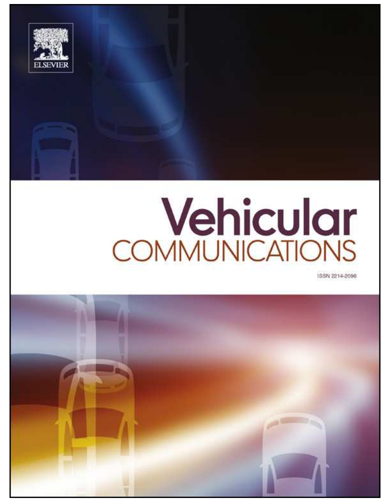
Article:

Bhattacharya, S, Qazi, BR, Muhtar, A et al. (2 more authors) (2017) Sleep-Enabled Roadside Units for Motorway Vehicular Networks. Vehicular Communications, 7. pp. 21-39. ISSN 2214-2096

https://doi.org/10.1016/j.vehcom.2017.01.001

Accepted Manuscript



Sleep-Enabled Roadside Units for Motorway Vehicular Networks

Samya Bhattacharya, Bilal R. Qazi, Adnan Muhtar, Wanod Kumar, Jaafar M.H. Elmirghani

PII: S2214-2096(16)30140-1
DOI: http://dx.doi.org/10.1016/j.vehcom.2017.01.001
Reference: VEHCOM 73

To appear in: *Vehicular Communications*

Received date: 6 October 2016
Revised date: 28 November 2016
Accepted date: 18 January 2017

Please cite this article in press as: S. Bhattacharya et al., Sleep-Enabled Roadside Units for Motorway Vehicular Networks, *Veh. Commun.* (2017), http://dx.doi.org/10.1016/j.vehcom.2017.01.001

Sleep-Enabled Roadside Units for Motorway Vehicular Networks

Samya Bhattacharya[1], Bilal R. Qazi, Adnan Muhtar, Wanod Kumar and Jaafar M. H. Elmirghani

Abstract

In this paper, we introduce a number of generic sleep mechanisms for energy saving at the vehicular roadside units (RSUs). Since random sleep cycles (sleep cycles type-I) were already introduced before, we term the introduced mechanisms sleep cycles (type-II, III, IV, V, VI). Each sleep cycles type arranges the service and sleep sequences distinctively to yield various levels of energy savings and average packet delay. A generic analytic model for the roadside unit (RSU) with such sleep cycles is proposed using G/G/1/K G-vacation queuing, where real vehicular traffic profiles and packet size measurements are utilised. The performance evaluation reveals that with one of the proposed sleep cycles (type-IV), the RSU achieves 68% energy savings and 7.3 ms average packet delay over the day, resulting in respective improvements of 10% and 28% compared to the existing random sleep cycles. These improvements have been achieved under a very conservative operating delay bound for audio conferencing applications. However, modern compression and codecs, due to their leniency on Quality of Service (QoS), would potentially enable higher energy savings through the proposed sleep cycles.

Index Terms

Embedded states, energy, quality of service (QoS), queuing, roadside unit (RSU), sleep cycles, queue vacations.

[1] Samya Bhattacharya and Jaafar Elmirghani are with the School of Electronic and Electrical Engineering, University of Leeds, Leeds LS2 9JT, U.K. email: {s.bhattacharya, j.m.h.elmirghani}@leeds.ac.uk. Bilal R. Qazi and Adnan Muhtar were with the School of Electronic and Electrical Engineering, University of Leeds, Leeds LS2 9JT, U.K. email: {bilalqazipk@gmail.com,adnanmuhtar@hotmail.com}. Wanod Kumar is with Department of Electronic Engineering, Mehran University, Pakistan. email: wanod.kumar@faculty.muet.edu.pk

1. INTRODUCTION

With the emerging trends of 'connected vehicles', the growth of vehicular communication networks comes at a critical time where existing communication technologies are already consuming significant amounts of energy, and environmental concerns are increasingly gaining importance [1]. On one hand, a looming spectrum shortage reduces the likelihood of the use of incumbent cellular networks to support multimedia communications in vehicular networks. On the other hand, extensive cellular base station (BS) deployment to support ubiquitous vehicular network coverage is rendered impractical due to the difficulty of providing high data rates at lower overall costs while maintaining the required quality of service (QoS) [2]. The QoS in a network is measured using certain parameters such as end-to-end latency, average packet delay, delay variation (jitter), and packet loss. Ubiquitous vehicular communication can nevertheless be achieved [3] through heterogeneous use of micro cells served by roadside units (RSUs) within a macro cell [4] where the RSUs enable higher data rates [2]. The roadside units (RSUs) are dedicated wireless communication devices which communicate with vehicles within short range and therefore provide each other with information, such as safety warnings, locations and traffic. Such architecture can become part of the 5G offerings. It can reduce the overall energy consumption of the network [2] where the locations and numbers of RSUs/BSs are pre-optimised from performance and energy perspectives.

However, further operational energy savings in an RSU/BS may be possible by utilising sleep mechanisms, where the transmitting circuitry is switched off to set the transceiver into low energy state [4], [5], [6], [7]. Such sleep strategies could be impractical for 'base station only' scenarios due to the large coverage, long resource activation time and high overheads that may arise from the ping-pong effect of consistently turning a BS ON and OFF [8].

Even though an RSU performs somewhat similar operations to that of a BS, it does so at a much smaller scale making it feasible to switch into low power state and then back to the fully operational state in a relatively small time. Thus, the switching overhead is remarkably low compared to that of a BS.

Please note that in this paper, we considered the primary types of sleep cycles developed taking into consideration the very basic types of traffic variations. There might be sleep cycles either with fixed time duration or sporadic nature or sleep cycles that follow some probabilistic distribution. However, these are all incremental and minor variations of the basic types proposed here.

The key contributions in this paper are:

1) Introduction of five new sleep mechanisms and their comparisons
2) Introduction of the new G-vacation concept to queuing theory
3) A workable (closed form) generic model for these sleep mechanisms
4) Analyses of these six sleep mechanisms (including the existing one) in a vehicular roadside unit

The remainder of this paper is organised as follows: Related work is presented in Section 2. In Section 3, the sleep cycles operating scenario is discussed in the context of a typical motorway vehicular network. Section 4 describes the delay minimisation technique at the CHs. In Section 5, different types of sleep cycles are introduced and subsequently an analytic model for the RSU is developed with the proposed sleep cycles. Section 6 discusses the performance results in terms of QoS and energy savings at the RSU. Finally, the paper concludes in Section 7.

2. Related Work

A number of studies showed that by considering a sleep strategy at a node during its inactivity, a certain amount of energy can be saved [2], [5], [9], [12], [13], [14], [15]. Sleep strategies in the recent past have been introduced as a solution to reduce the network power consumption as they do not need a complete overhaul of network devices and protocols. In [12], [13], the authors proposed sleep strategies for the line-cards in the routers depending upon the backbone traffic. The heuristic in [12] achieved 79% reduction in backbone energy consumption, which was achieved at the typical low link utilisation (30%) in the Internet Service Providers (ISPs) backbone. Such major reduction may not be feasible in wireless and mobile networks (e.g. cellular or vehicular) as they are not intrinsically over-provisioned and the link quality which depends upon the varying wireless channel, makes such networks susceptible to degraded QoS. Nevertheless, several research groups around the world are considering various sleep strategies to make cellular networks more energy efficient [2], [14], [15].

In [14], the authors proposed dynamic switching for a BS in low traffic conditions. However, fast switching may not be feasible to accommodate transient traffic behaviour because of the number of operations a large BS has to perform [8]. The problems are compounded in such an approach when considered for vehicular networks characterised by their very high mobility. In another study [15], the authors studied a periodic sleep strategy for cellular networks, which led to 46% reduction in operating energy expenditure. However, the architecture proposed was of the multi-layered type and deployed a cell breathing technique. Again, the cell breathing solution with its incurred overhead cannot accommodate fast user movement and variation in traffic demand, making it inapplicable for vehicular networks, especially in a motorway environment. Hence, a macro-micro cellular structure is considered in this paper, where a number of micro-cells are served by the Access Points (APs) / Roadside Units (RSUs) within a macro-cell served by a BS [4], making it feasible to operate various sleep strategies for energy savings at the APs/RSUs. One such strategy is sleep cycles where an RSU switches OFF only its transmitter part for a randomly distributed time duration when there is no request to serve [9]. The RSU remains in such mode for the entire time duration (randomly generated with a certain mean value) even if packets are waiting to be served, thus degrading the system performance. Similarly a random sleep strategy was utilised in [10], which resulted in an unacceptable average packet delay [10] for audio-conferencing applications [11]. Moreover, saving energy through sleep cycles at the RSU incurs wake-up overheads, associated with each sleep cycle [9]. Therefore, there is a need to introduce various types of sleep cycles which not only improve the QoS but also reduce the wake-up overhead while maximising energy savings at the RSU in a motorway vehicular environment. These redefine the performance analysis of such systems from both QoS and energy perspectives and call for the development of analytic vacation queuing models.

Security and privacy are important in vehicular networks as for the first time external remote users can gain access to vehicles and their systems. With RSUs responding to users' behaviour and demands through entering into / exiting sleep mechanisms, new attack scenarios can be envisaged and security and privacy in the presence of sleep mechanisms warrants further study. It is worth noting here for example the temporal and rate privacy concepts in [16].

Traditionally, queuing theory based models were extensively used in predicting the QoS of access networks with little emphasis on vacation queues. In [17], an M/M/c queue was analysed with queue length independent random vacations of the servers and their impact on average system delay and utilisation were studied. In [9], the authors studied two types of random vacations where the vacation duration was Negative exponential distributed using (1) queue length independent vacations to model the wireless channel impairments, and (2) queue length dependent vacations to model random sleep cycles (type-I). Both the arrival and the service discipline of packets were assumed to be memory-less Poisson type as a simplistic case. Hence, the respective scenarios were modelled as M/M/c and M/M/1/K queues with random vacations. Since both of these scenarios can be represented using simplistic Markov Chains, a Matrix Geometric Method (MGM) was adopted to solve these Markov Chains. In [10], the authors utilised the same random sleep cycles (type-I) at the CHs to save energy in a vehicle-to-vehicle (V2V) communication scenario. The arrival of the packets at the CHs was considered Poisson distributed. Since packet size distribution was considered random as a simplistic case, the service duration and vacation duration followed Negative exponential distributions. Hence, each CH was modelled as an M/M/1/K queue with random queue length dependent sleep cycles. However, to the best of our knowledge, General distributed packet arrival along with General distributed service discipline and various types of General distributed sleep cycles have not been considered or studied in conjunction.

Classification of Vacations

The authors in [9] proposed a sleep strategy which reduced the power consumption of the RSU by switching OFF only its transmitter part for a randomly distributed time duration when there is no request to serve. The RSU remained in sleep mode for a fixed time duration (randomly generated with a certain mean value) even if packets were waiting to be served. Upon waking up, the RSU served the arrived packets (if any) and switched back to sleep mode when the buffer became empty. This mechanism was called random sleep cycles. The same random sleep cycles were utilised in [10], which resulted in an unacceptable average packet delay for audio-conferencing applications [11]. For ease of reference, the random sleep cycles introduced in [9], [10] are termed sleep cycles type-I in this paper. These types of sleep cycles degrade the system performance when a large number of packets wait to be served. Since energy saving through sleep cycles of type-I was achieved at the expense of degraded QoS [10] and incurred wake-up overheads, associated with each sleep cycle [9], there is a need to improve QoS and reduce wake-up overhead while maximising energy savings. Hence, different types of sleep cycles need to be explored in the context of QoS and energy savings at the RSU.

The authors in [10] utilised sleep cycles type-I to save energy at the cluster head (CH) vehicles in a pure vehicle to vehicle (V2V) scenario while deteriorating QoS, and ignoring the effects on end-to-end performance. If the QoS bound is not met at the cluster heads (CHs), then the introduction of any sleep strategy at the RSU will be meaningless. Therefore, to study the end-to-end performance of a hybrid vehicular network we first introduce sleep cycles type-II at the cluster head (CH) within a cluster, which prematurely terminates a sleep cycle to improve the QoS at the CHs. Such a termination is crucial as it enables the study of different types of sleep cycles for energy savings at the RSU and their impact on QoS.

From the performance modelling perspective, sleep cycles can be represented by special types of vacations. Vacations can be synchronous or asynchronous. Synchronous vacations mean that all the servers go into vacations at the same time regardless of the arrival of the packets. Asynchronous vacation means that the servers can randomly go into vacations. An asynchronous vacation can be queue length dependent as well, where the server goes into a vacation only when there is no packet waiting in the queue. The vacations and sleep cycles types are illustrated in Figure 1. Descriptions of different types of sleep cycles are given in Section 3 (type-I and type-II) and Section 5 (type-III to V). It should be noted that the sleep cycles Type-I and II work at the cluster heads only while the other sleep cycles (type

III to VI) work at the RSU. The packet arrival at the cluster head is Poisson distributed while that at the RSU is General distributed. Sleep cycles Type-I need delay minimisation since the sleep duration is random and traffic independent. Thus, it is not suitable for the RSU. Sleep cycles Type-II follows the traffic (an ideal case) and therefore its switching overhead is high. Thus, it is also impractical for the RSU.

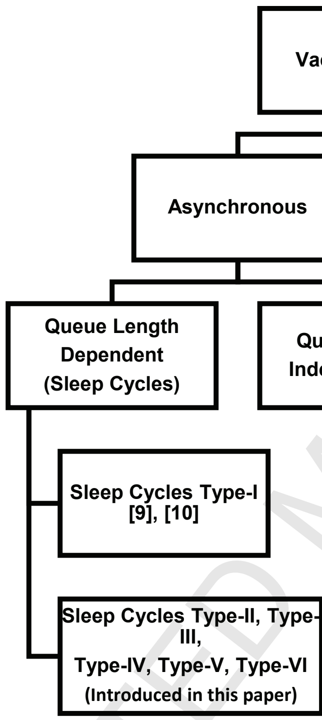
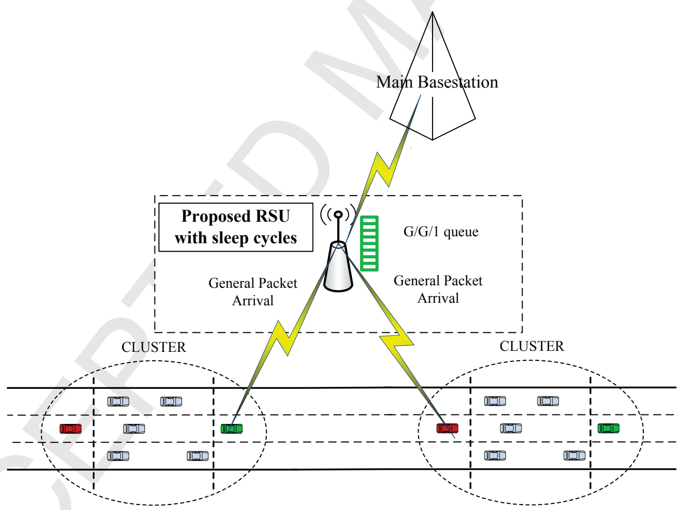


Figure 1. Classification of vacations.

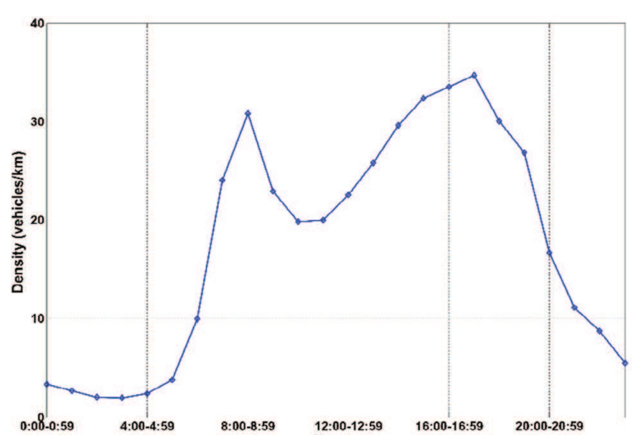
3. THE STUDIED SCENARIO

The studied vehicle to roadside (V2R) communication scenario, shown in Figure 2(a), uses micro/pico-cells served by the RSUs within a macro cell that enables higher data rates [2]. The RSUs are typically spaced 1 km apart along a 3 lane motorway stretch, which is in line with the Wireless Access for Vehicular Environment (WAVE) standard [18]. In [10], [19], the authors considered a V2V scenario where the CHs in a double cluster head (DCH) scheme were responsible for the formation/deformation, synchronisation and communication within the clusters. Such a setup makes the handovers between micro-cells transparent to the RSUs. In this double cluster head scheme, a vehicle (node) can perform four possible roles: 1) cluster member (CLM); 2) cluster head 1 (CLH1) (situated near the front end of the transmission range limit); 3) cluster head 2 (CLH2) (located near the rear end of the transmission range limit); and 4) undecided state (acts as an independent entity – not a part of any cluster). The transmission range, which governs the communication between the vehicles and the information exchange between them, is assumed to be "R" metres. The value of R can be varied in accordance with the amount of traffic at different times of the day. Vehicular traffic can be categorised into high, moderate and low vehicular traffic flow. Two or more nodes can form a cluster as long as they are all in each other's vicinity which is determined by the range R. Two nodes, within the range R, with the largest distance between them are chosen to be cluster head 1 (CLH1) and cluster head 2 (CLH2). All the other nodes within that range become cluster members. Another vehicle (e.g. Vehicle C) coming into the vicinity of any cluster head (e.g. CLH1) is checked if it is also in the range of the other cluster head (e.g. CLH2) of the same cluster. If it is and is found to have a larger distance compared to CLH1, then Vehicle C takes over the role of the cluster head (CLH1) and the previous CLH1 node becomes a cluster member. Otherwise, Vehicle C becomes a cluster member and communicates via CLH1. Each time a new cluster is created, the cluster heads allocate a cluster ID to the member vehicles in order to facilitate better communication.

In this paper, we focus on the end-to-end performance of the proposed scenario where a CH communicates with an RSU, which in turn connects to the main BS (Figure 2(a)), to access external resources such as the Internet. The vehicles in the motorway generate Poisson traffic with General distributed packet sizes [20]. Then a MAC (Medium Access Control) protocol operates at the CHs and each vehicle is allocated a time slot. The resultant packet stream at the output of the CHs is the input to the RSU. Since we aim to introduce energy savings at the RSUs while keeping the end-to-end QoS within acceptable bounds, any sleep strategy at the RSU will be meaningless as the maximum individual packet delay at the CH(s) was already unacceptable [10] for audio-conferencing [11]. Therefore, we first introduce sleep cycles type-II at the CHs, which improves the QoS, leading to the study of various types of sleep cycles at the RSU. Energy saving is maximised at the RSU while maintaining stringent bounds on the end-to-end QoS (packet blocking probability P_B^{E2E} within 5% [9] and maximum individual packet delay D_{max}^{E2E} within 32 ms [11]). Our approach enables us to evaluate the performance of the proposed sleep cycles under very stringent operating bounds (required for audio conferencing), though modern compression and codes allow more leniency on packet delay [21]. Without loss of generality, we only consider the up-link communication between the CH and the macro-BS (via RSU) as end-to-end, assuming that the macro-BSs are optically inter-connected. The coverage of the RSU is assumed to be 1 km, which is in line with the Wireless Access in Vehicular Environments (WAVE) standard [22]. Vehicular traffic densities from the M4 motorway, UK (Figure 2(b)) and real packet size distribution from [20] have been utilised to analyse the performance of the system. As we are interested in finding the fundamental limits of the end-to-end QoS and energy savings at the RSU through various types of sleep cycles, a collision and contention free MAC and idealistic wireless channel have been considered. In earlier work, we studied vehicular MAC performance [23] and channel impact [9].



(a) Proposed RSU architecture.



(b) Vehicular density.

Figure 2. The studied scenario.

4. AVERAGE PACKET DELAY MINIMISATION AT THE CLUSTER HEADS

In this section we first define sleep cycles type-I and type-II, which are used at the cluster heads, as follows:

1. **Sleep Cycles Type-I (Random sleep cycles):** In these types of sleep cycles, an entity (e.g. RSU) switches to sleep mode when there is no packet to serve. The sleep duration is Negative exponential distributed with a certain mean. An RSU with these sleep cycles can be modelled as an M/M/1/K queue with *M*-vacation where M stands for the memoryless property.

2. **Sleep Cycles Type-II (Follow the traffic):** In these types of sleep cycles, an entity (e.g. RSU) switches to sleep mode when there is no packet to serve. However, as soon as a packet arrives, it wakes up to serve the packet. These sleep cycles can be modelled as an M/M/1/K queue without any vacation. Since we use real packet sizes in this work instead of arbitrary packet sizes, an M/G/1/K queue is used instead.

In [10], the packet blocking probability was bounded at 5% at each hour of the day to maximise energy savings through random sleep cycles (type-I) at the CH. However, the average packet delay exceeded [10] the maximum individual packet delay bound for audio-conferencing (i.e. 32 ms [11]), which implies that the individual packets delay was substantially higher than the already out of bound average packet delay. Hence, the average packet delay at the CHs needs to be minimised in order to maintain the end-to-end maximum individual packet delay bound for audio-conferencing [11], one of the necessary and most demanding bounds. Since the RSUs have large buffer capacity, the packet blocking probability is negligible at the RSU.

For ease of flow, we first reproduce the model equations of [10] with explanations in Section 4-A, which correspond to (1)-(17). Secondly, a minimum buffer size is obtained considering real packet size distribution to maintain the packet blocking probability at 0.05. To improve the average packet delay, we introduce sleep cycles type-II, which readily terminates the sleep mode as soon as a packet arrives. Such a termination at the CH is crucial to decrease the average packet delay. The CH parameters for the studied scenario are shown in Table 1.

Table 1: Cluster Head (CH) parameters [10].

Parameter	Notation	Value
Up-link channel data rate	d_r	6 Mbps per CH [24]
No of vehicles in a cluster	V	3 to 36 [10]
Data generation rate per vehicle	d_t	320 kbps [25]
Average packet size	P_s	867.4 Bytes [20]
Average packet arrival rate per CH	λ	$d_t/(P_s \times 8 \times 2)$ packets/s
Average packet service rate	μ	$d_r/(P_s \times 8)$ packets/s

A. Cluster Head Model

In [10], an analytic M/M/1/K queuing model with random sleep cycles (sleep cycles type-I) for a CH was solved to obtain expressions for packet blocking probability (P_B) and average packet delay (W). In the above queuing model, the first M stands for Poisson distributed packet arrival, the second M stands for Negative exponential distributed service discipline, 1 corresponds to one server i.e. the transmitter of the CH and K stands for the buffer size (i.e. the number of packets including the packet in service) of the CH.

Each CH in [10] served $V = 2$ vehicles within the cluster with a mean rate of μ packets/s, where an negative exponential distributed service discipline was considered. The packet arrival at each CH was considered Poisson distributed with a mean rate of λ packets/s per vehicle. The states of the CH at the embedded transition points were represented by the number of packets in the CH (waiting and in-service) immediately after the embedded time point and the nature of the embedded point (i.e. whether it was either a service completion or a sleep cycle completion).

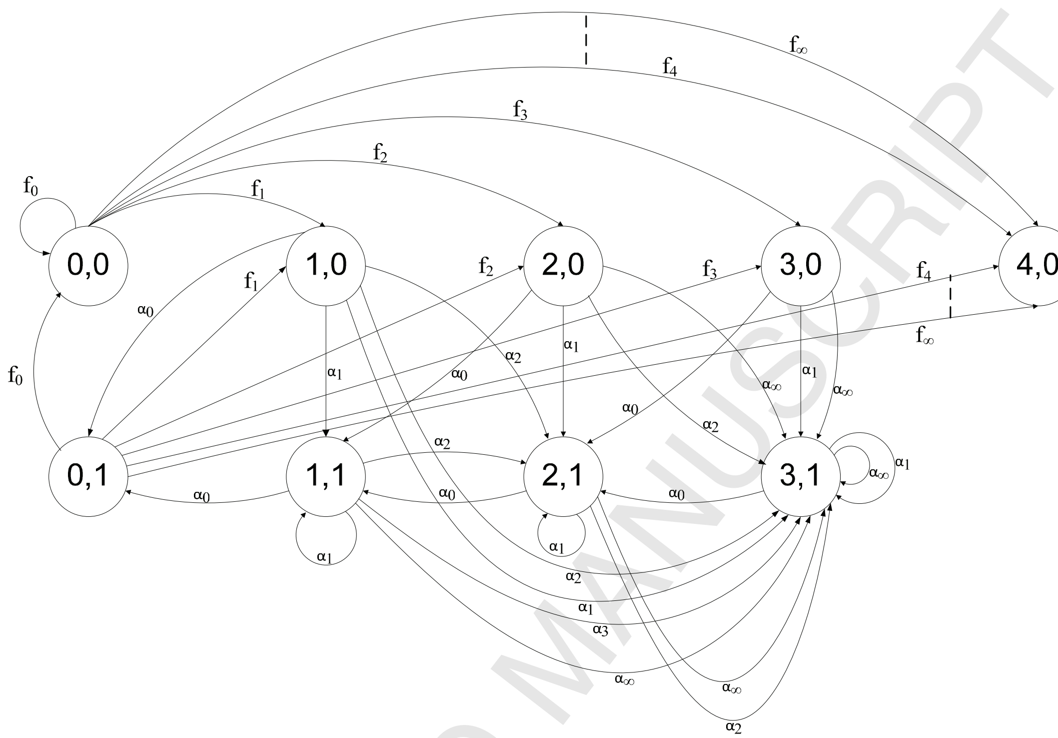


Figure 3. Embedded state diagram of the CH (for $K = 4$).

The CH state at the i^{th} embedded point is represented by (n_i, ϕ_i), where

n_i = number of packets in the CH just after the i^{th} embedded point and

ϕ_i = 0 if the i^{th} embedded point was a sleep cycle completion,

ϕ_i = 1 if the i^{th} embedded point was a service completion.

Considering the system in equilibrium, let q_k, $k = 0,1,\ldots,K$ be the probability of the state $(k, 0)$ and r_k , $k = 0,1,\ldots,(K-1)$ be the probability of the state $(k, 1)$ at the embedded points. Note that r_k is only defined till $k = (K-1)$ since the system, just after a service completion, cannot be in state K. Let f_j ($j = 0, 1, \ldots, \infty$) be the probability of j packet arrivals in the system within a sleep cycle. The probability f_j can be obtained as a continuous summation of the products of the probability of j packet arrivals during a sleep cycle duration t and the probability that a sleep cycle is of time duration t, where t varies from 0 to 1. The probability f_j can be expressed as

$$f_j = \int_0^{\infty} \frac{\left(\frac{V\lambda t}{2}\right)^j}{j!} e^{-\left(\frac{V\lambda t}{2}\right)} f_S(t)dt \qquad j = 0, 1, \ldots, \infty \tag{1}$$

where $f_S\,(t) = \frac{1}{\bar{S}}\, e^{-\frac{1}{\bar{S}}t}$ denotes the probability density function (pdf) of the Negative exponential distributed sleep durations (sleep cycles) with mean duration $\bar{S}$. Let $\alpha_j =$ $(j\ =\ 0,1,\dots,\infty)$ be similarly defined as the probability of j packet arrivals in a service time, which can be expressed as

$$\alpha_j = \int_0^{\infty} \frac{(\frac{V\lambda t}{2})^j}{j!}\, e^{-\left(\frac{V\lambda t}{2}\right)} b(t)dt \qquad j\ =\ 0,1,\dots,\infty \tag{2}$$

where $b(t) = \mu e^{-\mu t}$ denotes the probability density function (pdf) of the service durations. Equation (2) denotes a continuous summation of the products of the probability of j packets arriving during a service duration t and the probability that a service duration is of time period t, where t varies from 0 to 1. The transition equations are written as

$$q_k = (q_0 + r_0\,)\, f_k \qquad k\ =\ 0,1,\dots,(K-1) \tag{3}$$

$$q_K = (q_0 + r_0\,) \sum_{k=K}^{\infty} f_k \qquad k\ =\ K \tag{4}$$

$$r_k = \sum_{j=1}^{k+1} \left(q_j + r_j\right)\, \propto_{k-j+1} \qquad k\ =\ 0,1,\dots,(K-2) \tag{5}$$

$$r_{K-1} = q_K + \sum_{j=1}^{k-1} \left(q_j + r_j\,\right) \sum_{k=K-j}^{\infty} \propto_k \qquad k\ =\ K-1 \tag{6}$$

where q_0 and r_0 represent the probabilities of the system being empty after a sleep cycle completion and a service completion respectively. Equation (3) denotes the probability of k packet arrivals during a sleep cycle when the CH was initially empty. Similarly, (4) denotes the probability of K packets in the CH during a sleep cycle, which can only occur through K or more than K arrivals. Equation (5) denotes the probability of k packets in the system during a service period. This can only happen in $k\ +\ 1$ mutually exclusive ways, each of which denotes $k\ +\ 1 - j$ packet arrivals during a service period after the j^{th} embedded point. Equation (6) is obtained in a similar fashion. The embedded state diagram for four packets in the CH is shown in Figure 3, which helps us to visualise (3)-(6). Summing up the probabilities of all possible states, we obtain

$$\sum_{j=0}^{K} q_j + \sum_{j=0}^{K-1} r_j = 1 \tag{7}$$

To solve (3)-(7) recursively for q_k , r_k, we define an intermediate variable

$$\beta_k\ (k\ =\ 0,1,\dots,K-1)$$

as

$$\beta_k\ =\ \frac{q_k + r_k}{q_0 + r_0} \tag{8}$$

It is a ratio of the probability of the CH having k packets to the probability of the CH being empty. Using (3) and (5), (8) is recursively defined as

$$\beta_0 = 1,$$

$$\beta_1 = \frac{\beta_0 - f_0}{\alpha_0},$$

$$\beta_k = \frac{(q_0+r_0)f_k + \sum_{j=1}^{k+1} (q_j+r_j)\alpha_{k-j+1}}{(q_0+r_0)},$$

$$\beta_{k+1} = \frac{\beta_k - f_k - \sum_{j=1}^{k} \beta_j \alpha_{k-j+1}}{\alpha_0}. \qquad k = 2, \dots, K-1 \quad (9)$$

Substituting (3), (4), (8) in (7), we obtain

$$(q_0 + r_0) \sum_{k=K}^{\infty} f_k + \sum_{k=0}^{K-1} q_k + \sum_{j=0}^{K-1} r_j = 1$$

$$(q_0 + r_0) \left[\sum_{k=K}^{\infty} f_k + 1 + \sum_{k=1}^{K-1} \beta_k \right] = 1$$

Using $\beta_0 = 1$, the probability of the system being empty $(q_0 + r_0)$ is expressed as

$$q_0 + r_0 = \frac{1}{\left[\sum_{k=K}^{\infty} f_k + \sum_{k=1}^{K-1} \beta_k\right]} \quad (10)$$

Substituting the value of $q_0 + r_0$ from (10) in (3) and (4), we find q_k, $k = 0,1, \dots, K$. Further, using these values of q_k and values of k, we obtain r_k, $k = 0,1, \dots, (K-1)$ as

$$r_k = (q_0 + r_0)\beta_k - q_k \qquad k = 0,1, \dots, (K-1) \quad (11)$$

The probabilities, q_k and r_k, are now used to compute the QoS and energy parameters of the system. Let ρ_c be defined as the carried load, i.e. the probability that the CH is busy at an arbitrary time. Analysing all the intervals between successive embedded points over a long time duration (say T), we conclude that

$$\rho_c = \lim_{T \to \infty} \frac{\sum \text{service times in } T}{\sum \text{sleep cycle times in } T + \sum \text{service times in } T}$$

$$= \frac{(1-q_0-r_0)\bar{X}}{(q_0+r_0)\bar{S} + (1-q_0-r_0)\bar{X}} \quad (12)$$

where $\bar{X} = 1/\mu$ is the mean service time. The offered load, ρ in this case, is defined as

$$\rho = \frac{V}{2}\lambda\bar{X} \quad (13)$$

Using (12) and (13), the packet blocking probability, P_B, can be obtained as

$$P_B = \frac{\rho-\rho_c}{\rho} \tag{14}$$

In order to determine the other performance parameters like the average packet delay, W and the number of packets in the CH, N, we define a quantity, D, which is the mean time between successive embedded points of the above analysis, when the system is in equilibrium. The mean time can either be mean sleep duration with an occurrence probability equal to the probability that the CH is empty or mean service duration with an occurrence probability of CH not empty. Therefore, we write

$$D = (q_0 + r_0)\bar{S} + (1 - q_0 - r_0)\bar{X} \tag{15}$$

Using (15), N and W are, respectively, given as

$$N = \frac{2}{V\lambda D}\sum_{j=1}^{K-1} j r_j + K\left(\frac{\rho-\rho_c}{\rho}\right) \tag{16}$$

$$W = \frac{2N}{V\lambda(1-P_B)} \tag{17}$$

Since both P_B and W are dependent upon the buffer size as shown in the above model, it is evident that a small buffer results in a lower packet delay but a higher packet blocking probability whereas a large buffer results in a lower packet blocking probability but a higher packet delay. Therefore, we need to: (a) determine the minimum buffer size of the CH which maintains P_B at 0.05 throughout the day; and (b) minimise the average packet delay by introducing sleep cycles type-II (Section 4-B). As a first step to achieve (a), we substitute $q_0 + r_0$ from (10) in (12), then ρ_c from (12) in (14) and subsequently rearrange (14), in terms of buffer size K as

$$\sum_{k=K}^{\infty} f_k + \sum_{k=0}^{K-1} \beta_k = \frac{1-\frac{V\lambda}{2}(1-P_B)(\bar{X}-\bar{S})}{1-\frac{V\lambda}{2}\bar{X}(1-P_B)} \tag{18}$$

With the maximum number of vehicles ($V = V_{max}$) representing the busiest hour, the offered load ρ becomes highest i.e. $\rho_{\max}$. If we set $\bar{S} = 0$ by assuming CH does not sleep at the highest load, the minimum buffer size should be able to maintain $P_B = 0.05$. It is evident that with a lower load P_B is less than 0.05, so that the mean sleep cycle duration $\bar{S}$ can be tuned (maximised) to obtain $P_B = 0.05$. Thus, we set $\bar{S} = 0$ and $P_B = 0.05$ and substitute all the known parameters in (18) to solve for K, which belongs to the index of (18). Therefore, the equation needs to be iteratively solved to obtain the minimum buffer size. We denote the minimum buffer size as K_{min}^{exp} for Negative exponential distributed service discipline. For real packet size distribution [20], the service duration becomes General distributed. Hence, the minimum buffer size K_{min}^{gen} for General distributed service discipline cannot be deduced from the above mentioned queuing model of CH. Hence, we use a two moment approximation method [26], [27] to obtain K_{min}^{gen} in terms of K_{min}^{exp} as

$$K_{min}^{gen} = K_{min}^{exp} + NINT\ [\ 0.5\ (\ c_s^2 - 1)\ \sqrt{\rho_{\max}}\ K_{min}^{exp} \qquad (19)$$

where c_s^2 is the squared coefficient of variation of the service time distribution and $NINT$ refer to the nearest integer function. With the above mentioned parameters, we get $K_{min}^{exp} = 14$ and $K_{min}^{gen} = 10$. This can be explained as follows. Since, the variance of the real packet size distribution [20] is lower than that for the case of Negative exponential (random) packet size distribution for a given fixed data rate, the buffer has to accommodate much lower variations in the former case. Hence, the required minimum buffer size is smaller for real packet size distribution (i.e. $K_{min}^{gen} < K_{min}^{exp}$).

B. Sleep cycles type-II at the CH for delay minimisation

A sleep cycle can be described by two attributes: (1) The time epoch - the entity (CH/RSU) switches to sleep; and (2) The time duration - through which the entity remains in sleep. Both can be tuned for performance enhancement. Having determined the required minimum buffer size of a CH, we modify the random sleep cycle (sleep cycle type-I) [10] of a CH as follows:

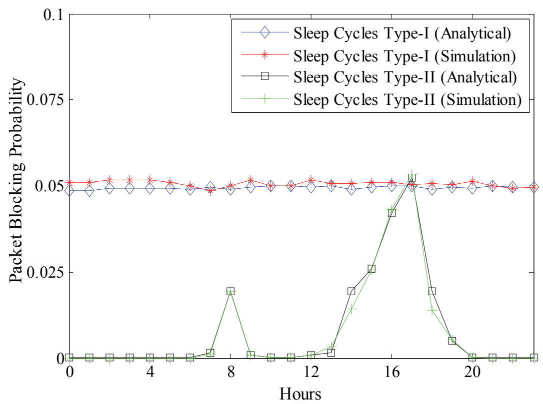
A CH switches to sleep mode if there is no packet in it and remains in sleep mode as long as no packet arrives. As soon as a packet arrives at the buffer of the CH, the CH immediately wakes up and starts serving. If a packet arrives during the service, the CH does not switch to sleep mode until it has served all the buffered packets. We term these types of sleep cycles *sleep cycles type-II.* Since sleep cycles type-II exactly follow traffic variation, these types of sleep cycles is transparent to both arrival and departure of traffic. From the above definition, it follows that the pdf of sleep cycles type-II is the pdf of the packet inter-arrival time at the CH. Therefore, a system with such sleep cycles behaves as though there are no sleep cycles operating, unless the switching overhead is considered when determining the energy savings. A cluster head is a vehicle having large battery (~40 Ah) that supplies electrical energy. Thus, the communication energy consumption of the vehicle is negligible compared to its electrical energy consumption. Therefore, the sleep cycles type-II is introduced only to improve the QoS at the CH. The sleep switching overhead is discussed in the context of RSU in Section 5-B and its impact on energy savings is evaluated in Section 6.

The performance, in terms of packet blocking probability P_B and average packet delay W, at a CH is evaluated with sleep cycles type-I and sleep cycles type-II as shown in Figure 4(a) and Figure 4(b). The analytic results are verified through simulations. These exhibit the same trend of vehicular density since *sleep cycles type-II* is transparent to QoS. With sleep cycles type-I, P_B was maintained around 0.05 throughout the day, however at the expense of higher W. Since the packets are served immediately in the case of sleep cycles type-II, P_B is considerably improved. Only at the busiest hour (17:00 hour), P_B reaches the threshold (i.e. 0.05). Note that at 17:00 hour, the behaviour of the CH becomes similar for both types of sleep cycles as CH seldom sleeps.

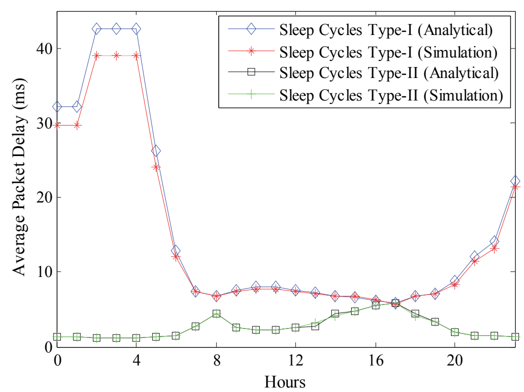
With sleep cycles type-I, W reduces with an increase in the number of vehicles because the CHs seldom sleep and thus serve all the buffered packets quickly. On the other hand W reaches $40\,ms$ during off-peak hours which is unacceptable for audio-conferencing [11]. With sleep cycles type-II, W reduces to a maximum of $5.8\,ms$ with an average of $2.6\,ms$ throughout the day. Thus, we infer that sleep cycles type-II improve both packet blocking probability and average packet delay compared to sleep cycles type-I, since the former follows traffic. However, if the packet sizes are small and the packet arrivals are frequent, then the number of sleeps i.e. the sleep count increases considerably. Since each sleep cycle incurs wakeup overhead energy, the energy savings reduce considerably. In fact, it may be possible that in some cases the sleep cycles achieve no energy savings or even negative energy savings i.e. increased energy consumption. This can occurs due to frequent switching, which results in considerable overheads. This motivates us to study different types of sleep cycles at the RSU for energy savings. Hence, subsequent analysis on the RSU is based on the output traffic from the CHs, which utilise sleep cycles type-II since sleep cycles type-I at the CHs led to unacceptable average packet delay (Figure 4(b)).

5. Energy Savings At The RSU

To achieve energy savings through sleep cycles, in this section we study different types of sleep cycles at the RSU, and develop their respective simulation and analytic models.



(a) Packet blocking probability at a CH.



(b) Average packet delay at a CH.

Figure 4. Performance at a CH with sleep cycles type-I and type-II.

Here, we formally define four types of sleep cycles, namely sleep cycles type-III, sleep cycles type-IV, sleep cycles type-V and sleep cycles type-VI. Their operation is shown in Figure 5. The timing diagram example of Figure 5 is carefully designed so that there are packet arrivals during service as well as no arrival during service. This distinguishes between the modified and unmodified versions of sleep cycles. Further, the figure clearly shows the sleep count (wake-up overhead) associated with each of these sleep cycles.

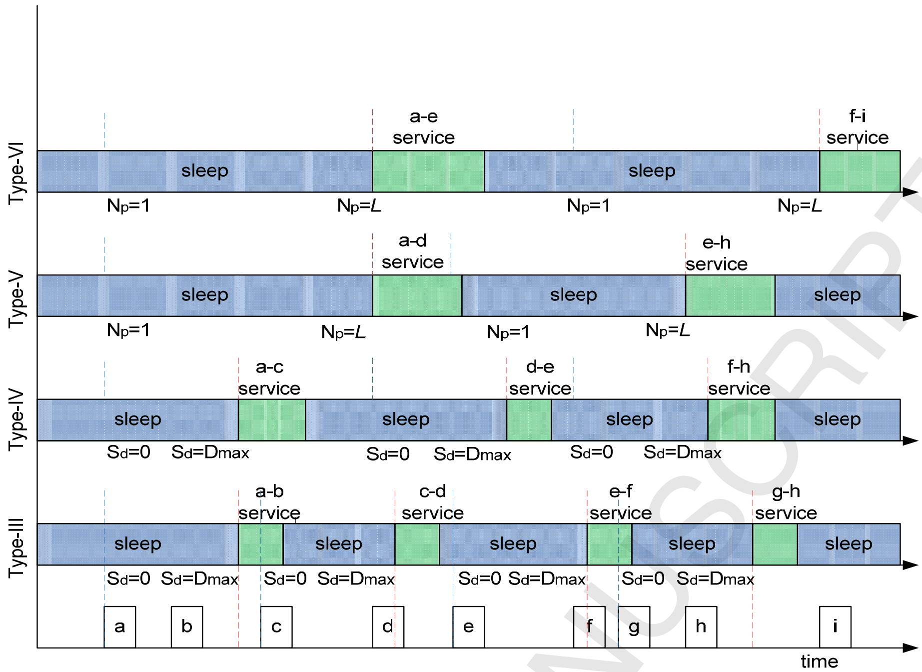


Figure 5. Timing diagrams of the proposed sleep cycles at the RSU.

Sleep Cycles Type-III (Time based burst accumulation): In these types of sleep cycles, an entity (e.g. RSU) switches to sleep mode when there is no packet to serve. However, as soon as the first packet arrives, it starts a timer of fixed duration but continues the sleep mode. As soon as the time elapses, the entity wakes up to serve the arrived packets. During service if another packet arrives, it (and all subsequent packets) have to wait for the next wake-up turn. Therefore, the packet arrival process at the RSU and the vacations are independent of each other and are General distributed. Thus, a G/G/1/K G-vacation queuing model is used for the RSU with sleep cycles type-III. Here, G stands for General distributed vacations, whose mean and variance are determined by the timer duration and packet arrival process.

The implemented sleep cycles type-III algorithm works as follows: If a packet arrives during a sleep cycle, the RSU accumulates the arrived packets for a certain time duration D_{max}^{RSU}, which starts from the arrival of the first packet of a new burst. During the accumulation process, the RSU remains in sleep mode. Upon reaching D_{max}^{RSU} of the first packet, the RSU wakes up and starts serving the burst (i.e. all the packets that arrived during the sleep mode (packet a and b in Figure 5). While the RSU is in service of the burst, if another packet arrives (packet c in Figure 5), the waiting timer starts. Upon service completion, if the waiting timer has reached time duration D_{max}^{RSU}, the RSU starts serving the waiting burst (packet c and d in Figure 5). Otherwise, it switches to sleep mode until the time duration D_{max}^{RSU} elapses. These types of sleep cycles limit the maximum individual packet delay.

Sleep Cycles Type-IV (Modified time based burst accumulation): In these types of sleep cycles, an entity (e.g. RSU) switches to sleep mode when there is no packet to serve. As soon as the first packet arrives, it starts a timer of fixed duration but continues the sleep mode. As soon as the time elapses, the entity wakes up to serve the packets. So far, this is identical to the sleep cycles type-III. However, the main difference between type-III and type-IV is that type-III switches back to sleep mode after serving a burst while type-IV continues serving until there is no packets waiting to be served before switching back to sleep mode. Similarly, the RSU with these sleep cycles are modelled as a G/G/1/K G-vacation queue as well but with different vacations mean and variance.

From the implementation point of view, these types of sleep cycles operate similar to the sleep cycles type-III with the following exception. Upon service completion, if the RSU finds even one packet in its buffer, it starts serving that packet and the subsequent packets (e.g. packet a, b ,c in Figure 5), which may have arrived during the service even if the time duration D_{max}^{RSU} for the first packet has not elapsed. Thus, unlike sleep cycles type-III, the RSU, while in service, does not switch to sleep mode until all the packets that have arrived are served regardless of the waiting timer reaching D_{max}^{RSU}. For example in Figure 5, packets a, b, c are served with type-IV instead of packets a, b in type-III.

Sleep Cycles Type-V (Length based burst accumulation): In these types of sleep cycles, an entity (e.g. RSU) switches to sleep mode when there is no packet to serve. As soon as the first packet arrives, it starts a counter to count the number of packets in the buffer but continues the sleep mode. As soon as the buffer length reaches a certain number of packets, the entity wakes up to serve that many packets and switches back to sleep mode until another burst of the packets is accumulated. The RSU with these sleep cycles are modelled as a G/G/1/K G-vacation queue with different parameters.

In sleep cycles type-V, the RSU accumulates L packets in its buffer before it starts processing them. During the accumulation process, the RSU remains in sleep mode. Upon service completion, if another L packets wait in the buffer, the RSU serves them as well. Otherwise, it switches to sleep mode until L packets are accumulated. It is intuitive that sleep cycles type-V deteriorate the delay of some sparsely arriving packets (at low load) at the expense of higher sleep duration (hence, higher energy savings). For example in Figure 5, the RSU serves the packets a, b, c, d and then switches to sleep mode. It wakes up only when another burst (packet e, f, g, h) is accumulated.

Sleep Cycles Type-VI (Modified length based burst accumulation): This is similar to type-V. However, after serving the predetermined number of packets, it does not switch to sleep mode till the buffer is empty. These sleep cycles are modelled as a G/G/1/K G-vacation queue with different parameters.

These types of sleep cycles operate similar to sleep cycles type-V with the following exception. Upon service completion, if the RSU finds even one packet in its buffer, it starts serving that packet (and subsequently all the remaining packets in the burst even if it did not accumulate L packets. Thus, it does not switch to sleep mode until all the packets that have arrived are served, similar to the case of sleep cycles type-IV. For example in Figure 5, the RSU serves the packets a, b, c, d, e and then switches to sleep mode. It wakes up to serve another burst (packet f, g, h, i).

Implementation setup for the RSU with various types of sleep cycles: The implementation of sleep cycles type-III/IV and type-V/VI at the RSU are shown in Algorithm 1 and Algorithm 2, respectively where the RSU can either be in service or in sleep mode. The algorithms accumulate individual packet delay d, sleep count N_S' and sleep duration S_d to obtain average packet delay W' and net energy savings E_{net}^{RSU}. The input to Algorithm 1 is the maximum individual packet delay D_{max}^{RSU} at the RSU whereas the input to Algorithm 2 is the number of accumulated packets L, required to wake up the RSU.

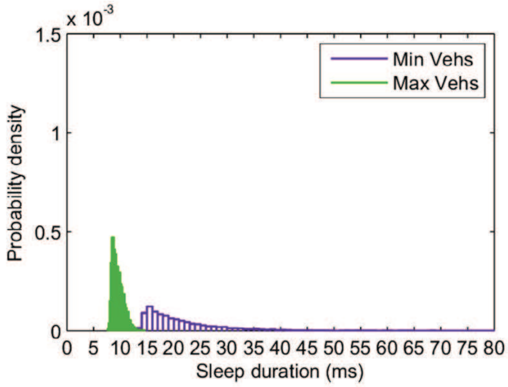
In Algorithm 1, the RSU remains in sleep mode accumulating sleep duration S_d until the first arrived packet has waited up to D_{max}^{RSU} . The RSU switches to service mode and starts serving the packet(s) in the buffer. After serving the burst, the RSU switches back to sleep mode regardless of newly arrived packets (while the RSU was in service) unless the newly arrived first packet has waited up to D_{max}^{RSU} in the case of sleep cycles type-III. In contrast in the case of sleep cycles type-IV, the RSU does not switch back to sleep mode until the buffer is completely empty (i.e. served all the packets that have newly arrived while in service regardless of their waiting times). Here, P_t^{RSU} is the transmission power while E_{w0}^{RSU} is the wakeup overhead.

As shown in Algorithm 2, the RSU remains in sleep mode until the buffer accumulates L packets. The RSU switches to service mode and starts serving L packets in the buffer. If the buffer accumulated another burst of L packets before the RSU finishes serving the previous L packets, the RSU will also serve that burst before switching back to sleep mode. If the number of accumulated packets was less than L at the time the RSU finished serving the previous burst, the RSU switches to sleep mode without serving newly arrived packets in the case of sleep cycles type-V. On the contrary in the case of sleep cycles type-VI, the RSU does not switch back to sleep mode until the buffer is completely empty (i.e. served all the newly arrived packets while in service regardless of their number). In each sleep cycles type, sleep counter N_S' determines the number of times the RSU switches to sleep mode. This accounts for the energy overhead associated with respective sleep cycles types.

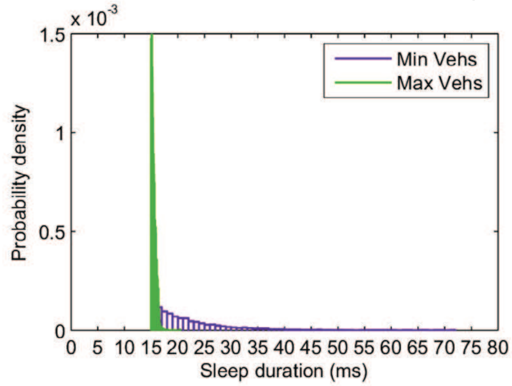
Algorithm 1: Sleep cycles type-III/IV.

```
input : D_{max}^{RSU}
output: W', N_S', E_{net}^{RSU}
begin
    S_d, N_S', d, t, NoOfPackets ← 0;
    while all packets NOT served do
        if RSU in SLEEP then
            increment S_d;
            if packet(s) arrived then
                increment t,d;
                if t = D_{max}^{RSU} then
                    set RSU to SERVICE;
                    t ← 0;
                end
            end
        else if RSU in SERVICE then
            start serving arrived packet(s);
            increment d;
            if arrived packet(s) served then
                set RSU to SLEEP;
                increment N_S';
            else
                if new packet(s) arrived then
                    case SLEEP TYPE-III:
                        increment t;
                        if t = D_{max}^{RSU} then
                            start serving newly arrived packet(s);
                            t←0;
                        end
                    end
                    case SLEEP TYPE-IV;
                        if arrived packet(s) served then
                            start serving newly arrived packet(s);
                        end
                    end
                end
            end
        end
        increment NoOfPackets;
    end
    compute W' = d / NoOfPackets;
    compute E_{net}^{RSU} = (S_d · P_t^{RSU}) − (N_S' · E_{wo}^{RSU});
end
```

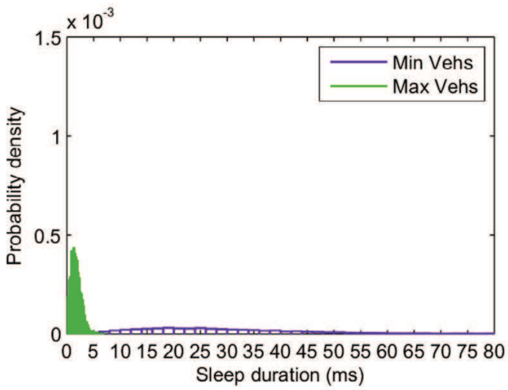
Models of sleep cycles type-III, type-IV, type-V and type-VI: The vehicles in the motorway generate Poisson traffic with General distributed packet sizes. A MAC (Medium Access Control) protocol was implemented by simulation in the CHs where each vehicle is allocated a time slot. This allowed the packets at the output of the CHs to be determined, which is the input to the RSU. Then the sleep cycles were implemented in the simulation to generate 3000 samples of sleep durations of each type (III, IV, V and VI) with minimum and maximum number of vehicles (which correspond to extreme load conditions) as shown in Figure 6.



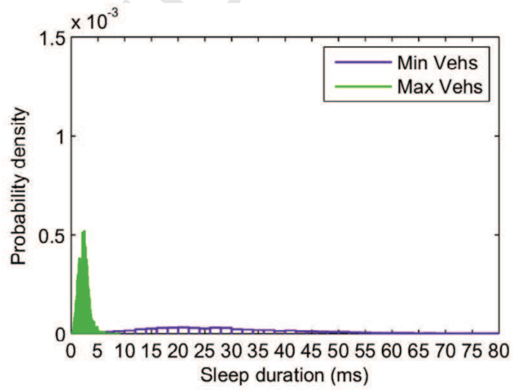
(a) Sleep Cycles Type-III.



(b) Sleep Cycles Type-IV.



(c) Sleep Cycles Type-V.



(d) Sleep Cycles Type-VI.

Figure 6. Sleep cycles data of the RSU. Minimum and maximum numbers of vehicles are 4 and 37 which corresponds to the lightest and heaviest traffic conditions respectively along the 1 km stretch of the motorway.

With the minimum number of vehicles, the inter arrival time for packets is large and randomly distributed (see *Appendix A*). Thus in such load, the first packet of a burst waits for a maximum of D_{max}^{RSU} (i.e. $15\ ms$) before being serviced in sleep cycles type-III and IV. Hence, the corresponding sleep distributions are spread beyond 15 ms as shown in Figure 6(a) and Figure 6(b). In case of sleep cycles type-III, a small portion of the sleep distribution spreads below $15\ ms$. This is due to the rare occurrence of packet arrival during service, which causes the RSU to sleep less than $15\ ms$. This is also illustrated in the timing diagram (Figure 5). In contrast, with the maximum number of vehicles, the probability of having a packet arrival as soon as the RSU sleeps is very high due to frequent packet arrival. This results in the distribution of sleep type-III to be steeper with a spread not exceeding $15\ ms$. Hence, the mean sleep duration becomes less than $15\ ms$. In sleep cycles type-IV, the RSU does not sleep until the buffer is empty as shown in Figure 5. This results in the RSU always sleeping greater than or equal to D_{max}^{RSU} (i.e. $15\ ms$), as shown in Figure 6(b), regardless of the traffic load i.e. the number of vehicles. Since the packet arrival is very frequent with the maximum number of vehicles, a packet arrives as soon as the RSU switches to sleep mode resulting in RSU mostly sleeping for D_{max}^{RSU} (i.e. $15\ ms$). Therefore, the sleep distribution is steep with a narrow spread, which indicates that the sleep process tends to be mostly deterministic. In sleep cycles type-V and type-VI, the RSU, while in sleep mode, accumulates a specific number of packets regardless of their arrival time (please refer to Figure 5). With the minimum number of vehicles in sleep cycles type-V and type-VI, the spread of the sleep distributions is large because the inter-arrival time is large and randomly distributed (see Appendix A), whereas, with the maximum number of vehicles, the RSU quickly accumulates L packets. Thus, the mean sleep duration reduces considerably as shown in Figure 6(c). With sleep cycles type-VI, the RSU cannot sleep until the buffer is empty. This marginally shifts the sleep distribution curve to the left as shown in Figure 6(d), resulting in a slightly lower mean sleep duration compared to that of the sleep cycles type-V.

Since the packet arrival process is General distributed (as shown in Appendix A) and the sleep mechanism depends upon arrival and service characteristics, the sleep durations of each type do not follow any specific distribution and are thus General distributed. Modelling sleep pdfs with accurate distributions increases computational complexity. Further, the model for the RSU loses its generality as it becomes sleep type dependent. Thus, we use a probabilistic mixture of Deterministic and Negative exponential (Det-Neg) distribution to represent the sleep pdf of these proposed types of sleep cycles. Although it is not a full re-creation of the distribution as it uses only the first two moments of a distribution rather than the whole distribution, but it makes the model generic and provides a workable engineered solution. The Det-Neg distribution of sleep cycles can be expressed as

$$f_S^{GEN}(\bar{S}, p, t) = p\delta(t - \bar{S}) + (1 - p)\frac{1}{\bar{S}}\, e^{-\frac{t}{\bar{S}}} \quad (20)$$

where p denotes the proportion of mixture $(0 \leq p \leq 1)$ i.e. the degree of superposition of Deterministic and Negative exponential pdf, and δ is the Dirac-delta function.

Algorithm 2: Sleep cycles type-V/VI.

```
input : L
output: W', N'_S, E_net^RSU
begin
    S_d, N'_S, d, N_p, NoOfPackets ← 0;
    while all packets NOT served do
        if RSU in SLEEP then
            increment S_d;
            if packet(s) arrived then
                increment N_p, d;
                if N_p = L then
                    set RSU to SERVICE;
                    N_p ← 0;
                end
            end
        else if RSU in SERVICE then
            start serving arrived packet(s);
            increment d;
            if arrived packet(s) served then
                set RSU to SLEEP;
                increment N'_S;
            else
                if new packet(s) arrived then
                    case SLEEP TYPE-V:
                        increment N_p;
                        if N_p = L then
                            start serving newly arrived packet(s);
                            N_p ←0;
                        end
                    end
                    case SLEEP TYPE-IV;
                        if arrived packet(s) served then
                            start serving newly arrived packet(s);
                        end
                    end
                end
            end
        end
        increment NoOfPackets;
    end
    compute W' = d/NoOfPackets;
    compute E_net^RSU = (S_d · P_t^RSU) − (N'_S · E_wo^RSU);
end
```

The mean and variance of the sleep cycles can be obtained as

$$Mean\ (f_S^{GEN}(\bar{S}, p, t)) \ = \ p\bar{S} + (1-p)\bar{S} = \bar{S} \tag{21}$$

and

$$Var\,(f_S^{GEN}(\bar{S},p,t)) \;=\; p.Var(\delta(t-\bar{S})) + (1-p)\bar{S}^2$$
$$=\; (1-p)\bar{S}^2 \qquad (22)$$

respectively. From the sleep cycles data, we compute mean and variance of each type of sleep cycles corresponding to minimum, average, maximum number of vehicles and in each case find out p by solving (21) and (22).

B. Analytic Model for the RSU with Various Types of Sleep Cycles

The packets generated by the vehicles are of different sizes i.e. 64 Byte (ACK), 580 Byte (Video), 1500 Byte (maximum Ethernet size) etc. [20]. They do not follow any specific distribution and are thus considered General distributed. As we recall from Figure 2(a), each CH is given a time slot to communicate in. Therefore, the traffic arriving at the RSU is a mixture of the two traffic streams, where each stream is General distributed. The arrival process is shown in *Appendix A.* Similarly, the service duration of the packets is also general distributed (non-identical to the arrival process). Hence, the RSU is modelled as a queue of type $G/G/1/K$ as shown in Figure 7. Furthermore, the sleep cycles at the RSU are also general distributed as discussed earlier. As mentioned earlier, these sleep cycles can be represented by General distributed vacations from a performance modelling perspective. The exact solution of such queuing model with General distributed vacations is not analytically possible, to the best of our knowledge. In the literature, only approximate bounds on average packet delay are obtained under different traffic conditions for much simpler $G/G/1$ queues and $M/G/1/K$ queues with vacations [28], [29]. Therefore, in this section, we obtain analytic bounds for the average packet delay and energy savings of the $G/G/1/K$ G-vacation queue (representing the proposed RSU).

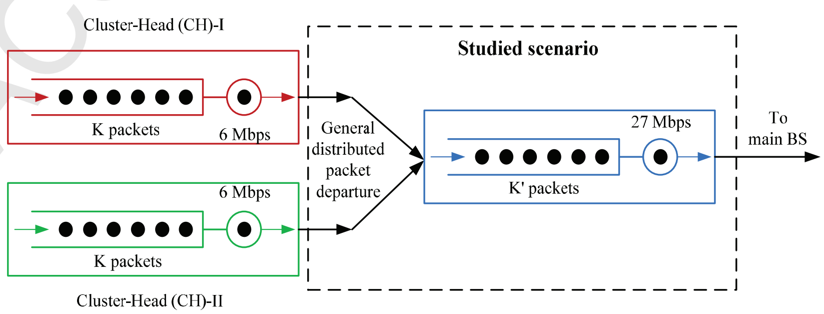


Figure 7. Queuing model for the RSU.

The analysis presented in Section 4-A yields the system (CH) states at the embedded points. Since the packet arrivals were considered memoryless Poisson process (holds the PASTA property [30]) and the random service durations were initially considered in Section 4-A, the system state at any arbitrary time point becomes identical to that of the embedded time point. However, in the case of General packet arrival and General service discipline, the state probabilities at any arbitrary time point are different from that of an embedded time point, therefore, need to be obtained.

For General distributed packet inter-arrival time, let the probability of the j^{th} packet arrival be expressed as $G_{IAT}\ (j, \lambda', t)$ where the mean arrival rate of packets is denoted as λ' so that the mean number of arrivals during time t becomes $\lambda' t$. The General distributed sleep cycle duration can be represented with mean $\bar{S}$, pdf $f_S^{GEN}\ (t)$ and cumulative distribution function (cdf) $f_S^{GEN}\ (t)$. Furthermore, the General distributed service duration of the packets can be represented with mean $\bar{X}\ = \mu^{-1}$, pdf $b\ ^{GEN}(t)$ and cdf $B\ ^{GEN}(t)$. We assume that the RSU has a large buffer, which can hold a maximum of K' packets (including the packet being served). Let the state probabilities, $Q_k\ (k\ =\ 0, 1, \ldots, K')$ and $R_k\ (k\ =\ 1, \ldots, K')$ be defined at any arbitrary time instant as

$$Q_k = Prob\ \{\ k \text{ packets in the system; given the server is currently in a sleep cycle}\}$$
$$R_k = Prob\ \{\ k \text{ packets in the system; given the server is currently in service}\}$$

The probability of j or more packet arrivals at the RSU during a sleep cycle can be expressed as

$$\begin{aligned} F_j &= \sum_{i=j}^{\infty} f_i \\ &= \sum_{i=j}^{\infty} \int_0^{\infty} G_{IAT}(i, \lambda', t) f_S^{GEN}(t) dt \\ &= \int_0^{\infty} G_{IAT}(j-1, \lambda', t) \left[\frac{1 - F_S^{GEN}(t)}{\bar{S}}\right] \lambda' dt \end{aligned} \tag{23}$$

Therefore,

$$\sum_{j=1}^{\infty} F_j = \sum_{j=1}^{\infty}\sum_{i=j}^{\infty} f_i$$
$$= (f_1 + f_2 + \cdots + \infty) + (f_2 + f_3 + \cdots + \infty)$$
$$+ (f_3 + f_4 + \cdots + \infty) + \cdots + \infty$$
$$= f_1 + 2f_2 + 3f_3 + \ldots + \infty$$
$$= \sum_{i=1}^{\infty} i f_i$$
$$= \lambda' \bar{S}$$

(24)

which is the expected number of arrivals during a sleep cycle. Since, the pdf of any time interval x within a sleep cycle can be expressed using the residual time argument as $\left[\frac{1-F_S^{GEN}(t)}{\bar{S}}\right]$. Similarly, the probability of j or more packet arrival at the RSU during a service duration can be expressed as

$$A_j = \sum_{i=j}^{\infty} \alpha_i$$
$$= \sum_{i=j}^{\infty} \int_0^{\infty} G_{IAT}(i, \lambda', t) b^{GEN}(t) dt$$
$$= \int_0^{\infty} G_{IAT}(j-1, \lambda', t) \left[\frac{1-B^{GEN}(t)}{\bar{X}}\right] \lambda' dt$$

(25)

Therefore,

$$\sum_{j=1}^{\infty} A_j = \sum_{j=1}^{\infty}\sum_{i=j}^{\infty} \alpha_i$$
$$= \sum_{i=1}^{\infty} i\alpha_i$$
$$= \lambda' \bar{X}$$
$$= \rho$$

(26)

where ρ is the offered load. Equations (23) and (24) represent new transition probabilities as the arrival is now not Poisson distributed. Therefore, the evaluation of the transition probabilities at the embedded points is not valid for all time points. In order to determine Q_k, let us consider any arbitrary time epoch τ within a sleep cycle such that there are k packet arrivals during the time interval x between the start of the sleep cycle and τ as shown in Figure 8. The probability of k arrivals in time duration x is $G_{IAT}\,(j, \lambda', x)$. The pdf of the time interval x can be expressed from the residual life property as $\frac{1-F_S^{GEN}(x)}{\bar{S}}$ [30]. The probability of selecting a sleep cycle is $(1-\rho_c)$, where ρ_c is the carried load.

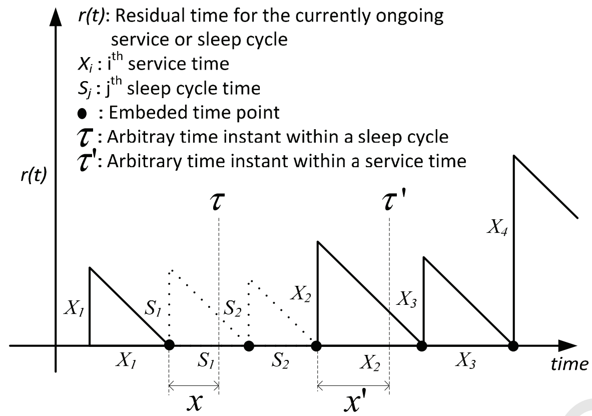


Figure 8. Timing diagram for the G/G/1 queuing model of the RSU.

Therefore,

$$Q_k = (1-\rho_c)\int_0^\infty G_{IAT}(k,\lambda',x)\left[\frac{1-F_S^{GEN}(t)}{\bar{S}}\right]dx \qquad k=0,\dots,K'-1 \qquad (27)$$

$$= (1-\rho_c)\sum_{k=K'}^{\infty}\int_0^\infty G_{IAT}(k,\lambda',x)\left[\frac{1-F_S^{GEN}(t)}{\bar{S}}\right]dx \qquad k=K \qquad (28)$$

$$= \left(\frac{1-\rho_c}{\lambda'\bar{S}}\right)F_{k+1} \qquad k=0,\dots,K'-1 \qquad (29)$$

$$= \left(\frac{1-\rho_c}{\lambda'\bar{S}}\right)\left(\sum_{k=K'+1}^{\infty}F_k\right) \qquad k=K' \text{ [from (23)]} \qquad (30)$$

Using (3) and (4), we write

$$\sum_{j=k}^{K'} q_j = (q_0+r_0)F_k \qquad k=1,\dots,K' \qquad (31)$$

Using (31), (30) can be rewritten as

$$Q_k = \left(\frac{1-\rho_c}{\lambda'\bar{S}}\right)\frac{1}{(q_0+r_0)}\sum_{j=k+1}^{K'} q_j \qquad k=0,\dots,K'-1$$

$$= \frac{(q_0+r_0)\bar{S}}{\lambda'\bar{S}[(1-q_0-r_0)X'+(q_0+r_0)]}\frac{1}{(q_0+r_0)} \qquad \text{[from (12)]} \qquad (32)$$

$$\sum_{j=k+1}^{K'} q_j = \frac{1}{\lambda' D}\sum_{j=k+1}^{K'} q_j \qquad k=0,\dots,K'-1 \qquad (33)$$

[from (15)]

Using (31), we simplify (30) for $k=K'$ to obtain

$$Q_k = \frac{1-\rho_c}{\lambda'\bar{S}}\left[\lambda'\bar{S} - \frac{1}{q_0+r_0}\sum_{k=1}^{K'}\sum_{j=k}^{K'} q_j\right] \qquad k=K' \qquad (34)$$

$$= 1-\rho_c - \frac{1}{\lambda' D}\sum_{j=1}^{K'} j q_j \qquad k=K' \text{ [from (12) and (15)]} \qquad (35)$$

To find R_k, we similarly consider any arbitrary time epoch τ' within a service duration such that there are k packets arrivals during the time interval x' between the start of the service period and τ' as shown in Figure 8. Now, the probability of selecting a service duration is ρ_c. Given that the service period has to start with a non-empty queue, the probability of it starting with j packets in the system is $\frac{q_j+r_j}{1-q_0-r_0}$ for $j = 1,\dots,K'-1$ or $\frac{q_{K'}}{1-q_0-r_0}$ for $j=K'$. The probability of $k-j$ arrivals of packets in the time interval x' can be given as $G_{IAT}\ (k-j,\lambda',x')$.The pdf of the time interval x' can be expressed from the residual life property [30] as $\frac{1-B^{GEN}(x')}{\bar{X}}$.

Therefore,

$$R_k = \rho_c \sum_{j=1}^{k} \left(\frac{q_j+r_j}{1-q_0-r_0}\right) . \int_0^\infty G_{IAT}(k-j,\lambda',x') \left[\frac{1-B^{GEN}(x')}{\bar{X}}\right] dx' \qquad k=1,\dots,K'-1 \tag{36}$$

$$= \rho_c \left(\frac{q_k}{1-q_0-r_0}\right) + \rho_c \sum_{j=1}^{K'-1} \left(\frac{q_j+r_j}{1-q_0-r_0}\right) \left(\sum_{k=K'-j}^{\infty} \int_0^\infty G_{IAT}(k,\lambda',x') \left[\frac{1-B^{GEN}(x')}{\bar{X}}\right] dx'\right) \qquad k=K \tag{37}$$

Using $\rho = \lambda'\bar{X}$ and (25), we simplify (37) as

$$R_k = \frac{\rho_c}{\rho} \sum_{j=1}^{k} \left(\frac{q_j+r_j}{1-q_0-r_0}\right) A_{k-j+1} \qquad k=1,\dots,K'-1 \tag{38}$$

$$= \rho_c \left(\frac{q_{K'}}{1-q_0-r_0}\right) + \left(\frac{\rho_c}{\rho} \sum_{j=1}^{K'-1} \left(\frac{q_j+r_j}{1-q_0-r_0}\right) \sum_{k=K'-j+1}^{\infty} A_k\right) \qquad k=K \tag{39}$$

Using the values of $q_j\ (j=0,\dots,K')$ and $r_j\ (j=0,\dots,K'-1)$ obtained earlier in Section 4-A, which denote the state probabilities at the embedded points, (33), (35) and (39), we obtain $Q_j\ (j=0,\dots,K')$ and $R_j\ (j=1,\dots,K')$ at any arbitrary time instant. For a system in equilibrium, we define state probability $p_j\ (j=0,\dots,K')$ as the probability of the system in state j at any arbitrary time instant. Therefore, p_j can be obtained as

$$p_j = \begin{cases} Q_j & j=0 \\ Q_j + R_j & j=1,\dots,K'. \end{cases} \tag{40}$$

These probabilities can now be used to find the usual QoS parameters. Therefore, by definition, the packet blocking probability can be expressed as

$$P_B' = p_{K'}. \tag{41}$$

Table 2: RSU Parameters

Parameter	Notation	Value
RSU data rate	d_r^{RSU}	27 Mbps [24][9]
RSU packet service rate	μ^{RSU}	$d_r^{RSU}/(P_S \times 8)$ packets/s
RSU max. operational power	P_{MAX}	$30W$ [24]
Min. operational power	P_{MIN}	$P_{\text{MAX}}/1.3548$ [5]
Transmit power	P_t^{RSU}	$P_{\text{MAX}} - P_{\text{MIN}}$ [5]
Energy for wake-up overhead	$E_{\omega o}^{RSU}$	$0.0175\ J$ [9]

The average number of packets in the system can be obtained as

$$N' = \sum_{j=1}^{K'} j\, p_j \tag{42}$$

Using Little's Theorem, the average packet delay is obtained as

$$W' = \frac{N'}{\lambda' (1-P_B')} \tag{43}$$

Since a fraction P_B' of the arrivals is blocked due to the buffer overflow at the RSU, the utilisation of the system can be obtained as

$$U' = \lambda' \bar{X}(1 - P_B') \tag{44}$$

The *gross transmission energy savings* of a RSU per hour through sleep cycles is expressed as

$$E_{gross}^{RSU} = (1 - U') \cdot P_t^{\mathrm{RSU}} \cdot 3600 \tag{45}$$

where P_t^{RSU} denotes the transmitter power of the RSU, as given in Table 2. The model developed for the RSU is generic and thus has expressions for packet blocking, delay and utilisation as well. For burst accumulation / traffic shaping, the RSU buffer should be very large, which is taken into account in the whole paper. Note that the blocking constraint (0.05) is there at the cluster head but not at the RSU. In RSU the blocking experienced is zero in the simulation for any case.

The energy model utilised in the paper is based on [31] where the energy per bit (in the transmission state) can be determined, if required, as the ratio between the transmitter power and the data rate [9]. The maximum operational power and minimum operational power (during sleep cycles) are also given in Table 2. The total energy overhead of a RSU in an hour depends upon two parameters, (i) wake-up overhead associated with each sleep cycle (E_{wo}^{RSU}) and (ii) the number of times the RSU sleeps and wakes up, i.e. sleep count (N_S'). The value of wake-up overhead, E_{wo}^{RSU} (see Table 2), accounts for system initialisation, frequency management, synchronisation, routing table updates etc. [9]. Since N_S' becomes a large quantity in an hour compared to E_{wo}^{RSU}, it plays a dominant part in the transmission energy efficiency of a RSU. Analytically N_S' can be obtained as

$$N_S' = \frac{(q_0 + r_0) \cdot 3600}{\bar{S}} \tag{46}$$

Considering the wake-up overhead for the RSU, the net transmission energy savings per hour through sleep cycles can be obtained as

$$E_{net}^{RSU} = (1 - \bar{U}) \cdot P_t^{RSU} \cdot 3600 - (E_{wo}^{RSU} \cdot N_S') \tag{47}$$

Since each wake-up process of an RSU incurs energy overhead [9], the sleep count for sleep cycles should be reduced to achieve maximum energy savings.

Note that the vehicles generate requests, which are routed to the roadside unit (RSU) through cluster head vehicles. Thus, the arrival process of the packet stream at the cluster head is Poisson distributed. However, due to the different packet sizes, the output traffic of the cluster heads are General distributed (at the RSU). Thus, we developed a Generic model, which is complex. The same model can be applicable to the ordinary case of Poisson distributed traffic and correspondingly the generic model can be simplified as well.

6. Results and Discussions

In this section, we evaluate the performance of the RSU in terms of average packet delay (W'), sleep count (N_S') and energy savings (E_{net}^{RSU}) for each type of sleep cycles discussed in Section 5-A. Note that the model of the RSU has expressions for packet blocking probability, delay and utilisation. For burst accumulation / traffic shaping, the RSU buffer should be very large, which is taken into account in the whole paper. In addition, note that the blocking constraint (0.05) is there at the cluster head but not at the RSU. In RSU, the blocking experienced is negligible for any case. We verified this through extensive simulation cases.

Hence, the packet blocking probability at the CH is also the end-to-end packet blocking probability, which was maintained within the 5% bound ($P_B \leq 0.05$) by introducing sleep cycles type-II in Section 4-B. As mentioned earlier an acceptable end-to-end individual packet delay D_{max}^{E2E} (i.e. $32\,ms$ [11]) is to be maintained to successfully deliver audio-conferencing. Therefore, first we need to find the maximum individual packet delay at the CH (D_{max}^{CH}) so that the maximum allowable individual packet delay at the RSU (D_{max}^{RSU}) can be obtained and accordingly energy savings at the RSU (E_{net}^{RSU}) can be maximised.

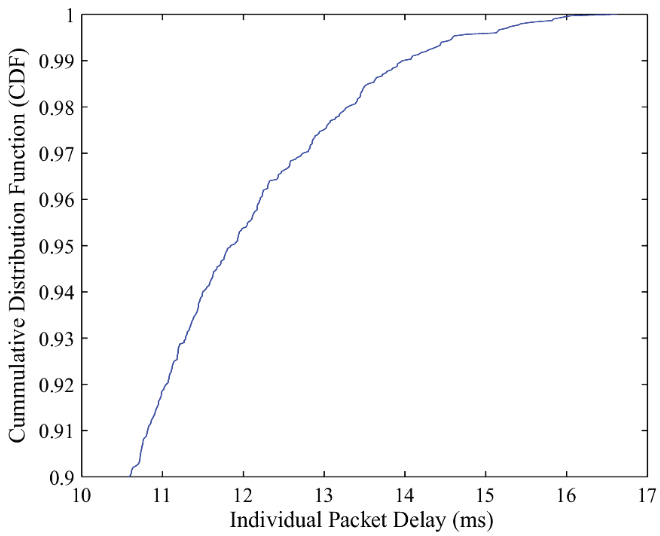


Figure 9. Individual packet delay at the CH for maximum number of vehicles.

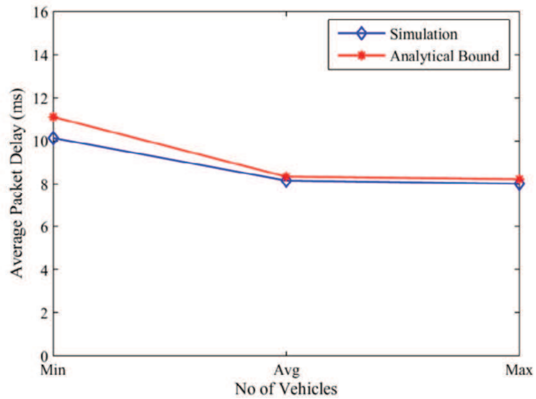
Figure 9 shows that all the packets experienced less than $16.5\ ms$ delay at the CH. Moreover considering the data rate of the RSU, the service time for the largest packet (1500 bytes [20]) is $\cong\ 0.5\ ms$. Therefore, D_{max}^{RSU} is set to $15\ ms$ to maximise energy savings at the RSU.

A. Model validation for sleep cycles type-III, type-IV, type-V, type-VI at the RSU

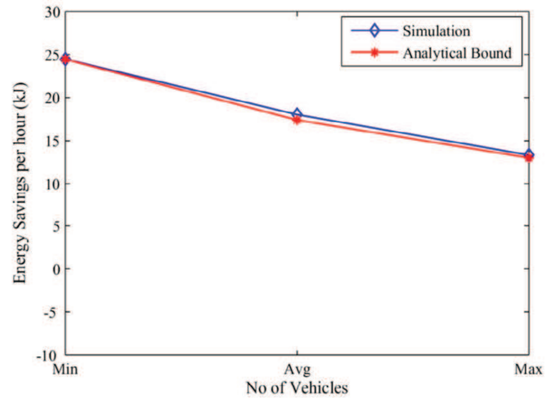
Without loss of generality, the packet arrival pdf at the RSU can be analytical bounded by an envelope of exponential pdf if we consider the worst case scenario (i.e. the maximum number of vehicles) to predict the bound on average packet delay and energy savings. We estimate the upper bound on average packet delay to maintain strict QoS. Since energy savings is inversely related to the QoS, the proposed model predicts a (marginally) lower bound on energy savings. The sleep pdfs for the different types of sleep cycles have already been discussed in Section 5-A.

With the parameters specified in Table 2 and $D_{max}^{RSU} = 15\ ms$, the performance, in terms of average packet delay and energy savings, of sleep cycles type-III and type-IV are respectively evaluated for minimum, average and maximum number of vehicles, as shown in Figure 10(a), Figure 10(b), Figure 10(c) and Figure 10(d). Noticeably, the average packet delay decreases with increase in load because the

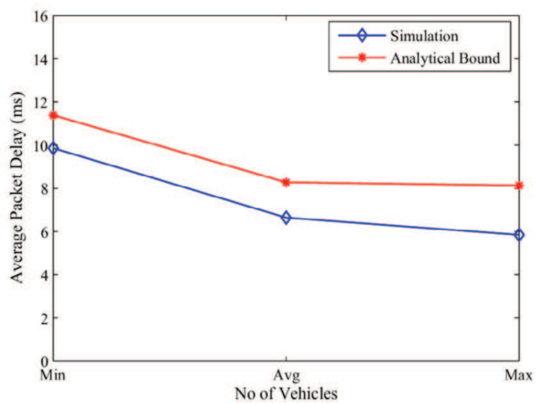
RSU sleeps for a shorter duration and switches to sleep mode less frequently resulting in a lower average packet waiting time. However as expected, energy savings decrease with an increase in load. In the case of sleep cycles type-IV, marginally lower average packet delay was achieved compared to the case of sleep cycles type-III. The explanation for this is discussed in Section 6-B.



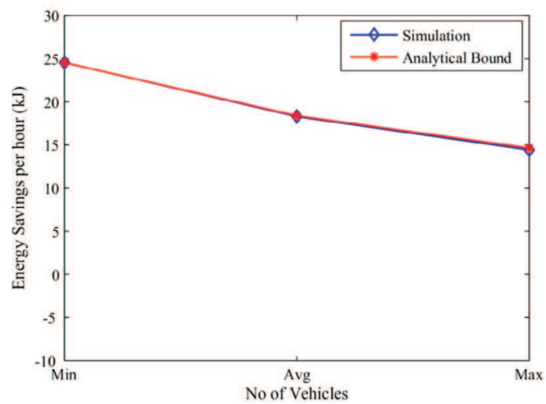
(a) Average packet delay (sleep cycles type-III).



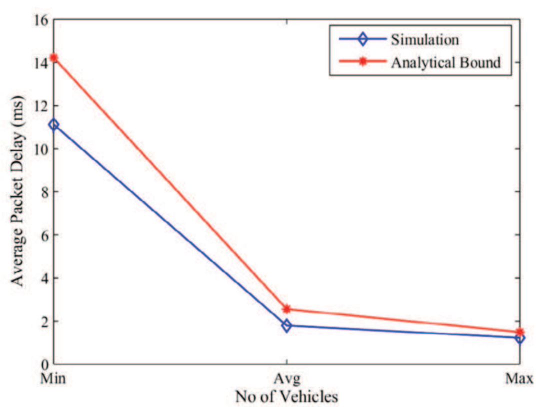
(b) Energy savings (sleep cycles type-III).



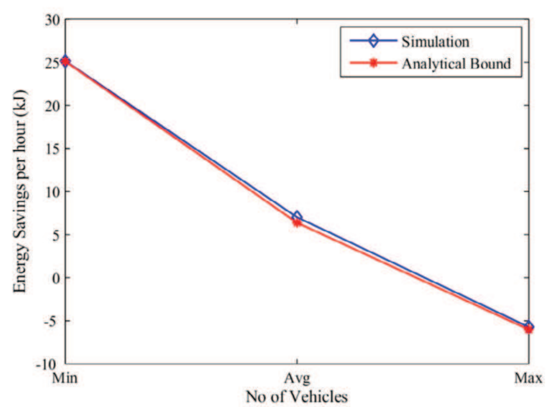
(c) Average packet delay (sleep cycles type-IV).



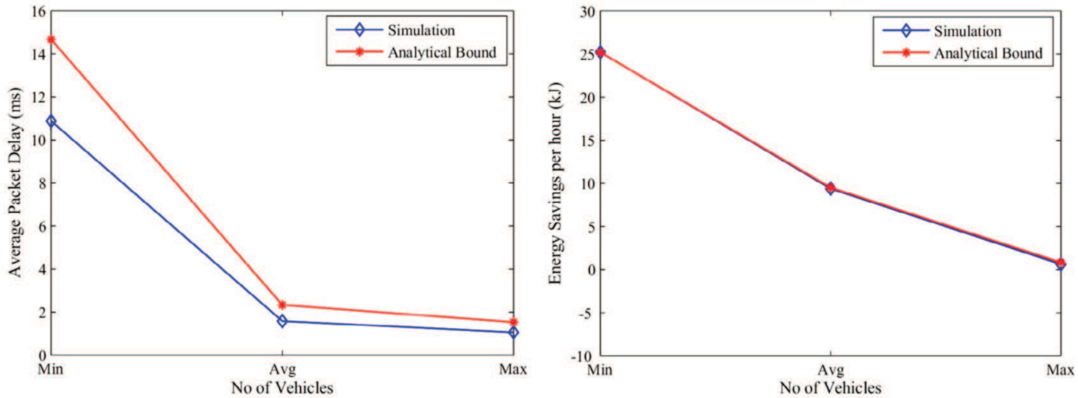
(d) Energy savings (sleep cycles type-IV).



(e) Average packet delay (sleep cycles type-V).



(f) Energy savings (sleep cycles type-V).



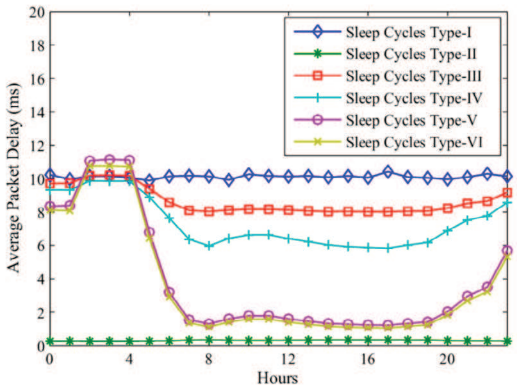
(g) Average packet delay (sleep cycles type-VI). (h) Energy savings (sleep cycles type-VI).

Figure 10. Model validation.

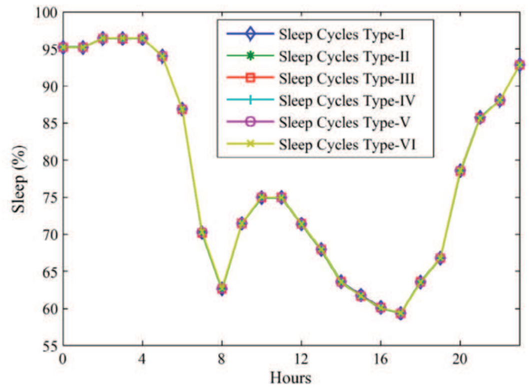
With $D_{max}^{RSU} = 15\,ms$, the highest W' observed with sleep cycles type-III was $10\,ms$ (in simulations). Hence, we set 10 ms average packet delay as a reference value. We use this to obtain the parameter $L = 4$ for type-V and VI for a fair comparison, as shown in Appendix B. It is evident that the results in Figure 10 for the different types of sleep cycles are nearly identical since we have chosen the parameters equivalent to each other (for type-III and type-V). The trends of average packet delay and energy savings in the case of sleep cycles type-V and VI (see Figure 10(e), Figure 10(f), Figure 10(g) and Figure 10(h)) are similar to that of type-III and IV.

B. Performance of sleep cycles type-III, type-IV, type-V, type-VI at the RSU

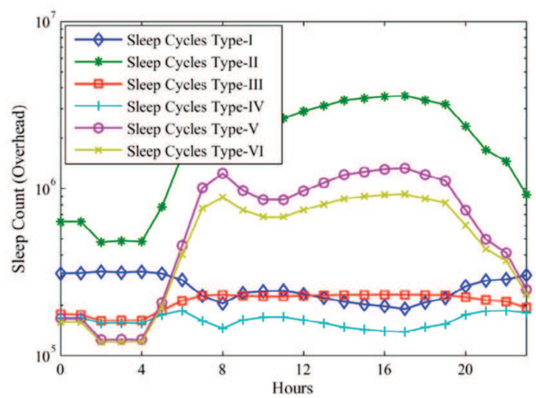
We evaluate the performance of the RSU in terms of average packet delay and energy savings for different types of sleep cycles (type-I to type-VI). With $D_{max}^{RSU} = 15\,ms$, the highest average packet delay (W') observed with sleep cycles type-III was $10\,ms$ (in simulations). Hence, we set $10\,ms$ average packet delay as a reference value to obtain the parameters $\bar{S} = 11\,ms$ for type-I, and $L = 4$ for type-V and VI for fair comparison, as shown in *Appendix B.*



(a) Average packet delay at the RSU.



(b) Sleep percentage at the RSU.



(c) Sleep count (overhead) at the RSU.

Figure 11. Performance of the RSU for sleep cycles type-I to type-VI.

Figure 11(a) shows the average packet delay at the RSU for sleep cycles type-I to type-VI for varying hourly vehicular load. At low traffic, all types of sleep cycles agree in terms of average packet delay due to the constraint introduced in the parameters requiring the average packet delay not to exceed a specific value. In the case of sleep cycles type-I, W' remains mostly unaffected throughout the day as the average hourly sleep durations are set to maximise energy savings. Therefore, the sleep duration is a function of the vehicular traffic density to yield maximum energy savings while maintaining a certain QoS. For example, at low vehicular traffic, longer sleep duration should be implemented to save maximum energy and vice versa. A detailed discussion can be found in Appendix B. In the case of sleep cycles type-II, the RSU becomes immediately available from sleep when a packet arrives, which means type-II follows the traffic. Therefore, W' in this case is very low (around $0.13 - 0.15\ ms$) throughout the day, as it only reflects the service duration or waiting, due to congestion at some instances. With sleep cycles type-III, W' reaches a maximum of $10\,ms$ during off-peak hours because the first packet in each burst waits D_{max}^{RSU}. However, W' decreases during peak hours due to the following. As the frequent packet arrivals from large bursts, the waiting times of the majority of packets within a burst is low compared to the case of off-peak hours where the burst size is much smaller. This results in a lower average packet delay compared to the case of off-peak hours. This trend is also reflected in the sleep pdf of sleep cycles type-III (Figure 6(a)) where a higher number of vehicles caused the RSU to sleep for shorter durations compared to the case of lower number of vehicles. In the case of sleep cycles type-IV, the RSU does not switch to sleep mode until the buffer is empty unlike the case of sleep cycles type-III. Due to the lower waiting times for newly arrived packets (while the RSU was in service), W' is lower compared to the case of sleep cycles type-III especially during the peak hours. This mechanism allows the RSU to achieve lower W' while sleeping longer with a minimum duration of D_{max}^{RSU}. In the case of sleep cycles type-V, the RSU during off-peak hours takes a considerably longer time to accumulate $L = 4$ packets. Therefore, W' during off-peak hours is significantly higher compared with its value at the peak hours. Sleep cycles type-VI achieve marginal improvements in W' as both type-V and VI are time independent and are governed by the number of accumulated packets. The marginal improvement is due to the fact that the RSU, while in service, continues to serve until the buffer is empty before switching to sleep mode.

Without any packet blocking all packets are eventually served. Therefore, the time to sleep is a function of the traffic pattern only i.e. the ratio of the gaps to the packets. Therefore, all sleep cycles result in the same sleep percentage. It is also to be noted that normal servers either serve or remain in idle state, whereas in this case the RSU either sleeps or serves. The gross energy savings (E_{gross}^{RSU}) at the RSU are inversely proportional to the utilisation, as shown in (50), but directly proportional to the sleep percentage. The sleep percentage denotes the proportion of time the RSU has been in sleep mode. The sleep percentage, as can be seen in Figure 11(b), is identical for all types of sleep cycles because the number of packets served and the data rate of the RSU remain same in all types. Thus, the gross energy savings E_{gross}^{RSU} (without considering any sleep overhead) are equal in each type.

However, the net energy savings (E_{net}^{RSU}), considering sleep overhead, vary with different types of sleep cycles due to the variation in the respective sleep count (N_S). Since, each wake-up incurs energy overhead E_{wo}^{RSU} (see Table 2), the sleep count N_S contributes negatively towards the net energy savings (E_{net}^{RSU}(52)). Revisiting the timing diagram in Figure 5 shows that each type of sleep cycles arranges the service and sleep mode uniquely, which would result in different N_S' for each type.

In sleep cycles type-I, N_S' decreases marginally during peak hours because the RSU switches to sleep less often. In sleep cycles type-II, the RSU wakes up at each packet arrival and switches back to sleep mode after service. This results in significantly higher sleep count compared to the other types as these types of sleep cycles follow packet-level traffic variation. Due to the time based burst accumulation, sleep cycles type-III reduce the sleep count significantly compared to sleep cycles type-II. Compared to sleep cycles type-I, sleep cycles type-III achieve lower N_S' during off-peak hours due to the following reasons: In sleep cycles type-I, the RSU remains in sleep mode for a fixed time duration regardless of packet arrivals or their waiting durations. On the contrary, the RSU in sleep cycles type-III remains in sleep mode until the first arrived packet of a burst has waited for a certain duration. This enables the RSU to switch to sleep mode infrequently compared to the case of sleep cycles type-I, however for longer durations. During peak hours, the mean sleep duration of sleep cycles type-III (see Figure 6(a)) closely matches that of the mean sleep duration of sleep cycle type-I (i.e. $11\ ms$), which results in a similar N_S'. With sleep cycles type-IV, the sleep count decreases considerably compared to sleep cycles type-III during the day because the RSU does not switch to sleep mode until the buffer is empty (as discussed earlier, see Figure 8). In sleep cycles type-V, the time to form a burst of L packets decreases with an increase in load, which in turn increases N_S' compared to sleep cycles type-III and IV. A lower N_S' is achieved during the day in sleep cycles type-VI compared to sleep cycles type-V. This is because the RSU does not switch to sleep mode until the buffer is empty. For a detailed description of sleep cycles types, please refer to Section 5-A.

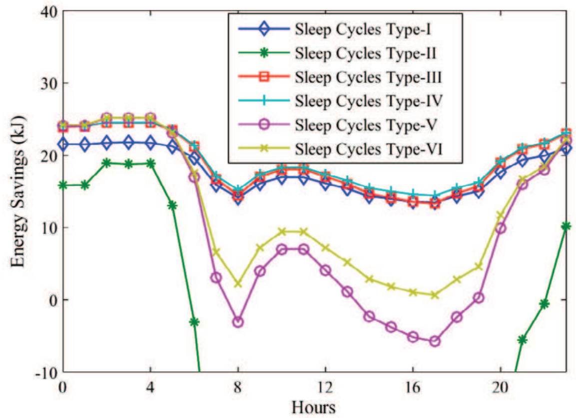


Figure 12. Net energy savings at the RSU.

Taking N_S' for sleep cycles type-I to type-VI and E_{wo}^{RSU} (see Table 2) into account, Figure 12 shows the hourly net energy savings E_{net}^{RSU} at the RSU. Sleep cycles type-I achieve maximum energy savings of $21.8\ kJ$ with an average of $17.6\ kJ$ (62.3%) throughout the day, though it decreases during peak hours to $13.4\ kJ$. With sleep cycles type-II, the energy savings become negative (-57.5% on average) during the majority of the day due to the high switching overhead (see Figure 11(c)). The negative energy savings indicate that the RSU with sleep cycles type-II consume more energy compared to the RSU without any sleep cycle. This reiterates the fact that although these types of sleep cycles improve the average packet delay, they are not suitable for an entity like an RSU due to the high associated switching overhead. Sleep cycles type-III and type-IV significantly improve the average energy savings (66.8% and 68.2%, respectively) compared to sleep cycles type-II. Although the improvement in energy savings of type IV compared to type I is only 10%, the improvement in average packet delay is substantial (i.e. 28%). Due to the very high switching overhead (at peak hour) associated with sleep cycles type-V, the energy savings become negative ($5.7\ kJ$). The mechanism of sleep cycles type-VI improves the energy savings compared to sleep cycles type-V and achieves positive energy savings throughout the day. For example, the average energy savings achieved through sleep cycles type-VI is 43.4% compared to 34.4% in the case of sleep cycles type-V.

Table 3 illustrates a summary of the comparisons of these sleep cycles based on our work.

Table 3

COMPARISONS OF DIFFERENT TYPES OF SLEEP CYCLES.

Sleep mechanism	Energy saving	Delay	Sleep count (for overhead)
1. Random (type-I)	Parameters of the sleep cycles can be adjusted to produce the same results of IV and VI since I, IV and VI do not sleep if there is a packet to serve	Comparable to III,IV with a suitably chosen sleep parameter; identical to II without sleep	Better than II but is comparable to III,IV with a suitably chosen sleep parameter
2. Follow traffic (type-II)	Worst among all for energy savings as sleep is interrupted with every arriving packets	Best among all six for delay	Highest sleep to wakeup overhead as the entity wakes up at every arriving packet
3. Time based (type-III)	Can be made identical to V by selecting a suitable time that generates a burst of N packets	Can be made identical to V by selecting suitable time that generates a burst of N packets	Better than all except IV; comparable to I in some cases
4. Modified time based (type-IV)	IV gives worse saving than III as it does not continue sleep after serving the last packet in burst	IV gives better delay than III	III yields less sleep count than IV
5. Length based (type-V)	Can be made identical to III by selecting N that is accumulated in time t	Can be made identical to III by selecting suitable number of packets that are accumulated in a given time	Dependent on traffic; thus can be made identical to III by selecting N that is accumulated in time t
6. Modified length based (type-VI)	VI gives better energy savings than V as it also serves the additional packets that arrive in service state thereby reducing	VI gives better delay than V	VI yields lower sleep count than V

	the sleep count		

7. Conclusions

In this paper, we introduced five different types of sleep cycles for energy savings at the roadside unit. Each type of sleep cycles arranges the service and sleep sequences distinctively which yields different levels of energy savings and average packet delay. Note that the sleep cycles are tuneable parameters of the RSU, which can be either pre-programmed (remotely) or auto-programmed. The wake-up over head depends upon the entity being put to sleep mode. Depending upon the bulkiness of the entity e.g. BS/RSU/CH, the wake-up overhead varies. Since sleep cycles type-I and II are applied at the cluster-head (vehicle) which performs very limited tasks, the over-head is negligible. On the contrary, the RSU performs synchronisation tasks with the cluster head and base station. Thus, we considered wake up overhead at the RSU. We have performed detailed simulation study and found that sleep cycles-I incurs significant overhead if applied at the RSU. The effect is more detrimental with sleep cycles type-II since it exactly follows the traffic. As can be concluded Sleep Cycles Type-II (Follow the traffic) are best for delay (QoS) and worst for energy savings. Sleep Cycles Type-I (Random), Type-IV (Modified time based burst accumulation), and Type-VI (Modified length based burst accumulation) can be made identical in operation by selecting suitable parameters. Unmodified versions i.e. Type-III (Time based burst accumulation) and Type-V (Length based burst accumulation) save more energy (i.e. sleep after serving a burst) and have worse delay (QoS) compared to their modified versions. Furthermore, Type-V (Length based burst accumulation) is a good strategy from the QoS perspective since the sleep duration is a function of traffic. At high (low) traffic, the packets are collected fast (slow) resulting in lower (higher) average packet delay and lower (higher) energy savings. Whereas, Type-III (Time based burst accumulation) is a good strategy from the energy savings perspective since a fixed time window means that the sleep mechanism adapts to traffic. It is also contained from the QoS perspective as the average packet delay is less sensitive to the traffic.

Average energy savings of 66.8% and 68.2% were achieved in sleep cycles type-III and type-IV, respectively. Although the improvement in energy savings of type IV compared to type I is only 10%, the improvement in average packet delay is substantial (i.e. 28%). Due to the very high switching overhead (at peak hour) associated with sleep cycles type-V, the energy savings become negative even though only $3.9\ ms$ average packet delay was achieved throughout the day. Sleep cycles type-VI alleviated the negative energy savings while marginally improving the average packet delay. For example, the average energy savings achieved through sleep cycles type-VI is 43.4% compared to 34.4% in the case of sleep cycles type-V. These improvements are achieved under a very stringent QoS bound, which indicates the suitability and generality of our proposed sleep mechanisms. Based on our findings in this paper, it can be concluded that each type of sleep cycles exhibits a few advantages and disadvantages. If the aim is to achieve the best QoS regardless of maximising energy savings, sleep cycles type-VI is ideal. On the other hand, if the aim is to maximise energy savings while maintaining the required QoS, sleep cycles type-IV should be adopted. Thus, it is inferred that the time based burst accumulation (type-III and IV) overall performs better in terms of energy savings. On the other hand, the length based burst accumulation (type-V and VI) overall outperforms the time based burst accumulation, in terms of QoS.

It is to be noted that most of the short range communication devices are now deploying some kind of sleep mechanisms to save energy and reduce carbon emission. With the evolution of the network and its convergence towards Internet of Things (IoT), the traffic is becoming localised and highly heterogeneous. In such scenarios, the devices can improve their energy efficiency with the chosen types of sleep cycles according to the traffic characteristics. These sleep algorithms would be auto programmed in the devices. For example, in case of safety messages, sleep cycles type-II is ideal as it exactly follows traffic. The practical limitation of such mechanisms lies in their electronic design, which determines the wakeup overhead in terms of latency, energy or any other concern. With the fast evolution of flash devices, we expect the above limitation to be negligible in the near future.

Acknowledgement

The authors acknowledge the support of the Engineering and Physical Sciences Research Council (EPSRC), UK for funding the INTelligent Energy aware NETworks (INTERNET) project under contract EP/H040536/1. All data are provided in full in the results section of this paper.

References

[1] "SMART 2020: enabling the low carbon economy in the information age," Tech. Rep., (Visited July 2013). [Online]. Available: http://www.smart2020.org/_assets/files/02_Smart2020Report.pdf

[2] F. Richter, A. Fehske, and G. Fettweis, "Energy efficiency aspects of base station deployment strategies for cellular networks," in IEEE 70th Vehicular Technology Conference, September 2009, pp. 1–5.

[3] M. Faezipour, M. Nourani, A. Saeed, and S. Addepalli, "Progress and challenges in intelligent vehicle area networks," Commun. ACM, vol. 55, no. 2, pp. 90–100, Feb. 2012. [Online]. Available: http://doi.acm.org/10.1145/2076450.2076470.

[4] A. Capone. (2013, May) Energy and mobility: Scalable solutions for the mobile data explosion. GreenTouch. Politecnico di Milano. [Online]. Available: http://www.greentouch.org/uploads/documents/4%20Antonio%20Capone_SEW%20Capone%20presentation.pdf.

[5] L. Haratcherev, M. Fiorito, and C. Balageas, "Low-power sleep mode and out-of-band wake-up for indoor access points," in IEEE GLOBECOM Workshops, Nov.-Dec. 2009, pp. 1–6.

[6] T. Han, Z. Zhang, M. Hu, G. Mao, X. Ge, Q. Li, and L. Wang, "Energy efficiency of cooperative base station sleep scheduling for vehicular networks," in Vehicular Technology Conference (VTC Spring), 2014 IEEE 79th, May 2014, pp. 1–5.

[7] S. Mostofi, A. Hammad, T. Todd, and G. Karakostas, "On/off sleep scheduling in energy efficient vehicular roadside infrastructure," in Communications (ICC), 2013 IEEE International Conference on, June 2013, pp. 6266–6271.

[8] S.-E. Elayoubi, L. Saker, and T. Chahed, "Optimal control for base station sleep mode in energy efficient radio access networks," in IEEE INFOCOM, April 2011, pp. 106–110.

[9] W. Kumar, S. Bhattacharya, B. Qazi, and J. Elmirghani, "A vacation-based performance analysis of an energy-efficient motorway vehicular communication

system," Vehicular Technology, IEEE Transactions on, vol. 63, no. 4, pp. 1827–1842, May 2014.

[10] "An energy efficient double cluster head routing scheme for motorway vehicular networks," in Communications (ICC), 2012 IEEE International Conference on, June 2012, pp. 141–146.

[11] G. Benelli, R. Fantacci, G. Giambene, and C. Ortolani, "Voice and data transmissions with a prma-like protocol in high propagation delay cellular systems," Vehicular Technology, IEEE Transactions on, vol. 49, no. 6, pp. 2126–2147, Nov 2000.

[12] W. Fisher, M. Suchara, and J. Rexford, "Greening backbone networks: reducing energy consumption by shutting off cables in bundled links," in Proceedings of the first ACM SIGCOMM workshop on Green networking, ser. Green Networking '10. New York, NY, USA: ACM, 2010, pp. 29–34. [Online]. Available: http://doi.acm.org/10.1145/1851290.1851297.

[13] M. Gupta and S. Singh, "Greening of the internet," in Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, ser. SIGCOMM '03. New York, NY, USA: ACM, 2003, pp. 19–26. [Online]. Available: http://doi.acm.org/10.1145/863955.863959.

[14] S. Zhou, J. Gong, Z. Yang, Z. Niu, and P. Yang, "Green mobile access network with dynamic base station energy saving," in Proc. ACM MobiCom (Poster Paper), Beijing, China, September 20-25 2009.

[15] G. Narlikar, S. Bhaumik, S. Chattopadhyay, and S. Kanugovi, "Green, energy-efficient network re-organization," Alcatel-Lucent India, Tech. Rep., 2011.

[16] M. F. A. M. Abdel-Shakour Abuzneid, Tarek Sobh and J. James, "Fortified anonymous communication protocol for location privacy in wsn: A modular approach," Sensor, Open Access, vol. 15, pp. 5820–5864, 2015.

[17] B. Vinod, "Exponential queues with server vacations," The Journal of the Operational Research Society, vol. 37, no. 10, pp. 1007–1014, 1986.

[18] R. Uzcategui and G. Acosta-Marum, "Wave: A tutorial," Communications Magazine, IEEE, vol. 47, no. 5, pp. 126–133, May 2009.

[19] B. Qazi, A. Muhtar, W. Kumar, and J. Elmirghani, "Performance of a double cluster head routing scheme in a motorway environment," in GLOBECOM Workshops (GC Wkshps), 2010 IEEE, Dec 2010, pp. 783–787.

[20] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and S. Diot, "Packet-level traffic measurements from the Sprint IP backbone," IEEE Network, vol. 17, no. 6, pp. 6–16, November-December 2003.

[21] Cisco. (2006) Understanding Delay in Packet Voice Networks. White paper. [Online]. Available: http://www.cisco.com/c/en/us/support/ docs/voice/voice-quality/5125-delay-details.html.

[22] Andre B. Reis, Susana Sargento, Filipe Neves, and Ozan K. Tonguz, IEEE "Deploying Roadside Units in Sparse Vehicular Networks: What Really Works and What Does Not", IEEE Transactions on Vehicular Technology, vol. 63, no. 6, July 2014.

[23] B. R. Qazi, H. Alshaer, and J. M. H. Elmirghani, "Analysis and design of a MAC protocol and vehicular traffic simulator for multimedia communication on motorways," IEEE Transcation on Vehicular Technology, vol. 59, no. 2, pp. 734–741, February 2010.

[24] "Smarter vehicles, safer roads MCNU R1551." kapsch, Tech. Rep., (Visited Feb. 2013). [Online]. Available: http: //ww1.prweb.com/prfiles/2008/10/22/915544/MCNUR1551.PDF.

[25] X. Wu, S. Wu, H. Sun, and L. Li, "Dynamic slot allocation multiple access protocol for wireless ATM networks," in IEEE International Communication Conference (ICC), vol. 3, June 1997, pp. 1560–1565.

[26] J. M. Smith, "Optimal design and performance modelling of M/G/1/K queueing systems," Mathematical and Computer Modelling, vol. 39, no. 9-10, pp. 1049–1081, 2004.

[27] F. R. B. Cruz, A. R. Duarte, and T. van Woensel, "Buffer allocation in general single-server queueing networks," Computers and Operations Research, vol. 35, pp. 3581–3598, November 2008.

[28] S. K. Bose, An Introduction to Queuing Systems. Kluwer Academic / Plenum Publishers, 2002.

[29] "Analysis of a M/G/1/K queue with vacations." [Online]. Available: http://home.iitk.ac.in/~skb/qbook/MG1K_Vacation.PDF

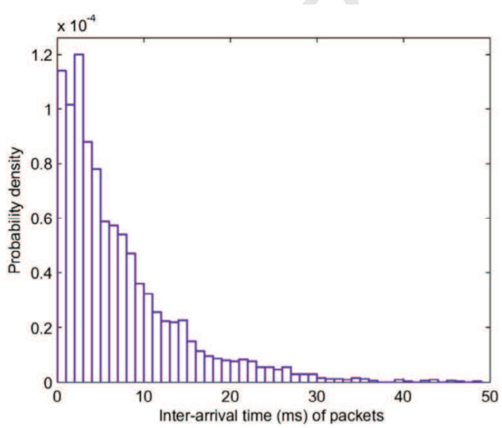
[30] W. J. Stewart, Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling. Princeton University Press, 2009.

[31] W. Kumar, A. Muhtar, B. R. Qazi, and J. M. H. Elmirghani, "Energy and QoS evaluation for a V2R network," in IEEE Global Communications Conference (GLOBECOM), December 2011.

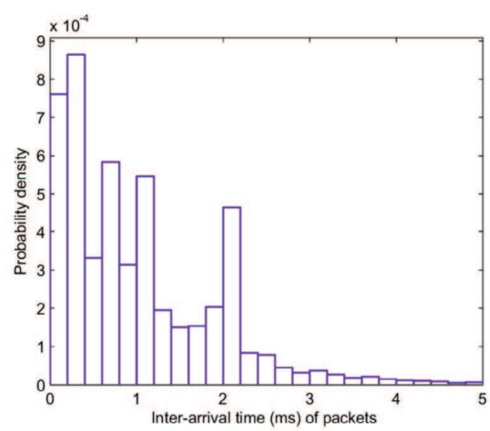
APPENDIX A

ANALYSIS OF INTER ARRIVAL TIME OF THE PACKETS AT THE RSU

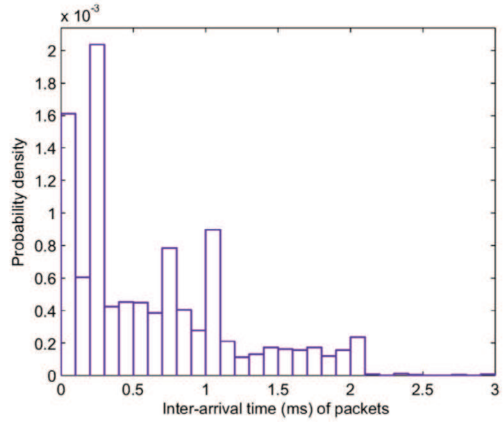
The packet arrivals at the cluster head consists of requests from individual vehicles. The requests from individual vehicles are Poisson distributed. As a result, the aggregated traffic stream at a cluster head is Poisson distributed. Now, the packet size distribution is not random (non-Poisson). Thus the output of the cluster head is treated as General distributed. Under light load condition, the output of the CH is Poisson distributed as the packet service time is small compared to the arrival gap. As the gap decreases with increase in load, the irregularity increases. These are shown in the subsequent figures (Figure 13(a), Figure 13(b), Figure 13(c)). Thus, a General distribution is appropriate to represent the traffic stream. Figure 13 shows the inter-arrival time (IAT) pdfs of packets arriving at the RSU for varying number of vehicles within a cluster. With sleep cycles type-II at CHs, a CH wakes up as soon as a packet arrives and switches to sleep mode as soon as it finishes serving the packet. Thus sleep cycles type-II are transparent to packet departure process. Therefore, the IAT pdf with minimum vehicles is shaped only by the packet service discipline. With minimum vehicles, the IAT of the packets is large. Hence, the packet size and in turn packet service time (which is General distributed) has little impact on IAT pdf. As a result, the IAT pdf with minimum vehicles follows Negative Exponential distribution as shown in Figure 13(a). With a higher number of vehicles, the IAT of the packets becomes smaller. Therefore, the General distributed service time of the packets governs the IAT pdf. Thus, the IAT pdf becomes General distributed as shown in Figure 13(b) and Figure 13(c).



(a) Min. no. of vehicles within a cluster.



(b) Avg. no. of vehicles within a cluster.



(c) Max. no. of vehicles within a cluster.

Figure 13. Inter-arrival time (IAT) pdf at the RSU.

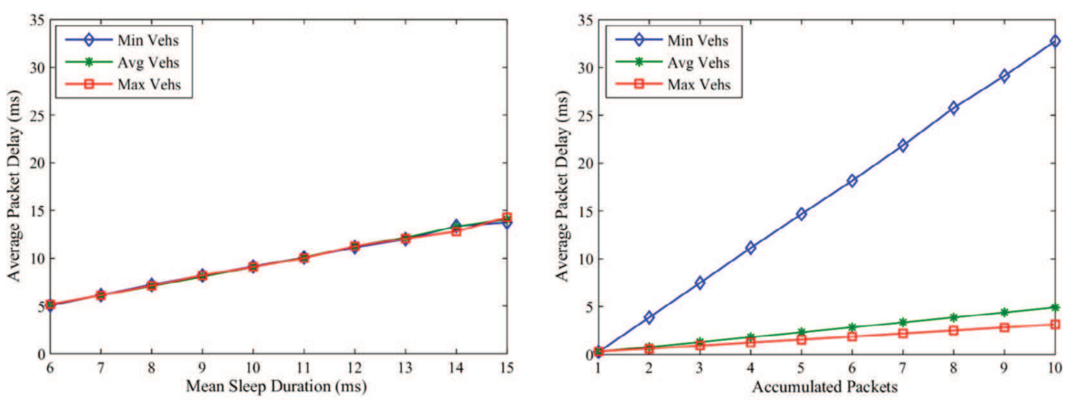
APPENDIX B

DETERMINATION OF $\bar{S}$ AND L FOR SLEEP CYCLES TYPE-I AND TYPE-V

To determine the mean sleep duration ($\bar{S}$) and the number of accumulated packets (L) for sleep cycles type-I and V respectively, we vary them individually to study their impact on the average packet delay in each case. The average packet delay for both sleep cycles types is evaluated with minimum, average and maximum number of vehicles, as shown in Figure 14.

With increase in the mean sleep duration for sleep cycles type-I, the RSU sleeps for a longer duration within a sleep cycle resulting in a higher waiting time for the packets that arrive during a sleep cycle. Therefore, the average packet delay increases with the mean sleep duration as shown in Figure 14(a). Once the RSU switches to sleep mode, it remains in that mode until the sleep duration elapses, regardless of the number of waiting packets or their waiting durations. Hence, the average packet delay remains the same under different loads for a given $\bar{S}$. From the figure, it can be noted that the average packet delay of sleep cycles type-I reaches the reference value of $10\ ms$ with an average sleep duration of $11\ ms$. Therefore $\bar{S}$ is set to $11\ ms$ for the performance analysis in Section 6-B.

With the increase in the number of accumulated packets in the case of sleep cycles type-V, the RSU sleeps for a longer duration within a sleep cycle. This results in a higher waiting time for the arrived packets during that sleep cycle. Therefore, the average packet delay increases with the number of accumulated packets as shown in Figure 14(b). However unlike the case of sleep cycles type-I, the number of accumulated packets here depends upon the packet arrival intensity, which is load dependent. Thus, the steepness of the increase in average packet delay considerably varies with minimum, average and maximum number of vehicles in sleep cycles type-V. Since with the minimum number of vehicles (i.e. low load), the time required to accumulate a specific number of packets is highest compared to average and maximum number of vehicles, hence the steepness is the highest, as shown in Figure 14b, resulting in significantly higher average packet delay. From the figure, it can also be noted that the average packet delay of sleep cycles type-V exceeds the reference value of $10\,ms$ when L lies between 3 to 4. Therefore, the nearest integer is chosen and set $L = 4$ for the performance analysis in Section 6-B for fair comparison.



(a) Sleep cycles type-I. (b) Sleep cycles type-V.

Figure 14. Average packet delay for sleep cycles type-I and type-V with varying mean sleep duration ($\bar{S}$) and accumulated packets (L).