical task offloading. In the multicast transmission, the local vehicle will adopt a transmission rate to suit the neighbouring vehicle with minimal channel quality. A V2V multicast group is formed by a local vehicle i, and a set of neighbouring vehicles Υ_i^{vv} ($i \in \mathrm{N}, \Upsilon_{\mathrm{i}}^{\mathrm{vv}} \in \mathrm{N}$). Let $H_{i,j}^{vv}$ present the channel coefficient between the local vehicle i and the neighbour vehicle j ($j \in \Upsilon_i^{vv}$), which is defined as $H_{i,j}^{vv} = \frac{d_{i,j}^{-2}|h_{i,j}|^2}{\sigma^2}$. Then, the transmission rate of the multicast group, R_i^{vv}, is calculated by,

$$R_i^{vv} = B_i \log_2\left(1 + p_i^{vv} h_{gr,i}^{vv}\right), \tag{4}$$

where $h_{gr,i}^{vv}$ is the minimal channel transmission of the vehicles in the multicast transmission group, computed by $h_{gr,i}^{vv} = \min_{j \in \Upsilon_i^{vv}} H_{i,j}^{vv}$.

A. Preliminary of SNC theory

Different from the traditional queueing theory, which targets obtaining the average system performance, the objective of SNC is to offer the stochastic upper bound of performance analysis in the network system. Specifically, the task arrival and departure at VEC nodes are expressed by two stochastic processes, $A(t)$ and $D(t)$. Herein, $A(t)$ represents the cumulative task arrivals and $D(t)$ denotes the cumulative task departures during the time interval $[0, t]$. Clearly, $A(t)$ and $D(t)$ are non-decreasing and $A(0) = D(0) = 0$. In SNC, $A(t)$ and $D(t)$ are linked by the service process, $S(t)$, as follows,

$$D(t) \geq \min_{\tau \in [0,t]} \{A(\tau) + S(\tau, t)\}, \tag{5}$$

where $S(\tau, t)$ represents the accumulative service of the VEC node during the time interval $[\tau, t]$. Similar to $A(t)$ and $D(t)$, $S(t)$ is also nonnegative, nondecreasing and $S(0) = 0$. Keeping in line with the works in [8], we assume VEC nodes exploit the First-In-First-Out (FIFO) scheduling scheme to execute the computation tasks. In this case, the delay of the VEC nodes at time t is expressed as,

$$W(t) \leq \min\left\{\omega \geq 0 : \max_{\tau \in [0,t]} \{A(\tau, t) - S(\tau, t+\omega)\}\right\}. \tag{6}$$

Eq. (6) provides the approach to calculate the latency of the VEC component at a particular time t. The goal of this work is to obtain a stochastic latency bound, $d(t)$, to satisfy the following equation,

$$P(W(t) > d(t)) \leq \varepsilon(t), \tag{7}$$

where $\varepsilon(t)$ represents the probability that the end-to-end latency is larger than the stochastic latency bound. The objective of the analytical model to be presented in the following section is to obtain a quantitative relationship between the latency bound and the violation probability defined in Eq. (7). This relationship could be exploited by the VEC system to optimise the resource allocation strategy to achieve more reliable and latency-guaranteed service provisioning.

IV. VEC Performance Analytical Model

This section presents the methodology of how to derive the worst-case analytical model to quantitatively study the stochastic latency bound of VEC systems. Specifically, we first create a stochastic envelope function to stochastically model the task generation process of vehicular applications and establish a bivariate Markov Chain to derive the effective task arrival of VEC nodes with bursty task generation and limited buffer configuration. Secondly, we build a statistical service curve to model the service process of VEC nodes and derive the leftover service curve to investigate the serious resource competition among multiple vehicular applications. Thirdly, empowered by the stochastic envelope functions and statistical service curves developed in the previous two steps, we derive

TABLE I: Key Notations in the Performance Derivation

Notations	Definitions
$A_{n,k}(t)$, $S_{n,k}(t)$, $D_{n,k}(t)$	Cumulative traffic arrival, traffic departure and service provided at the VEC nodes during the time interval of $[0,t]$
$W(t)$	Delay experienced by the arrival traffic at the time t
$d(t)$	Stochastic latency bound at the time t
$\varepsilon(t)$	Violation probability requirement at the time t
N_v	Number of the vehicles connected to the VEC server
N_a	Number of the applications at the vehicle n
$\mu_{d,n}$, $K_{d,n}$	Service rate and buffer size of the ODM module at the vehicle n
μ_{es}, K_{es}	Service rate and buffer size of the ES module
$\mu_{vp,n}$, $K_{vp,n}$	Service rate and buffer size of the VP module at the vehicle n
η, ϑ, ξ	Probabilities that the tasks are executed at the local server, neighbour vehicles and edge server
$R_n^{vv/ve}$	Transmission rates of V2V and V2I links
B_n, p_n	Bandwidth and transmission power of the vehicle n
h_n	Channel fading coefficient of communication links
σ^2	Received noise power
$H_{i,j}^{vv}$	Channel coefficient between the local vehicle i and the neighbour vehicle j
$h_{gr,i}^{vv}$	Minimal transmission rate of the vehicles in the multicast transmission group
$Q_{n,k}$	State transmission matrix of the MMPP process for the application n on the vehicle n
$\Lambda_{n,k}$	Rate matrix of the MMPP process for the application n on the vehicle n
$\varphi_{1,n,k}$, $\varphi_{2,n,k}$	Transmission probability of continuous-time Markov Chain from the state "On/Off" to the state "Off/On"
$\lambda_{n,k}$	Task arrival rate when the continuous-time Markov Chain is in the state "On"
$\alpha_{n,k}(\theta,t)$	Effective bandwidth of $A_{n,k}(t)$
$M_{A,n,k}(\theta,t)$	Moment generation function of $A_{n,k}(t)$
$\rho_{n,k}$,	Slope of the affine envelop of $A_{n,k}(t)$
$\sigma_{n,k}$	Burst parameter of the affine envelop of $A_{n,k}(t)$
θ	Free parameter of $M_{A,n,k}(\theta,t)$
$\bar{A_{n,k}}(t)$	Essential supremum of $A_{n,k}(t)$
$\chi_{n,k}$	Peak rate of the task arrival
$\pi_{1,n,k}$, $\pi_{2,n,k}$	Steady-state vector of $MMPP_{tG,n,k}$
Pb_d	Task loss probability of the ODM module
$MMPP_{d,n}^{e\to in}$	Effective task arrival of ODM module at vehicle n
$\Pi_{d,n}$, $\Upsilon_{d,n}$	Steady-state probability matrix and Transmission rate matrix of the Markov Chain
$p_{k,s}$	Probability that Markov Chain is in the state (k,s)
$N_{d,n}^s(t-\tau)$	Number of the tasks served by the ODM module at the time interval $[\tau,t]$
ν	Computation task size
Nc_n	Number of the vehicles participating in the task offloading operation at vehicle n

the upper latency bound of VEC systems, where both the horizontal and vertical offloading are taken into consideration. Table I lists the primary notations and parameters exploited in the model development.

A. Stochastic envelope function of the busty task arrival

1) Stochastic bound of the task generation process of VEC application: The MMPP process is exploited in this work to model the time-varying task generation of vehicular applications. Our choice is inspired by the fact that the arrival rates of the vehicular task generation vary randomly over time, exhibiting a high degree of burstiness [16]–[19]. In this regard, MMPP has been extensively used for modelling bursty traffic due to its superior capability of modelling time-varying traffic arrival and capturing the bursty correlation between the inter-arrival times, while the model analysis remains tractable. Specifically, for the application k on the vehicle n, the computation tasks are modelled as a two-state MMPP process, denoted as $MMPP_{tG,n,k}$. It is a doubly stochastic Poisson process, the arrival rates of which are controlled by a finite state continuous-time Markov chain. $MMPP_{tG,n,k}$ is parameterised by an infinitesimal generator $Q_{n,k}$ and a rate matrix $\Lambda_{n,k}$, expressed as,

$$Q_{n,k}=\begin{bmatrix}-\varphi_{1,n,k} & \varphi_{1,n,k}\\ -\varphi_{2,n,k} & -\varphi_{2,n,k}\end{bmatrix} \quad \text{and} \quad \Lambda_{n,k}=\text{diag}\left(\lambda_{\text{n,k}},0\right), \tag{8}$$

where $\varphi_{1,n,k}$ is the transmission probability of continuous-time Markov Chain from the state "On" to the state "Off" and $\varphi_{2,n,k}$ is from the state "Off" to the state "On". When the Markov Chain is in the state of "On", the tasks arrive at the peak rate of $\lambda_{n,k}$, and no tasks arrive in the "Off" state.

Let $A_{n,k}(t,\tau)$ denote the accumulative task arrivals of $MMPP_{tG,n,k}$. To obtain a tight upper bound of $MMPP_{tG,n,k}$, we adopt an affine envelope function to stochastically model $A_{n,k}(t,\tau)$ [29], which is defined as follows,

$$P\left(A_{n,k}(t,\tau)>\rho_{n,k}^{a}(t,\tau)+\sigma_{n,k}^{a}\right)\leq\varepsilon_{n,k}^{a}(t,\tau), \tag{9}$$

where $\rho_{n,k}^{a}$ and $\sigma_{n,k}^{a}$ represent the slope and burstiness of the affine envelope function, respectively. $\varepsilon_{n,k}^{a}(t)$ is the violation probability that $A_{n,k}(t,\tau)$ is larger than a given upper latency bound of $\rho_{n,k}^{a}t+\sigma_{n,k}^{a}$. By employing Chernoff theory [30], the left side of Eq. (9) can be transformed to,

$$\begin{aligned}&P\left(A_{n,k}(\tau,t)>\rho_{n,k}^{a}(t-\tau)+\sigma_{n,k}^{a}\right)\\&=P\left(e^{A_{n,k}(\tau,t)}>e^{\rho_{n,k}^{a}(t-\tau)+\sigma_{n,k}^{a}}\right)\\&\leq\frac{E\left[e^{-\theta A_{n,k}(\tau,t)}\right]}{e^{\theta\left(\rho_{n,k}^{a}(t-\tau)+\sigma_{n,k}^{a}\right)}},\end{aligned} \tag{10}$$

where $E\left[e^{-\theta A_{n,k}(\tau,t)}\right]$ represents the Moment Generation Function (MGF) of $A_{n,k}(\tau,t)$, denoted as $M_{A,n,k}(\theta,t-\tau)$. In SNC, the normalised log of $M_{A,n,k}(\theta,t-\tau)$ is equal to the effective bandwidth of $A_{n,k}(\tau,t)$, expressed as,

$$\alpha_{n,k}(\theta,t)=\frac{ln\left(M_{A,n,k}(\theta,t-\tau)\right)}{\theta t}. \tag{11}$$

Based on the Property (iv) in [31], $\alpha_{n,k}(\theta,t)$ is also constrained by

$$\frac{E\left[A_{n,k}(t)\right]}{t}\leq\alpha_{n,k}(\theta,t)\leq\frac{\bar{A_{n,k}}(t)}{t}, \tag{12}$$

where $\bar{A_{n,k}}(t)$ is the essential supremum of $A_{n,k}(t)$ and computed by $\bar{A_{n,k}}(t)=\max\left\{x:P\left\{A_{n,k}(t)\geq x\right\}>0\right\}$. By applying the essential supremum of $MMPP_{tG,n,k}$ to the right side of Eq. (12), $\alpha_{n,k}(\theta,t)$ is obtained by,

$$\begin{aligned}&\alpha_{n,k}(\theta,t)=\\&\frac{1}{\theta t}\log\left\{\left(\left(\pi_{1,n,k},\pi_{2,n,k}\right)\exp\left[\begin{pmatrix}-\varphi_{1n,k}+\theta\chi_{n,k} & \varphi_{1n,k}\\ \varphi_{2n,k} & -\varphi_{2n,k}\end{pmatrix}\text{t}\right]\right)\begin{pmatrix}1\\1\end{pmatrix}\right\},\end{aligned} \tag{13}$$

where $\chi_{n,k}$ is the peak rate of the task arrival, calculated by $\chi_{n,k} = \lambda_{n,k}$. Meanwhile, $(\pi_{1,n,k}, \pi_{2,n,k})$ is steady-state vector of $MMPP_{tG,n,k}$, computed by,

$$(\pi_{1,n,k}, \pi_{2,n,k}) = \frac{1}{\varphi_{1n,k} + \varphi_{2n,k}} (\varphi_{1n,k}, \varphi_{2n,k}). \tag{14}$$

Building upon the effective bandwidth of $A_{n,k}(t)$, we can readily obtain the MGF of $MMPP_{tG,n,k}$ from Eq. (10) with the parameters of $\sigma^a_{n,k} = 0$ and

$$\rho^a_{n,k} = \frac{1}{2\theta} \left(\theta\chi_{n,k} - \varphi_{1n,k} - \varphi_{2n,k}\right) + \sqrt{\left(\theta\chi_{n,k} - \varphi_{1n,k} + \varphi_{2n,k}\right)^2 + 4\varphi_{1n,k}\varphi_{2n,k}}. \tag{15}$$

2) Effective task arrival bound of ODM model: In this subsection, we present the methodology for deriving the effective envelope function for the tasks generated by the application k at the vehicle n. As depicted in Fig. 2, let $MMPP_{tG,n,k}$ denote the tasks generated by the kth application. As the tasks generated by multiple vehicular applications merge at the queue of the ODM module, the task arriving process at the ODM module is the superposition of multiple MMPP processes. Previous work in [32] has shown that the superposition of multiple MMPP processes leads to a new MMPP process, denoted as $MMPP^{in}_{d,n}$. Similar to the definition of MMPP process in Eq. (8), $MMPP^{in}_{d,n}$ can be represented by an infinitesimal generator $Q^{in}_{d,n}$ and a rate matrix $\Lambda^{in}_{d,n}$. By exploiting the MMPP superposition property, $Q^{in}_{d,n}$ and $\Lambda^{in}_{d,n}$ are calculated by,

$$\begin{aligned} Q^{in}_{d,n} &= Q_{tG,n,1} \oplus Q_{tG,n,2} \oplus \cdots \oplus Q_{tG,n,K-2} \oplus Q_{tG,n,K} \\ \Lambda^{in}_{d,n} &= \Lambda_{tG,n,1} \oplus \Lambda_{tG,n,2} \oplus \cdots \oplus \Lambda_{tG,n,K-2} \oplus \Lambda_{tG,n,K}, \end{aligned} \tag{16}$$

where "$\oplus$" is Kronecker sum operation. For queueing systems, the operation of Kronecker sum would lead to a large number of the state spaces for $MMPP^{in}_{d,n}$, making it intractable to analyse the queueing behaviour of the VEC system. To deal with this problem, we exploit an MMPP approximation approach [33] to create a new two-state MMPP process, denoted as $\widetilde{MMPP^{e\to in}_{d,n}}$, to approximate the original $MMPP^{in}_{d,n}$. This approximation ensures that the average arrival rate and task burstiness of $\widetilde{MMPP^{e\to in}_{d,n}}$ are similar to those of $MMPP^{in}_{d,n}$. We then apply a statistical mapping approach to obtain a new two-state infinitesimal generator $\widetilde{Q^{in}_{d,n}}$ and a two-state rate matrix $\widetilde{\Lambda^{in}_{d,n}}$. It is important to note that this step of MMPP approximation is crucial for analysing the stable queueing behaviour of VEC systems, enabling us to avoid the difficulty of exploring an analytical solution that may not be feasible given the complexity of VEC systems.

Since the buffer size of ODM is limited, newly arrived tasks will be dropped if the buffer becomes full, as shown in Fig. 2. Let Pb_d denote the task loss probability at the queue of the ODM module. Then, the effective tasks entering the ODM module are a fraction $(1 - Pb_d)$ of the total tasks generated by vehicular applications. We use the MMPP splitting principle [33] to generate a new MMPP process, denoted as $MMPP^{e\to in}_{d,n}$, which represents the effective task arrival at the ODM module of vehicle n. According to the MMPP splitting principle [33], the infinitesimal generator, $Q^{e\to in}_{d,n}$, and the rate matrix, $\Lambda^{e\to in}_{d,n}$ are calculated by,

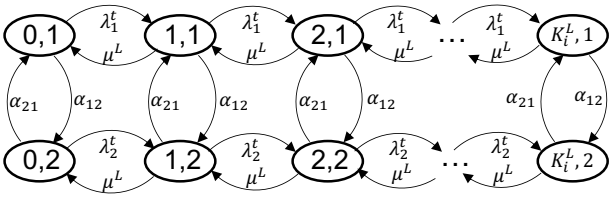


Fig. 3: Markov chain of the task offloading in the ith vehicle

$$Q^{e\to in}_{d,n} = \widetilde{Q^{in}_{d,n}} \quad \text{and} \quad \Lambda^{\mathrm{e\to in}}_{\mathrm{d,n}} = (1 - \mathrm{Pb_d})\,\widetilde{\Lambda^{\mathrm{in}}_{\mathrm{d,n}}}. \tag{17}$$

To derive the task loss probability of the ODM module in Eq. (17), we build a bivariate Markov Chain of $MMPP^{in}_{d,n}$ as shown in Fig. 3. Here, the state (k, s) represents that there are k tasks in the queue of the ODM module and the arriving process is at the state s. In the horizontal direction, λ^t_1 represents the transition rate from state (k, s) to state $(k+1, s)$, which are given by the rate matrix $\widetilde{\Lambda}^{in}_{d,n}$. The rate from state $(k + 1, s)$ to state (k, s) is $\mu_{d,n}$, which is the ODM module's service rate. In the vertical direction, α_{12} and α_{21} are the transmission rates between the state $(k, 1)$ and the state $(k, 2)$, obtained from the infinitesimal generator $\widetilde{Q}^{in}_{d,n}$.

Let $\Pi_{d,n}$ represent the steady-state probability matrix of the Markov Chain, and $p_{k,s}$ denote the probability that Markov Chain is in the state of (k, s). We define $P_k = (p_{k,1}, p_{k,2})$, then $\Pi_{d,n}$ is expressed as $\Pi_{d,n} = \left(P_1, P_2, ..., P_{K^l_n}\right)$. For a stable ODM queueing system with $\widetilde{\lambda^{in}_{d,n}} \le \mu_{d,n}$, we can build the equilibrium equations of the Markov Chain as follows,

$$\Pi_{d,n} \times \Upsilon_{d,n} = 0 \quad \text{and} \quad \Pi_{\mathrm{d,n}} \times \mathrm{e} = 1, \tag{18}$$

where $\Upsilon_{d,n}$ is the Markov Chain's transmission rate matrix. e is a unit vector with a length of $2K^l_n$. After solving Eq. (18), the steady-state probability matrix can be calculated by,

$$\Pi_{d,n} = u \times (I - \Phi + e \times u)^{-1}, \tag{19}$$

where $\Phi = I + \Upsilon_{d,n}/\min(\Upsilon_{d,n})$ and u is the first row of $\Upsilon_{d,n}$. Let p_k denote the probability that there are k tasks in the queue ODM module. Then, it is computed by,

$$p_k = \sum_{s=1}^{2} p_{k,s}. \tag{20}$$

As the newly arrived task will be dropped if it finds the queue of the ODM module full, the task loss probability is equal to the probability that the queue of ODM is full, denoted by $Pb_d = p_{K_{d,n}}$. After applying Pb_d to Eq. (17), we can obtain the effective infinitesimal generator $Q^{e\to in}_{d,n}$ and effective rate matrix $\Lambda^{e\to in}_{d,n}$. After inserting the parameters of $Q^{e\to in}_{d,n}$ and $\Lambda^{e\to in}_{d,n}$ into Eq. (15), the effective stochastic envelop of $A_{n,k}(t)$, $M_{A,n,k}(\theta, t)$, can be calculated by,

$$M_{A,n,k}(\theta, t) \le e^{t\left(\theta\zeta - \varphi^{e\to in}_{d,1,n} - \varphi^{e\to in}_{d,2,n} + \sqrt{\left(\zeta - \varphi^{e\to in}_{d,1,n} + \varphi^{e\to in}_{d,2,n}\right)^2 + 4\xi}\right)/2}, \tag{21}$$

where $\zeta = (1 - Pb_d)\,\lambda_{d,n}^{e\to in}$ and $\xi = \varphi_{d,1,n}^{e\to in}\varphi_{d,2,n}^{e\to in}$.

Recall that tasks from the ODM module will be sent to the local vehicle, neighbour vehicles or edge server for execution. Hence, the tasks arriving at these three destinations will be a fraction of the tasks departing from the ODM module, denoted as η, ϑ_i, and ξ. Let Pb_{v2i}, Pb_{es}, Pb_{v2v} and Pb_{vp} represent the task loss probabilities of V2I, ES, V2V, and VP modules, which can be obtained from Eqs. (16)-(21). Then, the effective arrival rates of V2I, ES, V2V, and VP modules can be calculated by,

$$\begin{aligned}
&\Lambda_{v2i,n}^{e\to in} = \eta\,(1 - Pb_d)\,(1 - Pb_{v2i})\,\Lambda_{d,n},\\
&\Lambda_{v2v,n,i}^{e\to in} = \vartheta_i\,(1 - Pb_d)\,(1 - Pb_{v2v})\,\Lambda_{d,n},\\
&\Lambda_{vp,n}^{e\to in} = \xi\,(1 - Pb_d)\,\left(1 - Pb_{vp}\right)\Lambda_{d,n},\\
&\Lambda_{es}^{e\to in} = \eta\,(1 - Pb_d)\,(1 - Pb_{v2i})\,(1 - Pb_{es})\,\Lambda_{d,n},\\
&Q_{v2i,n}^{e\to in} = Q_{es}^{e\to in} = Q_{v2v,n,i}^{e\to in} = Q_{vp,n}^{e\to in} = Q_{d,n}.
\end{aligned} \tag{22}$$

With the task loss probabilities, we can transform the VEC queueing systems with the limited buffer size into an equivalent system with unlimited buffer size. Then, the effective stochastic envelops of V2I, ES, V2V, and VP modules can be obtained by applying $\left(\Lambda_{v2i,n}^{e\to in}, Q_{v2i,n}^{e\to in}\right)$, $\left(\Lambda_{es,n}^{e\to in}, Q_{es,n}^{e\to in}\right)$, $\left(\Lambda_{v2v,n}^{e\to in}, Q_{v2v,n}^{e\to in}\right)$ and $\left(\Lambda_{vp,n}^{e\to in}, Q_{vp,n}^{e\to in}\right)$ into Eqs. (21) and (10).

B. Probabilistic service curve of the dynamic service provisioning at VEC nodes

This subsection aims to develop a service curve, which can probabilistically quantify the guaranteed service assigned to a task within a given time. Towards this aim, this subsection will establish the stochastic service curve for individual VEC nodes, derive the effective service curve assigned to interesting computation tasks in the presence of multiple competing computation tasks, and create a probabilistic service curve for VEC systems under dynamic service provisioning.

1) Stochastic service curve: Inspired by the work in [34], we exploit the Exponentially Bounded Fluctuation (EBF) model to express the stochastic service feature of the ODM module at vehicle n. It is defined as,

$$P\left(S_{d,n}(\tau,t) \le \rho_{d,n}^{s}(t-\tau) + b_{d,n}^{s}\right) \le \varepsilon\left(b_{d,n}^{s}\right), \tag{23}$$

where $\rho_{d,n}^{s}$ and $b_{d,n}^{s}$ are the slope and burstiness of the stochastic service curve. For an EBF model, the violation probability, $\varepsilon\left(b_{d,n}^{s}\right)$, is an exponential decay function, defined as $\varepsilon_s(b_s) = \alpha_s e^{-\theta b_s}$. By applying the Chernoff transformation, the left side of Eq. (23) can be transformed to,

$$\begin{aligned}
&P\left(S_{d,n}(\tau,t) \le \rho_{d,n}^{s}(t-\tau) + b_{d,n}^{s}\right)\\
&= P\left(e^{S_{d,n}(\tau,t)} \le e^{\rho_{d,n}^{s}(t-\tau)+b_{d,n}^{s}}\right)\\
&\le \frac{E\left[e^{-\theta S_{d,n}(\tau,t)}\right]}{e^{-\theta\left(\rho_{d,n}^{s}(t-\tau)+b_{d,n}^{s}\right)}},
\end{aligned} \tag{24}$$

where $E\left[e^{S_{d,n}(\tau,t)}\right]$ represents the MGF of $S_{d,n}(\tau,t)$, denoted as $M_{d,n}^{s}(-\theta, t-\tau)$. Different from $M_{d,n}^{a}(\theta, t-\tau)$, which offers the upper bound of the cumulative task arrivals, $M_{d,n}^{s}(-\theta, t-\tau)$ provides the lower bound of the cumulative service provision. Therefore, it is defined as the function of the parameter of $-\theta$ and is calculated by,

$$M_{d,n}^{s}(-\theta, t-\tau) \le e^{-\theta\left(\rho_{d,n}^{s}(t-\tau)+\gamma_{d,n}^{s}\right)}. \tag{25}$$

In the following, we will derive the parameters on the left side of Eq. (25). Let $N_{d,n}^{s}(t-\tau)$ denote the number of the tasks served by the ODM module over the time interval $[\tau, t]$. For the exponentially stochastic process, the probability distribution of $N_{d,n}^{s}(t-\tau)$ is given by [35],

$$P\left[N_{d,n}^{s}(t-\tau) = k\right] = e^{-\mu_{d,n}}\left[-\mu_{d,n}(t-\tau)\right]^{k}/k!. \tag{26}$$

Then the MGF of $N_{d,n}^{s}$, $M_{d,n}^{N}(-\theta, t-\tau)$, is calculated by

$$\begin{aligned}
M_{d,n}^{N}(-\theta, t-\tau) &= E\left[e^{-\theta N_{d,n}^{s}(t-\tau)}\right]\\
&= \sum_{k=0}^{\infty}\frac{e^{-\theta k}e^{-\mu_{d,n}(t-\tau)}\left[\mu_{d,n}(t-\tau)\right]^{k}}{k!}\\
&= \exp\left(\mu_{\mathrm{d,n}}(\mathrm{t}-\tau)\left(\mathrm{e}^{-\theta}-1\right)\right).
\end{aligned} \tag{27}$$

Given the task size is ν, the cumulative service process holds the relationship of $S_{d,n}(t-\tau) = \nu N_{d,n}^{s}(t-\tau)$. Then, the MGF of $S_{d,n}(t-\tau)$, can be obtained from Eq. (24) with $\gamma_{d,n}^{s} = 0$ and $\rho_{d,n}^{s} = \nu\mu_{d,n}(t-\tau)\left(e^{-\theta}-1\right)$.

2) Leftover service curve: To improve the resource utilisation of VEC systems, the computation nodes are usually shared by the tasks generated by different vehicular applications. However, this results in serious resource competition among computation tasks, making it difficult to guarantee the QoS for mission-critical vehicular applications. In this subsection, we focus on determining the amount of service that is left for the computation tasks of the interesting application where multiple competing vehicular applications exist. Recall that the cumulative task arrival, $A_{d,n}^{e\to in}$, and the cumulative task departure, $D_{d,n}^{out}(\tau,t)$, holds the relationship of

$$D_{d,n}^{out}(\tau,t) = \min_{\tau\in[0,t]}\left\{A_{d,n}^{e\to in}(\tau) + S_{d,n}(\tau,t)\right\}. \tag{28}$$

The right side of Eq. (28) obtains its minimal value when τ takes the beginning of the last busy period, $\tau*$. To derive the effective service curve, we divide $A_{d,n}^{e\to in}(\tau*)$ into three major parts: 1) the cumulative tasks that were successfully received by the ES module, the local vehicle server and the ith neighbour vehicle server, denoted as $A_{d,n,l}^{it->es}(\tau)$, $A_{d,n,l}^{it->ls}(\tau*)$ and $A_{d,n,i}^{it->ns}(\tau*)$, respectively; 2) the total cumulative tasks dropped in VEC systems from the interesting application l, denoted as $A_{d,n,l}^{it->dr}(\tau*)$; and 3) the total cumulative tasks generated by the competing application k, represented by $A_{d,n,k}^{cp}(\tau*)$. Then, $A_{d,n}^{e\to in}$ is expressed as,

$$\begin{aligned}
A_{d,n}^{e\to in}(\tau*) = A_{d,n,l}^{it->es}(\tau*) + A_{d,n,l}^{it->ls}(\tau*) + \prod_{i=1}^{Nc_n}A_{d,n,i}^{it->ns}(\tau*) +\\
A_{d,n,l}^{it->dr}(\tau*) + \sum_{k=1,k\neq l}^{N_a}A_{d,n,k}^{cp}(\tau*),
\end{aligned} \tag{29}$$

Meanwhile, the cumulative task departure, $D_{d,n}^{out}(\tau,t)$, can also be divided into,

$$D_{d,n}^{out}(\tau,t) = D_{d,n,l}^{it->es}(\tau,t) + D_{d,n,l}^{it->ls}(\tau,t) + \prod_{i=1}^{Nc_n} D_{d,n,i}^{it->ns}(\tau,t) + D_{d,n,l}^{it->dr}(\tau,t) + \sum_{k=1,k\neq l}^{N_a} D_{d,n,k}^{cp}(\tau,t). \tag{30}$$

By applying Eqs. (29-30) to Eq. (28), it can be rewritten as,

$$\begin{aligned} D_{d,n,l}^{it->es}(t) = {} & A_{d,n,l}^{it->es}(\tau*) + S_{d,n}(\tau*,t) \\ & - \left\{ D_{d,n,l}^{it->ls}(t) - A_{d,n,l}^{it->ls}(\tau*) \right\} \\ & - \left\{ D_{d,n,l}^{it->dr}(t) - A_{d,n,l}^{it->dr}(\tau*) \right\} \\ & - \sum_{i=1}^{Nc_n} \left\{ D_{d,n,i}^{it->ns}(t) - A_{d,n,i}^{it->ns}(\tau*) \right\} \\ & - \sum_{k=1,k\neq l}^{N_a} \left\{ D_{d,n,k}^{cp}(t) - A_{d,n,k}^{cp}(\tau*) \right\}. \end{aligned} \tag{31}$$

Because the total tasks departing from the ODL module are smaller than the effective tasks arriving at the ODM module, we can readily obtain $D_{d,n,l}^{it->ls}(t) \leq A_{d,n,l}^{it->ls}(t)$, $D_{d,n,i}^{it->ns}(t) \leq A_{d,n,i}^{it->ns}(t)$, $D_{d,n,l}^{it->dr}(t) \leq A_{d,n,l}^{it->dr}(t)$, and $D_{d,n,k}^{cp}(t) \leq A_{d,n,k}^{cp}(t)$. Then, it holds that

$$\begin{aligned} D_{d,n,l}^{it->es}(t) \geq {} & A_{d,n,l}^{it->es}(\tau*) + S_{d,n}(\tau*,t) - A_{d,n,l}^{it->ls}(\tau*,t) - \\ & A_{d,n,l}^{it->dr}(\tau*,t) - \sum_{i=1}^{Nc_n} A_{d,n,i}^{it->ns}(\tau*,t) - \sum_{k=1,k\neq l}^{N_n} A_{d,n,k}^{cp}(\tau*,t). \end{aligned} \tag{32}$$

Let $S_{d,n,l}^{it->es}(\tau,t)$ represent the effective service provided to the tasks successfully received by the ES module. By applying the relationship between the cumulative task arrivals and cumulative task departures to Eq. (32), $S_{d,n,l}^{it->es}(\tau,t)$ is expressed as,

$$\begin{aligned} S_{d,n,l}^{it->es}(\tau,t) = {} & S_{d,n}(\tau*,t) - A_{d,n,l}^{it->ls}(\tau*,t) - A_{d,n,l}^{it->dr}(\tau*,t) - \\ & \sum_{i=1}^{Nc_n} A_{d,n,i}^{it->ns}(\tau*,t) - \sum_{k=1,k\neq l}^{N_n} A_{d,n,k}^{cp}(\tau*,t). \end{aligned} \tag{33}$$

Then, the MGF of $S_{d,n,l}^{it->es}(\tau,t)$ can be calculated by,

$$\begin{aligned} & M_{d,n,l}^{S->es}(-\theta,t-\tau) = E\left[e^{-\theta S_{d,n,l}^{it->es}(\tau,t)} \right] \\ & = M_{d,n}^{S}(-\theta,t-\tau)\, M_{d,n,l}^{A^{it->ls}}(\theta,t-\tau)\, M_{d,n,l}^{A^{it->ns}}(\theta,t-\tau) \\ & \prod_{i=1}^{Nc_n} M_{d,n,i}^{A^{it->ns}}(\theta,t-\tau) \prod_{k=1,k\neq l}^{N_n} M_{d,n,i}^{A^{cp}}(\theta,t-\tau), \end{aligned} \tag{34}$$

where the MGFs of $A_{d,n,l}^{it->ls}(\tau)$, $A_{d,n,i}^{it->ns}(\tau)$, $A_{d,n,l}^{it->dr}(\tau)$ and $A_{d,n,k}^{cp}(\tau,t)$ are obtained from Eq. (19) with parameters of $\rho_{d,n,l}^{it->ls}$, $\rho_{d,n,i}^{it->ns}$, $\rho_{d,n,l}^{it->dr}$ and $\rho_{d,n,k}^{cp}$, and the MGF of $S_{d,n}(t)$ is obtained from Eq. (24) with the parameter of $\rho_{d,n}$. Then, the MGF of the effective service offered to the tasks successfully received by the ES module is calculated by,

$$\begin{aligned} & E\left[e^{-\theta S_{d,n,l}^{it->es}(\tau,t)} \right] \leq \\ & e^{-\theta\left[\rho_{d,n} - \rho_{d,n,l}^{it->ls} - \sum_{i=1}^{Nc_n} \rho_{d,n,i}^{it->ns} - \rho_{d,n,l}^{it->dr} - \sum_{k=1,k\neq l}^{N_n} \rho_{d,n,k}^{cp}\right](t-\tau)}. \end{aligned} \tag{35}$$

With $M_{d,n,l}^{S->es}(-\theta,t-\tau)$, the leftover service curve of $S_{d,n,l}^{it->es}(\tau,t)$ is calculated by Eq. (24) with the parameters of,

$$\begin{aligned} & \rho_{d,n,l}^{it->es} = \rho_{d,n} - \rho_{D,l,n}^{it->ls} - \sum_{i=1}^{Nc_n} \rho_{d,n,i}^{it->ns} - \rho_{d,n,l}^{it->dr} - \sum_{k=1,k\neq l}^{N_n} \rho_{d,n,k}^{cp} \\ & \gamma_{d,n,l}^{it->es} = 0. \end{aligned} \tag{36}$$

By exploiting the similar process of the leftover service curve derivation in Eqs. (29)-(36), we can calculate MGF functions and leftover service curves for V2V, V2I, VP and ES modules, characterised by the parameters of $\left(\rho_{v2v,n,l}^{it->es}, \gamma_{v2v,n,l}^{it->es}\right)$, $\left(\rho_{v2i,n,l}^{it->es}, \gamma_{v2i,n,l}^{it->es}\right)$, $\left(\rho_{vp,n,l}^{it->es}, \gamma_{vp,n,l}^{it->es}\right)$, $\left(\rho_{es,n,l}^{it->es}, \gamma_{es,n,l}^{it}\right)$, which are obtained from Eq. (36). Meanwhile, let $S_{net}(t)$ denote the end-to-end service process of VEC systems. By leveraging the associativity of min-plus convolution, $S_{net}(t)$ can be calculated by convoluting the individual service process of VEC nodes as follows,

$$S_{net}(t) = S_{d,n,l}^{it->es}(t) \otimes S_{v2i,n,l}^{it->es}(t) \otimes S_{es,n,l}^{it->es}(t). \tag{37}$$

Let $\left(\rho_{net,n,l}^{it->es}, \gamma_{net,n,l}^{it->es}\right)$ denote the slope and burstiness of the stochastic service curve of $S_{net}(t)$. Given $\gamma_{d,n,l}^{it->es} = 0$, the min-plus convolution operation in Eq. (37) results in $\gamma_{net,n,l}^{it->es} = 0$ and $\rho_{net,n,l}^{it->es}$ is calculated by $\rho_{net,n,l}^{it->es} = \min\left\{ \rho_{d,n,l}^{it->es}, \rho_{v2i,n,l}^{it->es}, \rho_{es,n,l}^{it} \right\}$. With $\left(\rho_{net,n,l}^{it->es}, \gamma_{net,n,l}^{it->es}\right)$, we can readily obtain the stochastic service curve for the end-to-end service process of VEC system from Eqs. (24-25).

C. Statistical latency bound of VEC systems

Built upon the stochastic envelope of the traffic arrival and stochastic service curve of the service process, this subsection derives the statistical latency bound of the VEC system in an end-to-end manner. Let $W_{net}(t)$ denote the end-to-end latency of the tasks processed at the VEC server, and d denote the latency bound of $W_{net}(t)$. If $W_{net}(t) > d$, it implies that the cumulative tasks successfully received by the VEC server over the time interval of $[0, t-d]$, $A_{es}^{e\to in}$, is larger than that of the departure process over the time interval of $[0,t]$, $D_{es}(t)$. Thus, it holds that

$$P\left(W_{net}(t) > d\right) = P\left(A_{es}^{e\to in}(t-d) > D_{es}(t)\right). \tag{38}$$

We assume VEC systems operate in a non-saturated state, which is a common assumption in most SNC studies [34] [36]. For a specific value of σ, the end-to-end service process, $S_{net}(t)$, satisfies the condition of $D_{es}(t) \geq \min_{\tau\in[0,t]}\left\{ A_{es}^{e\to in}(\tau) + [S_{net}(\tau,t) - \sigma]_+ \right\}$, where $[x]_+ = \max\{x, 0\}$. Then, the stochastic latency bound can be expressed as:

$$P\left(W_{net}\left(t\right)>d\right)\leq\varepsilon\left(\sigma\right)_{s}+$$
$$P\left(A_{es}^{e\rightarrow in}\left(t-d\right)>\min_{\tau\in[0,t]}\left\{A_{es}^{e\rightarrow in}\left(\tau\right)+\left[S_{net}\left(\tau,t\right)-\sigma\right]_{+}\right\}\right). \tag{39}$$

It is worth noting that the first item of the right side of Eq. (39) represents the probability that $S_{net}(\tau,t)$ does not satisfy the constraint of $D_{es}(t) \geq \min_{\tau\in[0,t]}\left\{A_{es}^{e\rightarrow in}(\tau)+\left[S_{net}(\tau,t)-\sigma\right]_{+}\right\}$, while the second item is the probability that the end-to-end latency is smaller than the pre-defined latency bound d.

Finally, by exploiting the stochastic envelope of $A_{es}^{e\rightarrow in}(\tau)$ with the parameters of $\left(\rho_{es,n,k}^{e\rightarrow in},\sigma_{es,n,k}^{e\rightarrow in}\right)$, and stochastic service curve of $S_{net}(\tau)$, with the parameters of $\left(\rho_{net,n,l}^{it->es},\gamma_{net,n,l}^{it->es}\right)$, the stochastic latency bound is calculated by,

$$\begin{aligned}
&P\left(W_{net}\left(t\right)>d\right)\\
&\leq\varepsilon\left(\sigma\right)_{s}+\sum_{\tau=0}^{t-d}\frac{E\left[e^{\theta\left\{A_{es}^{e\rightarrow in}(\tau,t-d)-S_{net}(\tau,t)\right\}}\right]}{e^{-\theta\sigma}}\\
&=\varepsilon\left(\sigma\right)_{s}+\sum_{\tau=0}^{t-d}\frac{e^{\theta\left[\rho_{es,n,k}^{e\rightarrow in}(t-d-\tau)+\sigma_{es,n,k}^{e\rightarrow in}\right]}}{e^{-\theta\sigma}e^{-\theta\left[\rho_{net,n,l}^{it->es}(t-d-\tau)+\gamma_{net,n,l}^{it->es}\right]}}\\
&=\varepsilon\left(\sigma\right)_{s}+\frac{e^{-\theta\left[\rho_{net,n,l}^{it->es}-\sigma\right]}}{\theta\left(\rho_{net,n,l}^{it->es}-\rho_{es,n,k}^{e\rightarrow in}\right)}.
\end{aligned} \tag{40}$$

V. Model Validation

In this section, we first present the parameter configuration for the performance evaluation. Next, we validate the accuracy of the proposed analytical model under diverse system configurations. Finally, we employ the proposed analytical model as a useful tool to shed light on improving resource allocation strategies in VEC systems.

A. Parameter configurations

To evaluate the effectiveness of the proposed analytical model, a VEC simulator was built based on an open-source Objective Modular Network Testbed in C++ (Omnet++) [37]. We consider a two-lane two-way road scenario with the size of 1000m*8m, where a Base Station (BS) resides in the middle of the road and has a coverage range of 2 km. $N_v = \{4-44\}$ vehicles are randomly scattered over the road. The speeds of the vehicles range from 15 km/h to 120 km/h. The channel bandwidth of BS is set to be 5 MHz. The transmission power is set to be $\{10, 100, 1000\}$ mWatts and the noise level is -100 dBm. For computation tasks, we consider object recognition in [38], where the data size ranges from 2M to 12M. The processing rates of ODM, VEC servers and onboard servers are 80 Mbps, 100 Mbps and 320 Mbps, respectively [39]. The transmission rates of V2V/V2I wireless channels are calculated from Eq. (4). The task arrival rates are set from 0 to 48 with an interval of 10% of the ODM service rate. Multiple violation error settings $\epsilon = \left\{10^{-1}-10^{-7}\right\}$ are adopted in the simulation experiments to reflect different reliability requirements. The buffer sizes of the VEC components are assigned from the set of $\{4, 8, 16, 32, 64, 128\}$. The task offloading probabilities are set from 0 to 1 with an interval of 0.1. Keeping in line with [8], we adopt a 95% confidence criterion to determine whether the VEC system reaches the steady state before any data collection. The data is collected and averaged for each simulation configuration through 10 simulation runs, each of which generates 10^9 computation tasks.

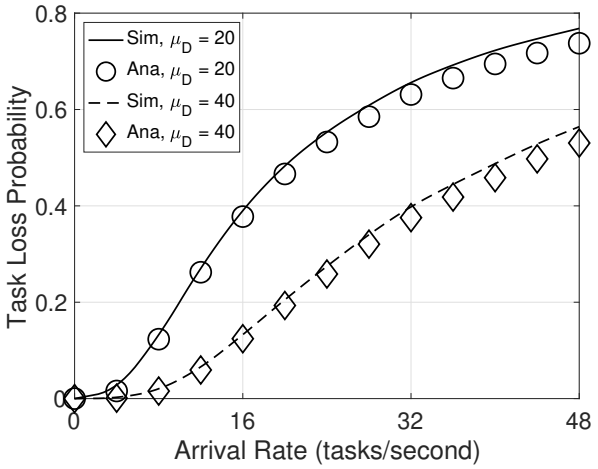
B. Model validation

In this subsection, the performance of the proposed model is evaluated with respect to the stochastic latency bound and packet loss probability by varying task arrival rates, service capabilities, task burstiness, vehicle speeds, violation error requirements and task offloading probabilities.

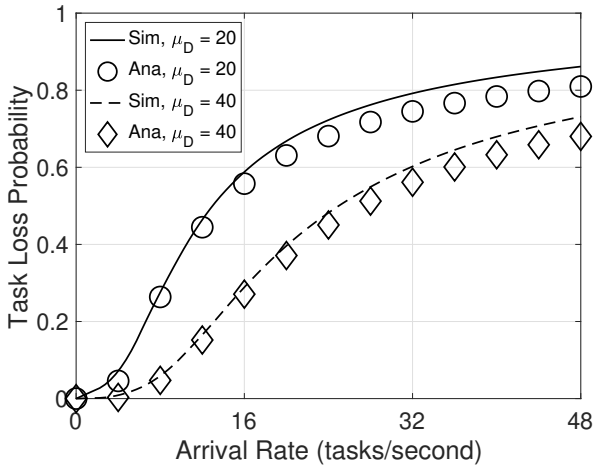
1) Prediction accuracy of the task loss probability by varying the task arrival rates: Fig. 4 presents the results of the task loss probability derived from the proposed analytical model plotted against those obtained from the simulation experiments with different task arrival rates. The horizontal axis of Fig. 4 represents the traffic arrival rate, $\lambda_{n,k}$, and the vertical axis denotes the task loss probability of VEC systems. It can be seen that the task loss probability obtained by the analytical model holds a reasonable degree of prediction accuracy, ranging from 93-99%. Meanwhile, when the VEC components operate in a steady state, where the average rate of task arrivals is less than the service capacity, the task loss probability increases with the task arrival rates. Once the VEC components become overloaded, the task loss probability approaches a limit value, which means that newly arriving tasks will be dropped and the QoS of service provisioning will be significantly affected.

2) Prediction accuracy of the statistical latency bound by varying the violation error requirements: Fig. 5 presents the statistical latency bounds obtained by the analytical model and those from the simulation experiments as the violation error requirements vary from 10^{-7} to 10^{-1}. In Fig. 5, the horizontal axis indicates the violation error requirements, while the vertical axis illustrates the statistical latency bound. From Fig. 5, we can observe that the end-to-end latency bounds obtained by the analytical model exhibit a reasonable degree of matching with the simulation experiment results, with prediction errors ranging from 8-17%. Compared to the SNC-enabled system performance analytical models reported in [40] [41], which suffer from 15-30% prediction errors, the 8-17% error range is superior. It is worth noting that the prediction error originates from the approximation operations employed in the model derivation, such as the MMPP approximation in Subsection IV.A and the MGF calculation in Eq. (35). Without these approximations, obtaining a closed-form upper latency bound would be intractable.

3) Prediction accuracy of the proposed analytical model by varying the task burstiness: This subsection aims to evaluate the prediction accuracy of the proposed model by varying the burstiness of the task arrivals. Specifically, for a given task arrival process, *e.g.,* $MMPP_{tG,n,k}$, the burstiness

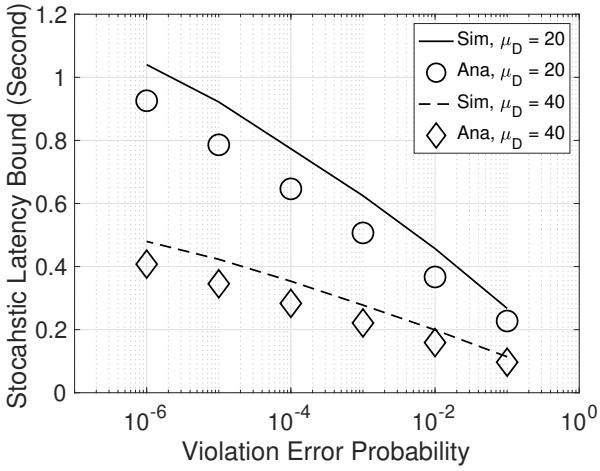


(a) $L_{vS} = 8$

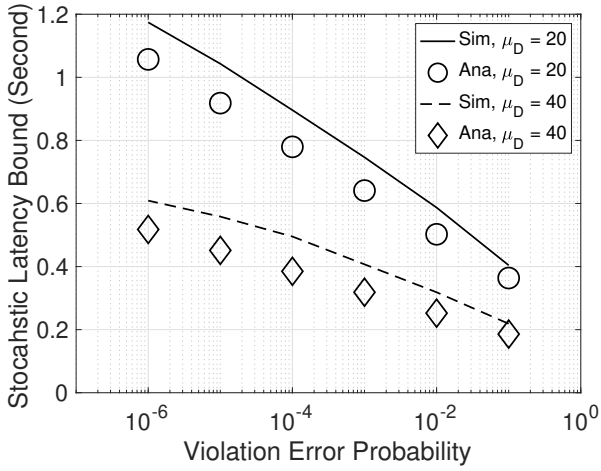


(b) $L_{vS} = 16$

Fig. 4: Packet loss probabilities predicted by the analytical model and obtained from the simulation experiments with $\varphi_{1,n,k} = 0.25$, $\varphi_{2,n,k} = 0.35$, $\lambda_{n,k} = \{0-48\}$, $L = 2$, $p = 100$, $Nc_n = 8$, $N_a = 4$, $\mu_{d,n}$=$\{20,40\}$, $\mu_{vp,n} = 50$, $\mu_{es} = 160$, $V = 15$, $\eta = 0.2$, $\vartheta = 0.35$, $\xi = 0.45$, $K_{d,n} = 4$, $K_{vp,n} = 16$, $K_{es} = 8$.



(a) $L_{vS} = 8$



(b) $L_{vS} = 16$

Fig. 5: Stochastic latency bounds predicted by the analytical model and obtained from the simulation experiments with $\varphi_{1,n,k} = 0.25$, $\varphi_{2,n,k} = 0.35$, $\lambda_{n,k} = 24$, $Nc_n = 8$, $N_a = 4$, $\mu_{d,n}$=$\{20,40\}$, $\mu_{vp,n} = 50$, $\mu_{es} = 160$, $V = 15$, $\eta = 0.2$, $\vartheta = 0.35$, $\xi = 0.45$, $K_{d,n} = 4$, $K_{vp,n} = 16$, $K_{es} = 8$.

of $MMPP_{tG,n,k}$ is measured by the Squared Coefficient of Variation, calculated by $C^2_{n,k} = 1 + \frac{2\lambda_{1n,k}\varphi_{1n,k}}{(\varphi_{1n,k}+\varphi_{2n,k})^2}$. In the simulation experiment, we change task generation burstiness while keeping the average rate constant by adjusting the parameters of $Q^{in}_{d,n}$ and $\Lambda^{in}_{d,n}$. This setting aims to study the relationship between traffic burstiness and the upper latency bound without being influenced by task arrival rates. The simulation and analytical results are shown in Table. II, demonstrating that the proposed analytical model provides an accurate prediction of the statistical latency bounds with simulation experiments under different burstiness levels of task arrivals.

4) Prediction accuracy of the proposed analytical model by task offloading probabilities: Efficient offloading decision-making plays a critical role for VEC systems to support

TABLE II: Stochastic latency bound by varying the task burstiness

					d (second)	
C^2_S	$\bar{\lambda}$	λ_1	φ_1	φ_2	Sim	Ana
2	32	32.5	0.0031	0.2	0.79235245	0.719621139
5	32	34.0	0.0125	0.2	0.81321485	0.723643471
10	32	36.5	0.0281	0.2	0.88412245	0.781949987
20	32	41.5	0.0594	0.2	0.90886842	0.781994308
50	32	56.5	0.1531	0.2	1.54842486	1.310132606
100	32	81.5	0.3094	0.2	1.67556854	1.402337206

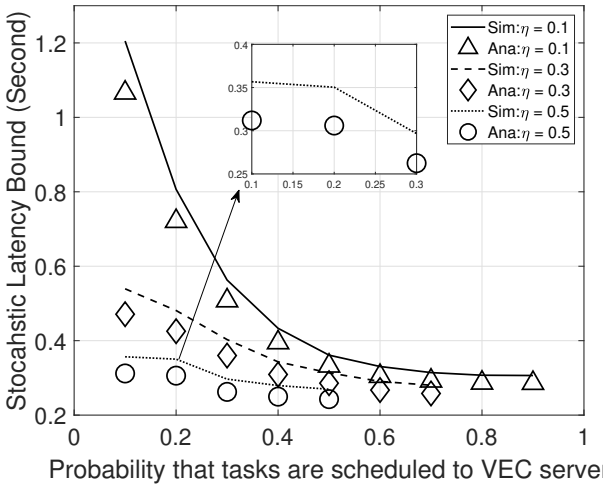


Fig. 6: Stochastic latency bounds predicted by the analytical model and obtained from the simulation experiments by varying task offloading probabilities with $\varphi_{1,n,k} = 0.8$, $\varphi_{2,n,k} = 0.3$, $\lambda_{n,k} = 24$, $Nc_n = 2$, $N_a = 3$, $\mu_{d,n}$=25, $\mu_{vp,n} = 50$, $\mu_{es} = 64$, $V = 15$, $K_{d,n} = 16$, $K_{vp,n} = 32$, $K_{es} = 128$.

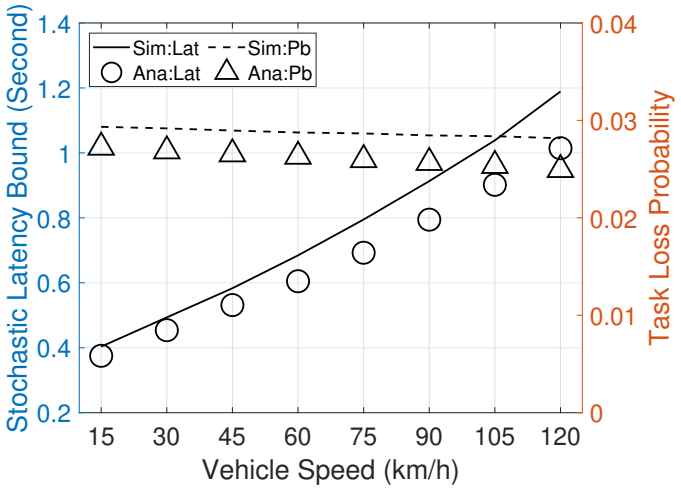


Fig. 7: Stochastic latency bounds and Task loss probability predicted by the analytical model and obtained from the simulation experiments by varying vehicle speeds with $\varphi_{1,n,k} = 0.25$, $\varphi_{2,n,k} = 0.35$, $\lambda_{n,k} = 12$, $Nc_n = 6$, $N_a = 4$, $\mu_{d,n}$=80, $\mu_{vp,n} = 50$, $\mu_{es} = 160$, $K_{d,n} = 16$, $K_{vp,n} = 32$, $K_{es} = 8$.

delay-sensitive and mission-critical vehicular applications. In this subsection, we evaluate the performance of the proposed model with respect to stochastic latency bound by varying the offloading probabilities (η, ϑ, ξ). Herein, we set the values of ϑ from 0 to 0.9 with an interval of 0.1 and η from the set of $[0.1, 0.3, 0.5]$. ξ is obtained from $\xi = 1-\vartheta-\eta$. Fig. 6 displays both the simulation and analytical results, from which we can see that the proposed analytical model provides an accurate prediction of the worst-case latency bound with different task offloading probabilities in VEC systems. Furthermore, Fig. 6 shows that the stochastic latency bound consistently decreases with the increase of the probabilities that tasks are executed at the edge server. This is because, due to factors such as energy consumption, deployment environment, and equipment costs, the VEC server usually has more powerful computation capabilities compared to vehicle servers. This strategy means that if the V2I links remain stable, increasing the number of tasks being offloaded to VEC servers could reduce the processing latency in VEC systems.

5) Prediction accuracy of the proposed analytical model by vehicle speeds: This section aims to evaluate the prediction accuracy of the proposed analytical model and simulation experiments with respect to the stochastic latency bound and task loss probability by varying the vehicle speeds. The simulation results are depicted in Fig. 7, which shows that the proposed analytical model provides a high degree of prediction accuracy with prediction errors ranging from 7.3% with $V = 15$ km/h to 14.7% with $V = 120$ km/h. From Fig. 7, we can observe that the higher vehicle speed pushes up the stochastic latency bounds of the VEC task offloading. This performance degradation stems from the deteriorated channel quality and the reduced V2V/V2I transmission rates when vehicles run at high speed. Contrastingly, as vehicular speed increases, VEC systems benefit from a marginal decrease in task loss probability. This is because the reduced V2V/V2I transmission rates when vehicles run at high speed, would reduce the task arrival rates of both edge servers and neighbouring vehicles, which alleviates their task computational load and results in a slight decrement in task loss probability. Therefore, Fig. 7 reveals that although the higher speed exacerbates the stochastic latency bound due to poorer channel quality, the higher mobility speed may simultaneously bring a marginal benefit of the lower task loss probability. This dual effect underscores the complex interplay between vehicle mobility and system performance of task offloading, which requests a more comprehensive approach to orchestrate the computation and communication resources of VEC systems.

6) Prediction accuracy of the proposed analytical model by varying the task size and transmission power: Table. III demonstrates the stochastic latency bounds derived from the proposed model alongside results from simulation experiments by varying both the task size, L, and transmission power, p. Specifically, the task size ranges from 2M to 12M and the transmission powers are set to be $p = 10, 100, 1000$mW. Table. III demonstrates that analytical results match well with those of the simulation experiments with a prediction error ranges from 7.06% with $L = 2$M and $p = 10$mW to 15.6% with $L = 12$M and $p = 1000$mW. Meanwhile, with the increase of the task size from 2M to 12M, the end-to-end latency bound is pushed up from 0.4157s to 4.4321s, over ten times of performance degradation. This performance degradation stems from the overloaded VEC system operation. This is because the larger task size requires more communication and computation resources for task transmission and processing, which makes the VEC system more overcrowded and leads to higher transmission and processing latencies. Furthermore, Table. III reveals that when the transmission power reaches

TABLE III: Stochastic latency bound with different task sizes and transmission power

	$p = 10mW$		$p = 100mW$		$p = 1000mW$	
L	Sim	Ana	Sim	Ana	Sim	Ana
2	0.4157	0.3883	0.3585	0.3381	0.3504	0.3286
4	0.9899	0.9038	0.8314	0.769	0.8148	0.7553
6	1.7161	1.5359	1.3329	1.1849	1.3226	1.1731
8	2.5513	2.2527	1.8617	1.6345	1.8273	1.5988
10	3.478	3.0433	2.4094	2.101	2.3696	2.0615
12	4.4321	3.8426	2.9443	2.5586	2.8789	2.4903

100 mW, further increasing the transmission power has a marginal impact on the end-to-end latency bound in VEC systems. Specifically, as the transmission power increases from 100 mW to 1000 mW, VEC systems only achieve 2.31% performance improvement of end-to-end latency bound reduction. This is because wireless channel transmission is determined by multiple factors, *e.g.,* bandwidth allocation, transmission power, distances among vehicles and edge server, path loss, antenna gain and noise level as shown in Eq. (1), therefore, solely increasing the contribution of one factor, *e.g.,* transmission power, in V2V/V2I transmissions does not yield substantial reductions of end-to-end latency bound for VEC systems, which requires more comprehensive resource allocation strategies to improve VEC system performance.

C. Performance Analysis

The aim of this subsection is to exploit the proposed model as a useful tool to investigate the resource allocation strategy for VEC systems.

1) Impact of the number of vehicles participating in the task offloading on the stochastic upper latency bound: Vehicle cooperation is a critical aspect of VEC systems for supporting vehicular applications. Although more vehicles involved in task offloading would bring more computation resources, it also results in a higher task arrival rate and computation burden for VEC servers. This subsection targets to investigate the impact of the scalability of vehicle cooperation on the service performance of VEC systems. Fig. 8 plots the stochastic upper latency against the number of vehicles, N_v, involved in vehicular cooperation under different violation error requirements. When N_v is growing from 4 to 16, the stochastic latency bound increases as the VEC system becomes denser. This performance degradation results from the higher number of tasks generated by vehicular applications as more vehicles participate in task offloading, causing VEC nodes to become overloaded, and finally yielding lower performance for VEC task processing. However, when N_v surpasses 16, the stochastic latency bound stabilises. This is because VEC nodes are overloaded and newly arriving tasks would be dropped, leading to higher task loss probability. As illustrated in the right side vertical axes of Fig. 8, the task loss probability increases from 23.76 to 42.36%. In this case, the latency bound does not consistently increase with the number of vehicles

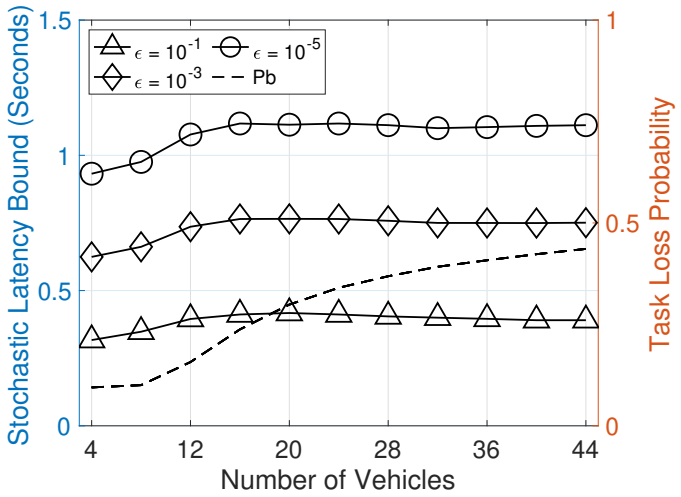


Fig. 8: Impact of the number of vehicles participating in the task offloading on the stochastic latency bound with $\varphi_{1,n,k} = 0.6$, $\varphi_{2,n,k} = 0.25$, $\lambda_{n,k} = 16$, $Nc_n = 4-44$, $N_a = 4$, $\mu_{d,n}$=30, $\mu_{vp,n} = 50$, $\mu_{es} = 80$, $V = 15$, $\eta = 0.2$, $\vartheta = 0.5$, $\xi = 0.3$, $K_{d,n} = 8$, $K_{vp,n} = 16$, $K_{es} = 64$.

involved in task offloading and it is preferable to have fewer vehicles participating in task offloading when VEC nodes are overwhelmed.

2) Impact of computation resource of VEC server on the stochastic upper latency bound: The computational capabilities of the VEC server are paramount for VEC systems to meet the stringent latency demands of vehicular applications. In this subsection, the proposed analytical model is exploited to investigate the relationship between the VEC computation resources and the stochastic latency bound. As observed in Fig. 9, the stochastic upper latency bound decreases inversely proportional to the service rates of the VEC server. Specifically, when $\mu_{es} < 32$, the stochastic latency bound gradually decreases with the increase in the service rates. This phenomenon occurs because the effective service allocated to the tasks successfully received by the VEC server is insufficient for processing the newly arrived tasks. Although a slight increase in service capability would alleviate the overloaded situation of the VEC server, the newly arrived tasks still need to wait for a long time in the queue before receiving computation services. When the effective service rate surpasses the arrival rate (with $\mu_{es} = 32$), there's a notable decrease in the latency experienced by tasks in the queue. Furthermore, when $\mu_{es} > 128$, the stochastic latency bound does not decrease rapidly, indicating that the VEC nodes are underloaded and further increasing the service rates does not significantly improve latency performance. In addition, we can observe that for a fixed service rate of the VEC server, the latency bound increases with the number of applications hosted at each vehicle. For example, with μ_{es}=128, the latency bound increases by 87.32% as the number of vehicular applications increases from 4 to 16. This is because the tasks offloaded to the VEC server are a proportion of the tasks generated by multiple vehicles. Therefore, increasing the number of vehicular applications per

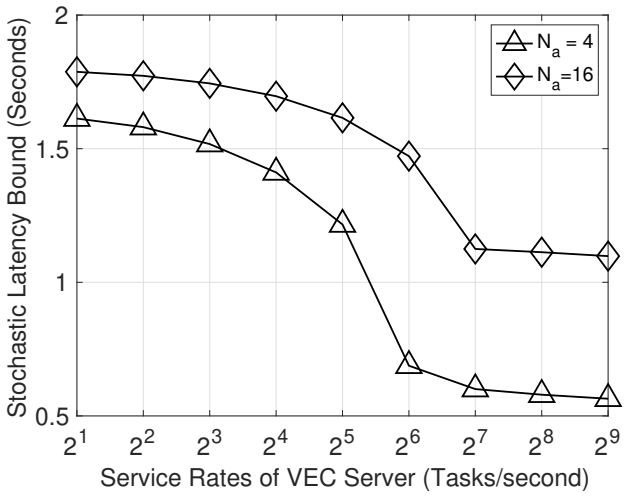


Fig. 9: Impact of the VEC service capabilities on the stochastic latency bound with $\varphi_{1,n,k} = 0.7$, $\varphi_{2,n,k} = 0.35$, $\lambda_{n,k} = 24$, $Nc_n = 6$, $N_a = \{4, 16\}$, $\mu_{d,n}$=40, $\mu_{vp,n} = 50$, $\mu_{es} = \{2^1 - 2^9\}$, $V = 15$, $\eta = 0.3$, $\vartheta = 0.25$, $\xi = 0.45$, $K_{d,n} = 16$, $K_{vp,n} = 16$, $K_{es} = 32$.

vehicle ultimately increases the overall rates of task arrivals at the VEC server, which potentially leads to deteriorated processing performance at the VEC server.

VI. Conclusion and Future Directions

Analytical models can be used as an effective method to study the performance limits of VEC systems. Existing work on the performance analysis of VEC systems primarily focused on deriving average performance metrics. To gain a deeper understanding of the worst-case performance of VEC systems, we presented a new analytical model in this work to quantitatively investigate the stochastic latency bound of intelligent VEC systems. Specifically, we formulated a new Markov chain to analyse the impact of finite buffer size on VEC system performance and derived the task loss probability to enable the proposed model to analyse the performance of VEC with limited buffer configurations. To enable cooperation among vehicles, a new leftover service curve was developed to calculate the available resources provided to tasks of the interesting application where multiple competing applications exist. Furthermore, to analyse the overall service capability of VEC systems, the end-to-end latency bound was obtained built upon the stochastic traffic arrival and dynamic service processing in VEC systems. By capturing the key features of VEC systems, such as bursty task generation, dynamic service provisioning, serious resource competition, and limited buffer configurations, the proposed analytical model has a high potential to offer tremendous value for VEC systems by pinpointing performance bottlenecks and quantitatively optimising overall system operations.

For future research, we plan to investigate the worst-case performance of VEC systems with the interference-rich communication scenarios, such as Non-Orthogonal Multiple Access (NOMA). Designing accurate analytical models with interference-rich cellular networks to investigate the worst-case end-to-end performance of VEC systems is intractable. The main challenge lies in how to mathematically capture interference-varying channels in the processes of the Markov-chain establishment, stochastic service curve development and MMPP-based steady-state performance analysis. Furthermore, we will work on developing novel analytical models to reliably analyse the worst-case performance of VEC systems under potential hardware failures, such as incompatible/failed firmware upgrades or failures due to cyber-security attacks. Although the proposed analytical model could potentially capture the impacts of potential hardware failures during VEC performance analysis by customising system parameters based on the failure consequences, it is still intractable for the proposed analytical model to analyse VEC system performance with cybersecurity-related failures, such as Distributed DoS (DDoS) attacks. This stems from that the network traffic features are usually unknown when VEC systems are under DDoS attacks, where a large amount of attack traffic pours into the VEC infrastructure. This makes it difficult to build accurate traffic models and conduct performance analysis. Therefore, more research endeavours are required to address how to predict traffic features and develop accurate analytical models for VEC systems under potential cybersecurity attacks.

References

[1] John Rydning. Worldwide idc global datasphere forecast, 2022–2026, 2022.

[2] Wenhan Zhan, Chunbo Luo, Jin Wang, Chao Wang, Geyong Min, Hancong Duan, and Qingxin Zhu. Deep-reinforcement-learning-based offloading scheduling for vehicular edge computing. *IEEE Internet of Things Journal*, 7(6):5449–5465, 2020.

[3] Ben-Jye Chang and Jhih-Ming Chiou. Cloud computing-based analyses to predict vehicle driving shockwave for active safe driving in intelligent transportation system. *IEEE Transactions on Intelligent Transportation Systems*, 21(2):852–866, 2020.

[4] Wenhan Zhan, Chunbo Luo, Geyong Min, Chao Wang, Qingxin Zhu, and Hancong Duan. Mobility-aware multi-user offloading optimization for mobile edge computing. *IEEE Transactions on Vehicular Technology*, 69(3):3341–3356, 2020.

[5] Huaming Wu and Katinka Wolter. Stochastic analysis of delayed mobile offloading in heterogeneous networks. *IEEE Transactions on Mobile Computing*, 17(2):461–474, 2018.

[6] Hyun-Suk Lee and Jang-Won Lee. Task offloading in heterogeneous mobile cloud computing: Modeling, analysis, and cloudlet deployment. *IEEE Access*, 6:14908–14925, 2018.

[7] Yixiao Gu, Bin Xia, Chenchen Yang, and Zhiyong Chen. The meta distribution of task offloading in stochastic mobile edge computing networks. *IEEE Transactions on Vehicular Technology*, 71(11):12402–12406, 2022.

[8] Li Liu, Sammy Chan, Guangjie Han, Mohsen Guizani, and Masaki Bandai. Performance modelling of representative load sharing schemes for clustered servers in multiaccess edge computing. *IEEE Internet of Things Journal*, 6(3):4880–4888, 2019.

[9] Alessio Bonadio, Francesco Chiti, and Romano Fantacci. Performance analysis of an edge computing saas system for mobile users. *IEEE Transactions on Vehicular Technology*, 69(2):2049–2057, 2020.

[10] Xiaojie Wang, Zhaolong Ning, Song Guo, and Lei Wang. Imitation learning enabled task scheduling for online vehicular edge computing. *IEEE Transactions on Mobile Computing*, 21(2):598–611, 2022.

[11] Wang Miao, Geyong Min, Xu Zhang, Zhiwei Zhao, and Jia Hu. Performance modelling and quantitative analysis of vehicular edge computing with bursty task arrivals. *IEEE Transactions on Mobile Computing*, 22(2):1129–1142, 2023.

[12] Jing Zhao, Yanbin Wang, HuaLin Lu, Zhijuan Li, and Xiaomin Ma. Interference-based qos and capacity analysis of vanets for safety applications. *IEEE Transactions on Vehicular Technology*, 70(3):2448–2464, 2021.

[13] Yan Zhu, Di Zhou, Min Sheng, Jiandong Li, and Zhu Han. Stochastic delay analysis for satellite data relay networks with heterogeneous traffic and transmission links. *IEEE Transactions on Wireless Communications*, 20(1):156–170, 2021.

[14] Jaya Prakash Champati, Hussein Al-Zubaidy, and James Gross. Transient analysis for multihop wireless networks under static routing. *IEEE/ACM Transactions on Networking*, 28(2):722–735, 2020.

[15] Wang Miao, Geyong Min, Yulei Wu, Haojun Huang, Zhiwei Zhao, Haozhe Wang, and Chunbo Luo. Stochastic performance analysis of network function virtualization in future internet. *IEEE Journal on Selected Areas in Communications*, 37(3):613–626, 2019.

[16] Navid Nikaein, Markus Laner, Kaijie Zhou, Philipp Svoboda, Dejan Drajic, Milica Popovic, and Srdjan Krco. Simple traffic modeling framework for machine type communication. In *ISWCS 2013; The Tenth International Symposium on Wireless Communication Systems*, pages 1–5, 2013.

[17] Chaabouni Sihem, Mounir Frikha, and Michael Meincke. Traffic models for inter-vehicle communications. In *2006 2nd International Conference on Information Communication Technologies*, volume 1, pages 773–778, 2006.

[18] Elena Grigoreva, Maximilian Laurer, Mikhail Vilgelm, Thomas Gehrsitz, and Wolfgang Kellerer. Coupled markovian arrival process for automotive machine type communication traffic modeling. In *IEEE ICC 2017 - IEEE International Conference on Communications*, pages 1–6, 2017.

[19] Zhenyun Zhou, Houjian Yu, Chen Xu, Zheng Chang, Shahid Mumtaz, and Jonathan Rodriguez. Begin: Big data enabled energy-efficient vehicular edge computing. *IEEE Communications Magazine*, 56(12):82–89, 2018.

[20] Zhi-Li Zhang, Don Towsley, and Jim Kurose. Statistical analysis of the generalised processor sharing scheduling discipline. *IEEE Journal on Selected Areas in Communications*, 13(6):1071–1080, 1995.

[21] Ralf Lübben, Markus Fidler, and Jörg Liebeherr. Stochastic bandwidth estimation in networks with random service. *IEEE/ACM Transactions on Networking*, 22(2):484–497, 2014.

[22] Wenhao Fan, Jie Liu, Mingyu Hua, Fan Wu, and Yuan'an Liu. Joint task offloading and resource allocation for multi-access edge computing assisted by parked and moving vehicles. *IEEE Transactions on Vehicular Technology*, 71(5):5314–5330, 2022.

[23] Yixiao Gu, Bin Xia, Chenchen Yang, and Zhiyong Chen. The meta distribution of task offloading in stochastic mobile edge computing networks. *IEEE Transactions on Vehicular Technology*, 71(11):12402–12406, 2022.

[24] Penglin Dai, Kaiwen Hu, Xiao Wu, Huanlai Xing, Fei Teng, and Zhaofei Yu. A probabilistic approach for cooperative computation offloading in mec-assisted vehicular networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(2):899–911, 2022.

[25] Lei Liu, Ming Zhao, Miao Yu, Mian Ahmad Jan, Dapeng Lan, and Amirhosein Taherkordi. Mobility-aware multi-hop task offloading for autonomous driving in vehicular edge computing and networks. *IEEE Transactions on Intelligent Transportation Systems*, 24(2):2169–2182, 2023.

[26] Ming Tang and Vincent W.S. Wong. Deep reinforcement learning for task offloading in mobile edge computing systems. *IEEE Transactions on Mobile Computing*, 21(6):1985–1997, 2022.

[27] Liang Huang, Suzhi Bi, and Ying-Jun Angela Zhang. Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks. *IEEE Transactions on Mobile Computing*, 19(11):2581–2593, 2020.

[28] Ping Lang, Daxin Tian, Xuting Duan, Jianshan Zhou, Zhengguo Sheng, and Victor C. M. Leung. Blockchain-based cooperative computation offloading and secure handover in vehicular edge computing networks. *IEEE Transactions on Intelligent Vehicles*, 8(7):3839–3853, 2023.

[29] Almut Burchard, Jorg Liebeherr, and Stephen Patek. A min-plus calculus for end-to-end statistical service guarantees. *IEEE Transactions on Information Theory*, 52(9):4105–4114, 2006.

[30] James Bucklew and Jone Sadowsky. A contribution to the theory of chernoff bounds. *IEEE Transactions on Information Theory*, 39(1):249–254, 1993.

[31] Ralf Lübben and Markus Fidler. Estimation method for the delay performance of closed-loop flow control with application to tcp. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9, 2016.

[32] Daniel Heyman and David Lucantoni. Modeling multiple ip traffic streams with rate limits. *IEEE/ACM Transactions on Networking*, 11(6):948–958, 2003.

[33] Wang Miao and Min Geyong. Performance modelling and analysis of software-defined networking under bursty multimedia traffic. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 12(5s):1–19, 2016.

[34] Markus Fidler. An end-to-end probabilistic network calculus with moment generating functions. In *14th IEEE International Workshop on Quality of Service*, pages 261–270, 2006.

[35] Pinsky Mark and Karlin Samuel. An introduction to stochastic modelling. *Academic Press*, 2010.

[36] Yuming Jiang. A note on applying stochastic network calculus. In *Computer Science*, 2010.

[37] Varga Andras. A practical introduction to the omnet++ simulation framework. In *Springer Innovations in Communication and Computin*, pages 261–270, 2019.

[38] Jian Wei, Jianhua He, Yi Zhou, Kai Chen, Zuoyin Tang, and Zhiliang Xiong. Enhanced object detection with deep convolutional neural networks for advanced driving assistance. *IEEE Transactions on Intelligent Transportation Systems*, 21(4):1572–1583, 2020.

[39] Alessio Bonadio, Francesco Chiti, and Romano Fantacci. Performance analysis of an edge computing saas system for mobile users. *IEEE Transactions on Vehicular Technology*, 69(2):2049–2057, 2020.

[40] Oscar Adamuz-Hinojosa, Vincenzo Sciancalepore, Pablo Ameigeiras, Juan M. Lopez-Soler, and Xavier Costa-Pérez. A stochastic network calculus (snc)-based model for planning b5g urllc ran slices. *IEEE Transactions on Wireless Communications*, 22(2):1250–1265, 2023.

[41] Lei Lei, Jiahua Lu, Yuming Jiang, Xuemin Sherman Shen, Ying Li, Zhangdui Zhong, and Chuang Lin. Stochastic delay analysis for train control services in next-generation high-speed railway communications system. *IEEE Transactions on Intelligent Transportation Systems*, 17(1):48–64, 2016.

Wang Miao received his PhD degree in Computer Science from the University of Exeter, United Kingdom in 2017. He is currently a lecturer in the Computer Science Department of the University of Plymouth, United Kingdom. His research interests focus on Vehicle Edge Computing, Unmanned Aerial Networks, Wireless Communication Networks, Software Defined Networking, Network Function Virtualisation, Applied Machine Learning, and Stochastic Performance Modelling and Analysis.

Geyong Min is a Professor of High Performance Computing and Networking in the Department of Computer Science within the College of Engineering, Mathematics and Physical Sciences at the University of Exeter, United Kingdom. He received the PhD degree in Computing Science from the University of Glasgow, United Kingdom, in 2003, and the B.Sc. degree in Computer Science from Huazhong University of Science and Technology, China, in 1995. His research interests include Future Internet, Computer Networks, Wireless Communications, Multimedia Systems, Information Security, High Performance Computing, Ubiquitous Computing, Modelling and Performance Engineering.

Zhengxin Yu received her Ph.D. degree in Computer Science from the University of Exeter, UK. She is currently a senior research associate at the Lancaster University, United Kingdom. Her research interests focus on Federated Learning, Deep Learning, Cyber Security and Multi-access Edge Computing.

Xu Zhang received the B.S. degree in communication engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 2012 and the Ph.D. degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 2017. He is lecturer with the School of Computing, University of Leeds, United Kingdom. His research interests include content delivery networks, network measurement, and cloud computing.