

On/Off Sleep Scheduling in Energy Efficient  
Vehicular Roadside Infrastructure

ON/OFF SLEEP SCHEDULING IN ENERGY EFFICIENT  
VEHICULAR ROADSIDE INFRASTRUCTURE

BY  
SHOKOUH MOSTOFI, B.Sc.

A THESIS  
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING  
AND THE SCHOOL OF GRADUATE STUDIES  
OF MCMASTER UNIVERSITY  
IN PARTIAL FULFILMENT OF THE REQUIREMENTS  
FOR THE DEGREE OF  
MASTER OF APPLIED SCIENCE

© Copyright by Shokouh Mostofi, August 2013

All Rights Reserved

Master of Applied Science (2013)  
(Electrical & Computer Engineering)

McMaster University  
Hamilton, Ontario, Canada

TITLE: On/Off Sleep Scheduling in Energy Efficient Vehicular  
Roadside Infrastructure

AUTHOR: Shokouh Mostofi  
B.Sc., (Electrical Engineering)  
Isfahan University of Technology, Isfahan, Iran

SUPERVISOR: Dr. Terence D. Todd

NUMBER OF PAGES: xi, 78

*To my beloved parents and brothers, the only ones that I always can rely on their  
unconditional love and support*

# Abstract

Smart downlink scheduling can be used to reduce infrastructure-to-vehicle energy costs in delay tolerant roadside networks. In this thesis this type of scheduling is incorporated into ON/OFF roadside unit sleep activity, to further reduce infrastructure power consumption. To achieve significant power savings however, the OFF-to-ON sleep transitions may be very lengthy, and this overhead must be taken into account when performing the scheduling. The OFF/ON sleep transitions are incorporated into a lower bound on energy use for the constant bit rate air interface case. An online scheduling algorithm referred to as the Flow Graph Sleep Scheduler (FGS) is then introduced which makes locally optimum ON/OFF cycle decisions. This is done by computing energy estimates needed both with and without a new OFF/ON cycle. The energy calculation can be efficiently done using a novel minimum flow graph formulation. We also consider the fixed transmit power, variable bit rate, air interface case. As before, a lower bound on RSU energy use is computed by formulating and solving an integer program. Results from a variety of experiments show that the proposed scheduling algorithms perform well when compared to the energy lower bounds. The algorithms are especially attractive in situations where vehicle demands and arrival rates are such that the energy costs permit frequent ON/OFF cycling.

# Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Terence D. Todd, for providing me with the opportunity to pursue my Master's degree at McMaster University. I am also thankful for the excellent job he did in guiding me through my research work. I am truly grateful for working under the supervision of a person with such a great amount of knowledge and experience.

I would also like to thank my committee members, Dr. Karakostas and Dr. Zhao, whose constructive comments helped to improve the quality of my thesis.

I am also thankful to my dear colleagues and friends in the Wireless Networking Lab, who I have greatly benefited from, through their companionship and useful suggestions regarding my research in the past two years. I would like to especially express my deepest appreciation to Amir Khezrian, Maryam Mohseni, Morteza Azimifar, Abdulla Hammad and Lei Qin.

My special thanks also goes to all my other friends in Hamilton, whose friendship and kindness made me feel at home.

Finally I would like to thank my parents who, without their endless love and support, none of this would have been possible.

# Notation and abbreviations

<b>VANET</b>	Vehicular Ad-hoc Network
<b>ITS</b>	Intelligent Transportation System
<b>RSU</b>	Road Side Unit
<b>OBU</b>	OnBoard Unit
<b>V2V</b>	Vehicle to Vehicle
<b>V2I</b>	Vehicle to Infrastructure
<b>LAN</b>	Local Area Network
<b>WLAN</b>	Wireless Local Area Network
<b>WAVE</b>	Wireless Access in Vehicular Environment
<b>InVANET</b>	Intelligent Vehicular Ad-hoc Network
<b>MANET</b>	Mobile Ad-hoc Networks
<b>QoS</b>	Quality of Service
<b>PSM</b>	Power Save Mode
<b>AP</b>	Access Point
<b>TSF</b>	Timing Synchronization Function
<b>PCF</b>	Point Coordination Function
<b>DCF</b>	Distributed Coordination Function
<b>TIM</b>	Traffic Indication Map

**DTIM**   Delivery Traffic Indication Map

**ATIM**   Ad-hoc Traffic Indication Map

**ASPD**   Automatic Power Save Delivery

**BS**   Base Station

**FGS**   Flow Graph Scheduler



# Contents

<b>Abstract</b>	<b>iv</b>
<b>Acknowledgements</b>	<b>v</b>
<b>Notation and abbreviations</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Energy Efficiency issues in VANETs . . . . .	2
1.2 Thesis Goal . . . . .	2
<b>2 Background and Related Works</b>	<b>4</b>
2.1 Vehicular Networks Architectures and Standards . . . . .	4
2.2 Vehicular Networks Applications . . . . .	6
2.3 Requirements and Challenges in Vehicular Networks . . . . .	7
2.3.1 Security . . . . .	7
2.3.2 Routing and Scheduling . . . . .	10
2.3.3 Energy Efficiency . . . . .	15
<b>3 On/Off Sleep Scheduling for Roadside Infrastructure</b>	<b>23</b>
3.1 Problem Description . . . . .	23

3.2	Constant Bit Rate Communication Model . . . . .	25
3.2.1	Model Description . . . . .	25
3.2.2	Lower Bound Formulation . . . . .	26
3.2.3	Online Scheduling Algorithms . . . . .	30
3.2.4	Flow Graph Sleep Scheduler (FGS) . . . . .	30
3.2.5	Greedy Scheduler . . . . .	35
3.3	Variable Bit Rate Communication Model . . . . .	36
3.3.1	Model Description . . . . .	36
3.3.2	Lower Bound Formulation . . . . .	37
3.3.3	Online Scheduling Algorithms . . . . .	41
3.3.4	LP Solver Scheduler (LPS) . . . . .	41
3.3.5	FCFS Sleep Scheduler (FCFSS) . . . . .	44
3.3.6	Comparator Sleep Scheduler (CSS) . . . . .	46
<b>4</b>	<b>Performance Results</b>	<b>49</b>
4.1	Constant Bit Rate Model . . . . .	49
4.2	Variable Bit Rate Model . . . . .	56
<b>5</b>	<b>Conclusions</b>	<b>69</b>

# List of Figures

3.1	Roadside Unit (RSU) Example. . . . .	24
3.2	Roadside Unit Timeline Example. . . . .	26
3.3	Roadside Unit ON/OFF Transition Example. . . . .	29
3.4	Minimum Energy Flow Graph Scheduler. . . . .	32
4.1	Total RSU Energy (J) vs. Vehicular Demand (time slots). $\epsilon_I = 100$ , $\epsilon_X = 500$ , and $\check{O} = 5$ time slots. Low vehicular arrival rate case. . . .	50
4.2	Total RSU Energy (J) vs. Vehicular Demand (time slots). $\epsilon_I = 100$ , $\epsilon_X = 500$ , and $\check{O} = 5$ time slots. High vehicular arrival rate case. . .	51
4.3	Total RSU Energy (J) vs. Vehicular Demand (time slots). $\epsilon_I = 100$ , $\epsilon_X = 100$ , and $\check{O} = 5$ time slots. Low vehicular arrival rate case. . . .	52
4.4	Total RSU Energy (J) vs. Vehicular Demand (time slots). . . . .	53
4.5	Total RSU Energy (J) vs. Vehicular Demand (time slots). Vehicle Arrival Rate = $1/50$ . . . . .	54
4.6	Total RSU Energy (J) vs. Vehicular Demand (time slots). Vehicle Speed = 10, 20 m/s. . . . .	55
4.7	Total RSU Energy (J) vs. Vehicular Demand (Mbits) . . . . .	58
4.8	Total RSU Energy (J) vs. Vehicular Demand (Mbits) . . . . .	59
4.9	Total RSU Energy (J) vs. Vehicle Arrival Rate (vehicles/sec) . . . . .	60

4.10	Total RSU Energy (J) vs. Vehicular Demand (Mbits). Vehicle Arrival Rate = 1/100. . . . .	61
4.11	Total RSU Energy (J) vs. Vehicular Demand (Mbits). Vehicle Arrival Rate = 1/100. . . . .	62
4.12	Total RSU Energy (J) vs. Vehicular Demand (Mbits). $\rho_I = 120$ . . . .	63
4.13	Total RSU Energy (J) vs. Vehicular Demand (Mbits). $\rho_I = 120$ . . . .	64
4.14	Total RSU Energy (J) vs. Vehicle Arrival Rate (vehicles/sec). $\rho_I = 120$ .	65
4.15	Total RSU Energy (J) vs. Vehicular Demand (Mbits) . $\rho_I = 70$ . . . .	66
4.16	Total RSU Energy (J) vs. Vehicular Demand (Mbits) . $\rho_I = 70$ . . . .	67
4.17	Total RSU Energy (J) vs. Vehicle Arrival Rate (vehicles/sec). $\rho_I = 70$ .	68

# Chapter 1

## Introduction

Vehicular Ad-hoc Networks (VANETs) are an important component of Intelligent Transportation Systems (ITS). ITS comprises a variety of applications and services which will improve safety and productivity in future transportation systems. These capabilities will be built, in part, by the use of wireless communication between vehicles and fixed roadside unit (RSU) infrastructure. RSU functionality is similar to that of wireless LAN access points, but may operate over much larger distances. In these systems, communication can occur directly between vehicles, i.e., vehicle-to-vehicle (V2V), or between vehicles and roadside infrastructure (V2I). One of the initial uses of vehicular networks will be for the transmission of safety messages, however they will eventually support numerous other types of applications, some of which are discussed in the next chapter.

## 1.1 Energy Efficiency issues in VANETs

Often in wireless ad-hoc networks, nodes have limited power resources. In VANETs however, energy efficiency for the vehicle nodes has not been an issue, because the radio communication is powered from the vehicle engine, which has enormous energy reserves. On the other hand, RSUs may sometimes be deployed in situations where no wired electricity is available, and therefore may be powered from alternative energy resources such as solar or wind. The cost of provisioning this type of roadside infrastructure is dependent on the average power consumption of the RSU, and can be significantly reduced by making the RSU operate in a more energy efficient fashion. One way to accomplish this is to reduce the power consumed by the RSU when transmitting on the downlink, which is the approach taken in this thesis.

## 1.2 Thesis Goal

Due to the wide range of temporal fluctuations in vehicular traffic, RSU workloads will vary dramatically. For this reason, switching off RSUs during quiescent periods can lead to large power savings. The thesis objective is to achieve this goal by developing energy efficient sleep schedules for green roadside infrastructure. Smart downlink scheduling is also incorporated into ON/OFF roadside unit sleep activity, to further reduce infrastructure power consumption. To achieve significant power savings however, the OFF-to-ON sleep transitions may be very lengthy, and this overhead must be taken into account when performing the ON state scheduling. Two types of downlink communication are considered: constant and variable bit rate air interfaces. First, the OFF/ON sleep transitions are incorporated into lower bounds

on energy usage that can be computed for given input sample functions. Then online scheduling algorithms are presented. Results from a variety of experiments show that the proposed scheduling algorithms perform well when compared to the energy lower bounds.

The remainder of the thesis is organized as follows. Chapter 2 presents some background information on VANETs, and also discusses some recent work in the context of vehicular networks. Chapter 3 contains our model descriptions and also formulates the On/Off Sleep Scheduling problem. The performance of the proposed algorithms is evaluated in Chapter 4, and finally Chapter 5 concludes the thesis.

# Chapter 2

## Background and Related Works

In this chapter some further information about VANETs is presented. First in section 2.1 the Network Architecture and the IEEE Standards which are proposed specifically for VANETs are discussed, then in Section 2.2 different kinds of VANET applications are explained. Finally Section 2.3 is dedicated to Requirements and Challenges in VANETs. Also mentioned in this chapter are some recent works in the vehicular networks context.

### 2.1 Vehicular Networks Architectures and Standards

This section first presents a brief description about the possible deployment architectures for vehicular networks and later some wireless technologies and standards which are specifically proposed for these networks are reviewed.

- **Network Architectures:**



Vehicular Networks can have one of these three main designs (Moustafa and Bourdon, 2008) : (1) Purely Ad-hoc, which means no roadside infrastructure exists and all the communications are in the form of V2V (Vehicle to Vehicle). (2) There is a fixed roadside infrastructure, and single-hop wireless connectivity exists between the vehicles and the infrastructure (V2I Communication). Classical WLANs use similar deployments. (3) Communication happens in a hybrid manner, i.e., it can be in the form of V2V or V2I. In this kind of deployment single and multi-hop communication can take place.

- **Wireless Technologies used in VANETs and related IEEE Standards:**

Specific characteristics of VANETs required some particular provisioning and enhancements to be made in the traditional IEEE 802.11 standards. IEEE 802.11p is an approved amendment to the IEEE 802.11 standard to add wireless access in vehicular environments (WAVE) and support ITS applications. IEEE 802.11p aims to provide specifications needed for MAC and PHY layers for the special needs of vehicular networks. IEEE 802.16a is another standard which introduces a mesh mode, enabling multi-hop communication and providing broadband radio coverage. It operates in the 2-11 GHz frequencies and covers up to about 50 km (Moustafa and Bourdon, 2008). IEEE 802.16e is an amendment to IEEE 802.16a that operates in the 2-6 GHz bands and aims to support subscriber stations moving at vehicular speeds. IEEE 1609 is also another family of standards which deals with issues such as management and security of the network. Furthermore, Intelligent Vehicular Ad-hoc Network (In-VANET) integrates multiple ad-hoc networking technologies, like WiFi IEEE

802.11p, WAVE IEEE 1609, WiMAX IEEE 802.16, Bluetooth, IRA, ZigBee, in order to improve accuracy, effectivity and simplicity of the communications in VANETs.

## 2.2 Vehicular Networks Applications

Future vehicular networks are expected to support a wide variety of applications. This section contains a review of some of the most important ones.

- **Safety messages, Decreasing the probability of accidents**

This category of applications constitutes one of the main motivations for vehicular networks development. Vehicles can be provided with information such as speed and location of other vehicles, position of intersections, possible hazardous locations, through the sharing of information between vehicles and roadside infrastructure (Karagiannis *et al.*, 2011). By the transmission of these kinds of information in the form of Safety Messages, VANETs can make a huge contribution to collision avoidance. Since these types of applications are related to human life, they are very time-critical and also the security of data transmission is more important here than in other types of VANET applications.

- **Contribution to Cooperative Driving**

By cooperation between the vehicles themselves and with the RSU, drivers can adjust their speed to different road and traffic situations, and also can be provided with updated information regarding local roads and maps, which all result in a smooth traffic flow.

- **Infotainment, Commercial and other services**

This group of applications consists of various types of services such as providing vehicles with information like notification about a specific location of interest, local electronic commerce, advertisement and media downloading. Global Internet services is another application, an example of which can be community services, such as insurance and financial services (Karagiannis *et al.*, 2011). An instance of the commercial products which is already in use is the installation of the black boxes in cars by insurance companies. This is done in order to collect information about the travelled distances and calculate the related insurance costs (Liu *et al.*, 2007).

## **2.3 Requirements and Challenges in Vehicular Networks**

There are some requirements and challenges associated with the implementation and operation of VANETs . In this section some of the main ones, such as Security, Routing, Scheduling and Energy Efficiency are presented.

### **2.3.1 Security**

The special characteristics of vehicular networks make them vulnerable to security attacks; therefore providing a secure communication environment is a very challenging task. It is especially critical in the context of safety applications,

where non-secure data transfer can endanger peoples' lives. In the following, some forms of security attacks, security requirements in VANETs and also a few of the proposed solutions, are explained.

- **Attacks:**

**Attacker Type:** There are various kinds of attackers, based on their identity, purpose of attack and activity. For example, an *insider* attacker is one of the internal compromised nodes of the network, as opposed to an *outsider* attacker, which is an entity from outside of the network. An *active* attacker modifies packets or diffuses malicious data into the network, as opposed to a *passive* one, whose activity is eavesdropping and stealing information (Liu *et al.*, 2007).

**Attack Type:** Attackers' activities also can be in different forms, such as injecting wrong information to affect driver's behavior for their own benefit, tracking vehicle locations through ID disclosure, and altering their perceived position, speed or direction, in an accident for example.

- **Security Requirements:**

**Source Authentication:** In order to prevent the attackers' malicious activities (either altering or stealing) to affect the network communications, the legitimacy of the sender should be assured; also a mutual authentication between every two communication parties is required.

**Data Verification:** In some situations, although the sender is legitimate, a packet may contain false information. One method for verifying the data is to investigate its consistency with similar messages like the ones generated in close time or space (Liu *et al.*, 2007).

***Confidential and Secure Data Transfer:*** Confidentiality is essential to ensure security of the users private information, especially in non-ITS applications.

***Real-time Constraints and Time Sensitivity:*** While transferring safety-related messages, it is very important that the transmission delay fits into the required bounds, which is assumed to be about 100 msec (Moustafa and Bourdon, 2008).

***Privacy and Non-traceability:*** This requirement, which is needed to ensure people's privacy against unauthorized observers, is a tough one to satisfy since sometimes observations should be available for concerned authorities, but not for the clients.

***Non-repudiation:*** In safety applications it is sometimes required to keep records of a sequence of exchanged messages (for determining the source of an accident, for example), therefore vehicles should not be able to reject the transmission of a message.

- **Solutions:**

In recent years some architectures and mechanisms have been proposed to provide secure communication in VANET environments. The simplest and yet the most effective method is the *Digital Signature*, in which a set of public/private key pairs is assigned to each vehicle, that allows the vehicles to be authenticated to the receivers by digitally signing the messages. The public keys should be issued and assigned by a trusted authority. However, attaching the digital signature and the certificate to each safety message increases the overhead; to

remove this issue, in (Liu *et al.*, 2007) it is proposed to use symmetric keys. They have considered two types of symmetric keys: pairwise and group keys. In (Raya and Hubaux, 2005) a security mechanism is devised for an urban environment, where service providers are the core entity for all authentication and access control operations, and mobile clients are authenticated by them.

### 2.3.2 Routing and Scheduling

Routing protocols are responsible for selecting the paths between source and destination nodes in communication networks. In VANETs, routing and scheduling of data packets is a challenging task due to the networks intrinsic characteristics, such as the high speed of vehicles, constant topology changes, restriction of vehicles mobility by the road layout, other vehicle's movements and traffic rules, network partitioning into clusters of vehicles and uneven network density (Lequerica *et al.*, 2010). In city environments the issue is intensified since there are many obstacles which cause blockage of the transmitted signals. All these factors make traditional MANET (Mobile Ad-hoc Network) routing protocols inappropriate for vehicular settings.

- **Routing Protocols:**

VANET Routing Protocols can generally be classified into two main categories: *Topology based routing* protocols and *Position based routing* protocols.

***Topology Based Routing Protocols:*** These types of routing protocols use link information within the network to forward the packets. They are divided into two groups of Proactive and Reactive routing protocols.

*Proactive:* In this kind of routing, each node preserves the information related

to the network paths in a table, which is updated whenever a change occurs in the network topology. The main advantage of proactive routing is that no path search and route discovery is required. Also it has low latency, which is suitable for real time applications. However, since the information of all the connections within the network should be maintained in the tables, even though the corresponding paths are not used in the communication, an overhead is imposed on the network (Paul *et al.*, 2011). FSR (Fisheye State Routing) (Pei *et al.*, 2000) is an example of this type of routing.

*Reactive:* In reactive routing, whenever a communication is required between two nodes, path search and route discovery is performed by flooding the network with query packets. Although this method reduces the burden on the network, it also has some disadvantages such as high latency due to the route search and also disruption of the node communication caused by excessive flooding. Some examples of these types of protocols are: Ad-Hoc On Demand Distance Vector routing protocol (AODV) (Perkins and Royer, 1999), Dynamic Source Routing (DSR) (Johnson and Maltz, 1996) and Temporally Ordered Routing protocol (TORA) (Park and Corson, 1997).

***Position based Routing Protocols:*** Position based routing protocols use geographical information, obtained by services like GPS, to determine the next hop in communication (closest to the destination). In this method no routing table or information exchange with neighboring nodes, regarding the link state, is required. It is also suitable for high node mobility patterns. However, the position determining services are not accessible everywhere, which limits the functionality of this type of routing protocol (Paul *et al.*, 2011).

VADD (Vehicle-Assisted Data Delivery), Geographical Opportunistic Routing (GeOpps) (Leontiadis and Mascolo, 2007), Greedy Perimeter Stateless Routing (GPSR) (Karp and Kung, 2000), Geographic Source Routing (GSR) (Lochert *et al.*, 2003), Anchor-Based Street and Traffic Aware Routing (A-STAR) (Seet *et al.*, 2004), Greedy Traffic Aware Routing protocol (GyTAR) (Jerbi *et al.*, 2007) and Connectivity-Aware Routing (CAR) (Naumov and Gross, 2007) are some examples of position based routing protocols.

- **Data Dissemination and Scheduling**

In a vehicular network environment vehicles usually move with high speed. Therefore they are within the RSU coverage area for a short period of time. This fact makes data scheduling a crucial task in VANETs. So far, a lot of work has been done in the context of data dissemination in VANETs, and many scheduling algorithms have been proposed as well, which try to serve as many requests as possible, while providing users with the required level of Quality of Service (QoS). The following two sections present a brief review of this work.

***Data dissemination Techniques:***

***Push-Based:*** This mechanism is mainly used in safety applications, where a large number of vehicles need to be informed of the same data, such as traffic conditions, i.e. speed, position and direction of the surrounding vehicles. There are two proposed approaches for achieving this goal (Zhao and Cao, 2006):

*Flooding*, in which every vehicle periodically broadcasts its data packets to other



vehicles in the network. A disadvantage of this approach is that it is not scalable, because in situations with high traffic, the network will be flooded by large numbers of messages.

*Dissemination*, where vehicles broadcast their own data packets as well as the ones from other vehicles; the vehicles update their information once they receive a broadcast from other vehicles and defer forwarding the information to the next broadcast period (Kumar and Chand, 2010). Unlike the previous method, this approach is scalable and removes the issue of flooding.

***Pull-Based:*** In this method, which is especially useful for non-safety applications, data dissemination happens in an on-demand manner, i.e., the RSU transmits data packets to individual vehicles based on their requests. This method is more scalable compared to the push-based method.

In addition to these data dissemination mechanisms some techniques have been designed to improve the performance of data delivery according to specific characteristics of VANETs. Some examples are:

*Opportunistic:* In order to deal with the issue of lack of connectivity, which exists in some VANET environments such as highways, opportunistic communication is proposed with cars used as data mules. Information is pulled from other vehicles or the infrastructure as the target vehicle encounters them (Ott and Kutscher, 2004), (Hadaller *et al.*, 2007).

*Vehicle-Assisted:* The carry-forward paradigm can be used to improve the quality of data transfer in VANETs. Vehicles carry the data packets as they move until an appropriate destination vehicle is found in their vicinity, and then they begin the forwarding process (Zhao and Cao, 2006).

### ***Scheduling Algorithms:***

Several scheduling algorithms have been proposed for VANETs, each considering a specific performance metric. Fairness, reliability, responsiveness, time constraints, data size, service ratio and data quality are some examples of these metrics. In the following, a brief review of some of these scheduling algorithms is presented (Hadaller *et al.*, 2007), (Wu and Cao, 2001), (Jiang and Vaidya, 1999), (Mi *et al.*, 2008) :

*First Come First Serve (FCFS)*: It is the simplest scheduling algorithm in which the request with the earliest arrival time will be served first. *First Deadline First (FDF)*: In FDF the request with the earliest deadline time (the most urgent request) has priority. *Smallest Data Size First (SDF)*: SDF serves the requests with smaller sizes first. *Longest Total Stretch First (LTSF)*: This algorithm, which introduces a new metric called *stretch*, reduces the waiting time. Stretch is defined as the ratio of the response time of a request to its service time. *Maximum Quality Increment First (MQIF)*: In this method, requests which result in more increment in quality of service are served first. *Least Selected First (LSF)*: LSF gives the highest priority to the requests that had the least opportunity of being previously served. *D\*S Scheduling Algorithm*: In this algorithm serving is performed based on request deadline and size. *D\*S/N Scheduling Algorithm* (Sayadi and Fathy, 2010): In the  $D * S$  algorithm, when several vehicles send requests for the same data, they will get service through a single broadcast. In these situations,  $D * S/N$  gives priority to the data with the highest number of pending requests. *D\*S/P Scheduling Algorithm* (Kumar and Chand, 2010):

This algorithm also works similarly to  $D * S$  with the difference that besides the deadline and size of the requests, it considers another metric as well, the Priority which is sent with the request according to the message type (Normal, Accidental, Emergency, Warning).

### 2.3.3 Energy Efficiency

In this section the issue of energy efficiency is first considered in general wireless communication networks. Then the problem is investigated specifically for VANETs, and some of the mechanisms that are proposed to make VANETs more energy efficient, are presented.

- **IEEE 802.11 power saving mechanisms**

Wireless LAN (WLAN) technology has become one of the most popular standards for wireless broadband internet access in homes, offices, and public hotspot locations. With the development of portable computing and mobile technology, energy conservation has become an important issue in these networks. Therefore some power saving mechanisms need to be established so as to improve the energy efficiency of WLANs. Since the idle receive state dominates the LAN adapter power consumption over time, one solution is to switch off the wireless communication devices. The IEEE 802.11 power saving mechanism is one of the most widely used protocols for energy conservation in wireless data networks; it uses the strategy of switching off the idle stations while maintaining network connectivity. The performance of this mechanism is briefly described in the

following section

### ***Power Saving in the IEEE 802.11 draft standard***

The main goal in the IEEE 802.11 power saving mechanism is to switch off the stations, while maintaining an active session, i.e., stations in power save (PS) mode should be able to receive data packets sent to them from other stations. This requires that all the stations in PS mode wake up simultaneously to listen to announcements from the sender station or the Access Point (AP), which has buffered the data packets. Therefore the standard defines a Timing Synchronization Function (TSF). In Infrastructure Networks, the AP is responsible to periodically generate beacons, which contain a time stamp besides other information. All the stations will adjust their timers to this time stamp.

In Ad-Hoc Networks, since there is no central infrastructure, the synchronization of stations happens in a distributed manner (Distributed Coordination Function - DCF). All the stations are responsible for generating beacons; after the beacon interval the stations will compete for transmission of their beacon, the first station is the winner, and other stations adjust their timers to the time stamp of the winning beacon (Rohl *et al.*, 1997).

While the stations are in the PS mode, APs buffer the data packets destined for them; the stations then wake up shortly before the end of the beacon interval. The AP transmits a Traffic Indication Map (TIM) within the beacon where all unicast messages are announced. Afterwards the stations poll the AP for receiving the packets, by sending power save polls (PS-Polls). The AP uses a More Data bit in the data frame to indicate whether there are more data

to be sent. A station goes to sleep only when it has retrieved all the pending data. The broadcast/multicast frames are first announced by a Delivery TIM (DTIM) and sent immediately afterwards (Rohl *et al.*, 1997), (Perez-Costa and Camps-Mur, 2010).

In DCF the power management is performed in a distributed fashion as well. The sender is responsible for buffering the data packets. After the beacon interval, there is a special interval called Ad-hoc TIM (ATIM) window, during which the sender transmits the ATIM and waits for acknowledgment from the receiver. After receiving the acknowledgment message, the data packet transmission begins.

IEEE 802.11e introduces an enhancement to the PS mode of IEEE 802.11, called the Automatic Power Save Delivery (ASPD), which reduces the required signaling load. Two APSD modes are available: unscheduled APSD (U-APSD) and scheduled APSD (S-APSD) (Perez-Costa and Camps-Mur, 2010).

- **Energy Efficiency in Ad-Hoc Networks**

In wireless ad-hoc networks nodes are usually battery powered, and they start with an initial amount of charge which is suitable for a limited period of time. Recharging or replacing the batteries is usually not possible. Thus energy efficiency becomes a very important issue that needs consideration in order to prevent network nodes from experiencing battery charge depletion. Energy efficiency in wireless ad-hoc networks has been investigated a lot by the scientific community in recent years. Most of these works have focused on power control mechanisms. Since energy consumption of a single node is proportional to the

power assigned to it, one method for minimizing the overall power consumption of the network is to select the proper level of transmission power for the network nodes. On the other hand, a sufficient degree of connectivity in the network and good communication between nodes should be guaranteed. This results in the introduction of a class of optimization problems, called the *range assignment* problems: Given a set of stations  $S$ , a range assignment for  $S$  is a function  $r : S \rightarrow \mathbb{R}^+$ . The cost of a range assignment  $r$  is the overall power consumption, that is (Clementi *et al.*, 2002)

$$cost(r) = \sum_{v \in S} [r(v)]^\alpha$$

where  $\alpha$  is the *distance-power gradient*. A range assignment  $r$  for a set  $S$  of stations yields a directed communication graph  $G_r = (S, E)$  such that, for each  $(u, v) \in S \times S$ , the directed edge  $(u, v)$  belongs to  $E$  if and only if  $v$  is at distance of at most  $r(u)$  from  $u$ .

Assuming various types of required properties (connectivity constraints between the network nodes) the class of range assignment problems with different levels of computational complexity can be obtained:

**Definition:** Given a graph property  $\Pi$ , the MIN-RANGE( $\Pi$ ) problem is defined as follows

*Input:* A set of points  $S$

*Output:* A range assignment  $r$  for  $S$  such that  $G_r$  satisfies  $\Pi$  and  $cost(r)$  is minimized.

The Strong Connectivity, the h-Strong Connectivity and the Broadcast are some examples of such properties.

The range assignment problem with Strong Connectivity was first investigated in (Clementi *et al.*, 2002). In (Clementi *et al.*, 2002) the network lifetime (power assignment) problem is also considered, while satisfying two other connectivity constraints: Symmetric connectivity and Broadcast.

In (Shpungin and Li, 2012) the authors look for an energy efficient (in terms of total energy and network lifetime) power assignment such that the communication graph is strongly connected, and also the capacity of the minimum capacity link is maximized. Two types of power assignment methods are considered in this work: homogeneous and heterogeneous. One special feature of the method in (Shpungin and Li, 2012) is that all nodes are transmitting at the same time. In (Berend *et al.*, 2008) the authors study asymmetric power assignments for which the induced communication graph is  $k$ -strongly connected, while minimizing the total energy assigned and maximizing the network lifetime.

Some other methods focus on the routing process in wireless ad-hoc networks, in order to achieve energy efficiency, e.g., in (Chang and Tassiulas, 2000) the goal is to select the routes and the corresponding power levels such that the time until the node battery drain-out is maximized. Also (Li and Halpern, 2001) presents a position-based algorithm to set up and maintain a minimum energy network.

- **Energy Efficiency in Base Station Equipment**

*Green wireless communication networks* have attracted a lot of attention in recent years, especially in the context of cellular networks there has been a lot of research on energy efficiency in Base Stations (BSs). The number of required

base stations in current cellular networks is increasing due to the fact that these networks need to provide service in almost 100% of geographical areas; expansion of smart phones and other wireless technologies also has caused an exponential increase in Internet access demand, which requires more BSs to be deployed to fulfill user requirements. This high number of BSs consumes a large amount of energy. In a typical energy consumption profile of a European mobile network, more than 50% of the total energy is attributed to the BS equipment (another 30% is consumed for mobile switching and core transmission equipment) (Vadgama and Hunukumbure, 2011). Since the carbon emissions of energy sources have a negative impact on the environment, and the price of energy is also increasing, energy consumption has become an important issue. In recent years a lot of research has proposed different strategies for making BSs more energy efficient. This work can be divided into two main categories: (1) Hardware, which includes improving devices and equipment such as digital signal processors and power amplifiers, in terms of power consumption; (2) Software, i.e., developing better algorithms for network operation, in order to make it more energy efficient.

One interesting idea that has been commonly used in software is to minimize power consumption by switching off unnecessary BSs. Usually the number of BSs to be deployed in a cellular network is determined based on the assumption of peak traffic load. But in reality the traffic intensity typically has major spatial and temporal fluctuations. For example the number of users during nighttime hours is low, hence there is not much traffic load, which means that many BS devices are under-utilized, but still consume energy. Therefore switching off can



lead to a significant power savings, especially because base components such as processing circuits and air conditioners are the most power consuming parts, and take up about 60 percent of the total consumption (Louhi, 2007). An issue that needs consideration in these scenarios, is that the radio coverage and service provisioning should be guaranteed for the users in the coverage area of the sleeping BSs. Much research has been conducted to achieve this goal; the main ideas in this work consist of increasing transmission power of the neighboring active BSs, wireless relay, and BS cooperation.

In (Niu *et al.*, 2010) a method, called ‘Cell zooming’, is proposed in which cell sizes change according to the traffic conditions. This is done by changing the BSs transmission power. The work done in (Kolios *et al.*, 2010) uses the Store-Carry and Forward (SCF) relaying method in which the relay nodes (vehicles) transfer traffic from cells with low utilization level to their neighbor cells, hence allowing the under-utilized BSs to be switched off. In this method there is no need for BSs to increase their coverage area/transmission power.

In VANETs, network nodes consist of OBUs and RSUs. Energy consumption has not been an issue for OBUs, since vehicle engines are a large energy supply. But RSUs are sometimes required to be deployed in non-urban areas with no access to wired electricity. When this is the case, one solution is to use renewable energy resources such as wind or solar power. The cost of this type of provisioning is dependent on the RSU power usage. Furthermore, in order to maintain a sufficient level of connectivity in the network and satisfy the required delay constraints (especially in time-critical applications), a larger number of RSUs need to be deployed to contribute to data dissemination as relay nodes.

Thus, for minimizing power consumption, designing energy efficient algorithms is an essential task. However, not much work has been done regarding this issue in VANETs. In (Sou, 2010) the authors investigate the impact on the connectivity metric when a power-saving model is applied to VANETs; their goal is to maximize the saved power while satisfying connectivity constraints. In (Zou *et al.*, 2011) a two stage algorithm is designed that provides an on/off schedule for the deployed RSUs, while assuring that every pair of vehicles are connected.

## Chapter 3

# On/Off Sleep Scheduling for Roadside Infrastructure

### 3.1 Problem Description

In certain vehicular scenarios, the location of vehicles passing through roadside unit radio coverage can be accurately predicted. When there is sufficient packet delay tolerance, smart schedulers can then use this information to reduce downlink infrastructure-to-vehicle energy communication costs (Hammad *et al.*, 2010). This is done by scheduling traffic when vehicles are in favorable energy locations as illustrated in Figure 3.1. In the figure, vehicle  $v$  is shown at two different times and distances from the RSU,  $(t_1, d_1)$  and  $(t_2, d_2)$ , respectively. Since  $d_1 \gg d_2$ , downlink communication with  $v$  may be far more preferable at time  $t_2$  since the energy costs are likely to be significantly lower. Scheduling algorithms for this were first proposed in (Hammad *et al.*, 2010).

In the coming sections it is shown that this type of scheduling can be incorporated

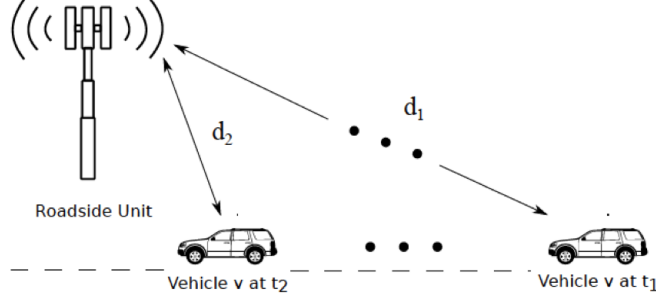


Figure 3.1: Roadside Unit (RSU) Example. Vehicle  $v$  is shown at two different times,  $t_1$  and  $t_2$ , and distances from the RSU, where  $d_1 \gg d_2$ . Communication at time  $t_2$  is preferred in terms of RSU energy cost.

into aggressive ON/OFF sleep cycles to further reduce infrastructure power consumption. To achieve significant power savings however, certain RSU subsystems must be completely switched off, and this results in OFF-to-ON transitions which may be very lengthy. This overhead must be taken into account when scheduling downlink transmissions of the kind considered in (Hammad *et al.*, 2010). The problem is investigated for two different types of downlink communication: Constant bit rate and Variable bit rate. First an integer program is formulated for each case that incorporates ON/OFF sleep transitions into a lower bound on energy usage that can be computed for a given input sample function. This bound is used for comparisons with online scheduling algorithms. The results show that the online algorithms perform well when compared to the energy lower bound and are especially attractive in situations where vehicle demands and arrival rates are such that the energy costs permit frequent ON/OFF cycling.

## 3.2 Constant Bit Rate Communication Model

### 3.2.1 Model Description

A roadside scenario is considered which consists of a single roadside unit that serves passing vehicles, an example of which is given in Figure 3.1. In the figure, vehicles are shown traveling in the same direction but this is not a requirement. It is assumed that the RSU has a single radio which can only communicate with one vehicle at a given time.

Channel time is assumed to be time-slotted and power control is used on the downlink (i.e., RSU-to-vehicle direction) so that each time slot can carry  $B$  bits, regardless of vehicle location within the RSU coverage area. This can be accomplished in a variety of ways such as by using a short two-way handshake prior to user data packet transmission. An example of a time-line is given in Figure 3.2, which shows two RSU OFF periods with durations,  $O_i$  and  $O_{i+1}$ . During an OFF period the RSU is in a very low power consumption state and is unable to operate its radio subsystems.<sup>1</sup> Figure 3.2 also shows that there is a minimum duration OFF interval, denoted as  $\check{O}$ . This duration corresponds to the sum of the overhead required to transition from a fully active mode to a deep sleep mode, and to return to an active mode at the end of the OFF period. This overhead is included in the OFF state duration since the RSU is not functional during these transitions. However, as discussed below, there is an energy cost associated with these transitions. In the figure,  $O_{i+1}$  is shown as a minimum duration OFF interval, and  $O_i > \check{O}$ . It is assumed that when a vehicle

---

<sup>1</sup>This low level of node activity is sometimes referred to as “deep sleep mode” where many node subsystems must be restarted before normal activity can resume. This is unlike the doze mode normally associated with air interfaces such as IEEE 802.11 where a node can reenter active mode almost instantly.

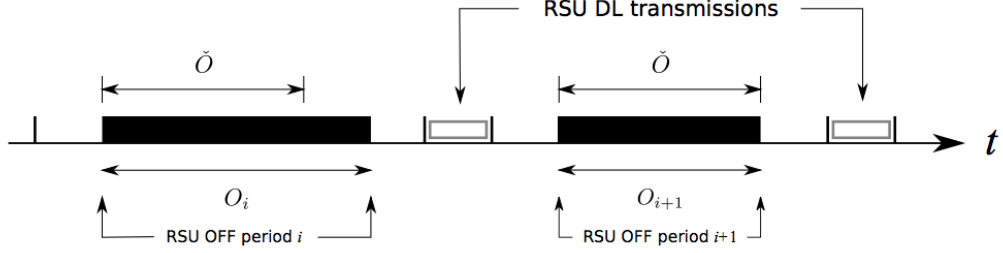


Figure 3.2: Roadside Unit Timeline Example. This example shows two RSU OFF periods of duration,  $O_i$  and  $O_{i+1}$ , respectively. The minimum OFF period,  $\check{O}$ , is shown to be 3 time slots in this example.

$v$  enters the coverage area of the RSU, it communicates its speed, direction and communication requirements, given by  $R_v$  bits, to the RSU. This is done using a medium access control protocol that permits the RSU to designate particular time slots, or mini-slots within particular time slots for uplink communication from newly arrived vehicles. The objective is to schedule incoming vehicular downlink requests so that RSU energy use is reduced as much as possible. For this reason, the RSU prefers to communicate with nearby vehicles rather than more distant ones, as shown in Figure 3.1. To use this option however, requires that there is sufficient packet delay tolerance. The scheduling must also be done in a way which guarantees the downlink packet reception requirements of the vehicles are fulfilled.

### 3.2.2 Lower Bound Formulation

Given an input sample function of arriving vehicles with known speed and traffic requests, a lower bound on the optimum energy scheduling is derived. It is assumed that there are  $N$  vehicles indexed by the set  $\mathcal{N} = \{1, 2, \dots, N\}$  and that there are  $T$  time slots given by the set  $\mathcal{T} = \{1, 2, \dots, T\}$  over which the scheduling is to occur.  $R_i$  is the downlink communication demand for vehicle  $i$  in bits. The following set of

binary scheduling variables is defined.

$$K_{i,t} = \begin{cases} 1 & \text{if the RSU transmits to vehicle } i \text{ in time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (3.1)$$

$\mathcal{E}_t$  is defined to be the energy consumed by the RSU in time slot  $t$  when it is on the ON state,  $\epsilon_{i,t}$  is the energy cost to transmit to vehicle  $i$  in time slot  $t$ , and  $\epsilon_I$  is the energy consumed during a non-downlink transmission time slot when the RSU is in the ON state, i.e.,

$$\mathcal{E}_t = \sum_{i \in \mathcal{N}} \epsilon_{i,t} K_{i,t} + \epsilon_I (1 - \sum_{i \in \mathcal{N}} K_{i,t}). \quad (3.2)$$

The first term calculates the energy used by the RSU to serve the downlink communication demands of the vehicles, while the second term computes the energy used by the RSU when it is in the idle/receive mode. Our objective is to assign time slots for downlink transmission and an ON/OFF schedule for the RSU that minimizes the total energy use over  $\mathcal{T}$ . A binary variable,  $\mathcal{O}_t$ , is defined as follows

$$\mathcal{O}_t = \begin{cases} 1 & \text{if the RSU is in the ON state in time slot } t, \\ 0 & \text{otherwise.} \end{cases} \quad (3.3)$$

A binary variable  $X_t$  is also defined, which is equal to 1 if the RSU transitions from the ON to the OFF state at time slot  $t$ . This can be written as

$$X_t = \begin{cases} 1 & \text{if } \mathcal{O}_{t-1} = 1 \text{ and } \mathcal{O}_t = 0, \\ 0 & \text{otherwise.} \end{cases} \quad (3.4)$$

$\epsilon_X$  is defined to be the energy required by the RSU during the ON-to-OFF and its associated OFF-to-ON state transition. Using these definitions the following integer program (IP) can be written, which gives a lower bound on the RSU energy required.

$$\underset{K_{i,t}, \mathcal{O}_t}{\text{minimize}} \quad \sum_{t \in \mathcal{T}} \left( \sum_{i \in \mathcal{N}} \epsilon_{i,t} K_{i,t} + \epsilon_I \left( 1 - \sum_{i \in \mathcal{N}} K_{i,t} \right) \right) \mathcal{O}_t + \epsilon_X \sum_{t \in \mathcal{T}} X_t \quad (3.5)$$

$$\text{subject to} \quad \sum_{t \in \mathcal{T}} K_{i,t} = \lceil R_i/B \rceil, \quad \forall i \in \mathcal{N} \quad (3.6)$$

$$\sum_{i \in \mathcal{N}} K_{i,t} \leq 1, \quad \forall t \in \mathcal{T} \quad (3.7)$$

$$K_{i,t} \leq \mathcal{O}_t, \quad \forall i \in \mathcal{N}, t \in \mathcal{T} \quad (3.8)$$

$$X_t \leq \mathcal{O}_{t-1}, \quad \forall t \in \mathcal{T} \quad (3.9)$$

$$X_t \geq \mathcal{O}_{t-1} (1 - \mathcal{O}_t), \quad \forall t \in \mathcal{T} \quad (3.10)$$

$$X_t \leq 1 - \mathcal{O}_{t-1} \mathcal{O}_t, \quad \forall t \in \mathcal{T} \quad (3.11)$$

$$\sum_{l=0}^{\tilde{O}-1} \mathcal{O}_{t+l} \leq M(1 - X_t), \quad \forall t \in \mathcal{T} \quad (3.12)$$

$$K_{i,t} \in \{0, 1\}, \quad \forall i \in \mathcal{N}, t \in \mathcal{T} \quad (3.13)$$

$$\mathcal{O}_t, X_t \in \{0, 1\}, \quad \forall t \in \mathcal{T} \quad (3.14)$$

In IP 3.5 to 3.14, the objective function is the total energy used by the RSU. Equation 3.5 contains two parts, first part is the energy expenditure from Equation 3.2 summed over those time slots when the RSU is in the ON state. The second part is the energy consumed during the OFF/ON transitions after awakening the RSU,



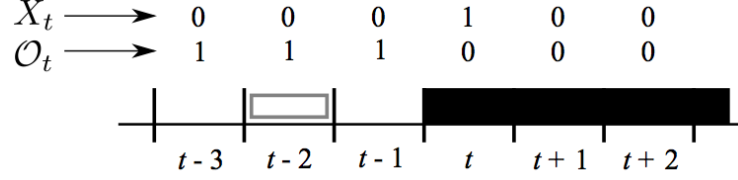


Figure 3.3: Roadside Unit ON/OFF Transition Example.

i.e., the summation over all  $X_t$  gives the number of OFF periods that the RSU incurred. Since the RSU is in a deep sleep during the OFF periods, it is assumed that power consumption associated with this is negligible compared with the other states, except for the aforementioned transition energy. Constraint 3.6 ensures that each vehicle communication requirement is fulfilled by summing the appropriate values of  $K_{i,t}$  and Constraint 3.7 ensures that at most a single packet is allocated to each time slot. Constraint 3.8 makes sure that the RSU can only assign time slots when it is in the ON state. Constraints 3.9 to 3.11 are used to properly define the  $X_t$  variables as given in Definition 3.4 and an example of this is shown in Figure 3.3. By definition,  $X_t$  is only equal to 1 when there is a  $1 \rightarrow 0$  transition in  $O_t$ . From Constraints 3.9 to 3.11, when  $(O_{t-1}, O_t) = (1, 0)$  as shown in Figure 3.3,  $X_t \leq 1$ ,  $X_t \geq 1$ , and  $X_t \leq 1$ , which gives  $X_t = 1$ . All the other combinations for  $(O_{t-1}, O_t)$  give  $X_t = 0$  as required by Definition 3.4. Constraint 3.12 ensures that when there is an ON/OFF transition, the RSU cannot be active for at least  $\check{O}$  time slots, i.e., the time which is required to re-awaken the RSU from OFF mode. When  $X_t = 1$ , Constraint 3.12 makes sure that at least the next  $\check{O}$  time slots are not usable. In this constraint,  $M$  is a constant where  $M \geq \check{O}$ . Constraints 3.13 and 3.14 define the binary variables.

IP 3.5 to 3.14 contains binary quadratic terms in the objective function and in Constraints 3.10 and 3.11. But because they are products of single binary variables, these

equations can be linearized by introducing supplementary variables (Ahuja *et al.*, 1993). This permits its solution using integer linear programming methods, such as branch and bound algorithms. These results are used for comparisons with online algorithms to be introduced in the next two sections.

### 3.2.3 Online Scheduling Algorithms

The optimization derived in Section 3.2.2 gives a lower bound on the downlink RSU energy needed to satisfy vehicular packet requirements. In order to compute the bound, the energy costs associated with a given packet transmission are used. Although it is difficult to precisely know this information in real-time situations, in certain scenarios excellent estimates of this cost can be made (Wang, 2005)(Sommer *et al.*, 2011). Accordingly, a highway scenario is considered where vehicles may travel at different speeds, but maintain their own speed throughout the RSU coverage areas (Khabazian and Ali, 2008). When vehicles enter the RSU radio coverage area they announce their location, direction and speed, which is information that can be used to accurately compute future energy transmission costs. This is done assuming distance dependent exponential path loss propagation (Wang, 2005)(Sommer *et al.*, 2011). If the RSU is in the OFF state when a vehicle arrives, it is assumed that this announcement occurs when the RSU returns to the ON state.

### 3.2.4 Flow Graph Sleep Scheduler (FGS)

In the FGS algorithm, the RSU makes locally optimum decisions about when to initiate OFF periods. This is done by computing an estimate of the energy needed to fulfill known vehicle communication requirements both with and without initiating a

new OFF period. These computations are made using all currently available vehicular information and remaining backlog, and is recomputed in each time slot when the RSU is ON. The algorithm is shown in Algorithm 1 and is described in detail as follows.

As shown in Step 5 at time  $t$ , when the RSU is in the ON state and there exist vehicles with unfulfilled backlog within RSU coverage, it first estimates the energy needed to serve all currently known vehicular backlog. This is done in Step 7 by solving ILP 3.15 to 3.18 (introduced below) using the vehicles currently in RSU coverage, i.e.,  $\mathcal{V}_t$ , for the time slots in  $\mathcal{T}_t$ , which is defined to be the union of all time slots for which vehicles in  $\mathcal{V}_t$  are within RSU coverage at that time. This ILP can be efficiently solved using a minimum cost flow graph (Hammad *et al.*, 2010)(Ahuja *et al.*, 1993), and is discussed in detail below. In Steps 8 and 9, ILP 3.15 to 3.18 is again solved but *assuming* an OFF period of duration  $T_s$  starting at time slot  $t$ . This is done by using the same known vehicular inputs but by removing those time slots which are used for sleeping the RSU. This gives the set of time slots  $\mathcal{T}'_t$ .

The ILP used in Steps 7 and 9 is similar to ILP 3.5 to 3.14 except that optimization of the OFF period is not included. It solves for the minimum energy schedule for the RSU using the currently available inputs at time  $t$ , namely  $\mathcal{V}_t$  and  $\mathcal{T}_t$ , and is given as

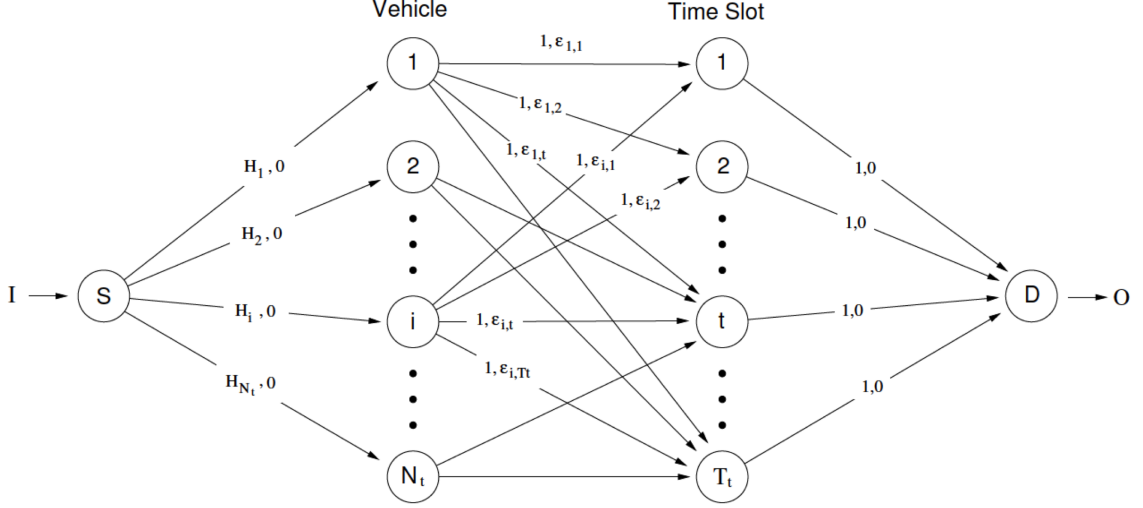


Figure 3.4: Minimum Energy Flow Graph Scheduler. Each edge is labeled with an ordered pair,  $(u_{i,t}, \epsilon_{i,t})$ , where  $u_{i,t}$  and  $\epsilon_{i,t}$  are the capacity and cost of using edge  $(i,t)$ . The input and output links,  $I$  and  $O$ , carry a flow of  $\sum_{i=1}^N H_i$  with a 0 edge cost.

follows.

$$\underset{K_{i,t}}{\text{minimize}} \quad \sum_{t \in \mathcal{T}_t} \sum_{i \in \mathcal{V}_t} \epsilon_{i,t} K_{i,t} \quad (3.15)$$

$$\text{subject to} \quad \sum_{t \in \mathcal{T}_t} K_{i,t} = \lceil \widetilde{R}_i / B \rceil, \quad \forall i \in \mathcal{V}_t \quad (3.16)$$

$$\sum_{i \in \mathcal{V}_t} K_{i,t} \leq 1, \quad \forall t \in \mathcal{T}_t \quad (3.17)$$

$$K_{i,t} \in \{0, 1\}, \quad \forall \{i \in \mathcal{V}_t, t \in \mathcal{T}_t\} \quad (3.18)$$

Constraint 3.16 satisfies the *residual* (i.e., remaining un-served) transmission requirement for vehicle  $i$ , given by  $\widetilde{R}_i$ . ILP 3.15 to 3.18 is in a form that can be solved in time complexity which is polynomial in the number of time slots, i.e.,  $|\mathcal{T}_t|$ , using a minimum cost flow graph (Hammad *et al.*, 2010)(Ahuja *et al.*, 1993). This is shown in Figure 3.4, where  $G = (V, E)$  is defined by a set  $V$  of vertices (nodes) and a set  $E$

**Algorithm 1** Flow Graph (FGS) Scheduler

---

```

1:  $\mathcal{V}_t$  = the set of vehicles with unfulfilled backlog within RSU coverage at time  $t$ .
2:  $\mathcal{T}_t$  = the union of all time slots for which vehicles in  $\mathcal{V}_t$  are within RSU coverage
   at time  $t$ .
3:  $S_t^a$  = the active schedule at time  $t$ .
4: for all  $t = 1 \rightarrow T$  do
5:   if RSU state is ON then
6:     if  $\mathcal{V}_t \neq \emptyset$  then
7:        $S_t$  = solve (ILP 3.15 to 3.18,  $\mathcal{V}_t$ ,  $\mathcal{T}_t$ ).
8:        $\mathcal{T}'_t \leftarrow \mathcal{T}_t - \{t' \mid t \leq t' \leq t + T_s\}$ 
9:        $S'_t$  = solve (ILP 3.15 to 3.18,  $\mathcal{V}_t$ ,  $\mathcal{T}'_t$ ).
10:      if  $\text{energy}(S_t) \leq \text{energy}(S'_t)$  then
11:         $S_t^a = S_t$ , Keep RSU in ON state and continue RSU-to-vehicle transmis-
          sion
12:      else
13:         $S_t^a = S'_t$ , Set RSU state to OFF for  $T_s$  time slots
14:      end if
15:    else
16:      Set RSU state to OFF, for a sleep duration of  $T_{off}$  time slots.
17:    end if
18:  else
19:    if RSU OFF period has expired then
20:      Set RSU state to ON.
21:    end if
22:  end if
23: end for

```

---

of edges (arcs) connecting the nodes. For each edge  $(i, j) \in E$  there is a capacity  $u_{i,j}$  that gives the maximum flow on the edge. Each edge  $(i, j)$  also has an associated cost,  $\epsilon_{i,t}$ , that denotes the cost per unit flow on that edge. These are written as ordered pairs,  $(u_{i,t}, \epsilon_{i,t})$ , on each graph edge in Figure 3.4.

The flow enters and exits the graph at dummy nodes  $S$  and  $D$ , respectively. The first column of nodes represents all vehicles in  $\mathcal{V}_t$ , where it has been assumed that  $N_t = |\mathcal{V}_t|$ . The second column represents all time slots in  $\mathcal{T}_t$ , where  $T_t = |\mathcal{T}_t|$ . Each vehicle node has edges connected to the time slot nodes during which the vehicle is

inside the RSU coverage area. The capacity for an edge from the source  $S$  to a vehicle node is the residual communication requirement for vehicle  $i$  in time slots, denoted by

$$H_i \triangleq \lceil \frac{\widetilde{R}_i}{B} \rceil. \quad (3.19)$$

The capacity for an edge from any time slot node to the destination  $D$  is 1 which prevents time slots from being used more than once. The edges between a vehicle  $i$  and a time slot also has a capacity of 1 which ensures that only one unit of transmission requirement can be assigned to a given time slot. The cost for using the edges originating from Node  $S$  or terminating at Node  $D$  is zero. Finally, the cost of the edges between the vehicle and time slot nodes is given by  $\epsilon_{i,t}$  which is the energy cost of communication to vehicle  $i$  at time  $t$ . Finding the minimum cost flow for graph  $G$  provides the minimum energy the RSU must consume to schedule vehicle transmission requirements for the given set of inputs. The Integrality Property Theorem (Ahuja *et al.*, 1993) ensures that provided input flows and capacities are integer, the resulting minimum cost flow will also be integer. Since our vehicle to time slot edge capacities are 1, the resulting flows are binary and gives the optimum values for  $K_{i,t}$ .

In Algorithm 1, once the schedule for the RSU has been updated in Steps 7 to 9, the minimum energy option is assigned as the active schedule, in Step 10.

As shown in Step 16, if there is no vehicle with unfulfilled backlog within RSU coverage, the RSU enters a sleep period with duration of  $T_{off}$  time slots. This step is necessary so that the RSU can be informed of newly arrived vehicles. When the RSU is in the OFF state, Steps 19 and 20 are used to test if the RSU should return to the ON state.

---

**Algorithm 2** Greedy Scheduler

---

```

1:  $\mathcal{V}_t$  = the set of unscheduled vehicles with unfulfilled backlog within RSU coverage
   at time  $t$ .
2:  $\mathcal{T}_t$  = the union of all time slots for which vehicles in  $\mathcal{V}_t$  are within RSU coverage
   at time  $t$ .
3: for all  $t = 1 \rightarrow T$  do
4:   if RSU state is ON then
5:     if  $\mathcal{V}_t \neq \emptyset$  then
6:        $S_t$  = solve (ILP 3.15 to 3.18,  $\mathcal{V}_t$ ,  $\mathcal{T}_t$ ).
7:        $\mathcal{T}'_t$  = the set of all time slots which are assigned in  $S_t$ .
8:        $t_c = \min \{j \mid j \in \mathcal{T}'_t\}$ 
9:       if  $t_c - t < \check{O}$  or  $(t_c - t)\epsilon_I < \epsilon_X$  then
10:        Stay in ON mode and continue RSU-to-Vehicle transmission.
11:       else
12:        Set RSU state to OFF until time slot  $t_c$ .
13:       end if
14:     else
15:       Set RSU state to OFF, for a sleep duration of  $T_{off}$  time slots.
16:     end if
17:   else
18:     if RSU OFF period has expired then
19:       Set RSU state to ON.
20:     end if
21:   end if
22: end for

```

---

### 3.2.5 Greedy Scheduler

In this algorithm, first time slots are assigned to vehicles that are within the RSU coverage area at time  $t$ . This is done by solving ILP 3.15 to 3.18. Then according to this schedule and by considering the minimum sleep duration constraint and also the energy usage of the state transition, the RSU makes ON/OFF decisions. The algorithm is shown in Algorithm 2 and is described in detail as follows.

As shown in Step 4 at time  $t$ , when the RSU is in the ON state and there exist vehicles with unfulfilled backlog within RSU coverage, first the energy needed to serve

all currently known vehicular backlog is estimated. This is done in Step 6 by solving ILP 3.15 to 3.18 using the vehicles currently in RSU coverage, i.e.,  $\mathcal{V}_t$ , for the time slots in  $\mathcal{T}_t$ . As in Section 3.2.4, this ILP is solved using a minimum cost flow graph. Afterwards, in steps 9 to 12, the RSU makes an ON/OFF decision based on the result of the comparisons shown in step 9. There are two necessary conditions for the RSU to be able to enter sleep mode: (1) the remaining time to the next closest serving time should be more than the minimum sleep duration time,  $\check{O}$ , and, (2) the energy consumed during this period as the RSU stays idle should be more than the turning-on (state transition) energy usage.

If there is no vehicle with unfulfilled backlog within RSU coverage, the RSU enters a sleep period with duration of  $T_{off}$  time slots, as shown in step 15. When the RSU is in the OFF state, Steps 18 and 19 are used to test if the RSU should return to the ON state.

Note that in this algorithm, unlike Algorithm 1, once a schedule is set, it won't be updated in the future, i.e., time slot assignments are never revoked. This is incorporated in Algorithm 2 by defining  $\mathcal{V}_t$  as the set of vehicles with unfulfilled backlog who have *not been scheduled*.

### 3.3 Variable Bit Rate Communication Model

#### 3.3.1 Model Description

In this section, we consider the case where no power control is used, but instead, variable bit rate communication is used. In this model we use recurring frames during which communication can occur. The frames are selected so that the bit rate



available to vehicles in a given frame is constant throughout the frame. However, in the simulation results we assume that all the frames have the same duration. As in the constant bit rate case, a minimum duration OFF interval is considered to account for overhead associated with RSU state transitions.

The objective is to design energy efficient scheduling algorithms that serve vehicles at good locations but also provides opportunities for the RSU to go into sleep mode. First a lower bound on the total energy usage of the RSU is computed, by solving an Integer Linear Program (ILP). Then three online algorithms are proposed. The first algorithm is based on solving the relaxed version of the same ILP used for computing the lower bound. The second, schedules vehicles in a first-come-first-served (FCFS) manner, and based on the resulted schedule determines the ON/OFF state of the RSU. The third algorithm uses an approach similar to the FGS algorithm in the constant bit rate case.

### 3.3.2 Lower Bound Formulation

As in the previous case, a lower bound on the optimum energy scheduling is derived, given an input sample function of arriving vehicles with known speed and traffic requests. It is assumed that there are  $N$  vehicles indexed by the set  $\mathcal{N} = \{1, 2, \dots, N\}$  and that there are  $F$  number of frames indexed by the set  $\mathcal{F} = \{1, 2, \dots, F\}$  over which the scheduling is to occur. Since this is an offline schedule, vehicle trace information, such as arrival time and speed, is known, and the frame set  $\mathcal{F}$  is also completely determined.  $R_i$  is the downlink communication demand for vehicle  $i$  in bits.  $K_{i,f}$  is a variable which is defined to be the amount of downlink flow in bits from the RSU to vehicle  $i$  in frame  $f$ .  $B_{i,f}$  is the bit rate of the downlink communication

from RSU to vehicle  $i$  at frame  $f$ . The duration of frame  $f$  in seconds is denoted by  $\tau_f$ .  $\rho_T$  and  $\rho_I$  are two constants which account for energy usage of transceiver and base equipment per second, respectively.  $\mathcal{E}_f$  is defined to be the energy consumed by the RSU in frame  $f$  when it is in the ON state, i.e.,

$$\mathcal{E}_f = \rho_T \sum_{i \in \mathcal{N}} K_{i,f} / B_{i,f} + \rho_I \tau_f \quad (3.20)$$

When the RSU is in the idle/receive mode in frame  $f$ , i.e.  $K_{i,f} = 0$ , the energy consumption is equal to the second term in equation 3.20 and when it is in the transmission mode, the energy consumption is equal to the summation of both first and second terms. The objective is to assign frames for downlink transmission and an ON/OFF schedule for the RSU that minimizes the total energy use over  $\mathcal{F}$ . As in the constant bit rate case, the binary variable  $\mathcal{O}_f$  is defined as follows

$$\mathcal{O}_f = \begin{cases} 1 & \text{if the RSU is in the ON state in frame } f \\ 0 & \text{otherwise.} \end{cases} \quad (3.21)$$

The binary variable  $X_f$  is equal to 1 if the RSU transitions from the ON to the OFF state at frame  $f$ . This can be written as

$$X_f = \begin{cases} 1 & \text{if } \mathcal{O}_{f-1} = 1 \text{ and } \mathcal{O}_f = 0, \\ 0 & \text{otherwise.} \end{cases} \quad (3.22)$$

$\epsilon_X$  is defined to be the energy required by the RSU during the ON-to-OFF and its associated OFF-to-ON state transition. Using these definitions the following integer

program (IP) can be written, which gives a lower bound on the RSU energy required.

$$\underset{K_{i,f}, \mathcal{O}_f}{\text{minimize}} \quad \sum_{f \in \mathcal{F}} (\rho_T \sum_{i \in \mathcal{N}} K_{i,f} / B_{i,f} + \rho_I \tau_f) \mathcal{O}_f + \epsilon_X \sum_{f \in \mathcal{F}} X_f \quad (3.23)$$

$$\text{subject to} \quad \sum_{f=1}^F K_{i,f} = R_i, \quad \forall i \in \mathcal{N} \quad (3.24)$$

$$\sum_{i \in \mathcal{N}} K_{i,f} / B_{i,f} \leq \tau_f, \quad \forall f \in \mathcal{F} \quad (3.25)$$

$$K_{i,f} \geq 0, \quad \forall i \in \mathcal{N}, f \in \mathcal{F} \quad (3.26)$$

$$K_{i,f} \leq N \cdot \mathcal{O}_f, \quad \forall i \in \mathcal{N}, f \in \mathcal{F} \quad (3.27)$$

$$X_f \leq \mathcal{O}_{f-1}, \quad \forall f \in \mathcal{F} \quad (3.28)$$

$$X_f \geq \mathcal{O}_{f-1} (1 - \mathcal{O}_f), \quad \forall f \in \mathcal{F} \quad (3.29)$$

$$X_f \leq 1 - \mathcal{O}_{f-1} \mathcal{O}_f, \quad \forall f \in \mathcal{F} \quad (3.30)$$

$$\sum_{l=0}^{\mathcal{O}-1} \mathcal{O}_{f+l} \leq M(1 - X_f), \quad \forall f \in \mathcal{F} \quad (3.31)$$

$$K_{i,f} \leq Z_{i,f} \cdot \mathcal{U}, \quad \forall i \in \mathcal{N}, f \in \mathcal{F} \quad (3.32)$$

$$K_{i,f} \geq Z_{i,f} \cdot \mathcal{L}, \quad \forall i \in \mathcal{N}, f \in \mathcal{F} \quad (3.33)$$

$$\mathcal{O}_f, X_f \in \{0, 1\}, \quad \forall f \in \mathcal{F} \quad (3.34)$$

$$Z_{i,f} \in \{0, 1\}, \quad \forall i \in \mathcal{N}, f \in \mathcal{F} \quad (3.35)$$

In IP 3.23 to 3.35, the objective function is the total energy used by the RSU. Equation 3.23 contains two parts. The first is the energy expenditure from Equation 3.20 summed over those frames when the RSU is in the ON state. The second part is the energy consumed during the OFF/ON transitions after awakening the RSU, i.e.,

the summation over all  $X_f$  gives the number of OFF periods that the RSU incurred. Since the RSU is in a deep sleep during the OFF periods, it is assumed that power consumption associated with this is negligible compared with the other states, except for the aforementioned transition energy. Constraint 3.24 ensures that each vehicle communication requirement is fulfilled by summing the appropriate values of  $K_{i,f}$ . Constraint 3.25 ensures that flow allocated to a single frame is such that the transmission time is less than or equal to the frame duration. Constraints 3.26 and 3.27 are necessary because negative values for flow variables,  $K_{i,f}$ , are not acceptable and the RSU can only assign frames when it is in the ON state. In constraint 3.27,  $N$  is a large constant. Constraints 3.28 to 3.30 are used to properly define the  $X_f$  variables, and as given in Definition 3.22, these constraints work similar to constraints 3.9 to 3.11 in the constant bit rate case. Constraint 3.31 ensures that when there is an ON/OFF transition, the RSU cannot be active for at least  $\check{O}$  frames, i.e., the time which is required to re-awaken the RSU from OFF mode. When  $X_f = 1$ , Constraint 3.31 makes sure that at least the next  $\check{O}$  frames are not usable. In this constraint,  $M$  is a constant where  $M \geq \check{O}$ .  $Z_{i,f}$  is a binary variable which is equal to 1 when RSU transmits to vehicle  $i$  at frame  $f$  and 0 otherwise. Constraints 3.32 and 3.33 ensure that flow variable  $K_{i,f}$  stays in an appropriate practical range.  $\mathcal{L}$  and  $\mathcal{U}$  are minimum and maximum acceptable values for the flow variable in bits. Constraints 3.34 and 3.35 define the binary variables.

IP 3.23 to 3.34 contains binary quadratic terms in the objective function and in Constraints 3.29 and 3.30. But these equations can be linearized by introducing supplementary variables. The obtained Lower Bound is used for comparisons with online algorithms to be introduced in the next section.

### 3.3.3 Online Scheduling Algorithms

The optimization derived in section 3.3.2 gives a lower bound on the downlink RSU energy needed to satisfy vehicular packet requirements. In real situations it is difficult to precisely know the energy costs associated with a given packet transmission, and therefore online algorithms are proposed in this section. The network assumptions are similar to the constant bit rate model. Vehicles may travel at different speeds, but maintain their speed throughout the RSU coverage areas. When vehicles enter the RSU radio coverage area they announce their location, direction and speed. If the RSU is in the OFF state when a vehicle arrives, it is assumed that this announcement occurs when the RSU returns to the ON state. In the online case, the frame set is not determined beforehand. Whenever any vehicle enters or leaves the coverage area a new frame begins. Also as vehicles move through the coverage area due to changes in distance, the bit rate of their communication with the RSU changes, which also causes the start of a new frame. Accordingly, the frame set is being continually updated.

### 3.3.4 LP Solver Scheduler (LPS)

This online algorithm is based on solving the relaxed version of the optimization problem formulated for obtaining the lower bound in Section 3.3.2. Since solving the original ILP is so time consuming, first a relaxation and rounding is performed on the optimization variables. The resulted problem is an LP (Linear Program) which can be easily solved. In each frame that the RSU is in the ON state, this LP is solved, given the currently available vehicular information and remaining backlog. The RSU then takes ON/OFF decisions based on the achieved solution. The algorithm, which is referred to as LP Solver Scheduler (LPS), is shown in Algorithm 3 and is described

---

**Algorithm 3** LP Solver Scheduler

---

```

1:  $\mathcal{V}_f$  = the set of vehicles with unfulfilled backlog within RSU coverage at frame  $f$ .
2:  $\mathcal{F}_f$  = the union of all frames for which vehicles in  $\mathcal{V}_f$  are within RSU coverage at frame  $f$ .
3: for all  $f = 1 \rightarrow F$  do
4:   if RSU state is ON then
5:     if  $\mathcal{V}_f \neq \emptyset$  then
6:        $S_f$  = solve (LP 3.36 to 3.46,  $\mathcal{V}_f$ ,  $\mathcal{F}_f$ ).
7:       Depending on  $S_f$ , either continue RSU-to-vehicle transmission or enter OFF mode (at frame  $f$ ).
8:     else
9:       RSU enters OFF mode for a sleep duration of  $F_{off}$  frames
10:    end if
11:  else
12:    if RSU OFF period has expired then
13:      Set RSU state to ON.
14:    end if
15:  end if
16: end for

```

---

in detail as follows.

As shown in Steps 4 to 6 at frame  $f$ , when the RSU is in the ON state and there exist vehicles with unfulfilled backlog, schedule  $S_f$  is found by solving LP 3.36 to 3.46 (introduced below) using the vehicles currently in RSU coverage, i.e.,  $\mathcal{V}_f$ , for the frames in  $\mathcal{F}_f$ , which is defined to be the union of all frames for which vehicles in  $\mathcal{V}_f$  are within RSU coverage at that frame. In Step 7, the RSU decides to either continue RSU-to-Vehicle transmission or enter the OFF state.

If there is no vehicle with unfulfilled backlog within RSU coverage, the RSU enters a sleep period with duration of  $F_{off}$  frames, as shown in Step 9. When the RSU is in the OFF state, Steps 12 and 13 are used to test if the RSU should return to the ON state. The LP used in Step 6 is achieved by solving a relaxation of ILP 3.23 to 3.35, i.e., the variables which were binary, are now allowed to take any value

in  $[0,1]$ . The result is a linear program (LP) which is shown in equations 3.36 to 3.46. The obtained fractional solutions must then be rounded to appropriate binary values. There are several approaches for doing the rounding. Here, a method is used in which the rounding is performed on the  $X_f$  variables, i.e., after the LP is solved, the largest value among the set of fractional solutions for  $X_f$  is taken, and rounded to 1. Consequently constraint 3.44 forces  $\mathcal{O}_f$  to be 0 for the next  $\check{O}$  frames. Given the new fixed values for some  $X_f$  and  $\mathcal{O}_f$  variables, the rounding process is repeated for multiple iterations, until all the  $X_f$  variables are binary.

$$\underset{K_{i,f}, \mathcal{O}_f}{\text{minimize}} \quad \sum_{f \in \mathcal{F}_f} (\rho_T \sum_{i \in \mathcal{V}_f} K_{i,f} / B_{i,f} + \rho_I \tau_f) \mathcal{O}_f + \epsilon_X \sum_{f \in \mathcal{F}_f} X_f \quad (3.36)$$

$$\text{subject to } \sum_{f \in \mathcal{F}_f} K_{i,f} = \widetilde{R}_i, \quad \forall i \in \mathcal{V}_f \quad (3.37)$$

$$\sum_{i \in \mathcal{V}_f} K_{i,f} / B_{i,f} \leq \tau_f, \quad \forall f \in \mathcal{F}_f \quad (3.38)$$

$$K_{i,f} \geq 0, \quad \forall i \in \mathcal{V}_f, f \in \mathcal{F}_f \quad (3.39)$$

$$K_{i,f} \leq N \cdot \mathcal{O}_f, \quad \forall i \in \mathcal{V}_f, f \in \mathcal{F}_f \quad (3.40)$$

$$X_f \leq \mathcal{O}_{f-1}, \quad \forall f \in \mathcal{F}_f \quad (3.41)$$

$$X_f \geq \mathcal{O}_{f-1} (1 - \mathcal{O}_f), \quad \forall f \in \mathcal{F}_f \quad (3.42)$$

$$X_f \leq 1 - \mathcal{O}_{f-1} \mathcal{O}_f, \quad \forall f \in \mathcal{F}_f \quad (3.43)$$

$$\sum_{l=0}^{\bar{O}-1} \mathcal{O}_{f+l} \leq M(1 - X_f), \quad \forall f \in \mathcal{F}_f \quad (3.44)$$

$$0 \leq \mathcal{O}_f \leq 1, \quad \forall f \in \mathcal{F}_f \quad (3.45)$$

$$0 \leq X_f \leq 1, \quad \forall f \in \mathcal{F}_f \quad (3.46)$$

### 3.3.5 FCFS Sleep Scheduler (FCFSS)

In this algorithm, frames are first assigned to vehicles that are within the RSU coverage area at the current time. This is done by applying a First Come First Serve (FCFS) scheduling algorithm, i.e., the vehicles with the earlier arrival time are scheduled first. Then according to this schedule, and by considering the minimum sleep duration constraint and also the energy usage for state transition, the RSU makes ON/OFF decisions. Algorithm 4 shows the algorithm, and the steps are explained in more detail as follows.

As shown in Steps 4 to 6, at frame  $f$ , when the RSU is in the ON state and there exist vehicles with unfulfilled backlog within RSU coverage, schedule  $S_f$  is found by



---

**Algorithm 4** FCFS Sleep Scheduler

---

```

1:  $\mathcal{V}_f$  = the set of unscheduled vehicles with unfulfilled backlog within RSU coverage
   at frame  $f$ .
2:  $\mathcal{F}_f$  = the union of all frames for which vehicles in  $\mathcal{V}_f$  are within RSU coverage at
   frame  $f$ .
3: for all  $f = 1 \rightarrow F$  do
4:   if RSU state is ON then
5:     if  $\mathcal{V}_f \neq \emptyset$  then
6:        $S_f$  = FCFS Schedule for  $\mathcal{V}_f$  and  $\mathcal{F}_f$ .
7:        $\mathcal{F}'_f$  = the set of all frames which are assigned in  $S_f$ .
8:        $f_c = \min \{j \mid j \in \mathcal{F}'_f\}$ 
9:       if  $f_c - f < \check{O}$  or  $\sum_{i=f}^{f_c} \tau_i \rho_I < \epsilon_X$  then
10:        Stay in ON mode and continue RSU-to-Vehicle transmission.
11:      else
12:        Set RSU state to OFF until frame  $f_c$ .
13:      end if
14:    else
15:      Set RSU state to OFF, for a sleep duration of  $F_{off}$  frames.
16:    end if
17:  else
18:    if RSU OFF period has expired then
19:      Set RSU state to ON.
20:    end if
21:  end if
22: end for

```

---

assigning frames to vehicles in  $\mathcal{V}_f$  with a FCFS schedule, i.e., earlier arriving vehicles are scheduled first. Afterwards, in Steps 9 to 12, the RSU makes ON/OFF decisions based on the result of the comparisons shown in Step 9. The conditions which are checked in this step are similar to the ones in Step 9 of Algorithm 2 in the constant bit rate case. In Step 15, if there is no vehicle with unfulfilled backlog within RSU coverage, the RSU enters a sleep period with duration of  $F_{off}$  frames. When the RSU is in the OFF state, Steps 18 and 19 are used to test if the RSU should return to the ON state.

### 3.3.6 Comparator Sleep Scheduler (CSS)

This algorithm is similar to the FGS algorithm in the constant bit rate case. The RSU makes locally optimum decisions about when to initiate OFF periods. This is done by computing an estimate of the energy needed to fulfill known vehicle communication requirements both with and without initiating a new OFF period of duration  $F_s$  frames, which is given as a parameter. These computations are made using all currently available vehicular information and remaining backlog, and are recomputed in each frame when the RSU is ON. In addition to using frames, there is another difference compared with Algorithm 1. Instead of using a Flow Graph for finding the schedule for the current  $\mathcal{V}_f$  set, an ILP is solved, which is given in equations 3.47 to 3.53. The algorithm is shown in Algorithm 5 and is described in detail as follows.

As shown in Step 5 at frame  $f$ , when the RSU is in the ON state and there are vehicles with unfulfilled backlog within RSU coverage, it first estimates the energy needed to serve all currently known vehicular backlog. This is done in Step 7 by solving ILP 3.47 to 3.53 (introduced below) using the vehicles currently in RSU coverage, i.e.,  $\mathcal{V}_f$ , for the frames in  $\mathcal{F}_f$ . In Steps 8 and 9, again the ILP 3.47 to 3.53 is solved but by *assuming* an OFF period of duration  $F_s$  starting at frame  $f$ . This is done by using the same known vehicular inputs but by removing those frames which are used for sleeping the RSU. This gives the set of frames  $\mathcal{F}'_f$ . The ILP used in Steps 7 and 9 is similar to ILP 3.23 to 3.35 except that the optimization of the OFF period is not included. It solves for the minimum energy schedule for the RSU using the currently

**Algorithm 5** Comparator Sleep Scheduler

---

```

1:  $\mathcal{V}_f$  = the set of unscheduled vehicles with unfulfilled backlog within RSU coverage
   at frame  $f$ .
2:  $\mathcal{F}_f$  = the union of all frames for which vehicles in  $\mathcal{V}_f$  are within RSU coverage at
   frame  $f$ .
3:  $S_f^a$  = the active schedule at frame  $f$ .
4: for all  $f = 1 \rightarrow F$  do
5:   if RSU state is ON then
6:     if  $\mathcal{V}_f \neq \emptyset$  then
7:        $S_f = \text{solve (ILP 3.47 to 3.53, } \mathcal{V}_f, \mathcal{F}_f \text{)}$ .
8:        $\mathcal{F}'_f \leftarrow \mathcal{F}_f - \{f' \mid f \leq f' \leq f + F_s\}$ 
9:        $S'_f = \text{solve (ILP 3.47 to 3.53, } \mathcal{V}_f, \mathcal{F}'_f \text{)}$ .
10:      if  $\text{energy}(S_f) \leq \text{energy}(S'_f)$  then
11:         $S_f^a = S_f$ , Keep RSU in ON mode and continue RSU-to-vehicle transmis-
        sion.
12:      else
13:         $S_f^a = S'_f$ , Set RSU state to OFF for  $F_s$  frames.
14:      end if
15:    else
16:      Set RSU state to OFF, for a sleep duration of  $F_{off}$  frames.
17:    end if
18:  else
19:    if RSU OFF period has expired then
20:      Set RSU state to ON.
21:    end if
22:  end if
23: end for

```

---

available inputs at frame  $f$ , namely  $\mathcal{V}_f$  and  $\mathcal{F}_f$ , and is given as follows.

$$\underset{K_{i,f}}{\text{minimize}} \quad \sum_{i \in \mathcal{V}_f} \sum_{f \in \mathcal{F}_f} \rho_T K_{i,f} / B_{i,f} \quad (3.47)$$

$$\text{subject to } \sum_{f \in \mathcal{F}_f} K_{i,f} = \widetilde{R}_i, \quad \forall i \in \mathcal{V}_f \quad (3.48)$$

$$\sum_{i \in \mathcal{V}_f} K_{i,f} / B_{i,f} \leq \tau_f, \quad \forall f \in \mathcal{F}_f \quad (3.49)$$

$$K_{i,f} \geq 0, \quad \forall i \in \mathcal{V}_f, f \in \mathcal{F}_f \quad (3.50)$$

$$K_{i,f} \leq Z_{i,f} \cdot \mathcal{U}, \quad \forall i \in \mathcal{V}_f, f \in \mathcal{F}_f \quad (3.51)$$

$$K_{i,f} \geq Z_{i,f} \cdot \mathcal{L}, \quad \forall i \in \mathcal{V}_f, f \in \mathcal{F}_f \quad (3.52)$$

$$Z_{i,f} \in \{0, 1\}, \quad \forall i \in \mathcal{V}_f, f \in \mathcal{F}_f \quad (3.53)$$

Constraint 3.48 satisfies the *residual* (i.e., remaining un-served) transmission requirement for vehicle  $i$ , given by  $\widetilde{R}_i$ . ILP 3.47 to 3.53 does not contain quadratic terms and therefore it has a much smaller size compared to ILP 3.23 to 3.35 and can be solved using integer linear programming methods. In Algorithm 5, once the schedule for the RSU has been updated in Steps 7 to 9, the minimum energy option is assigned as the active schedule, in Step 10. As shown in Step 16, if there is no vehicle with unfulfilled backlog, the RSU enters a sleep period with duration of  $F_{off}$  frames. When the RSU is in the OFF state, Steps 19 and 20 are used to test if the RSU should return to the ON state.

# Chapter 4

## Performance Results

In this section the performance of the proposed heuristic algorithms is evaluated. The lower bounds for RSU energy consumption that were derived in Sections 3.2.2 and 3.3.2 are also included and are referred to as Bound in the graphs. A highway environment is assumed where vehicles maintain constant speeds for relatively long time periods and as in references (Wang, 2005) (Khabazian and Ali, 2008) (Bilstrup *et al.*, 2008) Poisson process vehicle arrivals are assumed. Simulations are conducted for the two cases of Constant and Variable bit rate communication.

### 4.1 Constant Bit Rate Model

This section presents performance results for the proposed scheduling algorithms introduced in Sections 3.2.4 and 3.2.5. The online algorithms use knowledge of vehicle position and estimates of downlink transmission energy costs. In the results it is assumed that an accurate prediction of energy costs is possible based on a deterministic path loss scenario using a distance dependent exponential path loss model.

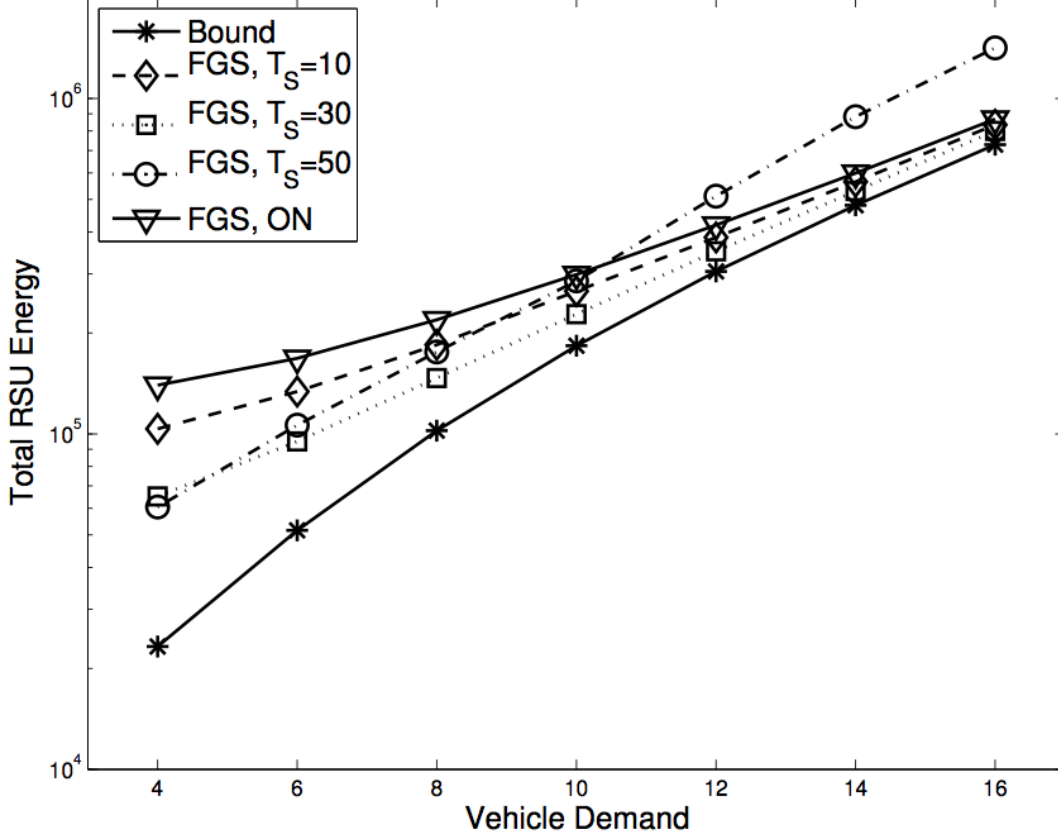


Figure 4.1: Total RSU Energy (J) vs. Vehicular Demand (time slots).  $\epsilon_I = 100$ ,  $\epsilon_X = 500$ , and  $\check{O} = 5$  time slots. Low vehicular arrival rate case.

Figures 4.1 and 4.2 illustrate the total RSU energy use versus vehicular demand for the FGS scheduler with different values of the  $T_s$  parameter. Also included in the graphs are the results that would be obtained without ON/OFF scheduling (i.e., the “FGS, ON” curve) but still using the flow graph scheduler for slot allocation. The analytic lower bound formulation from Section 3.2.2 is also plotted. The other parameters used are,  $\epsilon_I = 100$ ,  $\epsilon_X = 500$ , and  $\check{O} = 5$  time slots. The low and high vehicular arrival rates are 1/100 and 1/20 vehicles/sec, respectively.

In Figure 4.1 the vehicular arrival rate is low and it can be seen that there are

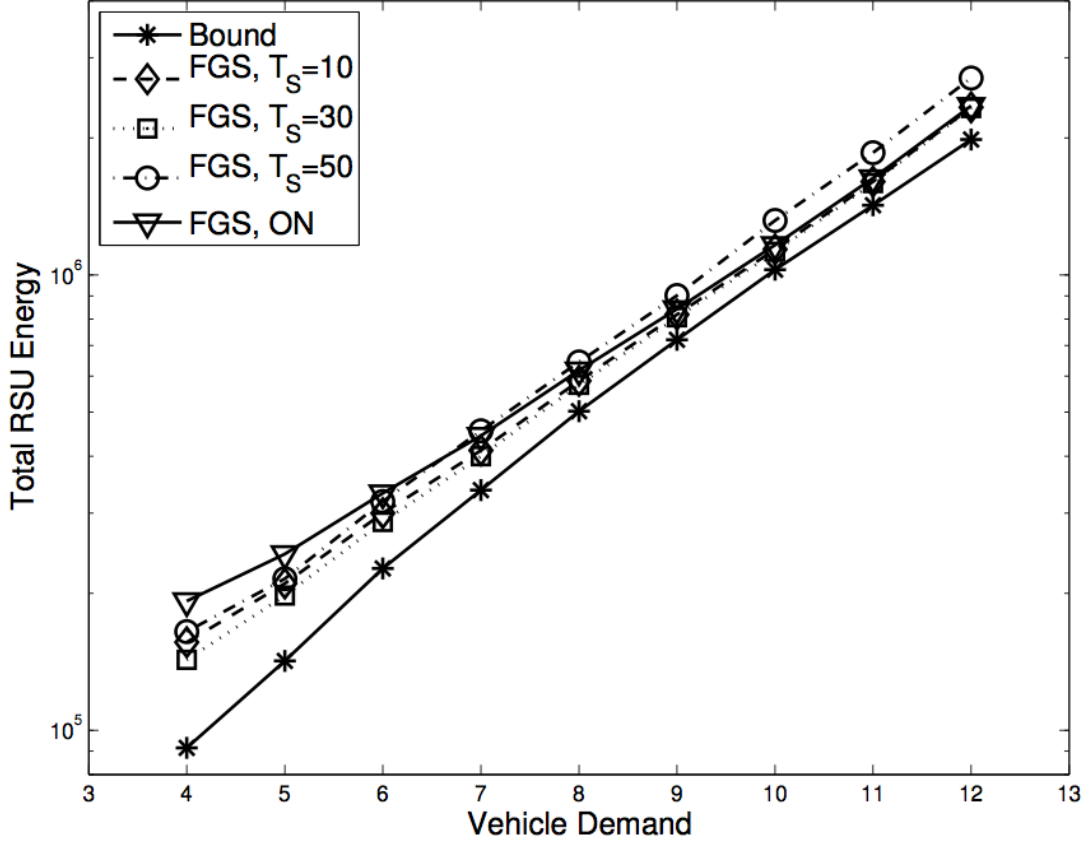


Figure 4.2: Total RSU Energy (J) vs. Vehicular Demand (time slots).  $\epsilon_I = 100$ ,  $\epsilon_X = 500$ , and  $\tilde{O} = 5$  time slots. High vehicular arrival rate case.

significant improvements in energy use, with the best performance obtained when  $T_s = 30$ . When  $T_s$  is decreased to 10 or increased to 50 the results become worse. The reason for this is that when  $T_s$  is too small, the repeated instances of ON/OFF cycles incurs energy penalties which are proportionately higher than would be the case if longer OFF periods are used. However, this effect does not continue indefinitely as can be seen for the  $T_s = 50$  case. Instead, when  $T_s$  is too large, the RSU remains in the OFF state too long and is unable to schedule vehicular time slots at the best times. Also, even when this is not the case, the RSU is forced to remain in the ON

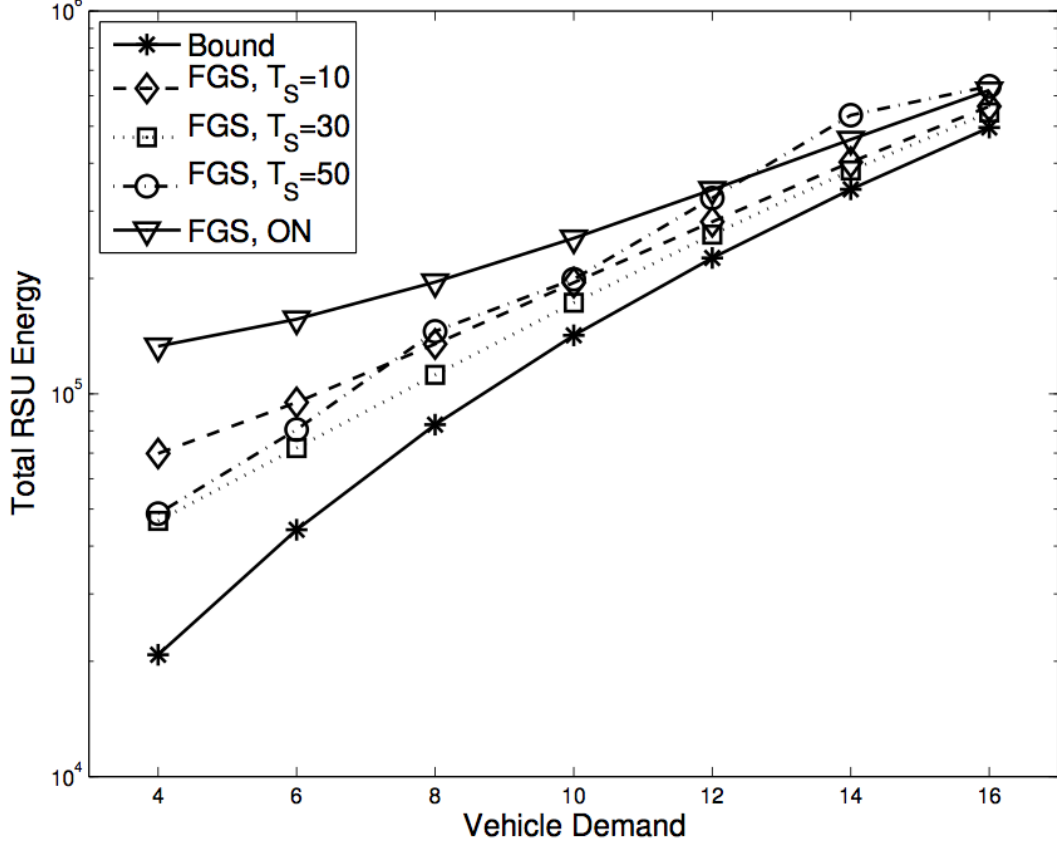


Figure 4.3: Total RSU Energy (J) vs. Vehicular Demand (time slots).  $\epsilon_I = 100$ ,  $\epsilon_X = 100$ , and  $\check{O} = 5$  time slots. Low vehicular arrival rate case.

state for lengthy time periods while vehicles move into more favourable positions. This is the reason that the curve for  $T_s = 50$  does increasingly worse as the vehicle demand increases.

It can be seen in general that as the vehicle demand increases, there is less opportunity for ON/OFF cycling and as a result, for large enough vehicular demands, FGS without sleeping performs the same as in the different  $T_s$  cases. In this case the curves become much closer to the Bound, which indicates that the flow graph scheduler is doing a good job of allocating time slots. At the extreme left of the graph it can be seen that



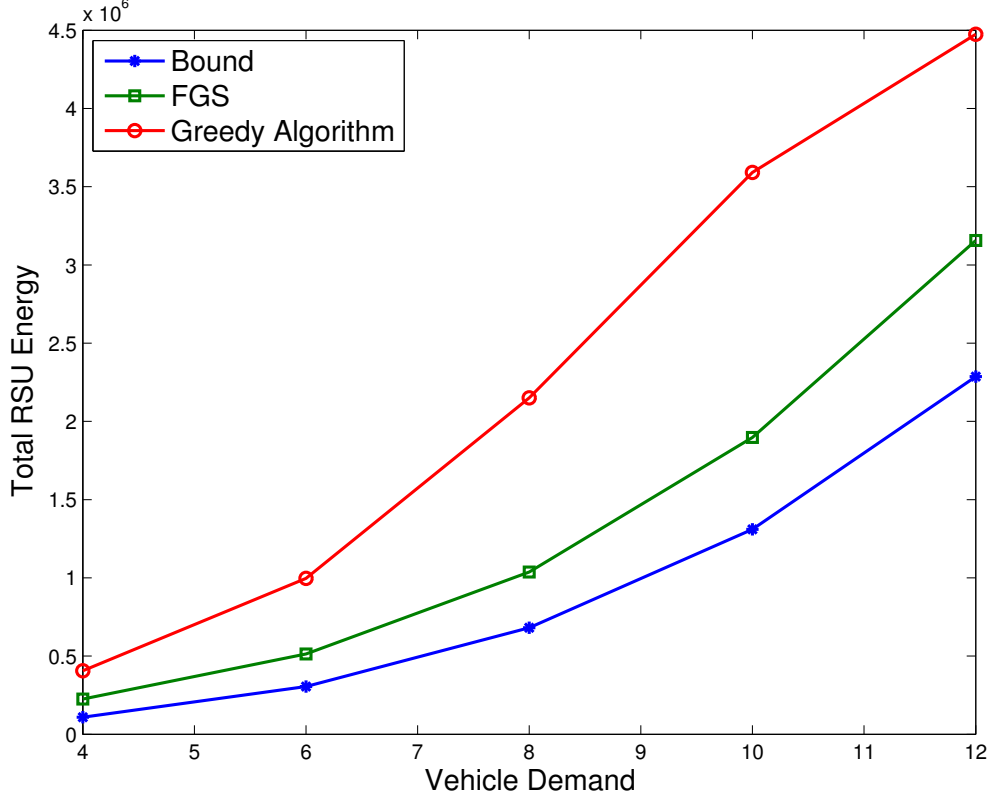


Figure 4.4: Total RSU Energy (J) vs. Vehicular Demand (time slots).

the FGS performance is quite a bit above that of Bound, suggesting that a scheduler with adaptive sleep periods may perform better. This is not certain however, since the Bound has non-causal knowledge of future vehicle arrivals and adjusts its sleep periods accordingly. Figure 4.2 uses the same parameters as Figure 4.1 except that the vehicle arrival rate is higher. This results in increased contention and less opportunity for RSU sleeping. However, the same trends occur.

In Figure 4.3 another example is shown with the same parameters as in Figure 4.1 except that the OFF/ON energy cost is significantly decreased to  $\epsilon_X = 100$ . It can be seen that doing so makes the value of OFF/ON cycling much more attractive

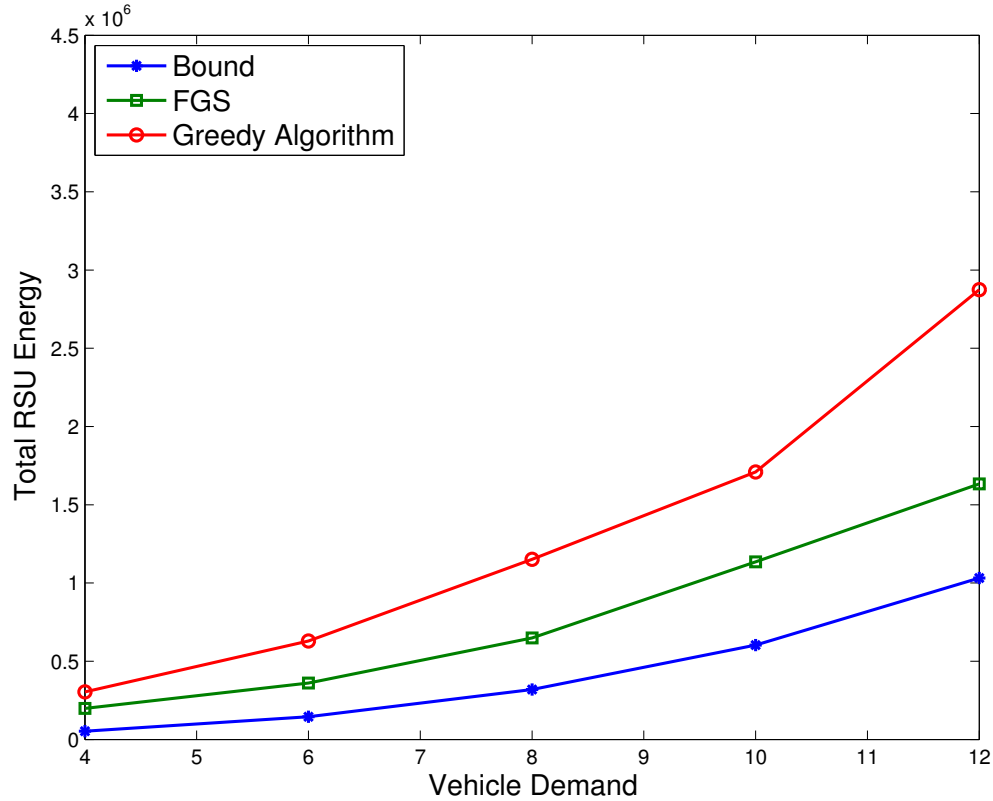


Figure 4.5: Total RSU Energy (J) vs. Vehicular Demand (time slots). Vehicle Arrival Rate =  $1/50$ .

resulting in much improved energy performance compared with the FGS algorithm without deep sleeping. As before, if the vehicle arrival rate significantly increases, the value of sleep cycling becomes less as was the case in Figure 4.2.

In Figures 4.4, 4.5 and 4.6, the performance of the FGS and the Greedy algorithm introduced in section 3.2.5, are compared. In Figure 4.4 it is assumed that the vehicle arrival rate is equal to  $1/30$  vehicles/sec, and the speed of each vehicle is randomly chosen from the set  $\{10, 20, 30\}$  m/s. Parameter  $T_s$  is also equal to 30 time slots in the FGS algorithm. It can be seen that when the vehicular demand is low, the

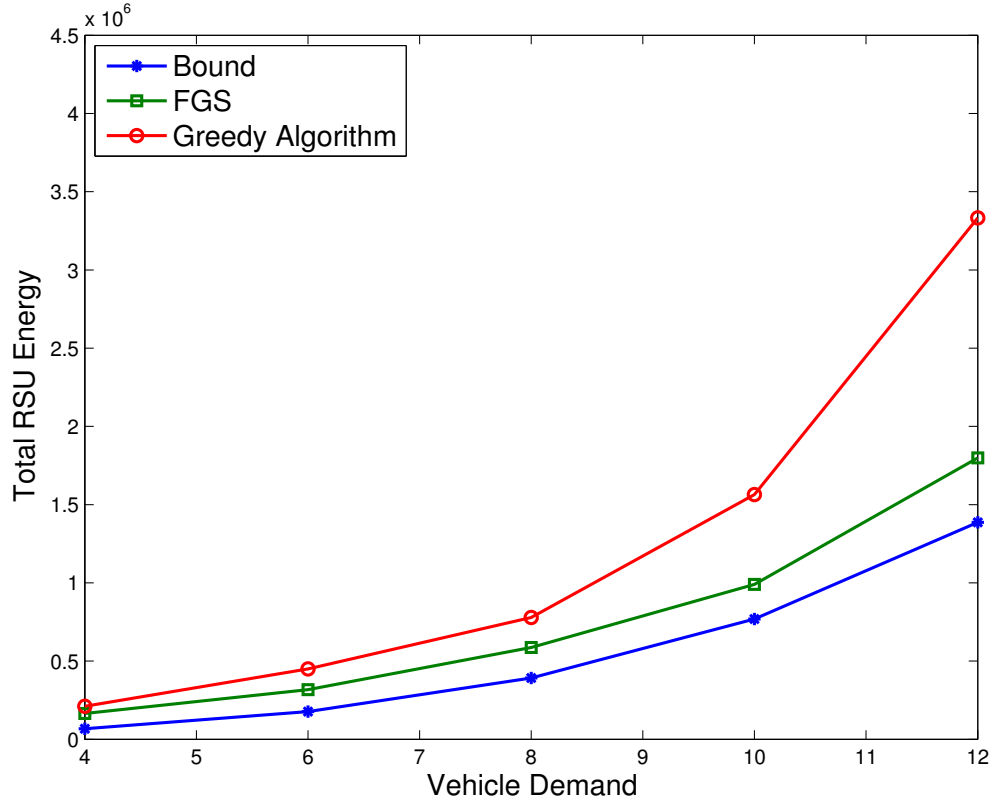


Figure 4.6: Total RSU Energy (J) vs. Vehicular Demand (time slots). Vehicle Speed = 10, 20 m/s.

Greedy algorithm performs close to the Bound and the FGS algorithm, but as the demand increases its performance degrades quickly.

In Figure 4.5 the vehicle arrival rate is decreased to 1/50 vehicles/sec. As expected, the total RSU energy use is decreased in all the cases. There is also less difference between the performance of the two heuristics in this case compared to the case with higher vehicle arrival rate.

In Figure 4.6, the same parameters as in Figure 4.4 are used, except for the speed set which is changed to  $\{10, 20\}$  m/s. In general the total RSU energy use is decreased.

It also can be seen that the gap between the heuristic curves and the Bound has become smaller.

## 4.2 Variable Bit Rate Model

This section presents performance results for the scheduling algorithms introduced in Sections 3.3.4, 3.3.5 and 3.3.6. The lower bound for RSU energy consumption that was derived in Section 3.3.2 is also included and is referred to as Bound in the graphs. Four different experiments have been performed. In the first experiment, the heuristics are tested with a base set of parameters and assumptions, shown in Table 4.1. Two types of graphs are plotted, which show total RSU energy use versus vehicular demand and vehicle arrival rate, respectively. These graphs are shown in Figures 4.7, 4.8 and 4.9.

In the second experiment the effect of vehicle arrival rate on the performance of the three heuristics is investigated. In order to do this all the parameters in Table 4.1 are kept the same except the Vehicle Arrival Rate, which is changed to 1/100 vehicles/sec. Figures 4.10 and 4.11 show the outcome of this experiment. Parameter  $\rho_I$ , which accounts for energy consumption of the RSU during a non-downlink transmission frame when it is in ON state, is assumed to be equal to 100 J/sec in the base parameters. When this parameter changes the energy that is consumed during idle periods is changed. In order to investigate the effect of this change on the performance of the heuristics, the third and fourth experiments were performed in which  $\rho_I$  is equal to 120 J/sec and 70 J/sec respectively. The related graphs are shown in two groups: Figures 4.12, 4.13, 4.14 and Figures 4.15, 4.16, 4.17.

Table 4.1: Base set of Simulation Parameters

Parameter	Value
Energy consumed per second, during a non-downlink transmission frame when the RSU is in the ON state ( $\rho_I$ )	100 joules
Energy consumed by transceiver equipment per second (when in downlink transmission mode) ( $\rho_T$ )	10 joules
Energy required by the RSU during the ON-to-OFF and its associated OFF-to-ON state transition ( $\epsilon_X$ )	50 joules/sec
Bit-Rate range	[6 27] Mbits/sec
Frame duration	50 seconds
The time which is required to re-awaken the RSU from OFF mode	1 frame
RSU Coverage area	4 km
Vehicle Arrival Rate	1/50 vehicles/sec
Vehicle Speed	10, 15 m/s
Duration of OFF period when there are no vehicles in coverage area ( $F_{off}$ )	2 frames

In Figure 4.7, the performance of the three proposed online algorithms are compared, the graphs show the total RSU energy use versus vehicle demand. The analytic lower bound formulation from Section 3.3.2 is also plotted. This figure is related to the first experiment where the base set of parameters from Table 4.1 is used.

It can be seen that LPS has the closest performance to the Bound, while FCFSS has the worst. The reason for this is that FCFSS gives priority to satisfying vehicular demands at the best possible frames in order to minimize the downlink *transmission energy*, therefore it sometimes makes poor On/Off decisions. In LPS however, the

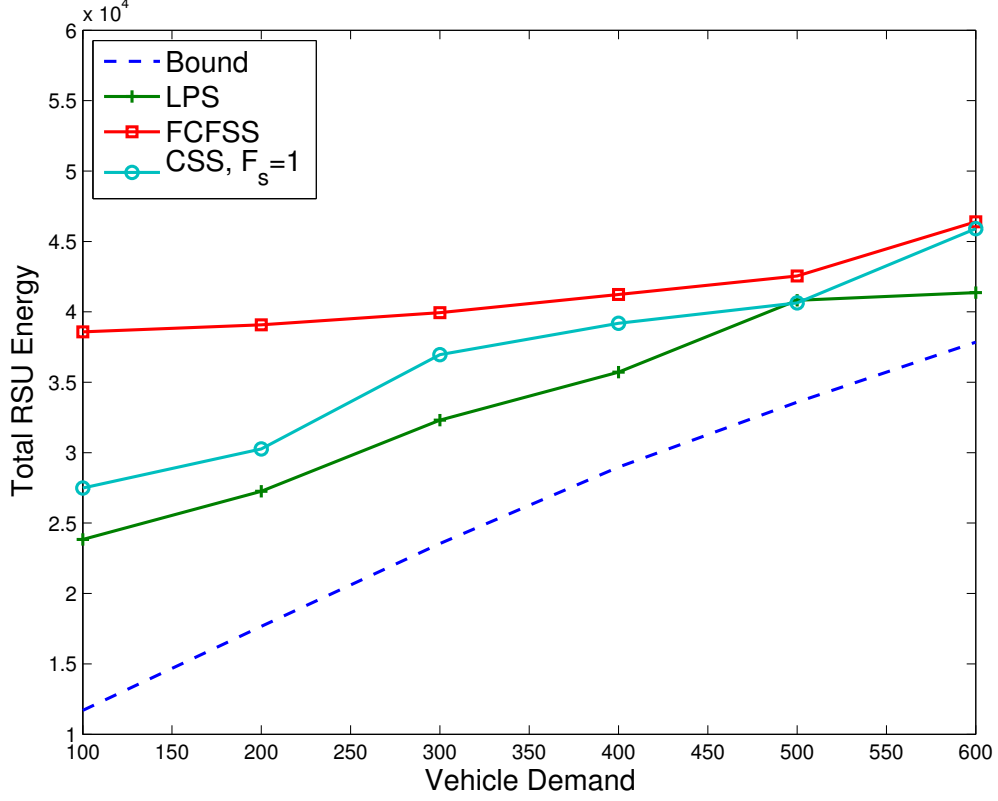


Figure 4.7: Total RSU Energy (J) vs. Vehicular Demand (Mbits)

RSU energy usage in transmitting and idle modes is considered at the same time. It can be seen in general that as the vehicle demand increases, the curves of different heuristics become closer to each other and also the gap between these curves and the Bound becomes smaller. The reason for this is that at higher demands there are not a lot of options for sleeping, therefore the heuristics and the lower bound make similar On/Off decisions. At the extreme left of the graph it can be seen that the performance of the heuristics are above that of Bound. In the case of LPS the reason for this difference is mainly the *state transition* energy usage. This cost is incurred due to frequent (every  $F_{off}$  frames) turning ON of the RSU to check the current traffic

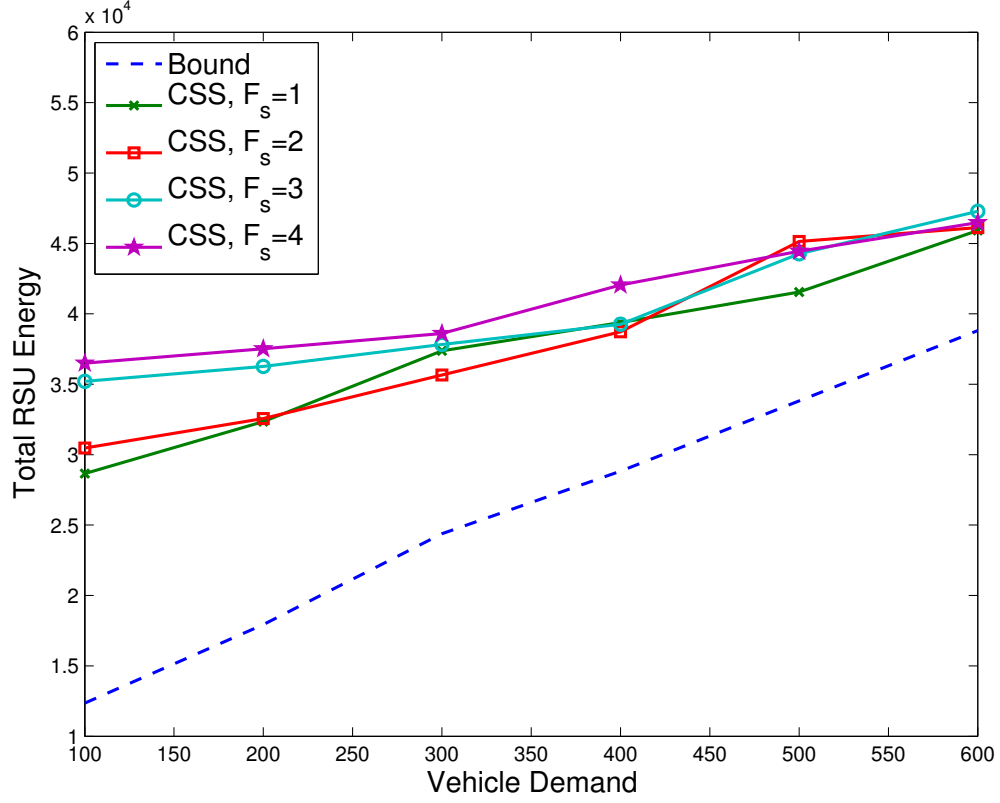


Figure 4.8: Total RSU Energy (J) vs. Vehicular Demand (Mbits)

conditions. This is not necessary for the Bound since it has non-causal knowledge of future vehicle arrivals. As for FCFSS and CSS, besides the above reason, there is the additional energy used during idle periods, because these two heuristics schedule the RSU to be in the ON state for larger numbers of frames. In Figure 4.8 the total RSU energy use versus vehicle demand is plotted for CSS with different values of the  $F_s$  parameter. It can be seen that in low vehicular demands, the heuristic has a better performance when  $F_s = 1$  and 2, However because of the same reason mentioned for Figure 4.7, as the vehicle demand increases, the performance converges, and even when  $F_s = 3$  or 4, lower energy levels are achieved at some points. The reason for this

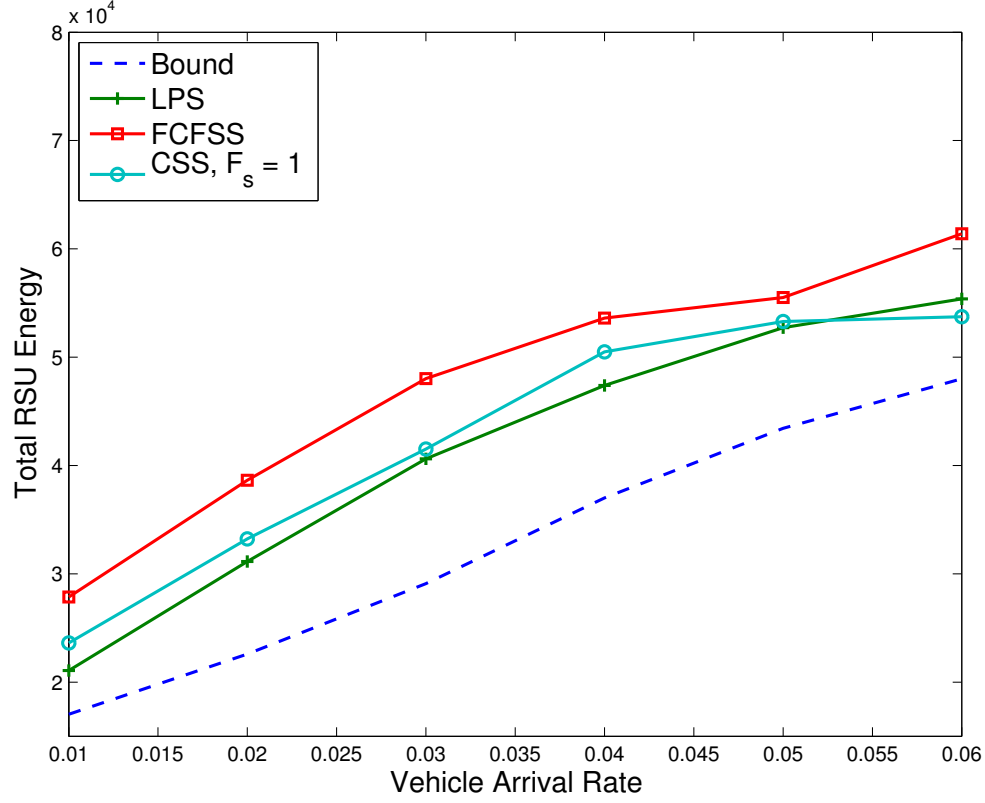


Figure 4.9: Total RSU Energy (J) vs. Vehicle Arrival Rate (vehicles/sec)

is due to the additional transition energy used when  $F_s = 1$  or 2. Figure 4.9 shows total RSU energy use versus vehicle arrival rate for the three proposed heuristics. In this case, the base set of parameters is used, except for the average vehicle demand which is assumed to be constant and equal to 300 Mbits. As before, the advantage of the LPS algorithm is clear in this figure. Again, it can be seen that as the vehicle arrival rate reaches higher values, different algorithms tend to perform similarly.

Figure 4.10, which is related to the second experiment, shows total RSU energy use versus vehicle demand for a lower vehicle arrival rate than in the first experiment, i.e., 1/100 vehicles/sec. This figure can be compared with Figure 4.7 and as expected,



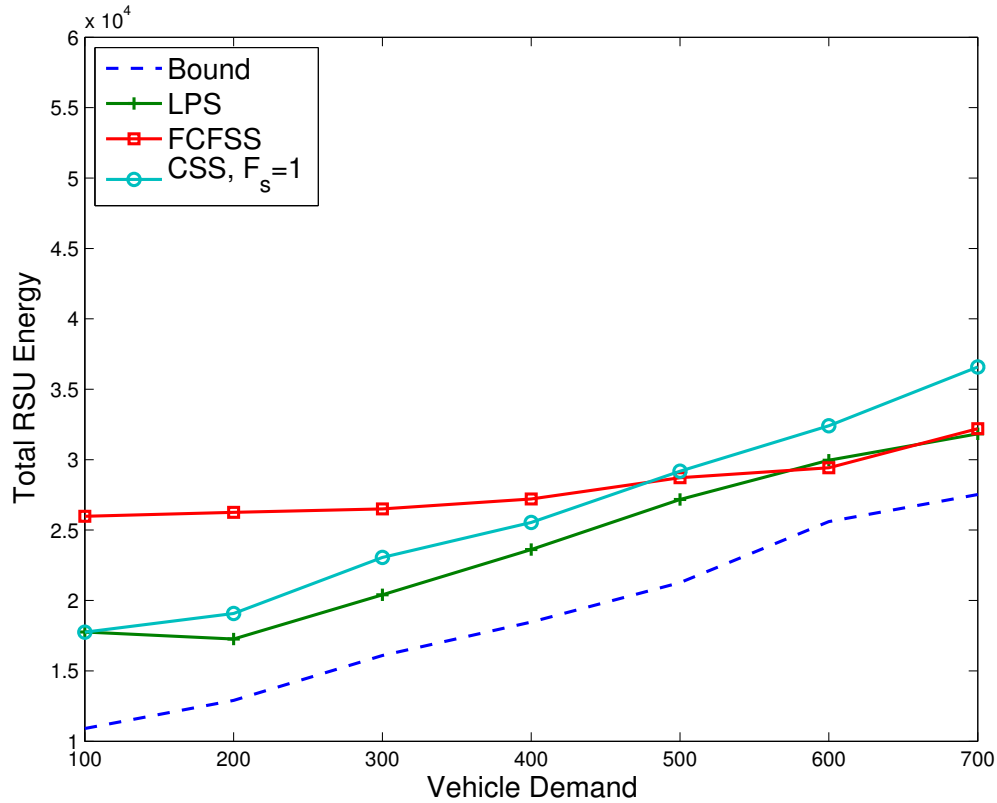


Figure 4.10: Total RSU Energy (J) vs. Vehicular Demand (Mbits). Vehicle Arrival Rate =  $1/100$ .

the RSU energy level is decreased in all the cases. The gap between the heuristics and the Bound has decreased considerably, which shows that the heuristics perform better under lower traffic load. The gap between the heuristic curves themselves also has decreased and the reason is that when there are less vehicles coming into the RSU coverage area, the scheduling of vehicular requests and On/Off frames is simpler and all the heuristics do a good job.

Figure 4.11 also shows the results of the second experiment, specifically for CSS. When compared with Figure 4.8 where the vehicle arrival rate is higher, it can be

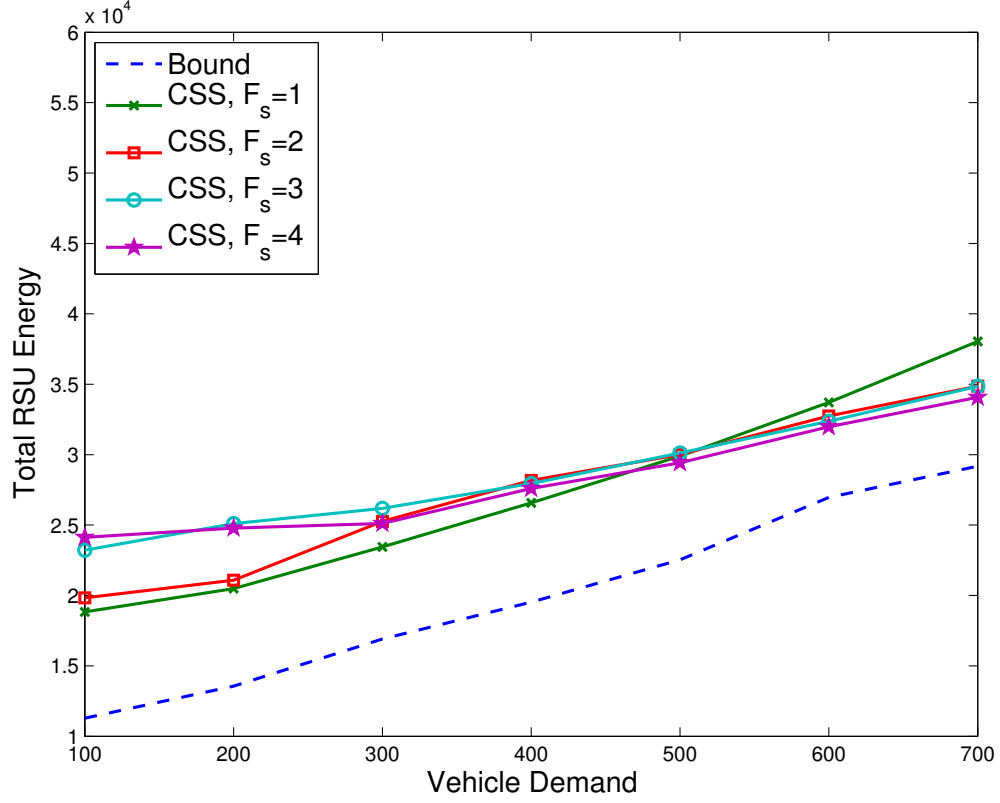


Figure 4.11: Total RSU Energy (J) vs. Vehicular Demand (Mbits). Vehicle Arrival Rate =  $1/100$ .

seen that the performance of the four cases has slightly changed. For example, when  $F_s$  is 3 or 4, the performance has improved. The reason for this is that when vehicle arrivals are infrequent, it is more beneficial to have longer OFF periods. Also the gap between the the Bound and the online algorithms is smaller here than in Figure 4.8, for all the cases.

Figure 4.12 is related to the third experiment, where parameter  $\rho_I$  is equal to 120 J/sec rather than 100 J/sec as in Table 4.1. The total RSU energy use versus vehicle demand is plotted in this figure for the three proposed online algorithms including

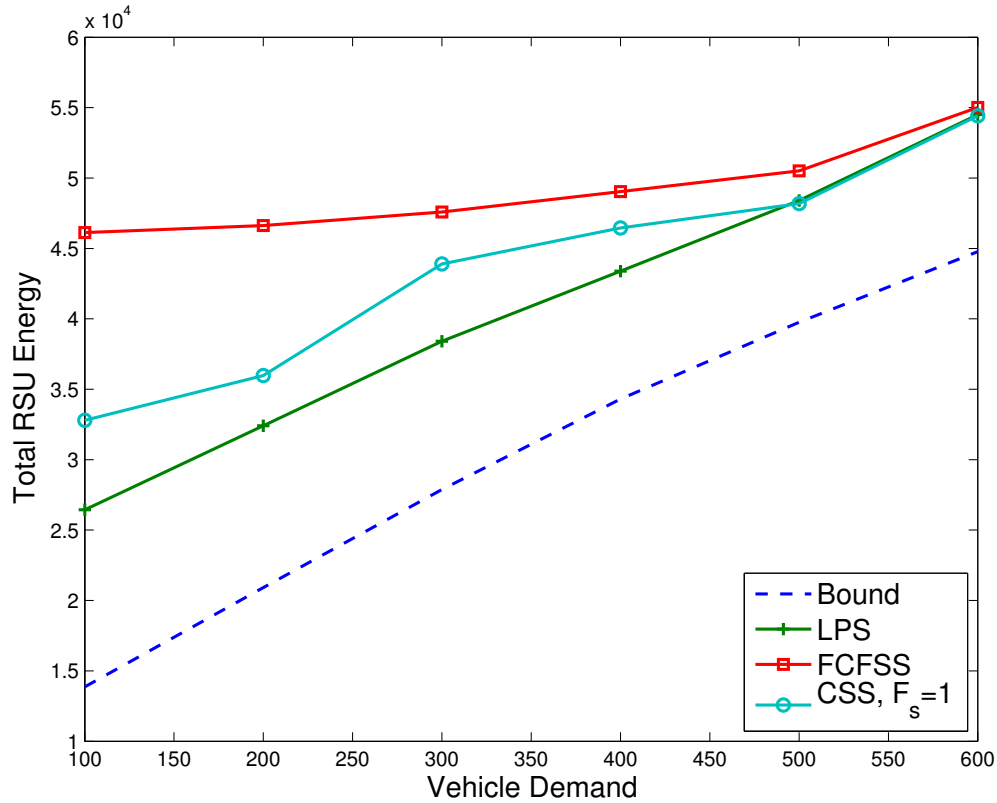


Figure 4.12: Total RSU Energy (J) vs. Vehicular Demand (Mbits).  $\rho_I = 120$ .

the Bound. By comparison with Figure 4.7, it can be seen that in this case the total RSU energy use is increased for all the algorithms. This is because the RSU uses more energy while it is in idle mode compared to the first experiment. Almost the same trend occurs here in terms of different online algorithm performance. However, the gap between the online algorithm curves has become larger, since here, with a higher energy consumption in idle mode, sleeping has more value and the advantage of better On/Off decision taking is more important. For example FCFSS, which is weaker in taking On/Off decisions than LPS, has resulted in a higher energy use, and the difference between these two heuristics has further increased.

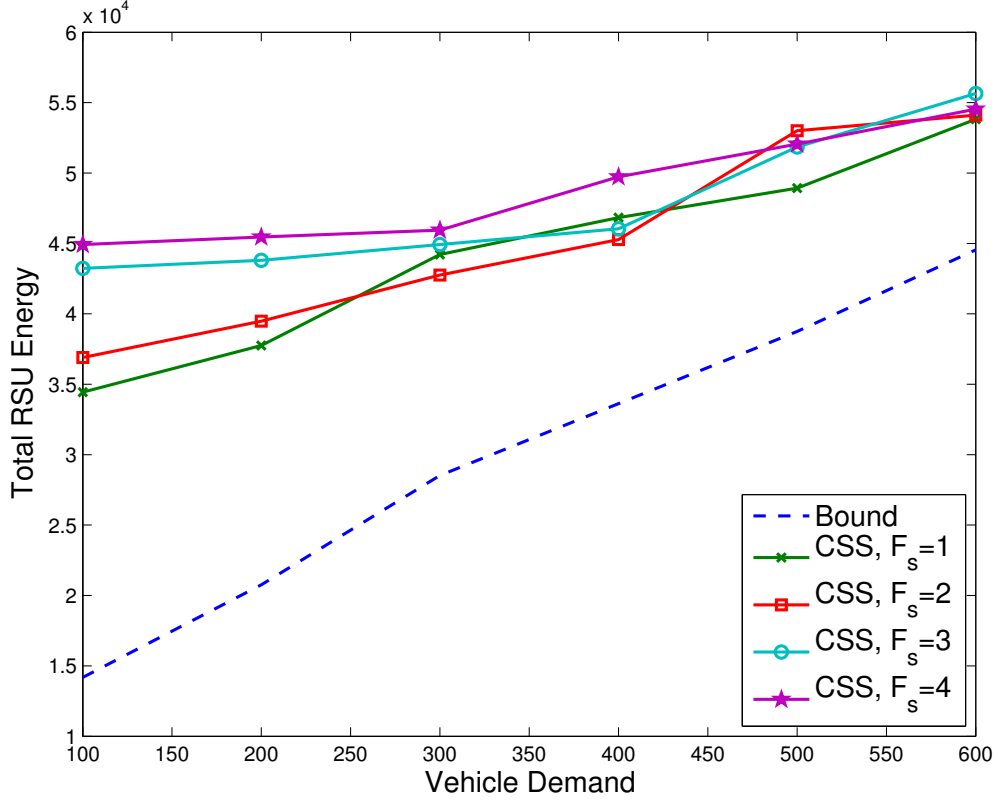


Figure 4.13: Total RSU Energy (J) vs. Vehicular Demand (Mbits).  $\rho_I = 120$ .

Figure 4.13 shows the results of the third experiment specifically for CSS. This figure can be compared with Figure 4.8 and it can be seen that the same trend occurs, only the gap between the curves related to different values of  $F_s$  is slightly increased. The reason for this is explained in the previous section, i.e., higher values of  $\rho_I$  causes the advantage of the better algorithms to become more pronounced.

Figure 4.14 illustrates the performance of the three proposed online algorithms in terms of the total RSU energy use versus vehicle arrival rate. This figure is related to experiment three where the parameter  $\rho_I$  is equal to 120 J/sec and it is comparable to Figure 4.9 from experiment one. The observations are similar to the ones mentioned

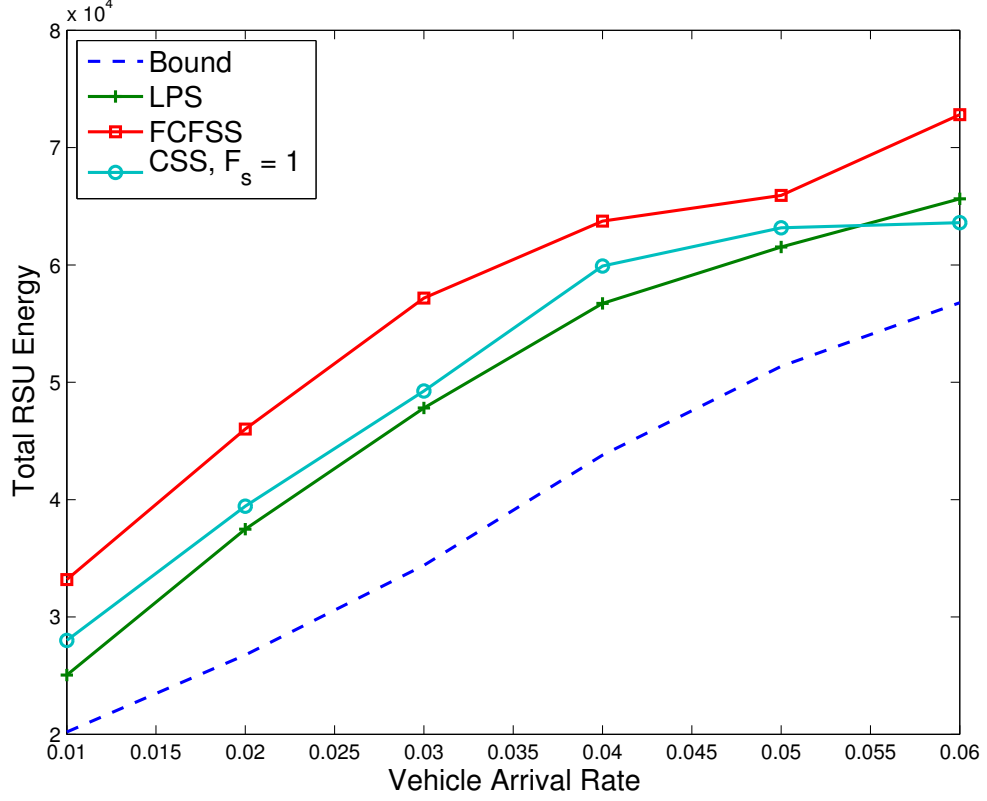


Figure 4.14: Total RSU Energy (J) vs. Vehicle Arrival Rate (vehicles/sec).  $\rho_I = 120$ .

in the last two figures. The same trend as in Figure 4.9 occurs, but the relative performance of the three online algorithms has slightly changed due to an increase in sleep value.

Figures 4.15, 4.16 and 4.17 show the results of experiment four in which parameter  $\rho_I$  is decreased to 70 J/sec. These sets of figures also can be compared with their counterparts in experiment one (Figures 4.7, 4.8 and 4.9) or in experiment three (Figures 4.12, 4.13 and 4.14). The comparisons lead to conclusions similar to the ones mentioned previously. Since the RSU energy consumption in idle mode has decreased, the total energy use has also decreased, and the gap between the online

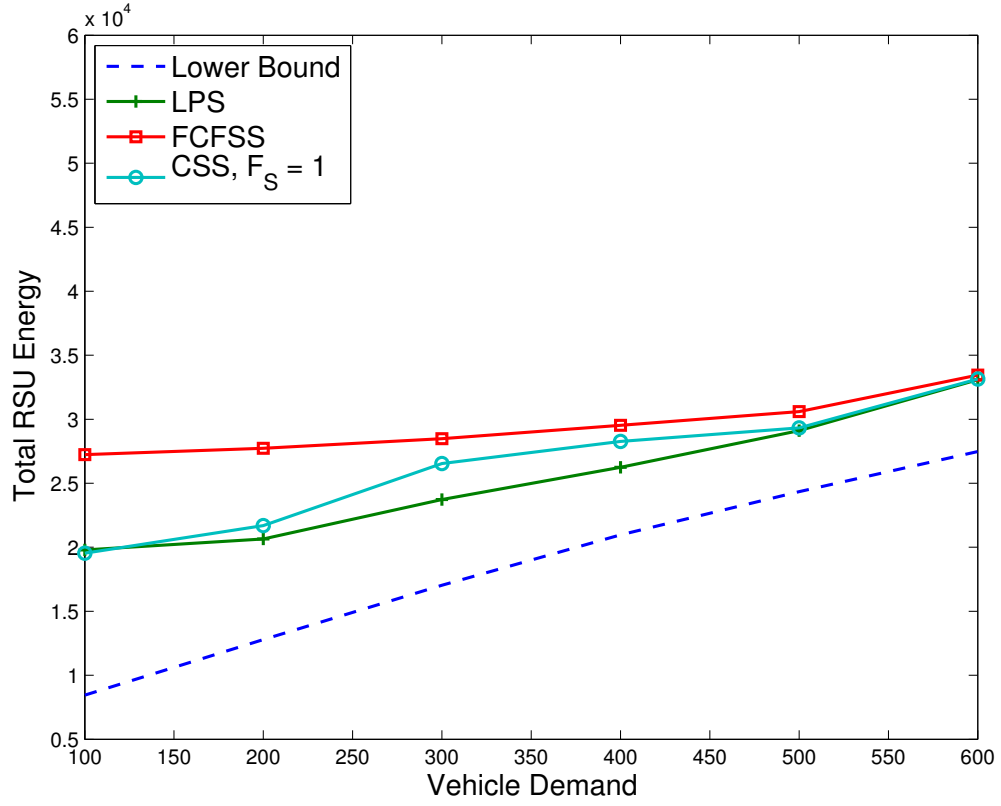


Figure 4.15: Total RSU Energy (J) vs. Vehicular Demand (Mbits) .  $\rho_I = 70$ .

algorithms has also become smaller.

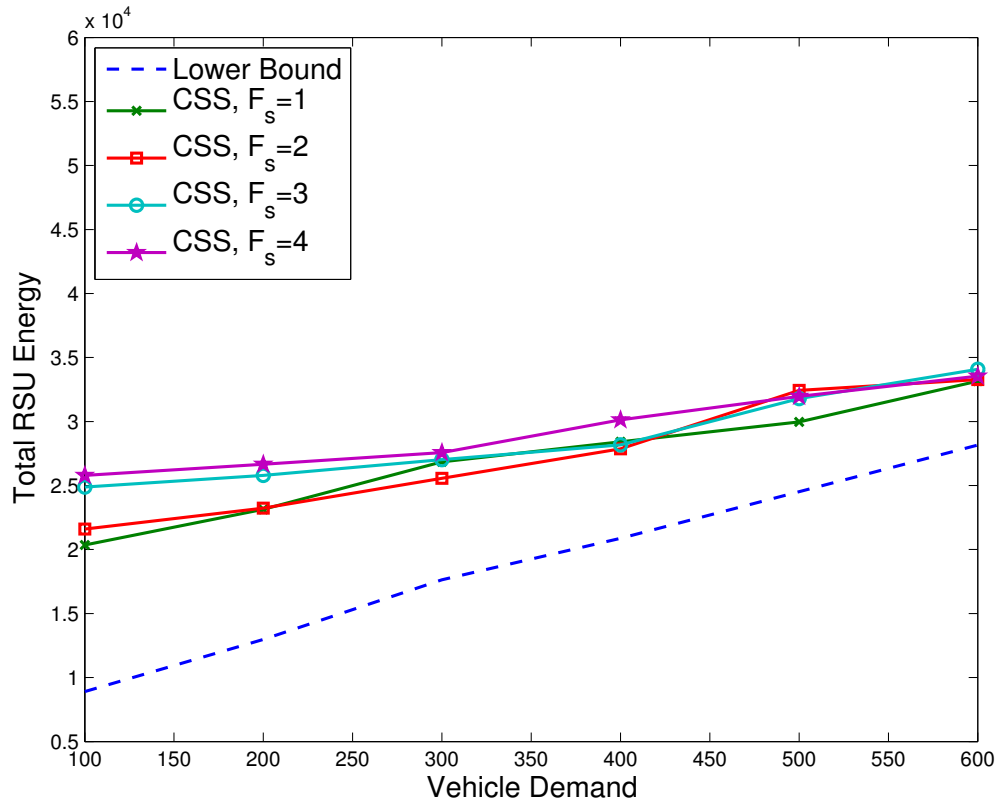


Figure 4.16: Total RSU Energy (J) vs. Vehicular Demand (Mbits) .  $\rho_I = 70$ .

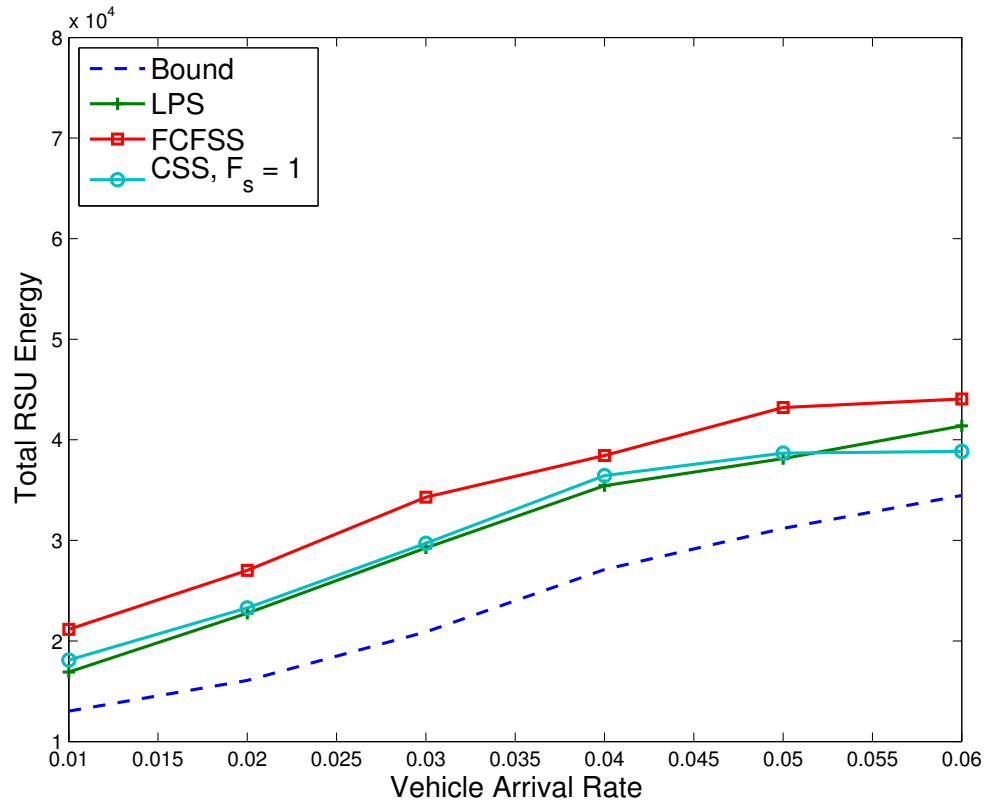


Figure 4.17: Total RSU Energy (J) vs. Vehicle Arrival Rate (vehicles/sec).  $\rho_I = 70$ .



# Chapter 5

## Conclusions

In this thesis the problem of energy efficient vehicular roadside unit scheduling in delay tolerant networks, is studied. OFF-to-ON roadside unit sleep transitions have been incorporated into minimum energy cost downlink transmission scheduling.

The problem is considered with two different assumptions about the network. In the first case it is assumed that power control is used for downlink communication, so that each time slot can carry a fixed number of bits, regardless of vehicle location within the RSU coverage area. In the second case no power control is considered and the bit rate of downlink communication varies according to channel quality. In both of these cases, integer programs were derived that give lower bounds on energy use that can be achieved for a given input sample function. Online scheduling algorithms were also proposed. For the constant bit rate model, two online algorithms were proposed, referred to as the Flow Graph Sleep Scheduler (FGS) and the Greedy Scheduler. In the FGS algorithm the RSU makes optimum OFF period scheduling decisions by computing an estimate of the energy needed to fulfill known vehicle communications requirements without initiating an OFF period, and by making a

similar computation based on initiating an OFF period with a duration of  $T_s$  time slots. On the other hand, the Greedy algorithm, which is much simpler, first considers scheduling vehicular time slots in order to minimize the energy consumed in packet transmission, and then schedules RSU OFF periods based on the resulting schedule. Simulation results show that this algorithm performs well only under light traffic load and generally the FGS gives better energy savings. The FGS algorithm was tested with different values for parameter  $T_s$  and the results showed that it is beneficial to have OFF periods with different durations under different traffic conditions. This is a subject for future work, i.e., to design schedulers that adaptively change this parameter based on traffic conditions.

For the variable bit rate model, three online algorithms were proposed. In the first, LPS, the relaxed version of the ILP used for obtaining the lower bound, was solved, and the fractional solutions were then converted to appropriate binary values by using a rounding approach. These computations are performed when there were newly arrived vehicles. In the second algorithm, FCFSS, first vehicular frames were scheduled in a FCFS manner, then decisions for the OFF periods were taken based on the resulting schedule. The third algorithm, CSS, is similar to FGS from the constant bit rate case. An estimate of the energy needed to fulfill known vehicle communication requirements is computed assuming two scenarios, i.e., without initiating an OFF period, or initiating an OFF period with a duration of  $F_s$  frames. Results from a variety of experiments show that the proposed scheduling algorithms perform well when compared to the energy lower bounds. This is especially true for the LPS algorithm, while FCFSS, which has less computational complexity, showed a weaker performance. The CSS algorithm, as in FGS discussed above, could be implemented

adaptively using current traffic conditions which would make it more robust to changes in loading. The work which is studied in this thesis can also be extended to the case of a vehicular network with multiple RSUs, so that higher power savings can be achieved by co-operation between neighboring RSUs. Traffic prediction schemes could also be incorporated into the algorithms so the network will be able to support higher traffic loading with reduced energy use.

# Bibliography

- Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ.
- Berend, D., Segal, M., and Shpungin, H. (2008). Power Efficient Resilience and Lifetime in Wireless Ad-hoc Networks. In *Proceedings of the 1st ACM international workshop on Foundations of wireless ad hoc and sensor networking and computing, FOWANC '08*, pages 17–24, New York, NY, USA. ACM.
- Bilstrup, K., Uhlemann, E., Strom, E., and Bilstrup, U. (2008). Evaluation of the IEEE 802.11p MAC Method for Vehicle-to-Vehicle Communication. In *Vehicular Technology Conference, 2008. VTC 2008-Fall. IEEE 68th*, pages 1–5.
- Chang, J.-H. and Tassiulas, L. (2000). Energy Conserving Routing in Wireless Ad-hoc Networks. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 1, pages 22–31 vol.1.
- Clementi, A. E., Huiban, G., Penna, P., Rossi, G., and Verhoeven, Y. C. (2002). Some Recent Theoretical Advances and Open Questions on Energy Consumption in Ad-Hoc Wireless Networks.

- Hadaller, D., Keshav, S., Brecht, T., and Agarwal, S. (2007). Vehicular Opportunistic Communication under the Microscope. In *Proceedings of the 5th international conference on Mobile systems, applications and services*, MobiSys '07, pages 206–219, New York, NY, USA. ACM.
- Hammad, A., Badawy, G., Todd, T., Sayegh, A., and Zhao, D. (2010). Traffic Scheduling for Energy Sustainable Vehicular Infrastructure. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*, pages 1–6.
- Jerbi, M., Senouci, S. M., Meraihi, R., and Ghamri-Doudane, Y. (2007). An Improved Vehicular Ad hoc Routing Protocol for City Environments. In *Communications, 2007. ICC '07. IEEE International Conference on*, pages 3972–3979.
- Jiang, S. and Vaidya, N. H. (1999). Scheduling Data Broadcast to "Impatient" Users. In *Proceedings of ACM MobiDE'99*, pages 52–59.
- Johnson, D. B. and Maltz, D. A. (1996). Dynamic Source Routing in Ad hoc Wireless Networks. In *Mobile Computing*, pages 153–181. Kluwer Academic Publishers.
- Karagiannis, G., Altintas, O., Ekici, E., Heijenk, G., Jarupan, B., Lin, K., and Weil, T. (2011). Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions. *Communications Surveys Tutorials, IEEE*, **13**(4), 584–616.
- Karp, B. and Kung, H. T. (2000). GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, MobiCom '00, pages 243–254, New York, NY, USA. ACM.

- Khabazian, M. and Ali, M. K. M. (2008). A Performance Modeling of Connectivity in Vehicular Ad Hoc Networks. *Vehicular Technology, IEEE Transactions on*, **57**(4), 2440–2450.
- Kolios, P., Friderikos, V., and Papadaki, K. (2010). Switching off Low Utilization Base Stations via Store Carry and Forward Relaying. In *Personal, Indoor and Mobile Radio Communications Workshops (PIMRC Workshops), 2010 IEEE 21st International Symposium on*, pages 312–316.
- Kumar, V. and Chand, N. (2010). Data Scheduling in VANETs: a Review. *International Journal of Computer Science and Communication*, **1**(2), 399–403.
- Leontiadis, I. and Mascolo, C. (2007). GeOpps: Geographical Opportunistic Routing for Vehicular Networks. In *WOWMOM*, pages 1–6. IEEE.
- Lequerica, I., Ruiz, P., and Cabrera, V. (2010). Improvement of Vehicular Communications by Using 3G Capabilities to Disseminate Control Information. *Network, IEEE*, **24**(1), 32–38.
- Li, L. and Halpern, J. (2001). Minimum-Energy Mobile Wireless Networks Revisited. In *Communications, 2001. ICC 2001. IEEE International Conference on*, volume 1, pages 278–283 vol.1.
- Liu, X., Fang, Z., and Shi, L. (2007). Securing Vehicular Ad Hoc Networks. In *Pervasive Computing and Applications, 2007. ICPCA 2007. 2nd International Conference on*, pages 424–429.
- Lochert, C., Hartenstein, H., Tian, J., Fussler, H., Hermann, D., and Mauve, M.

- (2003). A Routing Strategy for Vehicular Ad-hoc Networks in City Environments. In *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*, pages 156–161.
- Louhi, J. (2007). Energy Efficiency of Modern Cellular Base Stations. In *Telecommunications Energy Conference, 2007. INTELEC 2007. 29th International*, pages 475–476.
- Mi, J., Liu, F., Xu, S., and Li, Q. (2008). A Novel Queue Priority Algorithm for Real-Time Message in VANETs. In *Intelligent Computation Technology and Automation (ICICTA), 2008 International Conference on*, volume 2, pages 919–923.
- Moustafa, H. and Bourdon, G. (2008). Vehicular Networks Deployment View: Applications, Deployment Architectures and Security Means. *Ubiquitous Computing and Communication Journal, special issue on Ubiquitous Roads*.
- Naumov, V. and Gross, T. (2007). Connectivity-Aware Routing (CAR) in Vehicular Ad-hoc Networks. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 1919–1927.
- Niu, Z., Wu, Y., Gong, J., and Yang, Z. (2010). Cell zooming for Cost-Efficient Green Cellular Networks. *Communications Magazine, IEEE*, **48**(11), 74–79.
- Ott, J. and Kutscher, D. (2004). Drive-thru Internet: IEEE 802.11b for “utomobile” Users. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, pages –373.
- Park, V. D. and Corson, M. S. (1997). A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks. In *Proceedings of the INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications*

- Societies. Driving the Information Revolution*, INFOCOM '97, pages 1405–, Washington, DC, USA. IEEE Computer Society.
- Paul, B., Ibrahim, M., and Bikas, M. A. N. (2011). Article: VANET Routing Protocols: Pros and Cons. *International Journal of Computer Applications*, **20**(3), 28–34. Published by Foundation of Computer Science.
- Pei, G., Gerla, M., and Chen, T.-W. (2000). Fisheye State Routing: A Routing Scheme for Ad Hoc Wireless Networks. In *Communications, 2000. ICC 2000. 2000 IEEE International Conference on*, volume 1, pages 70–74 vol.1.
- Perez-Costa, X. and Camps-Mur, D. (2010). IEEE 802.11E QoS and Power Saving Features Overview and Analysis of Combined Performance [Accepted from Open Call]. *Wireless Communications, IEEE*, **17**(4), 88–96.
- Perkins, C. and Royer, E. (1999). Ad-hoc on-Demand Distance Vector Routing. In *Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA '99. Second IEEE Workshop on*, pages 90–100.
- Raya, M. and Hubaux, J.-P. (2005). The Security of Vehicular Ad Hoc Networks. In *Proceedings of the 3rd ACM workshop on Security of Ad Hoc and Sensor Networks, SASN '05*, pages 11–21, New York, NY, USA. ACM.
- Rohl, C., Woesner, H., and Wolisz, A. (1997). A Short Look on Power Saving Mechanisms in the Wireless LAN Standard Draft IEEE 802.11. In *IEEE 802.11. In The 6th WINLAB Workshop on Third Generation Wireless Systems, New-Brunswick, NJ*, pages 183–188.



- Sayadi, M. J. and Fathy, M. (2010). A New Approach in Packet Scheduling in the VANET.
- Seet, B.-C., Liu, G., sung Lee, B., heng Foh, C., and kee Lee, K. (2004). A-STAR: A Mobile Ad hoc Routing Strategy for Metropolis Vehicular Communications. In *Proc. NETWORKING 2004*, pages 989–999.
- Shpungin, H. and Li, Z. (2012). Throughput and Energy Efficiency in Wireless Ad Hoc Networks With Gaussian Channels. *Networking, IEEE/ACM Transactions on*, **20**(1), 15–28.
- Sommer, C., Eckhoff, D., German, R., and Dressler, F. (2011). A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments. In *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on*, pages 84–90.
- Sou, S.-I. (2010). A Power-Saving Model for Roadside Unit Deployment in Vehicular Networks. *Communications Letters, IEEE*, **14**(7), 623–625.
- Vadgama, S. and Hunukumbure, M. (2011). Trends in Green Wireless Access Networks. In *Communications Workshops (ICC), 2011 IEEE International Conference on*, pages 1–5.
- Wang, S. (2005). The Effects of Wireless Transmission Range on Path Lifetime in Vehicle-Formed Mobile Ad hoc Networks on Highways. In *Communications, 2005. ICC 2005. 2005 IEEE International Conference on*, volume 5, pages 3177–3181 Vol. 5.

- Wu, Y. and Cao, G. (2001). Stretch-Optimal Scheduling for on-Demand Data Broadcasts. In *Computer Communications and Networks, 2001. Proceedings. Tenth International Conference on*, pages 500–504.
- Zhao, J. and Cao, G. (2006). VADD: Vehicle-Assisted Data Delivery in Vehicular Ad hoc Networks. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–12.
- Zou, F., Zhong, J., Wu, W., Du, D.-Z., and Lee, J. (2011). Energy-Efficient Roadside Unit Scheduling for Maintaining Connectivity in Vehicle Ad-hoc Network. In *Proceedings of the 5th International Conference on Ubiquitous Information Management and Communication, ICUIMC '11*, pages 64:1–64:8, New York, NY, USA. ACM.