AN APPLICATION AWARE UTILITY BASED LIFETIME QUANTIFICATION FRAMEWORK FOR WIRELESS SENSOR NETWORKS

by

Bahri Atay Özgövde

B.S, in EE, Boğaziçi University, 1995

M.S, in ESC, Boğaziçi University, 1998

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

Graduate Program in Computer Engineering

Boğaziçi University

2009

ACKNOWLEDGEMENTS

Pursuiting a PhD degree has been a long and a crucial era of my life. I feel that I owe many people who made the whole process an unforgettable sequence of memories.

The most influential character of this episode has no doubt been my advisor, dear Cem Ersoy. I have learned and acquired a lot from him, maybe much more than I have consciously been aware of. During our time together, the material he presented to me covered much more than the technical subject matter we focused on, actually it covered the whole spectrum of life. I feel very lucky that I had the chance to experience this extraordinary and in a way magical kind of relationship for almost every day. What I have experienced have shaped me and I deeply feel that it will continue to do so for long long years.

During my time in the Computer Engineering department I had the chance to work with many genuine friends: İlker Demirkol, Rabun Koşar, Itır Karaç, Oya Aran, Onur Dikmen, Berk Gökberk, Albert Ali Salah, Aydın Ulaş, Hande Özgür, Koray Balcı, Ertan Onur, Kaan Bür, İlker Oyman, Burak Turhan, Mehmet Gönen, Ali Haydar Özer, Yunus Durmuş, Yunus Dönmez, Sinan Işık, Pınar Santemiz, Neşe Alyüz, İsmail Arı, Gül Çalıklı, Abuzar Yakaryılmaz and many others. They have always inspired me with their ways being.

I also have to mention the guidance and friendship of the NETLAB professors Tuna Tuğcu, M. Ufuk Çağlayan, Ayşe Bener, Lale Akarun, Haluk Bingöl, Fatih Alagöz. Their attitude have always made my life easier.

Many people on the way have genuinely been generous to me: Yücel Küçük, Hülya Törbik, Aygül Belek, İzzet abi (Yılmaz), Philippe Mazel, Birol Yamaç, Refik Aydın, Şeker Bilici, Muhteber Polater. I deeply appreciate their efforts.

I would like to thank to the jury members for the valuable comments and feedback they have provided.

I would like to send my sincere thanks to my father and mother who laid the foundations of all this happening.

Finally comes my dear family, Pınar, Elif and the *new member* without whom all this efforts would be meaningless.

This work has been supported by the State Planning Organization of Turkey under the grant numbers 03K120250 and 07K120610 (TAM Project), and by TÜBİTAK under the grant numbers 106E082 and 108E207. This research is also supported by Boğaziçi University BAP department under the grant numbers 04A105 and 09A101P.

ABSTRACT

AN APPLICATION AWARE UTILITY BASED LIFETIME QUANTIFICATION FRAMEWORK FOR WIRELESS SENSOR NETWORKS

Network lifetime is a novel performance metric that is used to evaluate networks comprised of nodes with irreplenishable energy sources. Wireless sensor networks (WSNs) are the primary examples of such networks. The network lifetime is a crucial performance metric since it indicates the amount of functionality obtained in return to the total investment including the sensor hardware, the deployment, and the administrative work. Unlike the legacy network performance metrics such as delay, throughput or jitter, the evaluation of *network lifetime* is not straightforward because of the application dependence involved. Application dependence is a recurring theme in the WSN domain that inhibits finding generalized solutions to the research problems, where the network lifetime quantification is no exception.

In this work, we devise a framework for incorporating the application dependence into the lifetime measurement process of the wireless sensor networks, thereafter via extensive experiments, demonstrate the significance of the lifetime metric itself in the quantification process for a variety of application scenarios including both scalar and video based wireless sensor networks. We show that the lifetime metrics that ignore application dependence fail in solving the network lifetime quantification problem in WSNs. Our proposed framework, weighted cumulative operational time (WCOT), combines two distinct mechanisms for realistic and application context aware network lifetime evaluation. Firstly, by introducing the *utility function* it enables the users of the network to inscribe their own application level requirements in a formal setting.

This clarifies the inherent subjectivity due to the application dependence involved in the WSN network lifetime quantification problem by transforming it into a form that renders further computation possible. The utility function denotes the total cumulative utility (usefulness) offered by the collaboration of the sensor nodes. Secondly, instead of offering a single cut-off threshold value for defining the point after which the network is assumed to be nonfunctional, WCOT framework makes use of the gradual change in the utility of the network and record how the network evolves over time in terms of functionality offered by keeping the weighted sum of the operational time. Unlike lifetime metrics that focus on a single threshold value, WCOT is able to differentiate network performances that differ in how the network evolves till the utility drops to zero.

ÖZET

TELSİZ ALGILAYICI AĞLAR İÇİN UYGULAMA BAĞIMLI FAYDA TABANLI AĞ ÖMRÜ BELİRLEME ÇATISI

Ağ ömrü, Telsiz Algılayıcı Ağlar (TAA) gibi kısıtlı ve yenilenemez enerji kaynağına sahip düğümlerden oluşan ağlar için kullanılan yeni bir başarım ölçütüdür. Kullanıcılarının bir TAA'dan edindikleri toplam faydays işaret etmesi bakımından ağ ömrü dikkatle değerlendirilmesi gereken önemli bir göstergedir. Bu bağlamda ağ ömrü bir TAA kullanıcısının yaptığı toplam yatırımın ne derece geri döndüğünü de ifade etmektedir. Ancak uygulamaya bağımlı olduğu için TAA'ların ağ ömrünü belirlemek, gecikme gibi geleneksel ağ başarım ölçütleriyle karşılaştırdığımız zaman daha karmaşık olmaktadır. Uygulama bağımlılığı TAA alanında sıklıkla karşılaşılan ve araştırma problemlerine genelleştirilmiş çözümler bulmayı engelleyen bir etmendir ve ağ ömrü ölçümü probleminde de durum benzerdir.

Bu çalışmada TAA'larda ağ ömrünün belirlenmesine uygulama bağımlılığı da katabilmek için bir çatı geliştirdik. Ağ ömrü ölçütünün ağ başarımını değerlendirmedeki önemi ve etkisini göstermek için ortaya koyduğumuz niceleme çatısını kullanarak video taşıyan ağ tiplerini de içerecek değişik uygulama senaryoları içeren deneyler gerçekleştirdik. Çalışmamızda uygulama seviyesindeki tanımlamaları dikkate almayan ağ ömrü ölçütlerinin TAA ağ ömrü nicelemesinde yetersiz kaldığını gösterdik.

Önerdiğimiz metod, ATİZ (Ağırlıklı Toplam İşlevsel Zaman), iki farklı mekanizmayı içiçe kullanarak gerçekçi ve uygulamaya bağlı ağ ömrü nicelemesini gerçekleştiremektedir. İlk olarak, ATİZ bir fayda fonksiyonu aracılığıyla ağın kullanıcılarının kendi uygulama seviyesindeki gereksinimlerini sistematik olarak ifade etmelerine olanak vermektedir. Böylece ATİZ ağ ömrü nicelemesi probleminde içsel olarak bulunan, uygulama

bağımlılığından kaynaklanan öznelliği, üzerinde matematiksel işlem yapılabilir hale getirerek bertaraf eder. Bahsi geçen fayda fonksiyonu ağdaki düğümlerin ortaklaşa ürettiği toplam yararı göstermektedir. İkinci olarak metodumuz, ağın yarattığı yararın kabul edilemez sınırların içinde olduğunu gösteren tek bir eşik tanımlayarak niceleme yapmak yerine değişen fayda seviyesini kullanıp ağ tarafından sunulan toplam işlevi zamanda ağırlıklı biçimde kaydetmek yoluyla daha yüksek çözünürlüklü ağ ömrü ölçümü yapmaktadır. Böylece sadece tek bir eşik değeri tanımlayan ağ ömrü ölçütlerinden farklı olarak ATİZ, ağın yararının bittiği noktaya kadar farklı yarar seviyelerinden geçerek gelen TAA'ların başarımlarını farklı rakamsal değerler ile ifade edebilmektedir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

e_{best}	Energy-wise best entry in the routing list
e_i	Energy cost of the i^{th} entry in the routing list
E_{rx}	Reception energy required per bit
E_{tx}	Transmission energy required per bit
$P(i)$	Probability that i^{th} entry is selected as next node
P_{max}	Maximum transmit power
P_{min}	Minimum transmit power
r_{comm}	Communication radius
$r_{sensing}$	Sensing radius
S_f	Final network state
S_i	Initial network state
U_f	Final network utility value
U_i	Initial network utility value
$U_{i,n}$	Utility of a WSN with n nodes deployed and i nodes remaining
α	Alive node ratio
δ	Sensing coverage
γ	Path loss index
ϕ_{base}	SMAC base duty cycle
Ψ	SMAC listen offset
σ	PEDER routing list size control parameter
Θ	Fixed prior selection probability assigned to energy-wise best entry by PEDER-III
ANR	Alive node ratio
CODA	Congestion detection and avoidance
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
DBF	Distributed Bellman-Ford
ECU	Energy cost update

ESA	Effective surveillance area
ESRT	Event to sink reliable transport
FB	Foreground background
FCFS	First come first served
FND	First node death
FoV	Field of view
GPSR	Greedy perimeter stateless routing
IEDR	Instantaneous event delivery ratio
ITR	Information transfer rate
LAS	Least attained service
LEACH	Low energy adaptive clustering hierarchy
LMM	Lexicographic max min
LND	Last node death
LP	Linear programming
MAC	Medium access control
MCF	Minimum cost forwarding
MER	Minimum energy routing
MHR	Minimum hop routing
MPEG	Moving pictures experts group
MTE	Minimum transmitted energy
MWSN	Multimedia wireless sensor network
PEDER	Probabilistic energy driven routing
QoS	Quality of service
RAM	Random access memory
RD	Reporting delay
RR	Round robin
SMAC	Sensor Medium Access Control
SPRT	Shortest remaining processing time
SQCIF	Sub quarter common interchange format
TFP	Trespassers' favorite path
TTLE	Traffic triggered lifetime extension

VSN	Video sensor network
VSSN	Video surveillance sensor network
WCOT	Weighted cumulative operational time
WMN	Wireless mesh network
WSN	Wireless sensor network

1. INTRODUCTION

Wireless sensor networks (WSNs) constitute a novel technology that allows distributed intelligence to penetrate into a variety of remote sensing and monitoring applications. This relatively new wireless networking technology is expected to go beyond the basic sensing the environment type of applications and actually become an enabling technology for the ubiquitous computing and "the internet of things" [1]. Nowadays, we witness the evolution of the sensor networks from conceptual test-bed level implementations to a more mature and usable technology with commercial products beginning to proliferate some examples being the eKo environmental and agricultural monitoring systems [2], PhyNet generic industrial and home control devices [3], Sensicast systems for structural health monitoring [4].

There are many factors that make the WSNs unique in the wireless communications domain. The strict energy constraint faced by the sensor nodes, which is also related to the subject matter of this thesis, gives strict emphasis on energy efficiency. Apart from the limited energy, the sensor nodes also suffer from the hardware resource constraints including the computational and the communication wise limitations. Small form factor and cost restriction required for most applications dictate this resource scarcity. Along with the constraints on the node capabilities, a highly specific traffic pattern is observed in a WSN which is inherently unbalanced due to the convergecast packet flows towards the sink node(s). Such many-to-one traffic type is hard to deal with as it creates bottlenecks around the sink node. More importantly, the WSNs are categorically different from the known types of networks as they are built from scratch to a geographic locality to perform an application specific task in a distributed manner. In that sense, they radically differ from the legacy wired or wireless networks which are built to give service to the application processes running on individual hosts where the shared infrastructure is optimized and standardized according to a specific range of applications and traffic flow character. In contrast, we can envision a WSN as a single logical entity in which the hardware and the software modules responsible for the communications, sensing and the computation collaborate and form a custom-made infrastructure streamlined for a specific application. In this context, WSN designs are bound to be application specific. This fact combined with the vast number and variety of existing and possible WSN applications make the term *application dependence* a recurring theme in the WSN research domain. Bringing all these together, the WSNs put forward a wealth of academical research problems as well as practical and technological challenges that need to be solved before they emerge as a robust and hassle free technology. Among these, the specific focus of this thesis is on how the inherent application dependence present in the WSNs can be resolved so that the network lifetime can be correctly and accurately quantified.

1.1. Utility and Lifetime of a Wireless Sensor Network

In the forms of networking that existed prior to the WSNs, it has always been a valid assumption that the network nodes sustain their physical operations continuously. Failure to do so is assumed to be intermittent in character such as the outage due to the excess traffic load, changing link quality, and the security attacks. The permanent loss of a network element is not envisioned in these legacy systems. In the extreme case of a hardware failure, the nodes are simply replaced and the transient loss of functionality is not generally taken into account in the formal problem formulation but rather considered as a technical maintenance issue. With the onset of the WSNs, however, we are faced with the fact that each node in the network (excluding the sink) is bound to eventually fail due to the fixed amount of initial energy deposited on the nodes. This creates a radically new perception of the service provided by the network in which the functionality offered gradually deteriorates. In this respect, an important research problem related to the WSNs is to both identify the functional duration of the network, i.e. the network lifetime, and to extend it. To be able to elongate the lifetime, the original design should incorporate methods to postpone the node failure times as far as possible. Energy aware algorithms need to be employed at all layers of the communications stack. Even more importantly, the communication stack layers are allowed to aggressively cooperate to achieve energy efficiency, which leads to the cross-layer efforts. All this is necessary to elongate the functional lifetime of the sensor nodes. The rationale behind this effort is to get more out of a WSN deployment.

Getting service from a WSN comprises many cost items including the hardware design and manufacturing costs if the COTS are not used, the development of the communications and sensing software employed on the nodes, the network design including the initial deployment phase and the R&D phase for the performance optimization, the operational and the administrative efforts and the node recollection phase due to the environmental regulations after the WSN operation is terminated. From the cost point of view, elongating the network lifetime amounts to decreasing the cost per bit of data obtained from the network.

Efforts to extend the network lifetime of a WSN actually extends the lifetime of the individual sensors. However the individual sensor lifetimes cannot directly be used as an indicator for the network lifetime. As a WSN operates, a temporal death sequence of the nodes is obtained. As will be thoroughly discussed throughout the thesis, it is not straightforward to map the death sequence vector of the nodes into a meaningful lifetime figure that actually reflects the amount of operational period the network goes through. This is basically because a WSN is a custom built solution to a monitoring problem, therefore its functional lifetime is unavoidably application dependent. As a remedy, we put forward the concept of the *network utility* both to measure the lifetime and also to guide efforts that perform WSN lifetime optimization. Our point is that the administrative user of the network who guides the custom-made design of the WSN in question has the full information on the application requirements and is able to describe the utility of the network at certain stages. With the help of the utility function the user of the WSN maps the network states into objective utility values. The term network state signifies the relevant utility indicators which are to be chosen by the administrative user of the network. Table 1.1 shows some utility indicators for a range of WSN applications.

In the literature, we observe that the cruciality of the network lifetime problem is acknowledged as many studies are employing the network lifetime as a performance metric for evaluation and optimization purposes. However, in practice the focus is given on proposing sophisticated schemes to increase the energy efficiency, whereas only rudimentary lifetime metrics are employed to evaluate the outcome of this effort

Table 1.1. Some WSN application categories and their possible utility indicators.

Application	Characteristics	Possible Utility Indicators
Precision Agriculture	Spatio-temporal redundancy of data. Low volume, delay tolerant periodic traffic.	Sensing Coverage
Health Monitoring	Highly critical data. Periodic + event triggered traffic.	Latency, Data Loss
Surveillance (scalar)	Critical data. Event triggered traffic. Low to medium volume traffic.	Latency, Event Loss, Coverage
Surveillance (video)	Critical data. Event triggered traffic. Temporally redundant data. Very high volume traffic.	Latency, Event Loss, Jitter, Coverage
Structural Monitoring	High frequency data acquisition. Periodic traffic. Temporally redundant data.	Data Loss, Coverage, Data synchronization
Environmental and Habitat Monitoring	Periodic + event triggered traffic. Delay tolerant traffic.	Event Loss, Coverage

which compromises the correctness of the results. Generally speaking, two problematic approaches are observed for the WSN network lifetime quantification. Firstly, the lifetime metric that is employed at a specific setting is not justified with regards to the application scenario at hand. Oversimplistic lifetime metrics such as the time till a certain percentage of the initially deployed nodes die are proposed and used with the intention of a generic lifetime metric. However, the actual lifetime of the network is dependent on the utility of the remaining nodes which cannot solely be captured by the sheer number of alive sensor nodes. The initial node redundancy, the deployment scheme, the application requirements in terms of the coverage or the average delay, the positional distribution of the sensor deaths among the network actually determine the point in time when the network actually becomes useless. Secondly, when evaluating the lifetime of a WSN, a specific milestone in the network evolution is assumed to trigger the death of the network, which is not realistic for many applications where the utility of the network gradually changes. For instance, the metric the time till the 30% loss of the sensing coverage implicitly assumes that the network is fully functional till it loses 30% of its coverage and thereafter suddenly become useless. Such a stepwise decrease in the network functionality may or may not be the case depending on the application and the network parameters. If the gradual change in the network utility as each sensor dies could be captured than it would be possible to differentiate the WSN lifetime performances in a much higher resolution. This is exactly what we aim for through the use of the utility function in our lifetime quantification framework, weighted cumulative operational time (WCOT). As the network operates, WCOT assess the utility of the network and use it to differentiate the usefulness of the operational periods the network goes through.

1.2. A Network Lifetime Case Study: Video Surveillance Sensor Networks

According to a broad categorization in terms of the sensing modality, we can classify the sensor networks as those that operate on the scalar data and the ones that convey multimedia data. The WSNs in which nodes measure scalar values such as the temperature or the humidity levels generally create less traffic when compared to the video sensor nodes. With less traffic volume, a WSN design that cares less about the realtime QoS issues can be afforded. Certain applications, however, cannot operate merely on the scalar data but require the existence of denser information content from the scene. Those are generally the type of applications in which not an attribute of the physical environment but the existence and/or the properties of the objects in the monitored area are to be identified. As an example, multimedia wireless sensor networks (MWSNs) provide sound and/or image information to the sink node so that surveillance type of applications can be accomplished. Despite its strengths in rich information content, carrying video streams over a sensor network is a challenging task. Even at low resolution, low frame rate video streams can easily overwhelm a sensor network given the stringent resource constraints. In this work, first, we identify the feasible operational range in terms of the camera frame, buffer size, communication duty cycle employed by the video sensor networks (VSNs), then enhance their performance using event based frame queue management algorithms. We observed that event-aware handling of the frames result in enhanced delay and event delivery characteristics.

It is more complex to evaluate the lifetime of a VSN due to the complicated relation between the aggregate packets received at the sink node and the information content actually conveyed. The received data is logically comparted into frames and in this setting not the absolute number but the distribution of the dropped packets within the frames actually determines the quality of the visual information. Moreover, in the surveillance networks generally the logical unit of messaging is an event as opposed to an image frame. Therefore, the way the event is reported becomes more important than the faith of the individual frames. Considering this, we employ an event based approach for a better lifetime quantification where the utility of the network is determined simultaneously by the sensing coverage and the instantaneous event reporting capability.

1.3. Key Problems Addressed and Contributions

In this thesis, we propose a lifetime quantification framework called WCOT for the WSNs. We define and elaborate on the concept of the network utility, which forms the fundamentals of our approach to the lifetime quantification. We show that the utility based approach lets users of the network to formally describe their application requirements which in turn is used in the lifetime calculation. This is in contrast to most of the existing works on the topic that omit the application dependence and treat the lifetime quantification as if it could be performed based on some generally applicable objective criteria. With extensive packet level simulation experiments carried out on realistic settings including MAC level communication sleep schedule, we show that lifetime metrics that disregard application dependence fail in correctly quantifying the network lifetime. This important result suggests that the energy efficient communication mechanisms and the network designs already proposed for the WSNs should be re-evaluated based on sound lifetime metrics, where WCOT may serve as a candidate.

In Chapter 2, we outline the general mechanical procedure that the WCOT applies when it is fed with the appropriate utility function and later study the lifetime behavior of both the scalar and the multimedia type sensor networks for different application scenarios. There are numerous WSN applications that are either commercially available or in prototyping/research phase. Given the vast range of applications, it is not possible to exhaustively study the performance of WCOT for all the possible cases. Instead, we focus on two generic family of WSN applications, namely the applications that rely on scalar data and the ones based on the visual data. For the scalar applications, the effect of the routing algorithm on the network lifetime is examined for the environmental monitoring scenario. To serve as the rival algorithm to the minimum energy routing (MER), we propose a family of probabilistic routing algorithms called probabilistic energy driven routing (PEDER) and show explicitly the dependence of the results on the lifetime metric.

We draw special emphasis on the sensor networks that carry visual data. VSNs are the novel branch of the WSNs. Visual information about the scenery contains valuable information and makes many application categories feasible that otherwise cannot be implemented using the scalar-only networks. However, this increased information content comes with a cost in terms of the increased traffic load as studied in Chapter 4.

To further understand and enhance the performance of the VSNs, we focus on a specific application, namely the video surveillance application. We implement event based approaches for queue management. By doing so, we achieve application level fairness which translates into increased performance for Video Surveillance Sensor Networks (VSSNs) in which the logical unit of messaging is an event. Event based queueing methods are shown to cause more visual information to be carried per event and in a more timely manner. In Chapter 6, the lifetime behavior of the VSSNs is explored where the effect of the target population size and the mobility on the network lifetime is tested. During the tests we observe that increased traffic load positively affected the lifetime of the nodes in the network. The dynamics and the underlying mechanisms of this counter intuitive phenomena that we call *Traffic Triggered Lifetime Extension* is studied in detail. We show that coverage or event reporting capabilities of the network by themselves are not sufficient to express the network utility. For a realistic lifetime quantification, we come up with a utility indicator which is a function of two independent variables, namely the sensing coverage, and the instantaneous event delivery ratio

(IEDR). In our tests, lifetime metrics based on IEDR and the sensing coverage result in lifetime figures which correctly identify the trends in terms of the event delivery performed by the network.

2. WCOT: A UTILITY BASED REALISTIC NETWORK LIFETIME QUANTIFICATION FRAMEWORK

2.1. Rationale

Wireless Sensor Network (WSN) research has one ultimate performance objective: prolonging the *network lifetime*. Being the decisive performance evaluation criteria, how the network lifetime is defined deserves extra attention. Basic functionality of a WSN is to *monitor* a region of interest. Therefore, a natural definition of the lifetime of a WSN is the time between the deployed sensor nodes start collecting data and the instance where the *monitoring quality* drops below an *acceptable threshold level.* Typical timeline of a WSN is depicted in Figure 2.1. The terms monitoring quality and acceptable threshold level have different interpretations for different application scenarios. For instance, for a WSN application that monitors the humidity of an agricultural area, the acceptable threshold might correspond to regular data being collected from at least 85 per cent of the crops, whereas for a military video-surveillance application, the maximum latency of two seconds for video packets may set the threshold for the monitoring quality. Given the wide range of WSN applications, it is not possible to give definitions for the *monitoring quality* and the *acceptable level of operation* that is generally applicable, therefore the network lifetime should be examined in an application dependent context. Existing lifetime metrics in the WSN literature, however, disregard

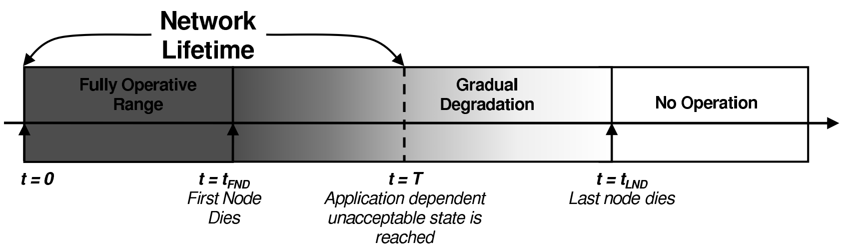


Figure 2.1. WSN timeline

the application dependence of the network lifetime and offer a single definition to be used for the whole range of WSN applications, which is simply not realistic. A lifetime metric that is not compatible with the specific application requirements causes two major problems: (i) lifetime measured would be incorrect, (ii) trying to optimize the network performance based on such a lifetime metric would result in a misallocation of the resources. Let us consider, as an example, the frequently used lifetime metric, *the time till the first node dies* [5–9]. A lifetime optimization based on this metric does not necessarily lead to the maximum lifetime for an application scenario that can tolerate a predefined percentage of loss of the initially deployed sensor nodes. Such an effort would dedicate all the network resources to delay the first node death without caring about what would happen at later stages which may lead to inefficiencies, hence shorter lifetimes, for the original scenario. Therefore, suitable lifetime definition that is in accordance with the application requirements is a must for correctly quantifying the useful lifetime so that it can further be optimized.

In this part of thesis, our aim is twofold: First, we propose a novel lifetime metric called Weighted Cumulative Operational Time (WCOT) for the realistic performance evaluation of the WSNs in an application dependent context. Second, with the help of WCOT, we show how a lifetime metric incompatible with the application requirements might result in misleading results. To achieve this, we focused on a specific WSN application area and conducted a series of simulation experiments whose performance is evaluated both by WCOT and the metric the time till the First Node Death (FND). We chose FND as the rival metric since it is widely used in works that optimize and/or compare WSN performances. The lifetime values obtained by the two metrics were either discrepant or contradictory for the cases studied. It is observed that FND values measured does not capture the actual lifetime behavior of the networks, whereas, WCOT realistically characterizes the application dependent operational lifetime of the networks involved. We believe that, with these results, it becomes evident that more care should be exercised on the lifetime metric which currently lacks in the WSN community.

2.2. Existing Lifetime Metrics

One can come up with various lifetime definitions for sensor networks in the literature. However, they show different directions for quantifying the operational lifetime of the network. Here, we shall go over some common lifetime definitions and discuss their capacity to represent the functional duration of a WSN. In our discussion, we will be in pursuit of a realistic lifetime metric that is in accordance with the application level requirements.

The most common lifetime definition encountered in the WSN literature is indisputably the time till the first sensor node death occurs (FND) [5–14]. FND, as lifetime metric, gives a lower bound on any possible lifetime definition - i.e. this lifetime definition cannot overestimate the operational lifetime of a WSN. Since the level of functionality offered by a sensor network drops with occurring sensor deaths and that there is no generally applicable threshold for the number of sensor nodes after which the network is considered to be dead, FND takes the simplistic approach and assumes the death of the network is triggered by the initial sensor node death. Another factor that FND owes its popularity to is that it can easily be incorporated into linear programming (LP) style problem formulations. It is considerably difficult to incorporate a more realistic lifetime measurement method into a linear optimization problem and most such efforts employ FND to quantify the network lifetime [5, 6]. Despite its wide use, confining the lifetime with the initial node death does not reflect the actual operational lifetime of a WSN, since in almost all WSN applications, initial sensor death causes only negligible deformation on the network functionality. Furthermore, trying to elongate the network lifetime in terms of FND is basically delaying the initial sensor node death which does not necessarily mean an extended lifetime for many possible applications.

Another class of lifetime definitions rely on the percentage of alive sensors contained in the network [15, 16]. Here, the network is assumed functional when the ratio of alive sensors to the initially deployed sensors is above a predetermined threshold. This is clearly a better lifetime definition when compared to FND, however the way the threshold is chosen is arbitrary and does not necessarily reflect the constraints of the application scenario. Also the importance of the sensors throughout the network is not homogeneous, therefore trying to determine the lifetime with a single percentage value is not realistic.

Yet another approach to quantify the network lifetime is in terms of coverage [17–19]. In general, the term coverage can be interpreted as sensing coverage or networking coverage. However, assuming $r_{sensing} = \frac{1}{2}r_{comm}$, sensing coverage implies networking coverage [20]. General practice is to assign a specific threshold coverage value to limit the lifetime of the network - e.g. the time till the coverage drops below 90 per cent [21]. The critique put forward for lifetime metrics based on the percentage of alive sensors applies here as well, that is in the works no justification is given for the specific coverage ratio assumed. Furthermore, functionality of the network is not only a function of coverage but also depends on the distribution of the coverage. A better way to incorporate coverage into the lifetime measurement is by assessing not only the coverage percentage but also taking care of how the loss of coverage is distributed among the region of interest. Maleki et al. [22] defines the lifetime as the time till the spatial distortion goes below a certain limit.

For two-tier WSNs, FND lifetime metric is categorized as $N-of-N$ lifetime [23] meaning that any N of the initial deployed N application nodes are equally important for the network. More realistic versions include $K-of-N$ and $m-in-K-of-N$ lifetime metrics in which K and $m-in-K$ indicate the vital nodes in the network [24].

Brown et al. formulate the maximum lifetime problem as the Maximum Flow Life Curve [25]. Here the authors try to elongate not only the initial node death but also the death times of all the sensor nodes in the network. A similar formulation is presented in [26], in which node death times are sorted as a vector and LMM (Lexicographic Max Min) optimal vector is sought. This definition is streamlined for Linear Programming style, and although their treatment seems to consider for the death times of all nodes, actually the earlier sensor deaths are decisive on the lifetime metric, first node death being the most decisive. Therefore, optimizations based on LMM optimality or Maximum Flow Life Curve still try to postpone the initial sensor death as much as possible. LMM based method does not take into account the distribution of the node death times but only focus on their sorted vector comparison , which can easily boil down to FND lifetime metric. To exemplify, a WSN whose first node death time is t_1 has shorter lifetime than a WSN whose first node time is t_2 whenever $t_1 < t_2$ holds.

According to LMM optimality this conclusion is reached irrespective of the distribution of the forthcoming sensor death times.

Blough et al. [27] give a more generic lifetime definition that is based on coverage, connectivity and the number of functional sensor nodes. Weight constants $(c1, c2, c3)$ are assigned to each of the feature to let users adjust the lifetime definition. In [28], for instance, (1,0,0) is utilized as weights meaning that coverage is considered only to measure the network lifetime. To the best of our knowledge, there are no studies that makes use of the three constants in a way justified by the application scenario requirements exist. Inadequacy of using FND as a lifetime metric is discussed in [29] where a generic concave function from the vector of node lifetime values to the network lifetime is defined, however practical considerations are missing. Another lifetime definition, on the other extreme, is the time till the last sensor node dies. Although not as common as the previous definition, there are studies that depend on this definition of network lifetime [7, 8, 30]. This definition implies that a WSN is functional even in the presence of a single sensor node. This is clearly an overestimation of the useful lifetime as the monitoring quality of a sensor network drops below acceptable threshold long before the number of remaining sensors is unity. A utility based lifetime definition is given in [31]. Adlakha et al. defined utility as the product of connectivity and accuracy and defined the lifetime as the time utility drops below a threshold. Here, the utility has a specific definition and the network performances above the threshold are not differentiated as we do in the case of WCOT. This is a necessity especially for networks that can operate till the utility drops to low levels. Nonetheless, work presented in [31] provides valuable insight into the lifetime measurement problem in sensor networks.

The motivation behind WCOT is based on the observation that existing lifetime metrics do not consider application related constraints and focus solely on a single point on the timeline of WSN, neglecting the evolution of the network throughout its operation. To further understand this, Figure 2.2 shows the performances of three hypothetical sensor networks via a remaining alive sensors vs time graph. These type of graphics are commonly used for denoting the performances of WSNs. Here, the networks A and B have identical lifetime values according to the metrics that measure

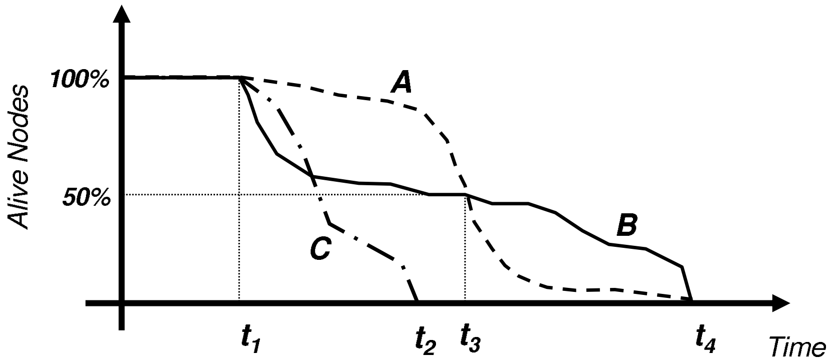


Figure 2.2. Sensor networks A and B experience the first sensor node death and the last sensor node death at identical times

the time till (i) the first node death , (ii) 50 per cent of the nodes die and (iii) the last node death. It is apparent, however, that the node death times are distributed rather differently for the two networks and that the overall utility obtained from each of them (depending on the application) are likely to be distinct enough to imply different lifetime values. Utility based, weighted lifetime calculation enables WCOT to quantify these differences in the performance which are visually recognizable but not captured by the existing measures. To achieve this, WCOT calculates a lifetime value which is a function of the complete history of the network states that a WSN goes through till the utility drops to zero.

2.3. The Utility Mapping

WCOT, to be able to assign a utility value, needs to asses the usefulness of the state that a WSN is currently in. Since the functionality obtained from a certain WSN state is application dependent, a scenario specific mapping that relates the network state to the corresponding utility value is needed. A utility function performs this mapping. Figure 2.3 depicts a sample utility function. The threshold state after which the network is assumed to be non-functional is denoted as S_f in the figure. Hence the utility value U_f corresponding to S_f is simply zero. U_i and S_i depicts the initial utility value and the network state respectively.

The input to the utility function are the network states. The *network state* is a generic term for expressing the measurable condition of a WSN. There are many

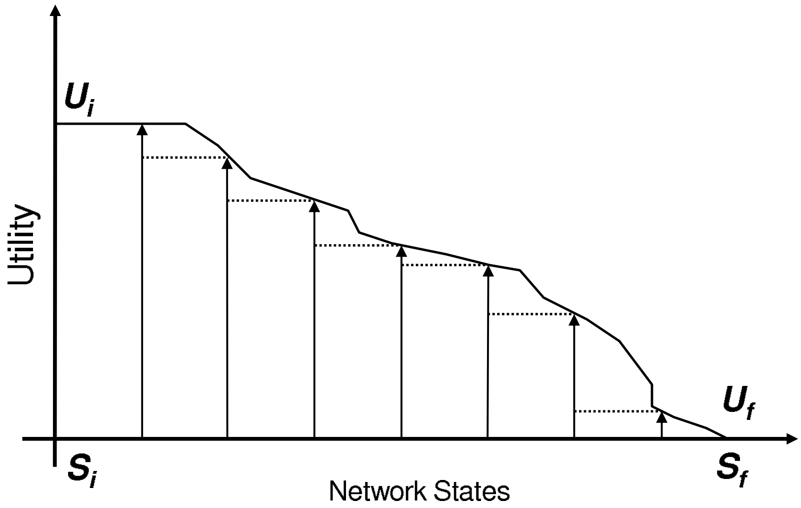


Figure 2.3. A sample utility function

indicators that can be used to quantify the network state, some of which are: coverage of the network, number of alive sensors, average residual energy/node, breach probability and maximum delay. In this work, the outcome of the utility function is normalized to the [0,1] interval. As shall be explained in Section 2.4, WCOT operates by assigning the utility values as weights to the time intervals and this normalization takes place so that the sense of time is not distorted in the resulting lifetime. For example, a duration of t_1 seconds which has a utility value one is reflected into the resulting lifetime as t_1 seconds, whereas an operational duration with 0.5 utility will be reflected as only half of the duration. Another utility interval would also function correctly but it would be harder to correlate the resulting WCOT value to the actual operational time.

Here, we want to emphasize that to be able to assign the utility mapping, the user need not have any expertise on the internal operation of a sensor network such as which routing and MAC protocol is used or what the initial battery capacity of the nodes are. The user takes the black box approach and does not concentrate on the network itself but rather on the requirements of his/her own application scenario. To exemplify, for a surveillance application, important factors for the end user is how much of the geographical area is covered by the sensor network, i.e. the size of the Effective Surveillance Area (ESA) and the delay experienced by the packets as they reach to the sink node, i.e. the Reporting Delay (RD). Therefore, the user would map various ESA and RD value combinations into utility. Both of these performance criteria are objective measurable conditions of the network which we have identified as the *network state* in the above discussion.

Throughout its operation, utility of the network dynamically changes and, generally speaking, it is a monotonic, non-increasing function of time, assuming no sensor redeployment. For the cases in which the utility of the network depends on the instantaneous traffic load the utility can fluctuate, however, in the long run the utility should have a decreasing trend. Note that the utility function does not contain any temporal information and that we need to have the performance of the network in terms of network states in time to further calculate the lifetime which is shown in the demonstrative example given in Section 2.4.2.

2.4. WCOT (Weighted Cumulative Operational Time)

To measure the lifetime, WCOT takes the reverse approach and instead of trying to give a single lifetime definition, it lets users to incorporate application specific requirements into the metric itself. WCOT is a utility based method, in which we abstract away the application dependent term *monitoring quality* with the notion of *utility*. Utility, as used in this context, denotes the extent of the collaborative monitoring activity performed by the sensors in a WSN. The dynamic utility of the network is captured by the user supplied *utility function*. The utility function, therefore, not only denotes the application specific threshold point after which the network is assumed to be dead, but also it specifies the continuous degradation of the network functionality that occurs due to eventual sensor deaths.

One major difference of WCOT from the current lifetime metrics is in the way it handles the degrading functionality of a WSN. Existing metrics, assume that the network is fully and equally functional throughout its lifetime, therefore they tend to measure the lifetime as the time between the initial and a final state. WCOT, on the other hand, divides the time into smaller durations and assigns different weights to operational intervals which have different utilities. The weighted lifetime calculation method reflects the changing utility of the network in time into the lifetime value.

This approach gives rise to an increased lifetime measurement accuracy, meaning that WCOT is able to assign different lifetime values to the WSN performances which cannot be differentiated by the existing measures.

2.4.1. Formal Definition

Definition (WCOT): Lifetime of a WSN is the sum of weighted subintervals of operation time where the weights are the utility offered by the network for the subinterval at hand.

Let Δt_i denote the duration in which the utility offered by the network is U_i and let us assume that the utility can take D different discrete levels. The network lifetime as defined above can be formulated as:

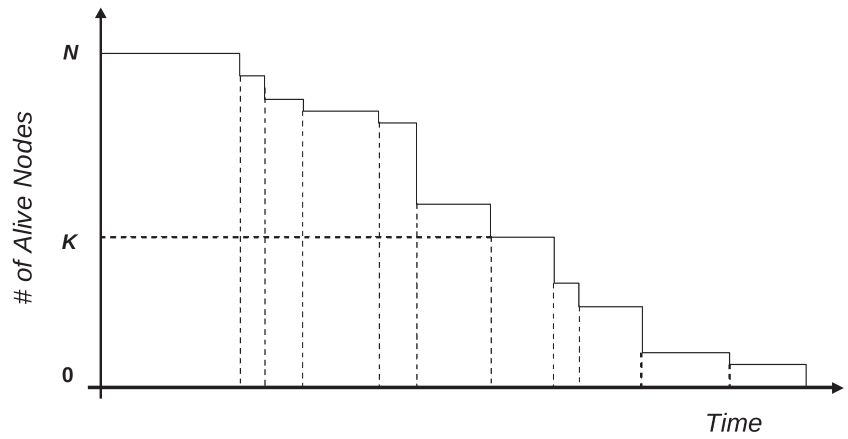
$$Network\ Lifetime\ (WCOT) \equiv \sum_{i=1}^{D} U_i \Delta t_i \tag{2.1}$$

Assuming discrete utility levels is based on the fact that a decrease in the utility of the network occurs only due to sensor deaths which is discrete in nature.

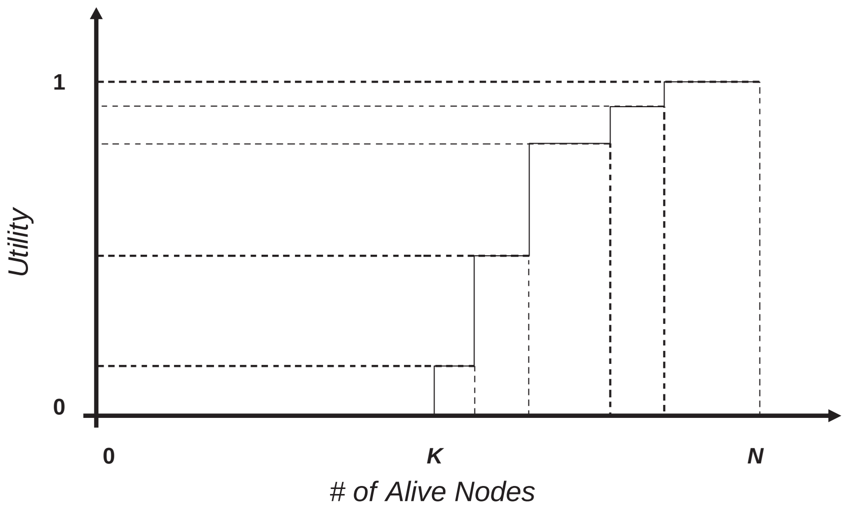
2.4.2. The Graphical Interpretation of WCOT: A Demonstrative Example

Traditionally the performances of the WSNs are shown in terms of the consecutive network states in time. An example graph is given in Figure 2.4(a) which shows the performance in terms of the number of alive sensors in time. Let us assume that the utility function specific to the WSN application in question is as depicted in Figure 2.4(b).

The network has initially N sensors deployed and is assumed to be functional at various levels till $(N - K)$ of its sensors die out. To calculate the lifetime of this WSN performance via WCOT, firstly, the graphs in Figure 2.4 need to be transformed into a *utility vs. time* graph. This is achieved by replacing the alive node numbers in



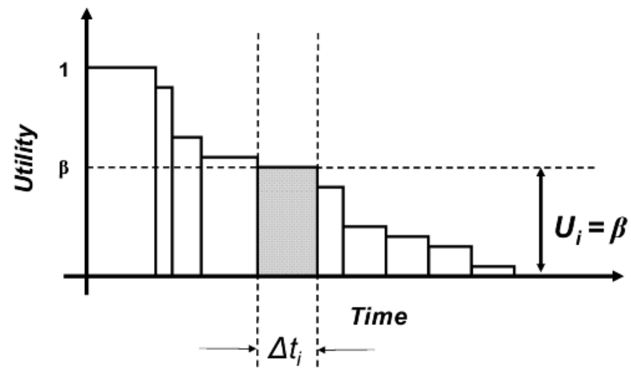
(a)



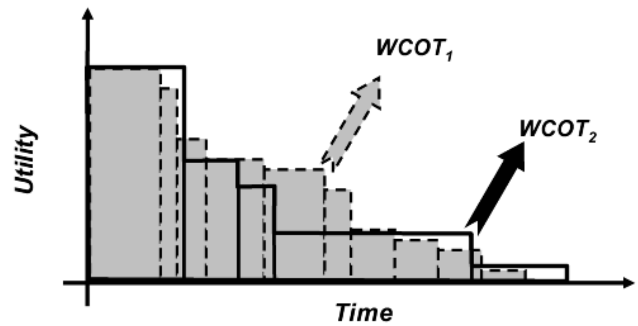
(b)

Figure 2.4. (a) Performance in terms of alive sensors in time (b) The utility function

Figure 2.4(a) with the corresponding utility values obtained from Figure 2.4(b). Figure 2.5(a) shows the resulting *utility vs. time* graph. The shaded region in the graph depicts the weighted lifetime calculation step for the duration Δt_i. Each duration is weighted with the utility value specific to the duration to obtain the effective resulting time. When all intervals are considered, WCOT actually performs a discrete integration over the graph. In Figure 2.5(b), performance of two different sensor networks are



(a)



(b)

Figure 2.5. (a) WCOT calculation shown graphically (b) WCOT lifetime values for two different sensor networks

quantified by the resulting areas computed by WCOT. The magnitude of the area is determined by the changing utility till the instance utility drops to zero. Therefore, the lifetime value obtained by WCOT is a reflection of the whole history of a WSN performance. To compare the performances of two sensor networks, we do the integration on each graph and compare the areas computed, as shown in Figure 2.5(b).

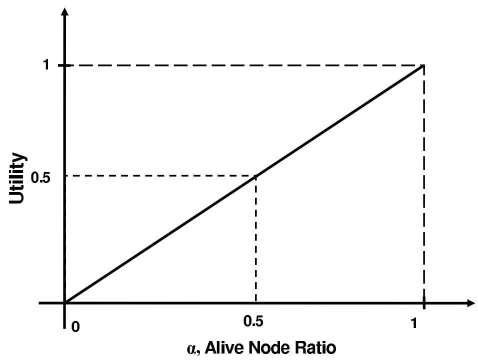
2.4.3. Alternatives for the Utility Function

To make the discussion independent of the absolute number of sensor nodes, we define the Alive Node Ratio (ANR), α, as the ratio of the number of alive sensor nodes to the number of nodes that were initially deployed. Thus, the utility functions presented here will have the [0,1] interval both as their domain and their range.

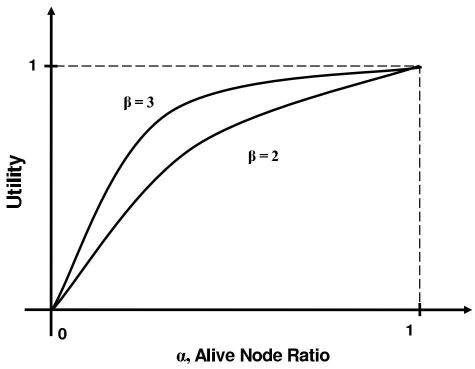
2.4.3.1. Homogeneous Utility Assignment:. If the utility of each initially deployed sensor node is assumed to be identical and static throughout the network operation time than we can talk about a homogeneous utility assignment among sensor nodes. In this type of utility assignment, death of a sensor node causes a degradation of $1/n^{\text{th}}$ of the initial utility where n is the number of initially deployed sensor nodes. $U_{i,n} = \alpha = i/n$ is an example utility function which is homogeneous. Figure 2.6(a) graphically shows how utility changes with respect to ANR for this type of utility assignment.

2.4.3.2. Biased Utility Assignment:. A sensor network application basically monitors a region of interest and monitoring quality drops not in proportion to available sensors, i.e., homogeneous utility distribution is not realistic. Therefore, when defining the utility function, a positive bias should be given to the network states where ANR is close to unity. $U_{i,n} = 1 - (1 - (\alpha))^{\beta}$ is an example utility function in which the initial sensor deaths are assumed to cause minor degradation in the monitoring quality whereas continuing deaths seriously disrupt the network functionality. Some example utility functions for different β values are depicted in Figure 2.6(b).

2.4.3.3. General Partial Utility Functions:. Existence of wide range of WSN applications means that there is a large group of different requirement sets that have to be represented by different utility functions. A single continuous function may not be adequate to represent the changing utility as a function of sensor node population.



(a)



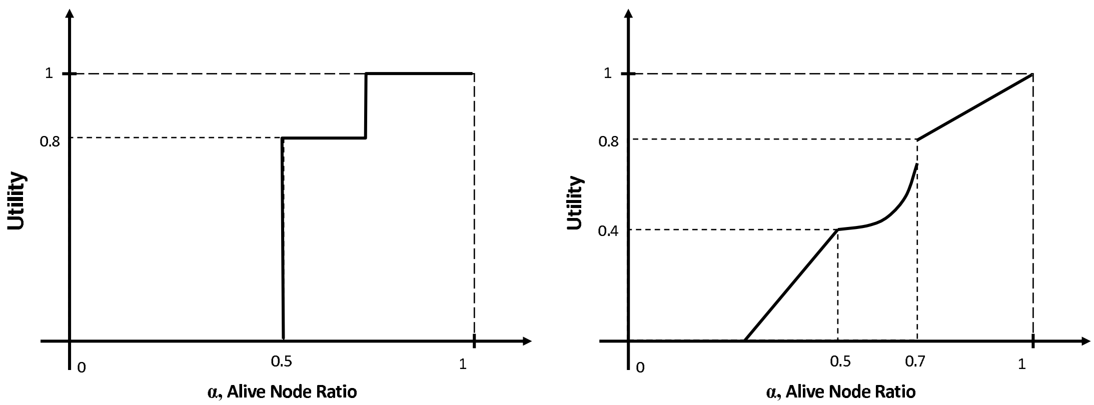
(b)

Figure 2.6. (a) Homogeneous utility assignment (b) Biased utility assignment example

As a remedy, we can employ partial functions that can model specific utility behavior of the sensor network application at hand. With partial functions, we can assign completely different utility characteristics corresponding to various ANRs. An example utility function that is defined as a partial function is given in Equation 2.2. Note that the utility is zero when ANR drops below 0.6.

$$U_{i,n} = \begin{cases} i/n & 1 \geq i/n \geq 0.8 \\ (i/n)^2 & 0.8 > i/n \geq 0.6 \\ 0 & 0.6 > i/n \geq 0 \end{cases} \tag{2.2}$$

Figure 2.7 depicts graphically two other utility functions that are defined in terms of partial functions. They reflect the needs of totally different WSN application scenarios.



(a) (b)

Figure 2.7. Various utility mapping examples

3. NETWORK LIFETIME EVALUATION OF SCALAR WIRELESS SENSOR NETWORKS

In this chapter, we devise a series of simulation tests in which the network lifetime is measured with both WCOT and FND. FND is, by far, the most frequently employed network lifetime metric in the WSN literature [5–14]. One other alternative lifetime metric we considered for the performance evaluation was the time till the Last Node Death (LND). However, during the tests LND, as expected, overestimated the operational lifetime by many factors. Also, the LND lifetime results showed high statistical variance which is not desired for a performance metric. Therefore, bearing in mind the popularity of FND, we confined the quantification of the lifetime results to the metrics WCOT and FND.

3.1. Application Scenario

We focused on environmental monitoring as our basic application scenario during the performance analysis tests. Environmental Monitoring is a frequently studied subject among WSN studies [32, 33]. Also with the foreseen proliferation of commercial products that are customized for environmental monitoring [2], the lifetime quantification will be a practical problem which is required for the performance analysis of sensor network solutions. In this work, we crudely classify the environmental monitoring scenarios as *periodic* and *event triggered* and perform lifetime quantification for both cases. Common features of both periodic and event triggered monitoring include delay insensitive communication, spatial and temporal redundancy of the data and sensor deaths being tolerable to some extent.

In the demonstrative example presented in Section 2.4, the utility is expressed as a function of the number alive sensor nodes. Although this simple choice of utility mapping gives a clearer understanding of how WCOT operates, it may not always reflect the actual utility of the network. This is due to the fact that not only the sheer number of active nodes but their distribution in the area is also important in assessing the utility. A better choice for the utility mapping is to use the sensing coverage. When compared with the number of alive nodes, coverage is a more direct indicator of the utility of a sensor network, since it is directly related to the monitoring quality. The utility function that we will adopt in all of the WCOT calculations in the rest of this chapter is given in Figure 3.1. Here, the sensing coverage is mapped to utility in a linearly degrading fashion such that coverage between 0.7 and 1.0 corresponds to utility values in [0.5,1]. Coverage below 0.7 indicates a nonfunctional environmental monitoring sensor network for our case.

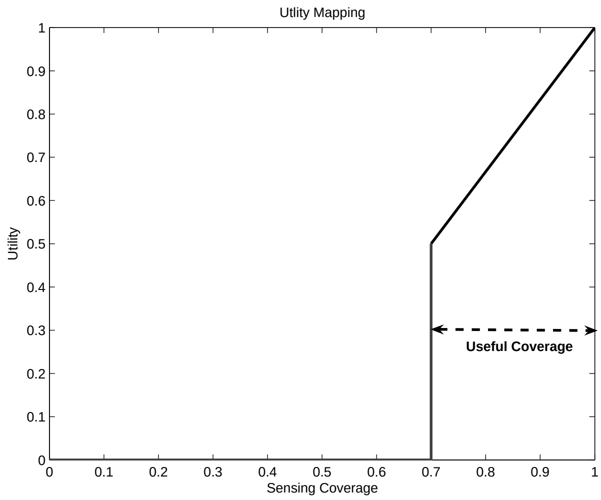
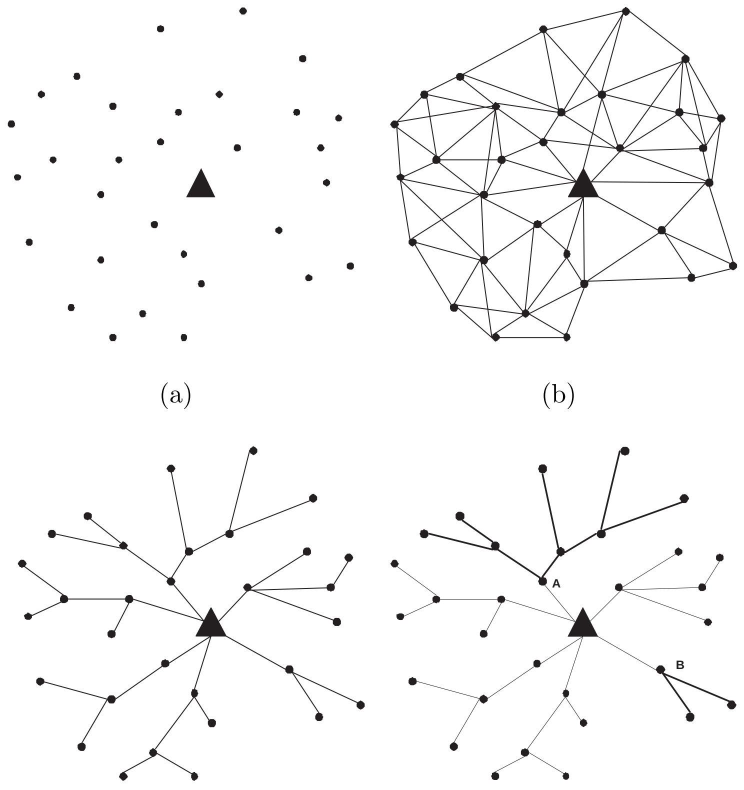


Figure 3.1. Utility function to represent the environmental monitoring application scenario considered

3.2. Probabilistic Energy Aware Routing (PEDER)

The routing protocol has crucial effect on the network lifetime since routing decisions determine how the energy cost of the routed packets is distributed among the network. One intuitive idea to consider in the routing layer is to use the Minimum Hop Routing (MHR) [34]. MHR is a greedy minimum cost routing algorithm in which the link costs are taken as unity. Pros of the MHR include the easy protocol setup and low delay due to the minimum number of transmissions a packet experiences till it reaches the sink node. One crucial problem, however, with MHR is that while trying to minimize the number of hops, it favors links among nodes with longer physical distances. When combined with the energy expenditure model that is exponentially related to the distance, MHR introduces energy inefficiency, hence shorter lifetime for the whole network.

Another idea frequently employed in the early WSN routing literature is to find the energywise least cost paths from each sensor node to the sink node [5]. This routing scheme is termed under different names such as Minimum Cost Forwarding (MCF) or Minimum Transmitted Energy (MTE) routing. In this work, we will term this type of routing as Minimum Energy routing (MER).



(a) (b) (c) (d)

Figure 3.2. Unbalanced load problem of MER (a) Random deployment (b) Possible communication paths (c) Minimum energy Paths due to MER (d) Different traffic loads MER puts on nodes

MER is a static routing algorithm that uses least energy paths from each node to sink. Therefore, in a sensor network that employs MER every packet causes least possible energy expenditure in the network as a whole. Thus, MER is an optimal routing algorithm when global energy expenditure is the concern. However, MER behaves suboptimal on a real WSN, as it lacks a global energy resource which MER implicitly assumes. Also, when combined with its static behavior, MER creates an unbalanced load on certain nodes and causes early deaths, which in turn implies serious reductions in network lifetime. Figure 3.2 depicts problematic behavior of MER.

In this work, we modify MER so that optimal flavor contained in it can still be useful in the distributed energy context of a WSN. To achieve this goal, we devised a family of probabilistic routing algorithms called Probabilistic Energy DrivEn Routing (PEDER). Instances of PEDER - like MER - are aware of the energywise best - i.e. globally optimum - path from each node to sink. On the other hand, PEDER - unlike MER - does not constantly use the optimum paths but occasionally use alternative paths in a probabilistic manner. The exact mechanism for path selection differs among PEDER variants, details of which are explained in Section 3.2.1.

3.2.1. Overview of PEDER

Basic objective of PEDER is prolonging the network lifetime by achieving load balancing. PEDER employs dynamic multipath routing with its probabilistic routing mechanism based on energy relations. For each packet generated in the network, cascaded probabilistic routing decisions are taken on its way to sink resulting in different paths for different packets generated at the same node. Assuming n entries in the routing list on the average, there is a theoretical maximum of n^m possible paths for a packet that travels to sink in m hops. This theoretical bound cannot be reached because of the geographical distribution of nodes in a typical WSN deployment scenario: dense deployment of nodes with a single sink as destination results in paths coinciding as packets move closer to the sink. Nevertheless, PEDER introduces considerable amount of multiplicity in the number of paths a packet can travel along when compared to MER. PEDER is a family of probabilistic routing protocols. Three variations of it are simulated and results are compared to MER and among variants themselves.

Run Distiributed Bellman-Ford Algorithm to populate the routing table
Process the routing list: Consider only the first N entries in ascending order. (*Window size = N*)
Node State ← *Altruist Mode*
if *Node Type* ≠ *SINK* **then**
  while ((Packet Generated = TRUE) OR (Packet Received = TRUE)) **do**
    if (Packet Received = TRUE) AND (Node State = *Selfish Mode*) **then**
      Ignore Packet !
    else if (Packet Received = TRUE) AND (Packet Type = Delete Me) **then**
      Re-process routing list: Remove Neighbor that wants to be deleted
      if Deleted entry is the first entry in the routing table **then**
        Broadcast an "Energy Cost Update" message with P_{max}
      end if
    else
      Probabilistically choose next hop based on energy relations
    end if
    if (Residual Energy < Energy Threshold) **then**
      Broadcast Delete Me packet with P_{max}
      Node State ← *Selfish Mode*
    end if
  end while
end if

Figure 3.3. The skeleton algorithm for all PEDER variants

3.2.2. Common Mechanisms for all PEDER Variants

The three instances of the generic protocol family PEDER differ in the way they select the next hop. However, other than the path selection, they share a set of common mechanisms which comprise the skeleton of PEDER whose pseudocode is presented in Figure 3.3.

3.2.2.1. Setup Phase. Distributed Bellman-Ford (DBF) [35] is the frequently used algorithm to find cost-based paths of a routing protocol. We assume that the sensor nodes are able to adjust their transmission power. Therefore, DBF is implemented as an expanding ring search. Starting from the sink node, nodes broadcast their energywise distance to sink in variable broadcast powers (between P_{min} and P_{max}). To limit the number of setup messages that will flow in the network, the adjustable transmission power range is discretized into some small finite levels. DBF is based on a label offering mechanism, in which nodes process the setup messages and sort the labels offered by the neighbors in ascending order to find the minimum energy path that links the node to the sink node. Nodes continue to broadcast their labels until all nodes have stabilized their minimum energy next hop. This is how typically the setup phase of MER is implemented. Differently, in the implementation of PEDER, we keep the all the offers that reaches to each node during the setup phase. Therefore, at the end of setup phase, each node will not only know via which neighbor the minimum energy path goes through (i.e. best label offered) but also will be informed about second best, third best etc offers. This is how the multipath behavior is introduced in PEDER. Since every offer indicates a different path, not necessarily disjoint though, nodes will have a multiplicity of routing alternatives at their disposal. Note that, there is no extra cost associated with gathering this additional set of paths as it readily reaches to each node during setup. In the case of MER these extra offers are simply discarded. In our algorithm, nodes initially do not discard any of the offers and populate a routing list that consists of neighbor id and cost of reaching sink via that neighbor. An example routing list is shown in Table 3.1.

Table 3.1. A typical routing table

Next_NODE	A	B	C	D
Energy to Sink	12	18	23	46

When setup phase is over, nodes start producing and relaying packets. Best entry - i.e. lowest energy offer - right after the setup phase in the routing list corresponds to the node that MER would statically relay packets to. In the case of PEDER, not just the lowest energy path but all entries in the routing list are candidates for a possible relaying operation.

3.2.2.2. Routing List Size. Final size of the routing list - i.e. number of entries it contains when setup phase is over - depends on the number of neighbors a node has, which in turn, depends on the communication radius and deployment density. PEDER limits the final size of the routing list by ignoring offers that are expensive in terms of the energy they offer. Rationale behind such a design is to prune paths which are known to cause high energy expenditure before actually starting to use the paths. PEDER decides how expensive an offered path is by comparing it with the best offer in the list - i.e. with the one that MER would use. PEDER uses an algorithm parameter σ to filter out high energy paths. Any entry i in the routing list not satisfying Equation 3.1 is simply deleted from the routing list.

$$e_i \leq \sigma.e_{best} \tag{3.1}$$

This implies that neighbor i does not become a candidate next hop for any future relaying operations. PEDER performs this possible reduction in routing list size right after setup phase is over, when list is stabilized. Figure 3.4 shows a sample node that ends up with five entries but two of the entries being deleted with $\sigma = 2$. Another aspect of limiting the routing list size is related to the question “ What will be the label offered to other nodes in the network in the setup phase?”. Label offered - i.e. energy offered - by a node in the setup phase reports others about the cost of reaching to sink via the offering node. In the case MER, this process is straightforward since there is

Neighbors	A	B	C	D	E
Energy	30	40	60	85	120

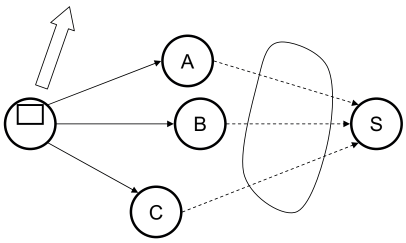


Figure 3.4. Limiting the routing list for $\sigma = 2$ (D and E get deleted)

only one entry that each node keeps record of. However, since we have a number of candidate next hops in PEDER, a single term that represents the whole list should be broadcast to the rest of the network. One alternative may be to broadcast the expected energy (weighted average of all entries in the routing list) as an offer. $\mathrm{E}[e_i] = \sum e_i.P_i$ where P_i depends on specific PEDER variant in use. Such an approach is followed in [9]. Although, expected energy of the whole list is a precise way of calculating the label to be offered, it has the drawback that any alteration to the list during setup phase changes the expectation, hence triggers a new broadcast. Assuming an N level transmit power each such broadcast triggers new broadcasts in descendant nodes.

A simpler and more energy saving approach is utilized by PEDER: nodes only broadcast the best offer in their routing list. Since the worst possible entry in the list is within a constant factor (σ) of the best entry, best entry not only represents a single value but rather a range of values. Therefore, it somehow represents the routing list itself. Although, this representation may not be as accurate as in the case of weighted averaging the whole list, it still works correctly and it is much more cheaper since only a change in the best entry - rather than the whole list - triggers a new broadcasts.

3.2.2.3. "Delete Me" Messaging. Due to the inherent traffic characteristics of a WSN, nodes near by sink are always heavily used. MER, which constantly uses static paths, when combined when combined with this inherent asymmetry causes over utilization of certain nodes in the network.Even when a multipath protocol like PEDER is used nodes nearby sink are bound to carry intense traffic.

As a partial remedy to this problem, PEDER lets nodes that drop below a certain threshold of their initial energy to stop relaying others' packets to sink. Such a node notifies all of its neighbors with maximum transmission power that it goes into *selfish mode*. Nodes that hear this advertisement check their own routing list and simply delete the corresponding entry, if it exists. Note that not all neighbors contain each other in their routing list due to the possible deletions at the end of setup phase, as explained in previous section.

3.2.2.4. "Energy Cost Update" Messaging. If a node receives a "Delete Me" message that causes its energywise best entry to get deleted, then it needs to inform the descendant nodes that may be using itself as next hop. In this case a specific type of control message called "Energy Cost Update" (ECU) message is broadcast to all nodes informing others about the changed cost of the path the node is the root of. Please note that an ECU message may trigger new ECU messages in the descendant nodes depending on whether the node that initially sends an ECU massage is itself a best entry in the receiving nodes. If the average energy method mentioned in Section 3.2.2.2 was used than every ECU message would trigger a new ECU message in all receiving nodes. ECU messaging when combined with limited routing list size, enables to dynamically update the cost in the network with minimal overhead.

3.2.3. PEDER Variation I

PEDER-I is not actually energy driven. It has been included among variants to better see the effect of energy driven approaches coming in other variants. PEDER-I simply does the routing with equal probability among neighbors. If there happens to be n entries in the routing list then probability that any neighbor is selected as next hop is $\frac{1}{n}$.

3.2.4. PEDER Variation II

PEDER-II probabilistically chooses next hop based on the energy fields of the entries in the routing list. As explained in Section 3.2.2.1, energy field of the entries in the routing list corresponds to cost of reaching sink using the node in the entry. PEDER-II assigns probabilities to each node in the routing list based on this energy cost till sink. We want the probability that a certain node is chosen as next hop to increase as the energy cost offered by the node decreases. With this requirement, best offer - i.e. the neighbor that MER would constantly choose - has always a greater chance over other candidates. However, as minimum energy path is not statically employed, PEDER-II distributes load more evenly than MER. Probability assignment for individual neighbors are done using Equation 3.2 and Equation 3.3. Equation 3.2 assumes an assignment which is inversely proportional to energy with an exponent parameter.

$$P(i) \quad \sigma \quad \frac{1}{(e_i)^\beta} \tag{3.2}$$

$$\sum_i P(i) = 1 \tag{3.3}$$

If $\beta = 0$, then PEDER-II degenerates to PEDER-I. As β increases, PEDER-II behaves more biased towards minimum energy route. When $\beta >> 1$ PEDER-II begins to degenerate to MER.

3.2.5. PEDER Variation III

Since PEDER is a heuristic algorithm, an extra flexibility in probability assignment may lead to better algorithms in the search space. In this context what PEDER-III does is to assign an extra capability to give inclination towards the least energy paths. To achieve this, we define a new protocol tuning parameter Θ which enables the users to assign a fixed prior probability to the best entry in the routing list.

$$P_{PEDER-III}(i) = \begin{cases} (1-\Theta)P_{PEDER-II} & \text{if } i \neq 1 \\ \Theta + (1-\Theta)P_{PEDER-II} & \text{if } i = 1 \end{cases} \tag{3.4}$$

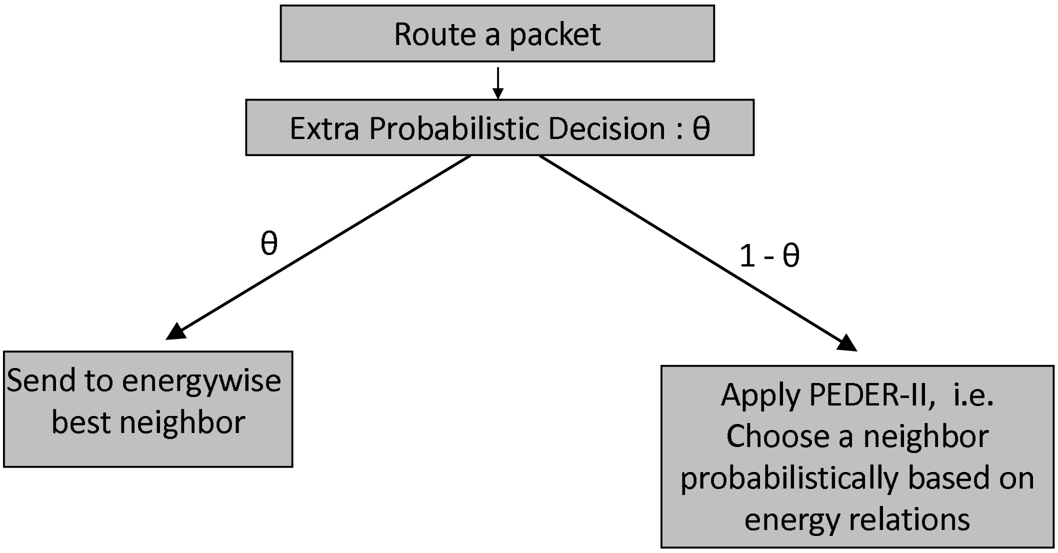


Figure 3.5. Additional probabilistic decision layer introduced by PEDER-III

3.3. Quantifying the Lifetime in an Application Dependent Context: Periodic Traffic Case

An example scenario for the periodic monitoring is *precision agriculture* in which temperature and humidity of the crops are reported [36]. A characteristic feature of periodic environmental monitoring schemes is the low traffic load. The order of data generation rate varies between tens of minutes to several hours [37]. For this type of traffic we can safely assume contention free communication at the MAC level. This fact puts the emphasis on energy efficient routing that keeps the coverage at acceptable levels as long as possible. In the simulation experiments MER and PEDER are used to guide the multihop communication between the sensor nodes and the sink node. MER is a static routing protocol in which the least energy paths are formed during the setup period and are used throughout the WSN operation. PEDER belongs to a class of routing algorithms called *Probabilistic Multipath Routing* which try to even out the load on the nodes by using alternate paths.

The choice of the routing algorithms is based on clarity rather than performance. Here, we want to concentrate on the effect of the network lifetime metric on the performance evaluation for a WSN scenario. In this context, a more advanced routing algorithm, such as LEACH [7] or TEEN [38] would complicate the discussion with their operational details. Due to the light traffic load, a perfect MAC layer in which nodes do not contend for the medium is assumed.

3.3.1. Simulation Settings

The network is assumed to be comprised of a single sink node and numerous sensor nodes. Exact number of sensor nodes deployed depends on the targeted node density for the deployment. Sensor nodes are uniform randomly deployed on a square geographic area with lateral length being l. Sink node is placed in the middle $(l/2, l/2)$. Nodes are assumed to have identical hardware including transmission capabilities and initial battery energies they posses. Also, nodes have the ability to individually adjust their transmission powers - thus their transmission ranges. We adopt the energy model given in [7] and [17] which can be expressed as:

$$E_{tx} = \kappa_1 + \kappa_2 . d^{\gamma} \tag{3.5}$$

$$E_{rx} = \kappa_3 \tag{3.6}$$

Here, E_{tx} and E_{rx} denote the energy per bit required for transmission and reception respectively. κ_1 is the distance independent part of the transmission energy, d is the distance between communicating sensor nodes and γ is the *path loss index* which is assumed to be two in our model. Energy required for the reception is independent of the distance d and κ_1 and κ_3 are assumed to be identical [7]. In this work κ values are taken as $50nJ/bit$ for κ_1 and κ_3, and $100pJ/bit/m^2$ for κ_2.

The experiment design is comprised of a setup in which the performance of a sensor network is studied as the node density is varied. Detailed packet level simulations are carried out with OPNET [39] in which multihop communication among nodes including the initial self organization period has been considered in the energy consumption relations. Common simulation parameters are presented in Table 5.1.

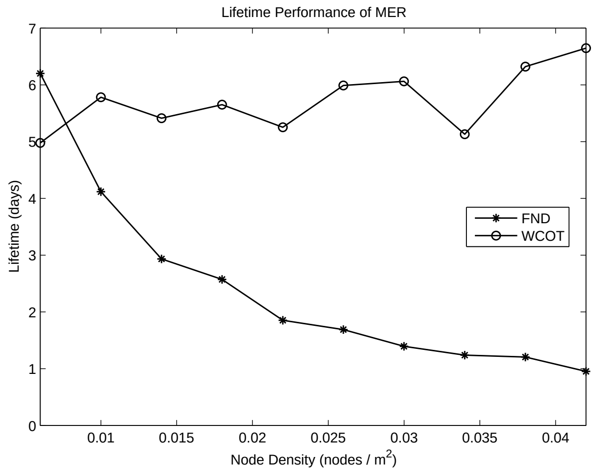
Table 3.2. Common Simulation Parameters

Paremeter	**Value**
Node Deployment	Uniform Random
Area Size	100 x 100 m^2
Node Density	0.006 to 0.042 nodes/m^2
Data Rate	20 kbps
Packet Size	1024 Bits
Trans. Range (min,max)	22 m , 102 m
Trans. Power (min,max)	2 mW , 22 mW
Sensing Radius	10 m
Initial Energy per Node	1 J
Sensing Period	15 minutes
Traffic Type	Periodic
Packet Generation Rate	1 Packet/15 min. (per node)

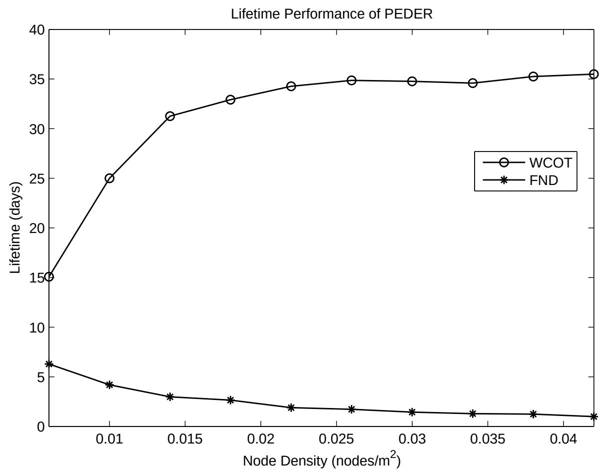
A total of 200 simulation runs were carried out in which each routing algorithm was tested under 10 different random sensor networks for each of the 10 different node density values. Results presented depict the average values.

3.3.2. Node Deployment Density and Network Lifetime

How WCOT and FND quantify the effect of the increased deployment density for the routing algorithms involved is explored in Figure 3.6(b). As seen in the figure, FND not only underestimates the operational lifetime of the network, but also gives the misleading impression that denser deployment results in shorter lifetime values. Error in lifetime quantification is more pronounced for the PEDER case, since PEDER is a dynamic routing algorithm that uses energy more efficiently, hence has an elongated utility degradation period. To further understand this phenomenon, Figure 3.7(a) shows where the *first node death time* corresponds to on the overall lifetime cycle of the network. FND, as a milestone, does not represent a network-wide characteristic that can serve as a lifetime metric since the utility of the whole network does

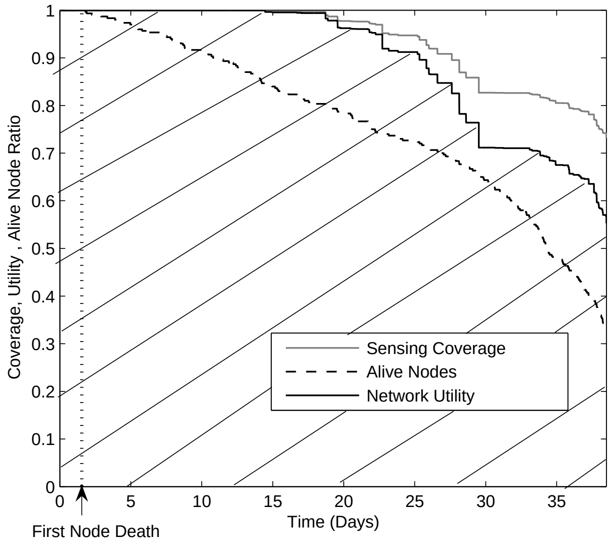


(a)

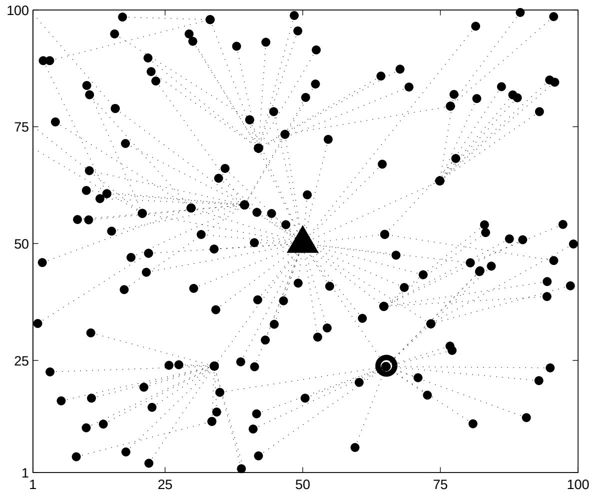


(b)

Figure 3.6. Performance evaluation in terms of WCOT and FND (a) MER Case (b) PEDER Case



(a)



(b)

Figure 3.7. (a) Evolution of the network parameters (300 Nodes, PEDER case) (b) Static routing paths of MER

not necessarily gets affected on the initial sensor death. The same argument can be extended to other lifetime metrics that focus on a single point for the determination of the non-functionality threshold, such as the time for 50 per cent of the initial nodes die. The problem with such metrics is based on the fact that the functionality of the network is dependent on more complicated variables, such as node redundancy and other application dependent factors. WCOT, on the other hand, incorporate these factors via a utility function and record the degrading utility by performing a discrete integration over the utility vs. time graph which corresponds to the shaded area in Figure 3.7(a). Since utility values are in [0,1] interval, integration does not affect the unit of the lifetime.

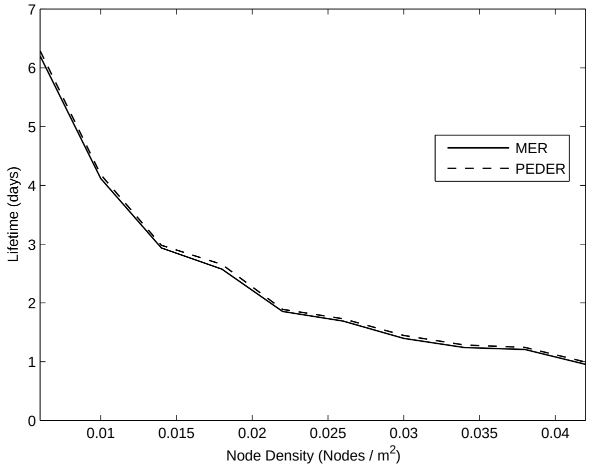
Focusing on Figure 3.7(b) enables us to realize how local the information obtained from the *first node death time* is. On the figure, the circled sensor node is the first node to die since it has the highest energy-wise load. In MER, FND always corresponds to the lifetime of such a heavily loaded node, instead of the lifetime of the whole network.

3.3.3. Comparative Evaluation of PEDER and MER

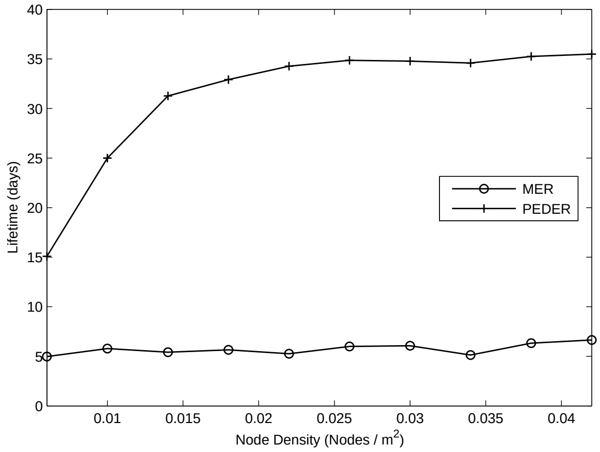
Similar results are obtained when WCOT and FND are used for the comparative evaluation of the routing algorithms at hand. Figure 3.8(a) depicts that FND is actually incapable of differentiating the lifetime performances of the routing algorithms for the density range studied. A more realistic picture is presented in Figure 3.8(b), in which PEDER performance first increases rapidly then saturates due to the inscalability of the algorithm itself. MER, on the other hand, generates smaller lifetime values and show only slight increase which is due to the fact that using static routing paths make descendant nodes unreachable in the case of eventual sensor deaths.

3.4. Lifetime Quantification: Event Triggered Traffic Case

For the event triggered environmental monitoring, an example scenario is the wild life monitoring [32]. Here, the target species are monitored and traffic created is a function of the movement of the animals. We will term the habitat agents that



(a)



(b)

Figure 3.8. Performance comparison of the routing algorithms MER and PEDER (a) Lifetime metric is FND (b) Lifetime metric is WCOT

will be monitored as targets. Due to the proximity of the sensor nodes in a typical deployment, a moving target triggers events that are sensed by many sensors. The target, as it moves along, sweeps the area continuing to trigger event based traffic. This type of traffic inevitably create packet collisions which needs to be addressed by the MAC layer. Also, the events can be rare, therefore communication level sleep schedule needs to be introduced at the MAC level for increased lifetime performance.

3.4.1. Simulation Settings for the Event Triggered Scenario

The monitored geographical area is assumed to be of rectangular form in which 140 sensor nodes and a single sink node is deployed. The deployment is uniform random and sink node is placed at the center. The node density is varied by changing the area length. Homogeneous hardware, including the initial battery capacity, is assumed for the sensor nodes. The sink node does not have energy or computational restrictions. In the experiments, the binary sensing model is used where a target within the sensing radius of a node triggers an event. Nodes create one packet in each second till the target leaves the sensing range. The case in which more than one target is detected is interpreted as a single event.

The communication stack used is comprised of SMAC [40] as the MAC layer and Minimum Hop Routing for the network layer. SMAC with its CSMA/CA based behavior is able to resolve the locally concentrated traffic in the areas where target(s) are detected. SMAC also employs a sleep schedule to take advantage of the energy reduction by periodically going into the sleep state. In the sleep state, only negligible energy is consumed by the nodes. We use the term *SMAC Duty Cycle* to mean the ratio of the awake period to the complete SMAC cycle, i.e. the sum of sleep and awake periods plus the period used for control messaging. In this context, a low duty cycle means longer sleep periods for the nodes. A summary of the simulation parameters used is presented in Table 3.3. Packet level simulations are carried out in OPNET [39], in which 10 different random network is created for each node density, duty cycle combination. The results shown are the average values.

Table 3.3. Simulation Parameters for the Event Triggered Monitoring Case

Paremeter	**Value**
Node Deployment	Uniform Random
Number of Nodes	140
Number of Targets	5
Mobility Model	Random Waypoint (v=1m/s)
Area Length	200 m to 500 m
Area Width	400 m
Node Density	0.7 to 1.75 nodes/1000m^2
Data Rate	250 kbps
Packet Size	1024 Bits
Max. Trans. Range	80 m
Max. Trans. Power	20 mW
Reception Power	10 mW
Idle Power	10 mW
Sensing Radius	30 m
Initial Energy per Node	1 J
Traffic Type	Event Triggered
Packet Generation Rate	1 packet/second
MAC Duty Cycle (awake)	5, 10, 40 (per cent)

3.4.2. Node Density, Sleep Duty Cycle and Network Lifetime

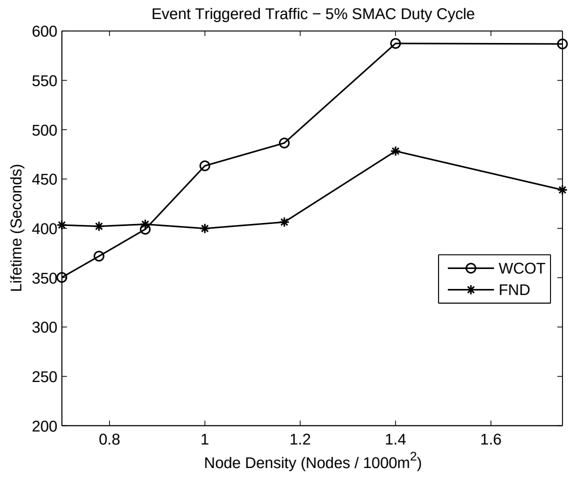
Lifetime quantification performed by WCOT and FND for the event triggered traffic scenario is depicted in Figure 3.9. Here, the two simulation variables that affect the lifetime are the node deployment density and the SMAC duty cycle values. Assuming that networking and sensing coverage requirements are met, further increasing the node density is a method to increase the energy content, hence the lifetime of the network. In the tests, node density is more than doubled by increasing it from 0.7 nodes/1000m^2 to 1.75 nodes/1000m^2. However, Figure 3.9 depicts that FND does not capture the increased lifetime due to the increase in the node density. Table 3.4 also numerically outline the difference in the lifetime measurement of the metrics involved.

Table 3.4. Increase in Lifetime as density is increased from 0.7 nodes/1000m^2 to 1.75 nodes/1000m^2.

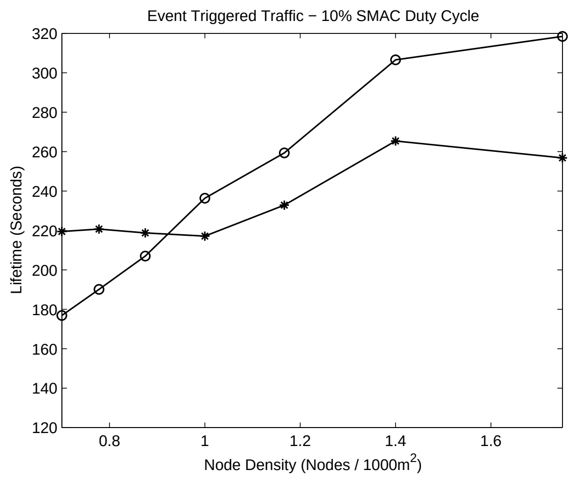
SMAC Duty Cycle	5 per cent		10 per cent		40 per cent	
Lifetime Metric	WCOT	FND	WCOT	FND	WCOT	FND
Increase in Lifetime	0.6759	0.0884	0.8006	0.1702	0.5434	0.0622

To better assess the results, we need to stress that changing the sleep schedule affects the energy consumption patterns of the sensor nodes. For low duty cycle operations, less power is consumed in the idle state and the main energy consumption source becomes the radio communication. Figure 3.9 (a) and (b) show the lifetime performances for the low duty cycle case. For the medium duty cycle values, however, the idle energy consumption is comparable to the communication energy consumption. This fact causes all nodes, irrespective of the volume of traffic they carry, to consume a fixed energy which shortens the lifetime as seen in Figure 3.9(c).

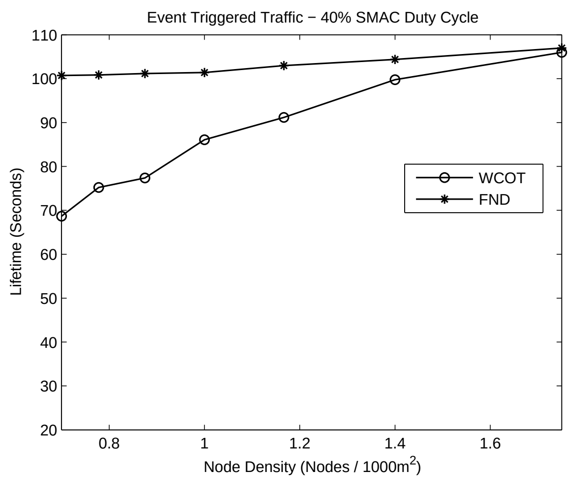
Due to occurring sensor deaths in the network, coverage hence the utility of the network drops in time. Figure 3.10 visually compares the degrading utility of the network over time with the first node death instance. WCOT value is found by integrating the area under the network utility line, as previously explained in Section 2.4.2. The time instance where ANR drops to zero represents the lifetime according



(a)



(b)



(c)

Figure 3.9. Performance comparison of WCOT and FND for duty cycle values (a) 5 per cent (b) 10 per cent (c) 40 per cent

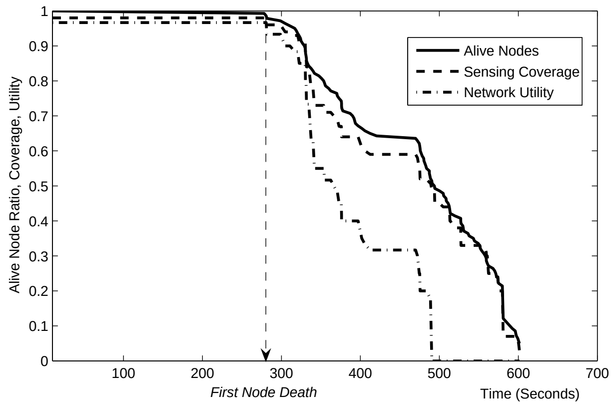


Figure 3.10. Crucial network parameters presented in time for Duty Cycle = 5 per cent and density = 0.7 nodes/1000m^2

to the LND lifetime metric. Please note that, utility, ANR and coverage values are all normalized to [0,1], therefore WCOT gives a realistic lifetime value between FND and LND according to the application requirement. For the extreme cases in which the application requirement dictates a lifetime metric similar to FND or LND, the utility function can easily be modified to comply with the application.

4. EXPLORING THE CAPABILITIES OF IMAGE BASED WIRELESS SENSOR NETWORKS

4.1. Handling Image Streams in Sensor Networks

Video Sensor Networks (VSNs) are the new members of the Wireless Sensor Network (WSN) family in which multirate streaming data traffic and related Quality of Service (QoS) requirements result in new problems that require novel solutions. Traditional wireless sensor networks are generally tuned for scalar data that is being relayed through multihop routes towards the data sink. Therefore, previously proposed WSN protocols may be insufficient for VSNs as the video streams require very large bandwidth compared to scalar data such as temperature readings. Additionally, due to the nature of the video, the streams have always realtime requirements. Moreover, since the logical unit of the communicated data becomes video frames, either successful delivery of all or a large percentage of packets of a video frame are required to be delivered to the sink node. Majority of the available video coding schemes such as MPEG are designed to have computationally intensive video processing at the sender and less computational effort at the receiving side. However, the requirements in WSNs are exactly the opposite. Sensor nodes have less computational power and energy capacities, on the other hand, the data sink is usually assumed to be computationally more powerful and has unlimited source of energy. This makes the complicated inter-frame coding based video processing techniques infeasible for the VSNs [41]. For that reason, a very low frame rate video is assumed which is basically a sequence of images to be transferred to the sink. However, in order for individual images to be useful for tracking or identification purposes, a certain percentage of the packets are required to be delivered to the sink.

Introducing a sleep schedule is required to increase the energy efficiency of a WSN. For traditional scalar type of data traffic, lowering the duty cycle result in a higher energy efficiency at the expense of increased delay [42]. However, in the context of video traffic, changing the duty cycle not only affects the delay but also the throughput of the system, which in turn affects the object identification or tracking quality. In general, due to the congestion in the network and the limited buffers of the sensor nodes, not all of the packets will be delivered to the data sink. For that reason, increasing the sensor video quality generated at individual nodes does not necessarily entail an increase in the received video quality at the data sink. In this work, we explore the limitations on VSNs in terms of the carried traffic rate and application level requirements. We run simulations with realistic parameters to explore the effect of the duty cycle and the frame rate on the performance of VSNs. It is shown that higher video quality can increase application level performance only within a bounded operational region.

4.2. System Model and Simulation Parameters

To asses the performance behavior of VSNs, simulations are run under OPNET simulation environment [39] with realistic parameters reflecting the hardware and software capabilities that are currently available. The deployment is done with single sink node located in the geometric center of the surveillance area. Nodes are equipped with image modules composed of cameras capable of producing and compressing video images [43], [44]. Raw image format is software adjustable and in our simulations Sub-Quarter Common Interchange Format (SQCIF) (128×96) format is assumed. The image module employs intra-frame encoding which results in compressed images of size 10 Kbits. Predictive encoding alternatives such as ISO MPEG or H.26X cannot practically be used in VSNs due to the high complexity involved [45]. Distributed source coding techniques are promising alternatives for encoding video in VSNs as they exploit the inter-frame redundancy with affordable complexity in the sensor nodes [41]. However, due to the lack of practical implementations yet available, we resort to the JPEG compression available on the image module. Software controlled frame rate feature allows video streams with rates between $1 - 12$ fps to be introduced to the network by each individual sensor node. Event triggered data generation is simulated where the triggering event is the visual detection of the target. Since the cameras employed support the background subtraction feature, they only produce an image when the scenery changes significantly. Triggering occurs when the target is within camera detection range of 30 m and is within the Field of View (FOV) of 52 degrees. The target is assumed to move within the surveillance area according to the Random Waypoint Mobility model where the target velocity is set to 10 m/s and pause time is set to zero seconds. Crucial simulation parameters are tabulated in Table 5.1. Data transfer at the frame level to the sink is assumed to be done in the application layer whereas packet level communications at the network and MAC layers are handled with GPSR [46] and SMAC [40], respectively.

Table 4.1. Simulation parameters

Paremeter	**Value**
Surveillance Area	400 x 400 m^2
Network Size	60 Nodes
Deployment Type	Uniform random
Video Frame Size	10 Kbits
Packet Size	1 KBits
Camera Frame Rate	1 to 12 fps
Field of View	52°
Camera Detection Range	30 m
Bandwidth	250 Kbps
Buffer size	20 Kbits
Target Mobility Model	Random Waypoint

4.3. Effect of Sleep Schedule and Frame Rate in Video Sensor Networks

Several simulation runs are performed with different duty cycles and the sensor camera frame rates. At each run, the total number of frames created is recorded along with the number of received video frames at the sink. Table 4.2 shows the average aggregate frame traffic rates achieved versus the camera frame rate based on the target detection scenario described in Section 4.2.

Table 4.2. Average video traffic triggered in the network

Camera Frame Rate (fps)	**Average Traffic Created (fps)**
1	0,1502
2	0,2994
5	0,7431
8	1,2002
10	1,4805
12	1,7844

4.3.1. Effective Traffic Carried in the Network

Generally, higher *video quality* is required for better VSN application performance. Video quality can be adjusted in the system by varying the image resolution and the camera frame rate. In our case, we fix the image resolution since a lower resolution may not be tolerated by the identification application, whereas a higher resolution results in frame sizes that cannot effectively be carried in the network. Therefore, in the simulations the frame rate of the cameras on the sensors is varied to alter the video quality throughout the network. As depicted in Figure 4.1, increasing the video quality in the sensors only pays-off up to a saturation point, after which the throughput drops, hence the average frame rate received at the sink decreases. To show the limiting factors on the throughput, the experiments are repeated for three different duty cycle values. Figure 4.1 exhibits that the saturation point is dependent on the duty cycle of the system. A higher duty cycle value enables higher network service rate by handling more packet transmission per unit time. Here, an implicit energy trade-off is also observed since increasing the duty cycle increases the energy consumption in the network.

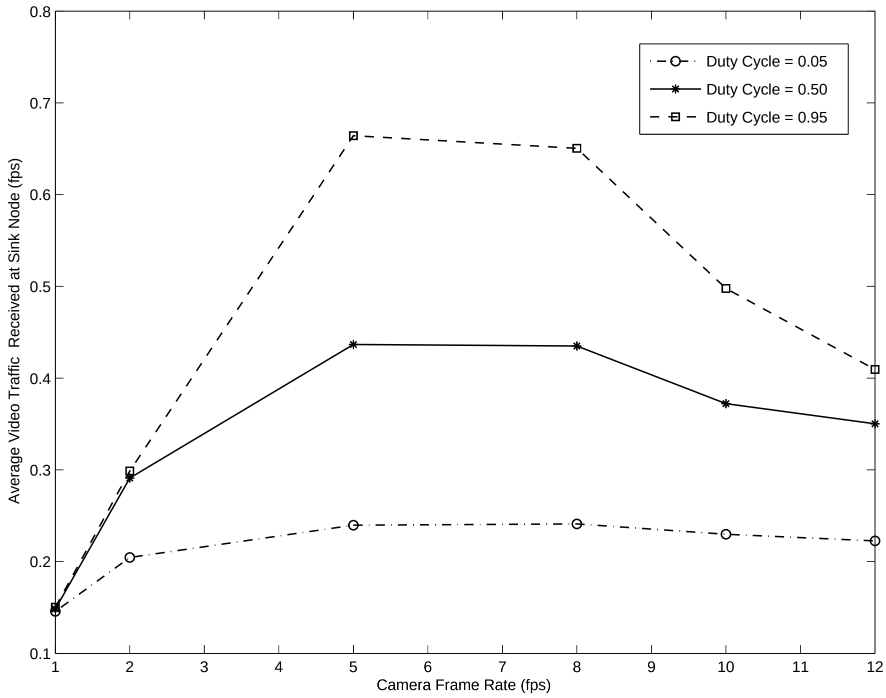


Figure 4.1. Effect of sensor video quality (frame rate) on the received frame rate at the sink

4.3.2. Delivery Ratio

As the compressed video includes dense information, a frame can be defined as lost after a certain drop percentage for the packets that belong to that frame. In our simulations we set that threshold to 10 per cent, i.e. if more than 10 per cent of packets that belong to a frame are dropped, then that frame could not be recovered and is labeled as a lost frame. Figure 4.2 shows the successful frame delivery ratio for different sensor frame rates under different duty cycles. Depending on the QoS requirements of the application, the maximum allowed sensor frame rate can be extracted from this figure. For instance, if the application requirement is 90 per cent successful frame delivery ratio, then for 50 per cent duty cycled network operation, the sensor frame rate must be 2.5 fps or less to achieve that QoS requirement.

To further understand how the duty cycle introduced by the sleep schedule affects the application level performance, we need to examine the main cause of the packet

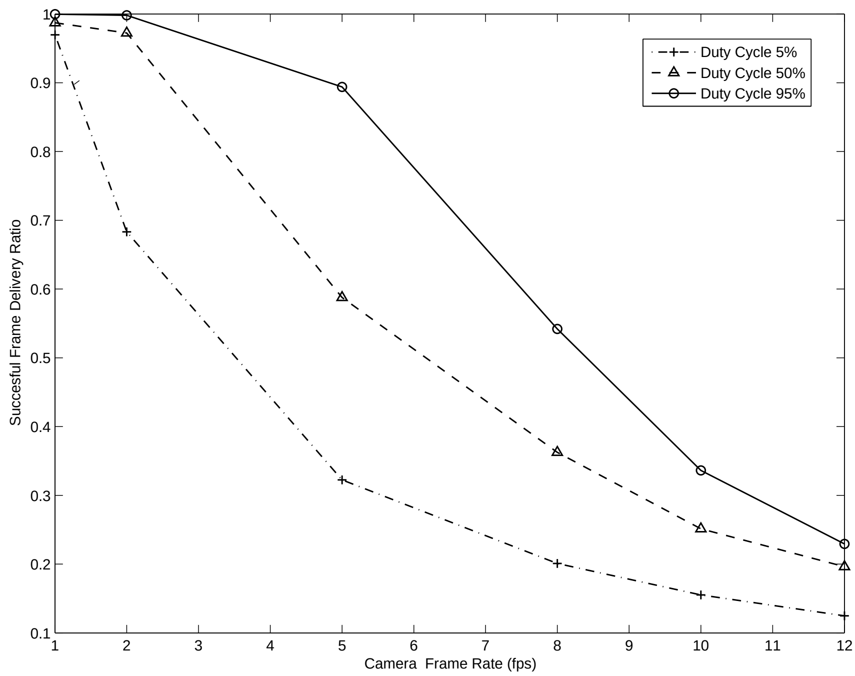


Figure 4.2. Successful frame delivery ratio

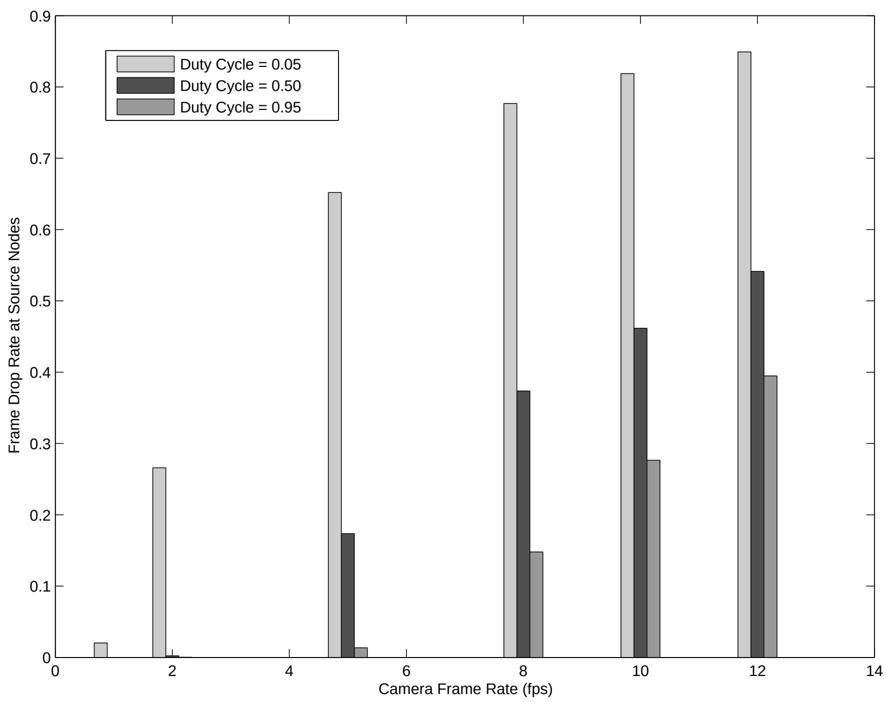


Figure 4.3. Ratio of aggregate dropped traffic at source nodes to aggregate created traffic

drops experienced in the network. Figure 4.3 depicts that as the duty cycle is lowered, considerable amount of video traffic is dropped at the source nodes. A sensor node operating at a low duty cycle, upon detecting the target begins to accumulate video frames as the probability to have a neighbor awake gets lower and buffer overflow occurs. More concisely put, by decreasing the duty cycle one actually limits the video traffic that can be introduced into the system.

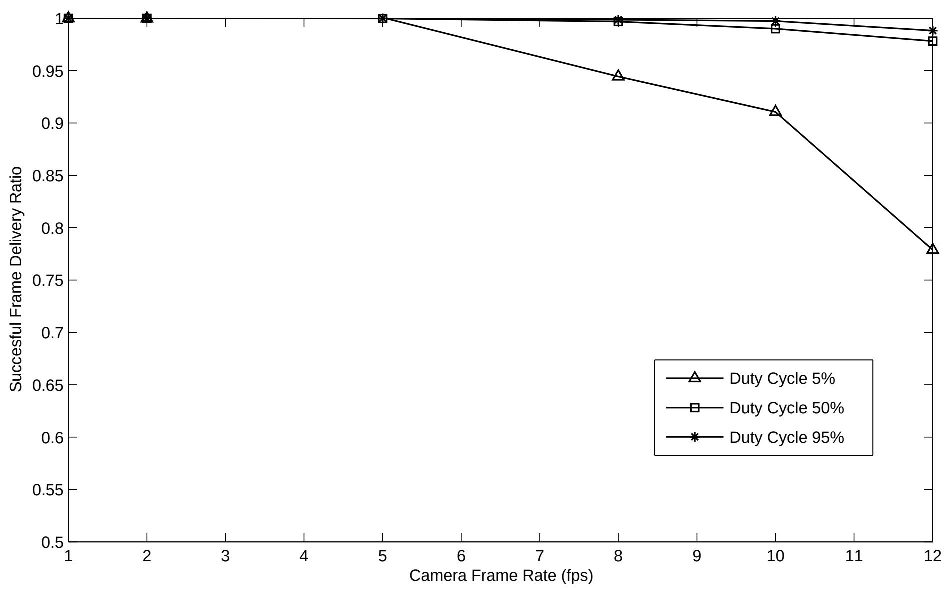


Figure 4.4. Successful frame delivery ratio obtained when buffer size is increased to 250 Kbits

4.3.3. Effect of Buffer Size

Current sensor nodes generally have around 60 Kbits of RAM available [47]. Our choice of 20 Kbits buffer size is based on the assumption that available RAM area that is not used by the communications stack and the application code can be allocated as a buffer to handle the images conveyed from the camera module. However, it is possible to increase the physical RAM size to create more buffer for image handling at the expense of increased costs. Here, we explore the effect of increasing buffer size to alleviate the overflow problem. The previous simulation runs are repeated for the buffer size value of 250 Kbits.

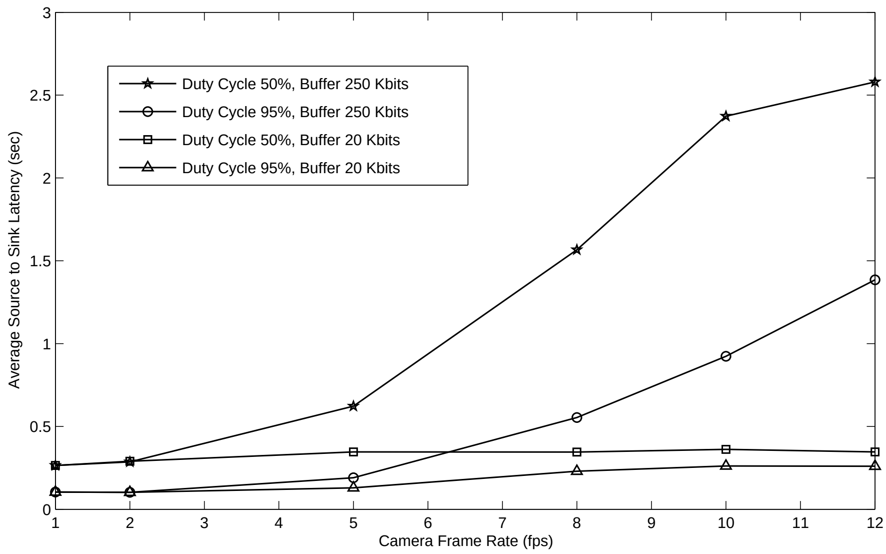


Figure 4.5. Effect of increased buffer size for duty cycle values of 50 per cent and 95 per cent

As shown in Figure 4.4 successful delivery ratio is considerably higher for the new buffer size. The major reason for this behavior is that high drop rates at source nodes are now eliminated, i.e. the video traffic is now being let into the network with a much less loss rate. How the increased buffer affects average delay in the system is shown in Figure 4.5 and Figure 4.6. Here the delay experienced by the packets as they travel from source to sink is measured. As expected, increased delay is observed, however, duty cycle is also an important factor. For low buffer size, maximum source to sink delay is bounded by 0.25 seconds, which is sufficient for realtime applications. For increased buffer size, different delay characteristics are observed as a function of SMAC duty cycle applied. For duty cycle values of 50 per cent and 95 per cent, delay is observed to be bounded by three seconds, which still may be considered as acceptable for many applications. The delay also varies with the increased frame rate and applications requiring stricter delay bounds should operate at lower frame rates.

The delay observed for the lowest duty cycle value is depicted in Figure 4.6. Here, it is clearly seen that increasing buffer size puts the system in a non-functional state, as the camera frame rate goes above one fps. Although, successful delivery ratio depicted in Figure 4.4 indicates that more than 0.75 of the frames are received at the sink even

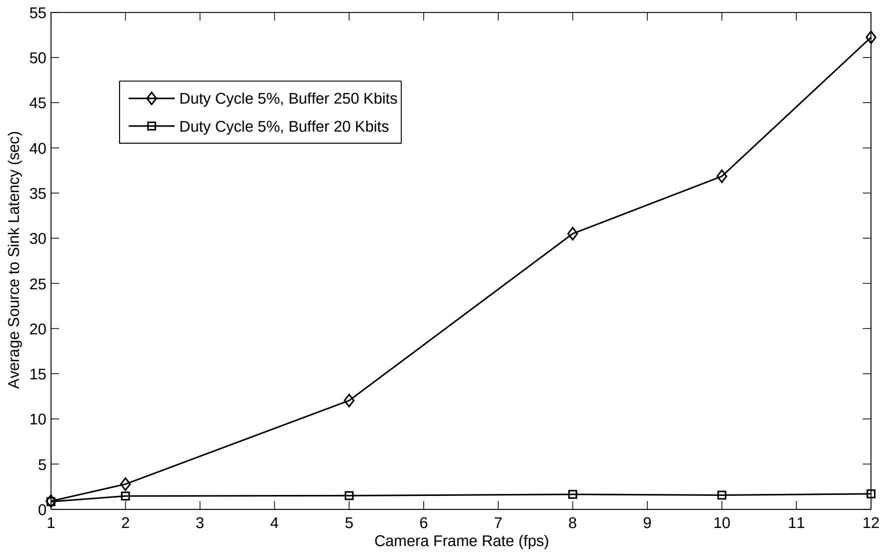


Figure 4.6. Effect of increased buffer size for duty cycle value of 5 per cent

for 12 fps, due to high buffering capacity combined with the limited communication capacity imposed by the low duty cycle, packets are received at sink with unacceptable delays.

4.4. Chapter Summary

In this chapter, capabilities and limitations of a VSN which is implemented with the currently available technology is explored. It is observed that sending images more frequently from individual motes can achieve better application level quality only within a bounded operational region. This region is determined by the available constraints on the mote hardware (communication bandwidth, buffer size) and also by the sleep schedule introduced. Simulation runs with realistic parameters are conducted to show the limits of the carried video traffic in relation to the application level requirements.

5. APPLICATION DEPENDENT QUEUE MANAGEMENT SCHEMES FOR THE MULTIMEDIA SURVEILLANCE SENSOR NETWORKS

5.1. Introduction

In this chapter we are looking for enhancements to video transmission over sensor networks. In the light of the results obtained in Chapter 4, we identify the main problems as follows: buffer overflow, congestion, low delivery ratio. Low profile hardware provided by the sensor nodes is only partially responsible for the problematic video transmission. We implement event based buffer management method to improve the video transmission.

5.2. Event Based Buffer Management

When compared with legacy Wireless Sensor Networks (WSNs) that operate on scalar data such as humidity and temperature, the visual information provided by Multimedia Wireless Sensor Networks (MWSNs) in general and Video Surveillance Sensor Networks (VSSNs) in particular, increase the accuracy of event identification and decrease the false alarm rate considerably. However this enhanced identification performance comes with the additional complexity of increased traffic volume that needs to be processed according to the realtime QoS requirements. This is very challenging since, in spite of the increased application and networking level complexity, VSSNs are typically implemented on similar hardware designed for scalar WSNs [48].

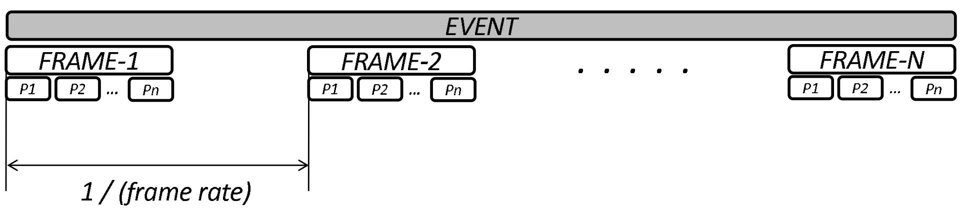


Figure 5.1. A sample event reported in frames and network packets

In a scalar WSN, the logical unit of messaging used by the application and the unit physical data conveyed in the network coincide to be the network packet. For a MWSN, however, the logical unit of messaging becomes a video frame which is almost always larger than the network packet size, therefore it is fragmented into packets before being pumped into the network. In this work, we add a further abstraction of messaging unit specific to VSSNs: *Event*. An event is identified as the sequence of image frames produced by the same source node triggered by the detection of a target. A sample event composition is shown in Figure 5.1. Taking the logical messaging unit as an *event* and not as a *frame* forms the basis of our approach of application level fairness since it is the sequence of correlated frames belonging to the same event that makes its identification easy and accurate at the sink node. The number of frames contained in an event is variable and as shown in Figure 5.2, it is a function of the target speed, V, the camera frame generation rate, K, and the path length, D_{AB}, covered inside the Field of View (FoV). Due to the spatio-temporal redundancy of visual information among consecutive frames, it is not easy to define the minimum required number of frames for the healthy reception of an event in the sink node. However, it can intuitively be concluded that the more frames received belonging to an event, the better recognition and identification take place at the center. Among the frames of an event, first arriving ones have more importance as their incremental contribution in terms of information content is larger than the ones arriving later. Incremental contribution of the first frame arriving to the sink (need not necessarily be the actual first frame created at the source node) is very large, since it makes the system aware of the event. Therefore, we will put further focus on the arrival time of the first frame.

In this chapter, we propose an event based queue management scheme to enhance the performance of VSSNs. Our goal is to have as many frames from all events as possible in the sink node. In a VSSN application, number of events generated in time can easily go over the total capacity of the network, especially when the number of targets that trigger the events increases. Even in the case of a single target, as depicted in Figure 5.3, when the target moves around the monitored area, it continues to trigger events which should still be conveyed to the sink node for localization and position estimation purposes.

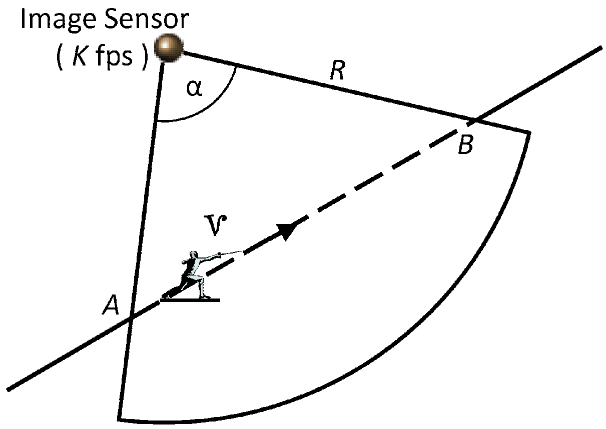


Figure 5.2. Target detection performed by a single node

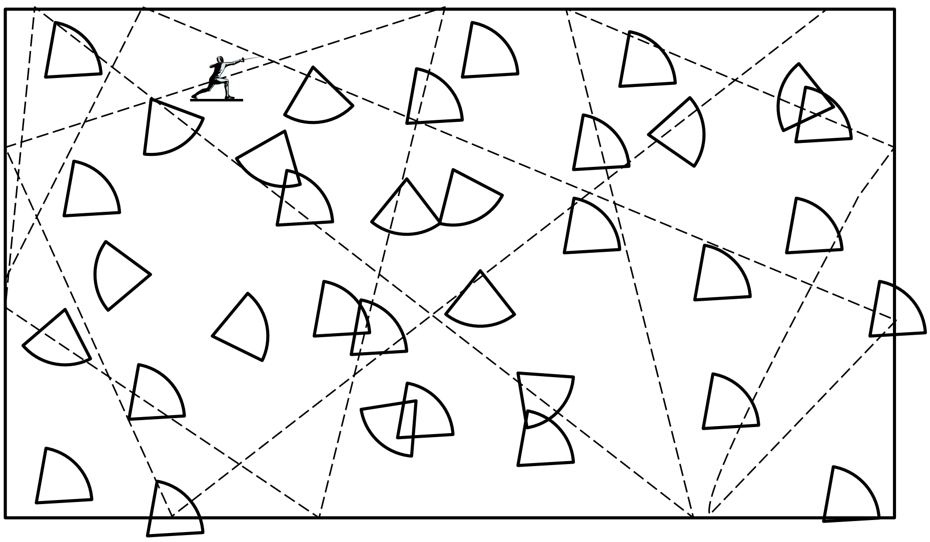


Figure 5.3. Random paths created by a single target in a sample VSSN deployment

Legacy FCFS queue management does not differentiate between packets belonging to different frames and events. However, for receiving the maximum visual information content at the sink, we seek for application level fairness and try to achieve a fair distribution of network resources among events. Fairness has been studied before in the context of wired and wireless networks. Most of the attention has been drawn to the node-level fairness, in which the total bandwidth is shared among flows fairly with respect to a required traffic matrix. For a VSSN application, what is more important than the node level fairness is the application level fairness [49] which translates to having fairness among events.

To implement the proposed event based queue management, we employ two different fair queueing algorithms, namely, Round Robin (RR) Fair Queueing and Least Attained Service (LAS) scheduling. In the simulation experiments, both RR and LAS is shown to perform better than FCFS queueing where frames are serviced in their order of arrival. RR operates on the snapshot of the queue, hence, provides fairness among flows whose frames currently exist in the queue. Event-flows in a VSSN show an intermittent behavior, therefore, steady state flow rates may not always be attained. In this respect, LAS, which considers not only the current queue content but also the service history of the event-flows, is more successful in dealing with short-lived event-flows, hence provides better event level fairness. Operationally, when event-flows coincide on a sensor node, LAS gives priority to the flow who has sent least number of frames so far. This functionality has two implications: (i) when a packet is dropped due to overflow, it is guaranteed to belong to a flow that has sent the maximum number of frames so far, i.e. the drop will decrease the information content of a flow that already transmitted maximum visual information, (ii) in cases when no overflow is experienced, least sequence numbered frames of an event will have priority over frames of other event flows. Therefore, delay of the initial frames will be decreased.

5.3. Related Work

Fairness is well-studied in the context of wired networks [35, 50, 51]. For wireless communication, fairness is generally discussed according to the OSI level that fairness is supported. For instance, [52] advocates that MAC level fairness alone does not necessitate the fairness of the whole wireless network, although MAC support can increase the efficiency for the fairness provided at the network layer. An option to make the network independent of the fairness issues at lower layers is to achieve fairness at the transport flow layer. A centralized max-main fairness approach for wireless mesh networks (WMNs) which strives to achieve end-to-end fairness at the transport layer is presented in [53]. The centralized solution discussed is justified for WMNs but it is not applicable for the WSN case whether it be a scalar or a video based WSN. There are studies that specifically address fairness in WSNs [54–56], among which Rangwala et al. proposes IFRC that combines fair bandwidth allocation with rate control [54]. In [57], feedback based congestion control mechanisms to enhance data delivery such as ESRT, CODA and SPEED [58–60] are classified as reactive and the authors come up with a collision-free scheduling that provides max-min fairness in a proactive and distributed manner. In that sense, our work also can be characterized as proactive. A similar work by Tassiulas et al. proposes a scheduling scheme which achieves max-min fairness without giving the implementation level details of the MAC protocol [61]. Vaidya *et al.* proposes a variant of 802.11 DCF MAC protocol which incorporates SCFQ [62] to achieve max-min fairness distributedly [63].

Among feedback based mechanisms, ESRT [58] is a transport protocol designed to guarantee the reliable event delivery and reduce the energy consumption. The designed protocol tries to carry the optimum number of packets from an event with a feedback mechanism from the sink to the nodes. However, the effectiveness of ESRT depends on the length of decision intervals ($\approx$ 10 sec) and the feedback latency. If the duration of the event is short as in surveillance applications and the feedback latency is high (network diameter is high), the notification may arrive to the source after the end of the event. Therefore the protocol cannot be able to avoid the congestion. Moreover, ESRT is not designed to decrease the reporting delay of the events. As mentioned in a recent survey on Least Attained Service (LAS) [64] and proven in [65] Shortest Remaining Processing Time (SRPT) is optimal for minimizing the mean response time. SRPT gives precedence to the jobs with shortest remaining time left by assuming that the queue dispatcher is aware of the residual size of the job that does not arrived yet.

However in blind systems as in WSNs although the job size may not be known, a job's age is always known therefore instead of SRPT, a more practical policy LAS scheduling is a better choice.

In the literature there are many studies on LAS scheduling in different names, such as Foreground-Background (FB) [66] and Shortest Elapsed Time (SET) [67]. Among them the performance of LAS with respect to the variability of the job size is analyzed in [68]. Authors indicate that while 99 per cent of the jobs encounter a reduced conditional mean slowdown under LAS, less than one per cent of the largest jobs experience a negligible increase of their conditional mean slow down. In [69], LAS is shown to outperform Processor Sharing (PS) with respect to mean response time and mean slowdown when the job size distribution has a decreasing failure rate. In another work [70] of the same authors, a classification of scheduling policies considering the unfairness is presented. They show that LAS is always unfair, since jobs whose sizes are greater than a certain size have higher mean response time under LAS than under PS. Furthermore, the effect of LAS on heavy-tailed traffic in wireless networks are presented in [71]. The authors compares LAS with Round Robin (RR) based scheduling and shows that LAS outperforms RR in a single bottleneck link and also in a one hop wireless shared link.

5.3.1. Motivation

When we focus on the contents of an event-flow, we observe that there is spatio-temporal redundancy among consecutive frames. This is mainly because the camera module of the sensor node takes continuous snapshots of the scene with a certain frame rate. It is not possible to generically define the number of frames to be received at the sink for healthy reception of the event. This depends on the type of detection method run on the back end. This could range from simple event detection in which only the existence of the event is notified to classification of the target or identification of the target. Also the frames received could be an input to an image recognition engine or to an human operator. Another factor is the specific positioning and movement of the target within the visual sensing range. A target closer to the camera module

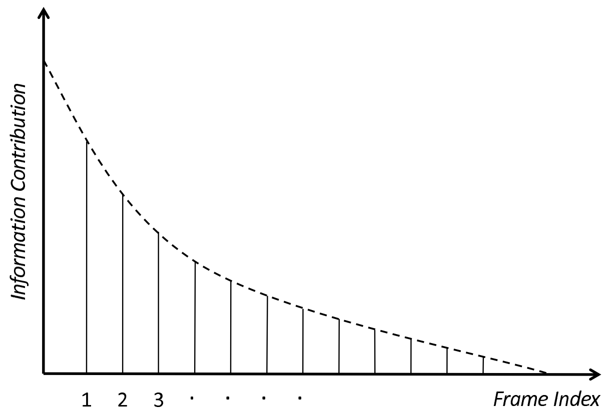


Figure 5.4. Information contribution of individual frames of a generic event-flow

takes a bigger portion of the picture, however assuming target is mobile, proximity to the sensor also implies shorter residence time inside sensing range, hence a shorter event-flow. Therefore we can crudely conclude that event-flows as they become larger, they contain more frames of the scene and likely to have more redundancy among frames. The information contribution of the individual frames of a generic event-flow is depicted in Figure 5.4.

With this observation in mind, we conclude that irrespective of the duration of the events initial frames of an event deserves special care. That is because they contain much of the visual information and also the delay experienced by them directly affects the reporting delay. We give priority to the frames via application level queue management. There are other methods available which can be used independently or in conjunction with our method such as adaptive contention window adjustment.

5.3.2. Round Robin Based Implementation

Round Robin based queue management implementation strives to give fair service to all events that are currently enqueued in a VSSN node. A node can uniquely identify the frames in its queue according to the events they belong to by using the source node id and the local event id assigned by the source node. The queue is composed of frames received from the network for relaying purposes and the frames received from the application layer, i.e. the video frames produced by the node itself. RR operates by servicing frames of events in a round robin manner, one frame from each event at a time. Internally, RR dynamically forms logical queues for each event and gives service to each queue in a time-shared manner. The duration in which all event queues are served once is called an *epoch*. During an epoch, the available bandwidth is equally divided among each event. The overall service rate an event gets from RR depends on the length of the event (in terms of frames) occupied in the main queue, the total number of events in the queue, the length of each event in the queue and the congestion level experienced at the MAC level (available effective bandwidth). When the incoming frames are more than the capacity of the node, buffer overflow occurs. In that case, RR drops the frame from the longest event queue. With this scheme a received frame that arrives at the full main buffer need not be dropped unless it belongs to the event that currently has the longest logical queue. When compared with the FCFS behavior, RR provides fair bandwidth allocation to events and also gives priority to events with fewer frames. This latter property is especially more pronounced in the case of buffer overflows in which frames of events with longer queues are dropped. In that sense, RR tries to homogenize the service rate among events according to the snapshot of the queue.

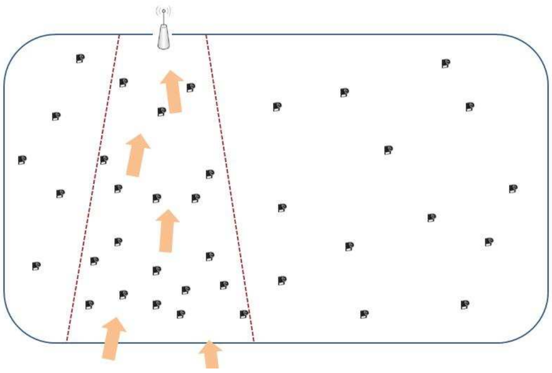
One point to note in the above discussion is that the queue manipulation is done in terms of frames and not packets. Normally, it is straightforward to process a queue in terms of frames as each packet has a packet identifier, however there is no guarantee that a frame will be received completely from the neighbors due to packet drops. In our VSSN implementation, *SMAC* [49] with *Message Passing* feature is used as the MAC layer. Message Passing allows frames to be passed among nodes intact which makes our assumption about frame based queue manipulation possible.

5.3.3. LAS Based Implementation

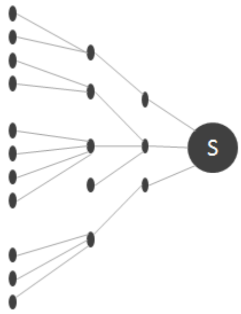
The main idea behind LAS Based implementation is that an event is a sequence of frames flowing in the network and at a specific time instance, only a portion of it may be contained in the buffer of a VSSN node. This is due to the buffer size limitations and earlier frame drops that an event may experience. RR operates on the instantaneous snapshot of the buffer and provide fairness among events according to what is currently present. In this respect, a way to provide better fairness among events is to consider not only the current buffer composition but also to take into account the frames of an event that has been relayed previously. LAS, like RR, forms logical queues of frames per event and service one frame from each queue in an epoch in a round robin fashion. However, unlike the RR implementation, LAS keeps track of the sent frames and inserts a virtual frame to the event queues as place holders for each frame of an event that is relayed. Therefore, a logical queue for an event contains both real frames that are waiting to be send and virtual frames that are already sent. In every logical event queue virtual frames are placed in the front of the queue, therefore, when deciding on the next frame to get relayed, LAS, gives explicit priority to events that have fewer frames sent.

5.3.4. Experimental Setup

We examine the effect of RR and LAS using the OPNET simulation environment [39]. In order to observe the improvements on the reporting latency and the video quality of events in detail, a border surveillance scenario is constructed, in which intruders follows a favorite path called Trespassers' Favorite Path(TFP) [72]. These paths (Figure 5.5(a)) are practical preference for intruders because of the geographical topology, checkpoint locations etc. Since they are frequently used, the sensors should be more densely deployed with respect to the rest of the border and consequently the traffic is mostly generated by them. Therefore we have simulated only the TFP part of the border as presented in Figure 5.5(b). In video surveillance applications the volume of the data traffic is related to the dwell time of the target, the camera frame rate and the compression algorithm. The duration of an event and number of frames varies according to these parameters. In our tests we represent the variation in the number of frames with Normal distribution and assume that the interarrival time between the events is Exponentially distributed. Furthermore, frames are generated in uniform intervals according to the camera frame rate (Figure 5.1). Since in WSNs, a typical practice for minimizing the energy consumption is by introducing a sleep schedule, we performed our tests on SMAC [49] with different duty cycle values. The list of the simulation parameters are presented in Table 5.1.



(a) Trespassers' Favorite Path (TFP)



(b) Detailed view of the TFP multi-hop simulation scenario

Figure 5.5. Deployment scenario where intruders follow the favorite path in which the sensors are deployed more densely

5.3.5. Results

A histogram summarizing the events according to the number of successfully received frames at the sink is presented in Figure 5.6. Out of the 2364 events generated, for FCFS case, for instance, around 130 events are reported only with a single frame whereas around 100 events are reported with 14 frames. The number of events reported for all queueing mechanisms are close to each other 2133, 2220 and 2215 for FCFS, RR and LAS respectively. However, it is observed that the variance in the frequency of the frames per event is decreased by RR and LAS. In the one to three fps interval and $10-20$ fps interval, the number of events are less in RR and LAS case than FCFS since LAS and RR decreases the over reported events and share the available excess bandwidth among the under reported events. Thus, most of the events are reported similarly which is due to the fair treatment of frames according to the related events.

Total number of frames required to be received at the sink in order a triggered event to be considered as *detected* depends on the application. However, as previously pointed out, it is the initial frames of an event that contribute more to the visual information received at sink. In order to observe the effect of LAS and RR on reliable event reporting, we plot the ratio of missed events in Figure 5.7. As expected, when the required frames per event is increased, missed event ratio also increases in all

Table 5.1. Simulation parameters

Parameter	**Value**
Event Size	Normal distributed with ($\mu = 15$,$\sigma^2 = 7$) Frames
Video Frame Size	10 Kbits
Packet Size	1 KBits
Frame Interarrival Time	Uniformly distributed with $\mu = 1/3$ sec
Event Interarrival Time	Exponentially distributed with $\mu = 25$ sec
Duty Cycle	$10, 20^*, 30, 40$ (per cent)
Bandwidth	250 Kbps
Buffer size	100 Kbits
MAC layer	SMAC [49] with Message Passing feature

(*) Unless otherwise specified, 20 per cent duty cycle is the default in the experiments.

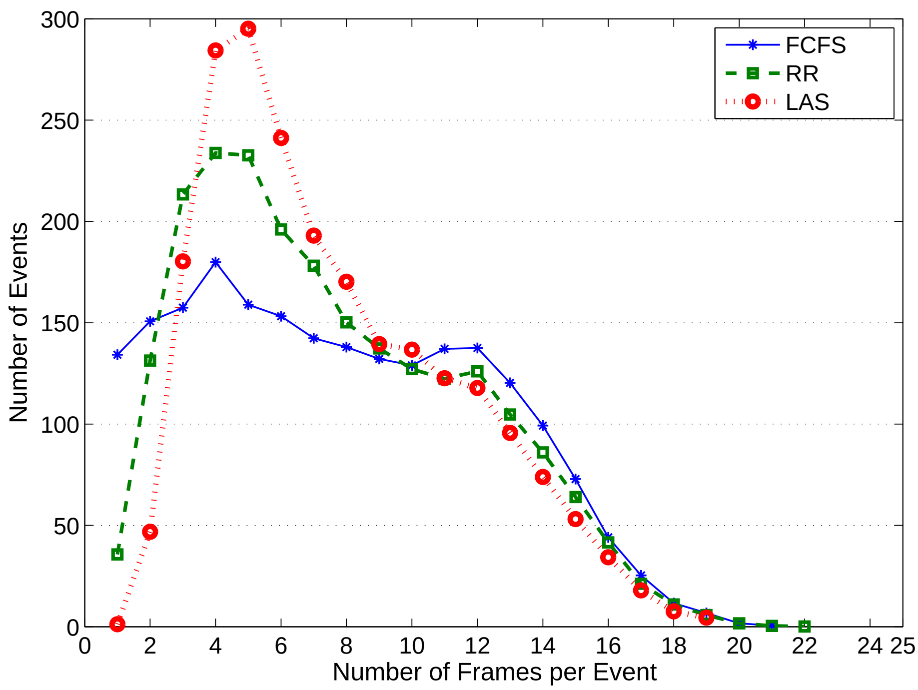


Figure 5.6. Histogram of the number of frames received at the sink for each event

queueing techniques. However, the ratio of missed events are clearly less in RR and LAS cases compared to the FCFS case. For instance, when the required frames for event identification is set to four, while FCFS misses 28 per cent of the events, RR misses 22 per cent and LAS misses less than 16 per cent. Additionally, the difference

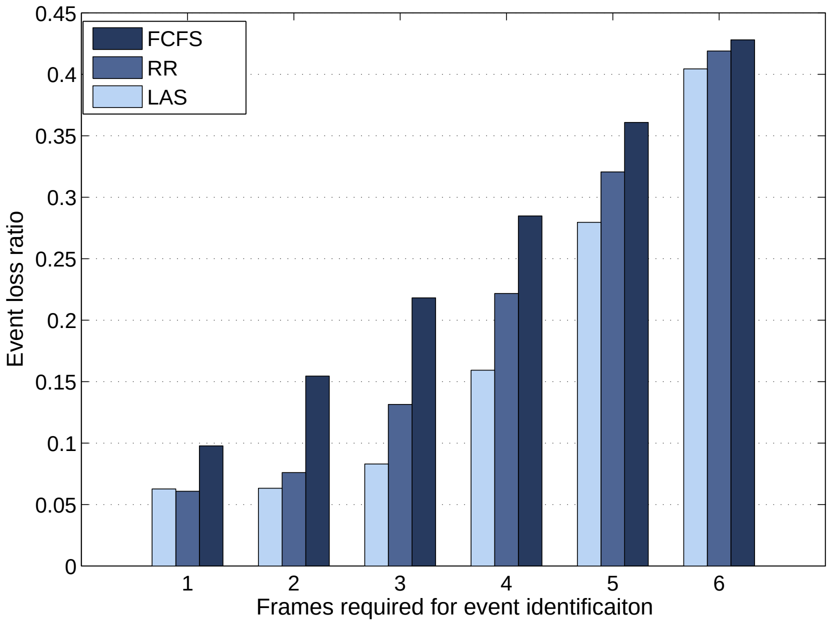


Figure 5.7. Missed event ratios with different number of frames required for the identification

between the FCFS, RR and LAS systems decreases as the number of required frames increases. The reason is that LAS punishes the large events while giving precedence to smaller ones especially when the network load becomes high.

Besides the video quality, the reporting latency of the events are also important. Especially the frame of an event which arrives first to the sink has the most significant contribution for the event reporting since it makes the sink aware of that event. Figure 5.8 presents the average of the first frame latency of the events with various duty cycles and indicates that RR and LAS improve the event reporting delay significantly with compared with FCFS.

To have a more general understanding on the latency behavior of the events, Figure 5.9 depicts the average delay a certain frame of an event experiences, e.g. the average latency of the 8^{th} frame of the events. It is observed that LAS decreases the delay for all frames of the events, whereas RR performs better than FCFS up until the 4^{th} frames of the events. In other words LAS leverages the mean response time of the events.

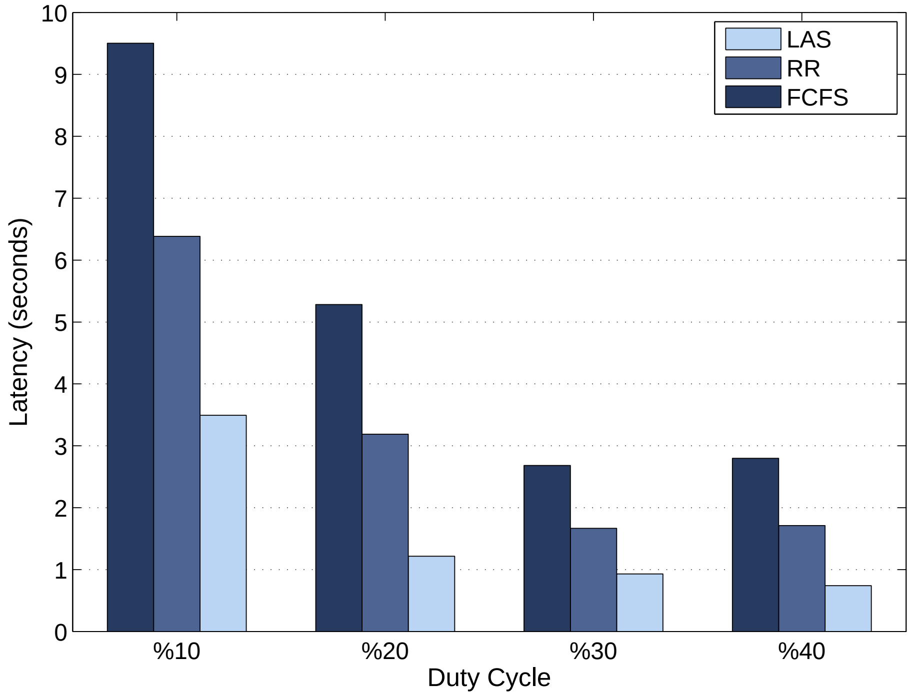


Figure 5.8. Average delay for the first arriving frame of an event using various duty cycles

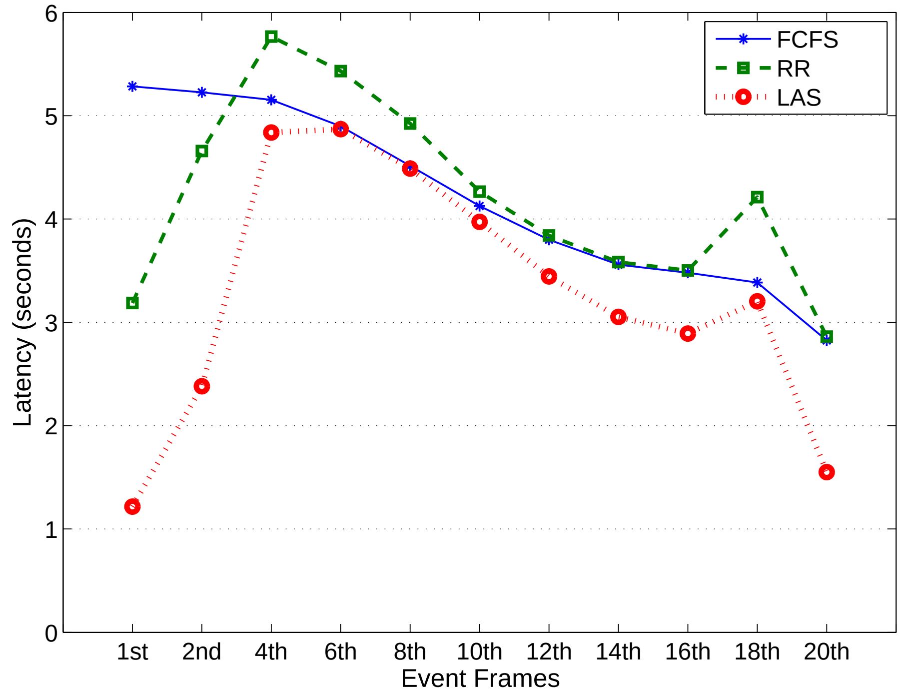
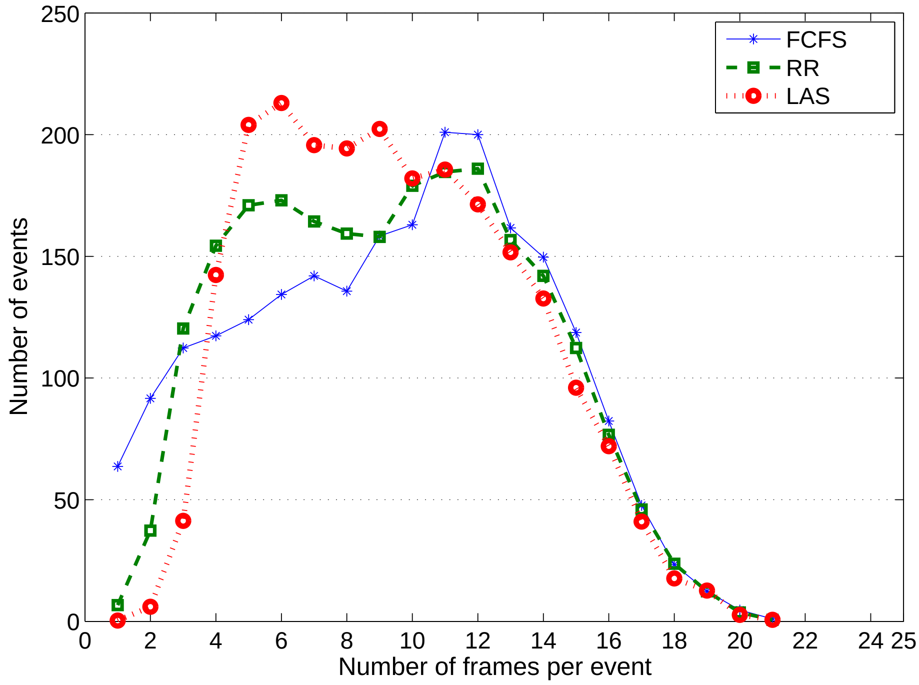
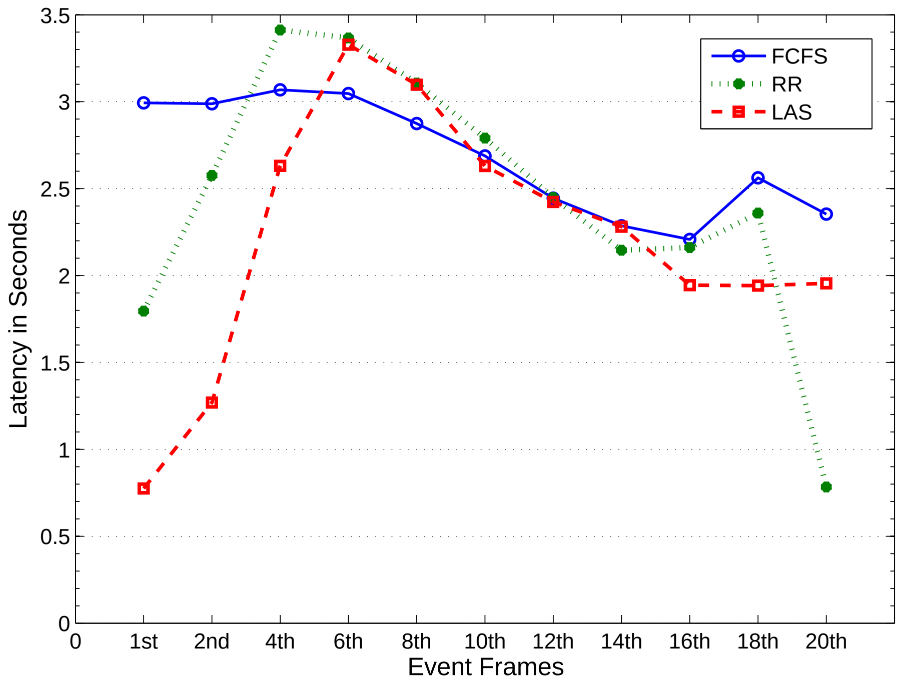


Figure 5.9. Average latency of frames



(a) Histogram of the number of frames reached to sink for each event



(b) Average latency of frames

Figure 5.10. Latency and histogram of the events (40 per cent duty cycle)

On the other hand, when the duty cycle is increased most of the frames arrive to the sink. Therefore as in Figure 5.10(a), the difference between the intelligent queueing techniques and the FCFS decreases. However, as observed in 5.10(b), compared with FCFS, RR and LAS can relay all frames without a significant sacrifice in terms of latency.

5.4. Contributions of Event Based Queueing

In this chapter, an event based queue management scheme for Video Surveillance Sensor Networks (VSSNs) is introduced. Main goals of the scheme are to increase the event reception ratio at the sink node and to decrease the initial reporting delay of the events. There is inherently high volume of traffic in a VSSN when an intruder(s) is present. Most of the time the traffic produced is more than that can effectively be carried by the network. To be able to achieve its design goals, the event based queue management scheme defines an *event*, as opposed to a video *frame* or a network *packet*, to be the logical messaging unit in a VSSN and introduces the fair treatment of events which are application level entities. Two different implementations, namely RR and LAS, are compared with the legacy FCFS style queue management. The proposed scheme is shown to not only increase the number of events that are reported properly but also to lower the initial reporting delay considerably. As a future work, we plan to implement complementary solutions that are implemented in the routing and the transport layer to further enhance VSSN functionality.

6. UNDERSTANDING THE LIFETIME BEHAVIOR OF THE VIDEO SURVEILLANCE SENSOR NETWORKS

Challenges of conveying video traffic over WSNs and enhancing their performance using event based techniques are explored in Chapter 4 and Chapter 5, respectively. In this chapter, we explicitly focus on the lifetime behavior of the Video Surveillance Sensor Networks (VSSNs). Two facts about the VSSNs that require further focus are the increased traffic load compared to scalar sensor networks and the necessity to consider events as opposed to image frames or network packets as the logical messaging unit. To be able to effectively quantify the lifetime behavior of a VSSN, both factors should be taken into account. In this chapter, first, we perform lifetime measurements using the legacy metrics based on the number of sensors and the sensing coverage. Later, to realistically quantify the lifetime, we present a new utility indicator called Instantaneous Event Delivery Ratio (IEDR). We show that legacy utility indicators do not suffice to quantify the lifetime of a VSSN. We obtained sound lifetime results when the utility function is expressed in terms of both the sensing coverage and IEDR.

6.1. A Surveillance Scenario for Video Sensor Networks

In this chapter, we focus on an outdoor surveillance application scenario in which fixed amount of target nodes randomly wander inside the monitored geographical area. Sensor nodes are equipped with camera modules where the image resolution and encoding method are as reported in Section 4.2. We assume a single sink node which is computationally and energy-wise unconstrained. An event is characterized by the visual assessment of the images taken by the camera module of the sensors. Due to the resource scarcities, it is not feasible for a node to identify an event in detail. Rather, we expect a node to perform an elementary processing to filter out the noise from an actual physical presence in the scene. It is the sink node that receives the image sequences sent by the nodes and performs classification/recognition. The ultimate decision about a local event can therefore only be given in the sink node. For a VSSN there are various methods for the nodes to detect an event. One means is to perform simple image processing such as background subtraction on the nodes. Alternatively, a low energy scalar WSN can be overlaid for presence detection purposes, which triggers video transmission in nearby VSSN nodes. We assume that a target entering the sensing area of a sensor node triggers an event. The sensing area of a node is determined by the sensing radius $r_{sensing}$ as well as by the field of view of the camera lenses. The values of the relevant experiment parameters used for the performance evaluation are summarized in Table 6.1.

Table 6.1. Simulation parameters

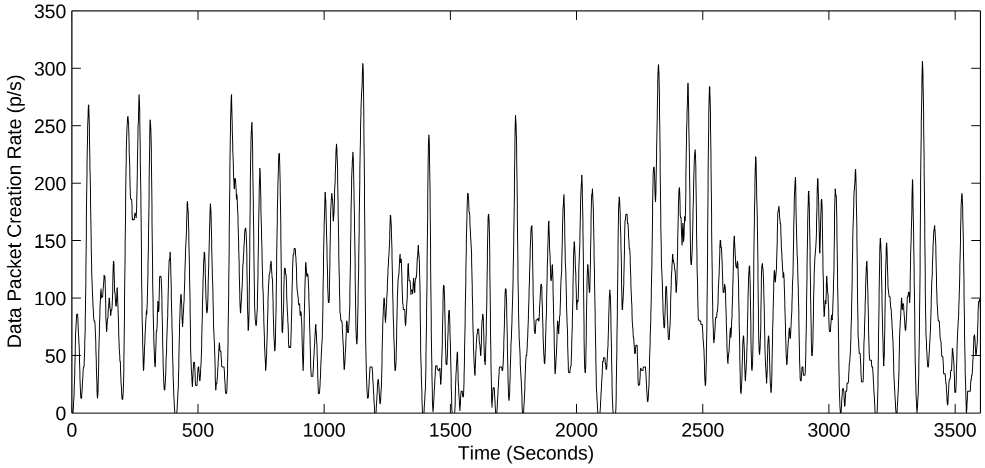
Paremeter	**Value**
Number of targets	0,3,6,9
Target Speed	0.33 m/s, 3m/s
Surveillance Area	400 x 400 m^2
Network Size	180 Nodes
Deployment Type	Uniform random
Video Frame Size (compressed)	10 Kbits
Packet Size	1 KBits
Camera Frame Rate	4 fps
Field of View	150°
Camera Detection Range	40 m
Bandwidth	250 Kbps
Buffer size	20 Kbits
Target Mobility Model	Random Waypoint
Initial Node Energy	1.5 J
Communication Stack	SMAC, GPSR
MAC Sleep Schedule Duty Cycle	20 per cent (Awake)

6.1.1. On the Scalability of the Simulation Experiments

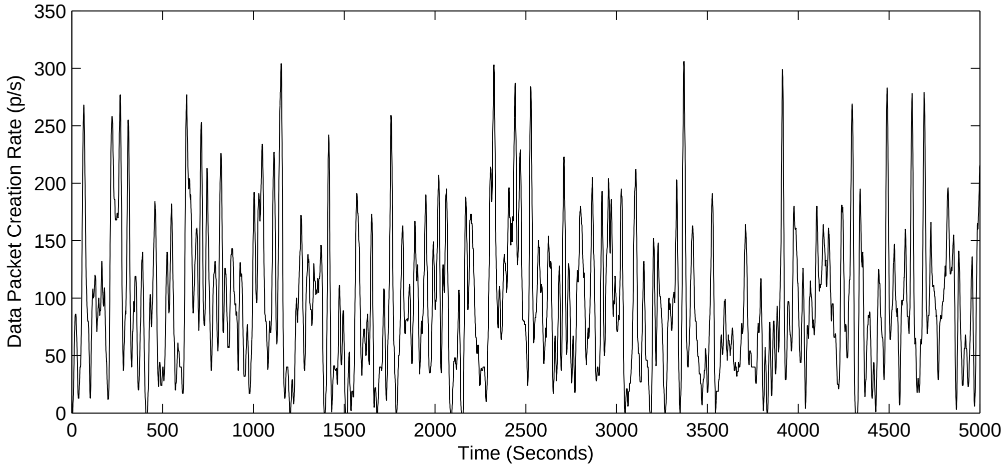
We use the detailed packet level simulation capabilities offered in OPNET Modeler to carry out the simulations. Our experiment code performs a detailed VSSN operation in which physical level collision detection, virtual cluster based MAC protocol with synchronization messaging and communication sleep schedule, a greedy geometric routing, a network size of 180 nodes and up to nine targets with mobility models are implemented according to the realistic settings. This fact combined with the repetitions required for statistical significance increases the running times considerably. Therefore, to decrease the running time of the simulation experiments, a scaled down battery capacity of 1.5 joules is employed as the initial energy of the sensor nodes. Such a scaling down does not alter the relative magnitudes and trends of the results we obtain because of the traffic generation pattern triggered by the target nodes. The target nodes start moving inside the monitored area triggering event based traffic just at the beginning of the simulation. Therefore, the generated data traffic swiftly saturates the nodes and we observe a random but repetitive pattern for the generated traffic across the network. The traffic related patterns do not get altered for the elongated simulation runs. As an example, Figure 6.1 denotes the traffic generation rate for the simulation runs with nodes assigned an initial energy level of 15 Joules and 50 Joules. The pattern observed is similar for other initial battery capacity assignments that are over 1.5 Joules.

6.2. Lifetime Evaluation Based on the Number of Nodes and the Sensing Coverage

Decreasing number of nodes and the sensing coverage are used by a large group of researchers as indicators of the functionality provided by a sensor network. For certain application scenarios, such metrics can be applicable when tuned correctly according to the application. In this section, we explore the compatibility of such legacy lifetime metrics with the VSSNs. The lifetime metrics used are the FND and the WCOT in which the sensing coverage is employed as the utility indicator.



(a)



(b)

Figure 6.1. Traffic generation pattern for extended initial energy assignment of (a) 15 Joules (b) 50 joules

6.2.1. Coverage Based Utility Function for WCOT

As thoroughly discussed previously, in the WCOT framework, the utility function is the means for the user of the network to formally describe the eventually decreasing network functionality in terms of the relevant utility indicators. Specific to the experiment setup carried out in this section, we define the utility function in terms of the sensing coverage as shown in Figure 6.2.

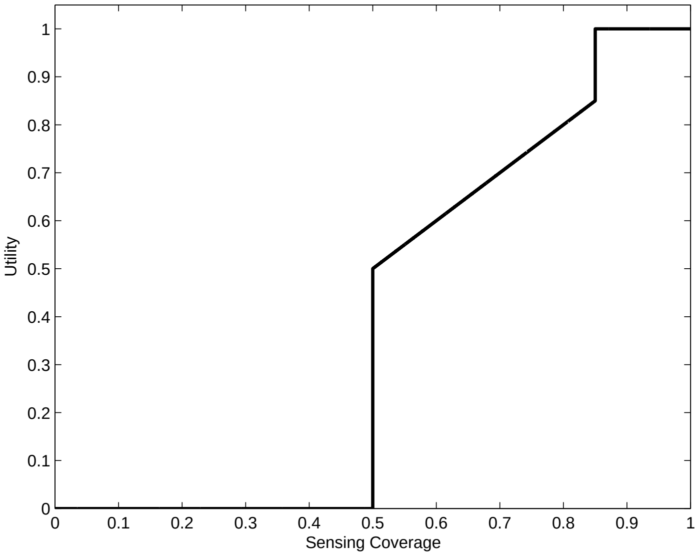


Figure 6.2. Utility function based on the sensing coverage

As an alternative expression, the utility function can be described as a partial function in terms of the sensing coverage, δ, as in Equation 6.1. The reader should note that the utility drops to zero for all coverage values smaller than 0.5. The reason behind this choice is that our scenario is an event triggered scenario in which the target nodes have mobility.

$$U_\delta = \begin{cases} 1 & 1 \geq \delta \geq 0.85 \\ \delta & 0.85 > \delta \geq 0.5 \\ 0 & 0.5 > \delta \geq 0 \end{cases} \tag{6.1}$$

As δ drops during the network operation, the targets experience longer durations of undetected presence, however, they still pass through the sensor nodes' sensing area though less occasionally. Therefore, up to a certain threshold value, decreasing sensing coverage amounts to getting less information intermittently from a target but not necessarily losing its track which is captured in our utility function by the linear utility drop when $\delta \in [0.5,0.85)$.

6.2.2. Effect of the Target Population Size and Mobility on the Lifetime

In Chapter 4, the feasible operational parameters were sought for a generic video sensor network application. Now, we explore how this tuned video conveying infrastructure behaves in a more specific application scenario, namely the video surveillance. Here, we vary the number of targets moving in the monitored area and their mobility levels. By doing so, the created traffic load, the number of events observed and their physical distribution among the monitored area are varied. The simulation parameters used in the experiments are depicted in Table 6.1. OPNET Modeler is used in the simulation experiments in which packet level simulations are carried out with 10 repetitions for each data point presented. The lifetime results obtained are shown in Figure 6.3 and Figure 6.4. Both figures depict a positive correlation between the number of targets in the area and the network lifetime.

To further understand dynamics of the network, Figure 6.5(c) presents the traffic load created in the network and Figure 6.5 shows how the individual sensor death times and the sensing coverage of the network are altered for the cases studied. It is clearly put forward in Figure 6.5(a) that irrespective of the metric, the lifetime of the individual nodes in the network are elongated as a result of the increased traffic load. This is a counter intuitive behavior for sensor networks in which the packet transmission and reception are the primary sources of energy consumption. We term this unexpected phenomenon as Traffic Triggered Lifetime Extension (TTLE) and dedicate a separate section for exploring its root causes.

Revisiting Figure 6.3 and Figure 6.4, to conclude that FND and coverage based approaches realistically quantify the VSSN network lifetime, we need to assess whether they actually measure the application specific functionality offered. Primary function-

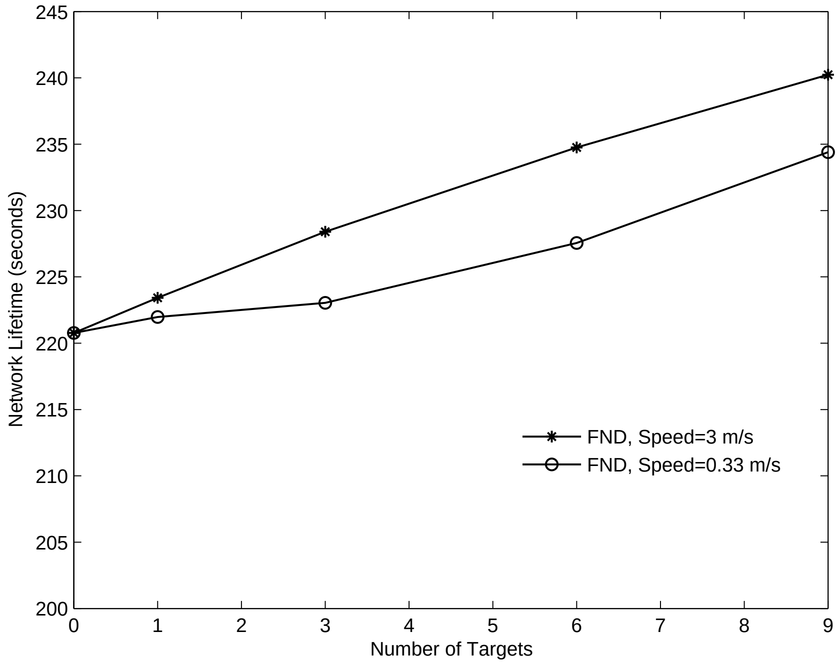


Figure 6.3. The effect of the number of targets and the mobility on the network lifetime when the metric is FND

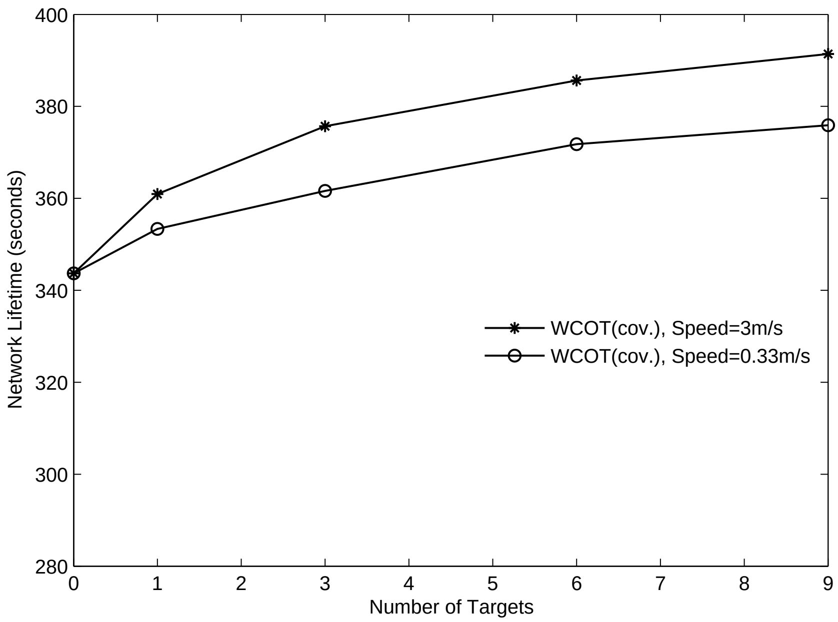


Figure 6.4. The effect of the number of targets and the mobility on the network lifetime when the metric is coverage based WCOT

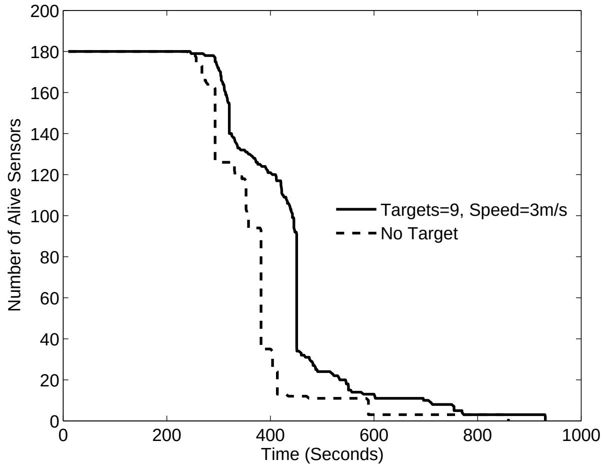
ality of the video surveillance sensor networks is the event reporting using visual data. Therefore, a sound lifetime quantification metric should produce results correlated to the level of event reporting performed by the system. Figure 6.6 shows the event delivery behavior observed in the experiments. When focused on event delivery, it can be concluded that the functionality of the network deteriorates as the traffic load increases. In this respect, the metrics used in the experiment settings do not capture the actual lifetime behavior of a VSSN and draw misleading conclusions.

6.3. Traffic Triggered Lifetime Extension (TTLE) Phenomena due to the Disseminated Sleep Schedule Pattern

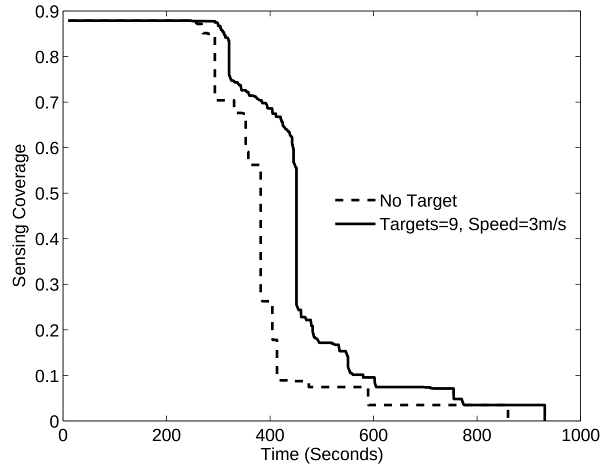
To understand the causes of the TTLE phenomenon, we need to see, at the individual node level, why the increased traffic load positively affects the lifetime. The reader should note that this examination should be carried out in the context of the employed MAC protocol, which is SMAC for our experiment settings. SMAC is one of the earliest works in sensor networks on introducing a sleep schedule for the energy efficiency. Similar approaches are used after SMAC, which are either enhancements of the same concept or modifications to the original proposal. To conform to the strict energy efficiency requirements of a typical WSN, employing a communication duty cycle is a logical choice. However, this may complicate the relations between the various network dynamics and therefore its effect may be different than mere scaling down the global energy consumption.

6.3.1. Effective SMAC Communication Duty Cycle Behavior

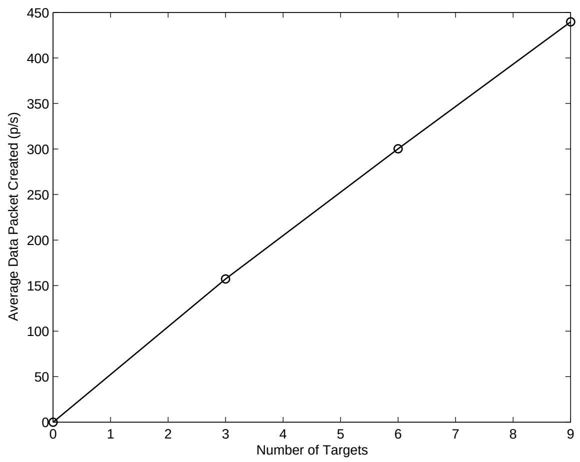
In SMAC operation, nodes have a *listen* period and a *sleep* period which together constitute the SMAC period, T_{SMAC}. Figure 6.7(a) depicts sample communication sleep schedule patterns. Technically, the listen period is further composed of two sub parts in which the synchronization messaging and the data messaging occur. However, for the scope of the on going discussion, we consider the listen period as a monolithic duration in which the node is in high energy consumption mode. The durations of both the listen and the sleep periods are fixed for all nodes in the network and a loose



(a)

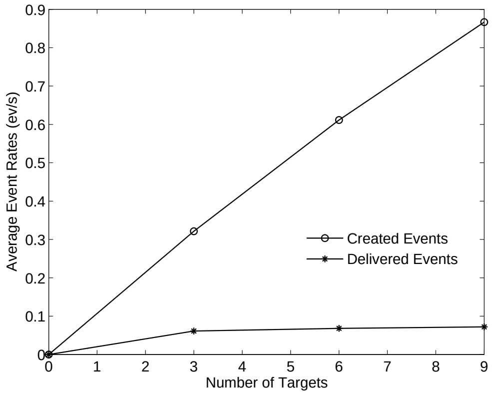


(b)

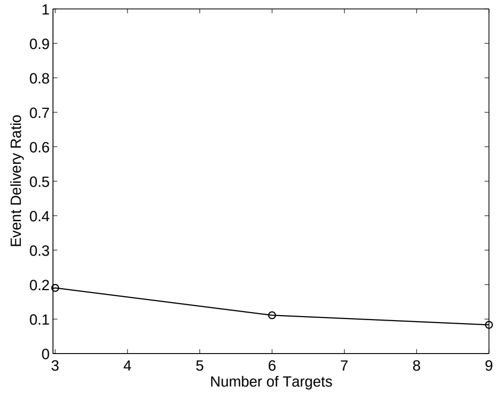


(c)

Figure 6.5. (a) Alive sensors in time (b) Sensing coverage in time (c) Aggregate traffic load (Target speed = 3 m/s)



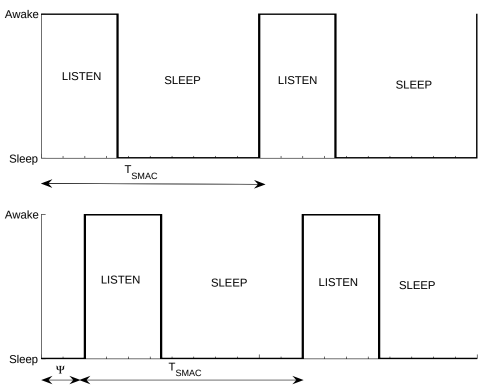
(a)



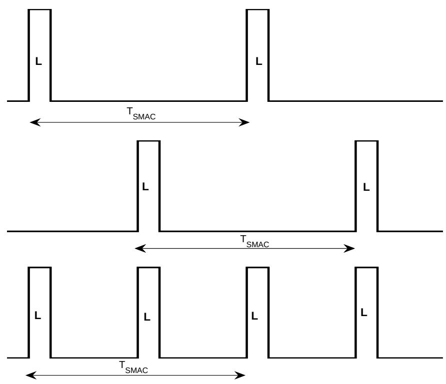
(b)

Figure 6.6. Event delivery for target speed = 3m/s (a) Created and delivered events (b) Event delivery ratio

time synchronization is maintained via periodic SYNC messages. The base duty cycle, ϕ_{base}, of any node equals to t_{listen}/T_{SMAC}.



(a)



(b)

Figure 6.7. (a)Listen offset Ψ, $\Psi \in [0, T_{SMAC})$ (b) Single and double sleep schedule patterns

<u>6.3.1.1. SMAC Listen Offset and Its Dissemination.</u> Despite fixed duty cycle and communication period, nodes do not necessarily start their listen periods simultaneously. The time difference between the starting points of two nodes' SMAC periods according to an assumed time frame of reference is called the *listen offset*, Ψ, as shown in Fig-

ure 6.7(a). Any two nodes in the network,to communicate, should obey the same sleep schedule pattern, i.e., they should have the same listen offset values. A self-organized listen offset dissemination procedure is employed in SMAC in which a node selects a random offset during the setup and waits for a random amount of time to broadcast its listen offset. If a node hears an offset broadcast message before announcing its own, it cancels the broadcast and records the acquired offset value and broadcasts it. Once the listen offset is announced by a node (either its own or the acquired offset) it becomes fixed for that node. The neighboring nodes, thus, have the necessary information about when to start data contention when they have data destined to the node. Collection of nodes that have both physical connectivity and obey to the same sleep schedule, i.e., have the same listen offset, are said to form a *virtual cluster*. A node may hear offset broadcasts after its listen offset has been fixed. In that case, the node adopts to both sleep schedules, that is, it has multiple listen offset values. This is the case for nodes that are located between virtual cluster boundaries, though, this is not the only possibility for a node to end up having multiple sleep schedules. Figure 6.7(b) presents example sleep schedule patterns belonging to nodes with single and multiple listen offset values. Clearly, nodes having multiple schedules have accordingly increased energy expenditure rates. Figure 6.8 shows how the random sleep schedule dissemination algorithm results in nodes with different sleep schedule patterns. This means that irrespective of the base communication duty cycle imposed by SMAC, the network ends up with nodes having various energy consumption characteristics. The sleep pattern drawn for the nodes in Fig 6.8 belong to a network with SMAC duty cycle value of 20 per cent, however due to the multiple schedules their effective duty cycle is higher. In the figure, the effective duty cycle of the nodes 43 and 31 are 35 per cent and 57 per cent respectively.

6.3.1.2. Multiple Sleep Schedules, Traffic Load and Node Lifetime. Assuming similar power dissipation rates for the the idle, transmit and receive states, the lifetime of a node with single sleep schedule is not affected by the varying traffic load. That is because during the listen period, a node performs one of the three operations. For a node with multiple sleep schedules, however, packet transmission started after one

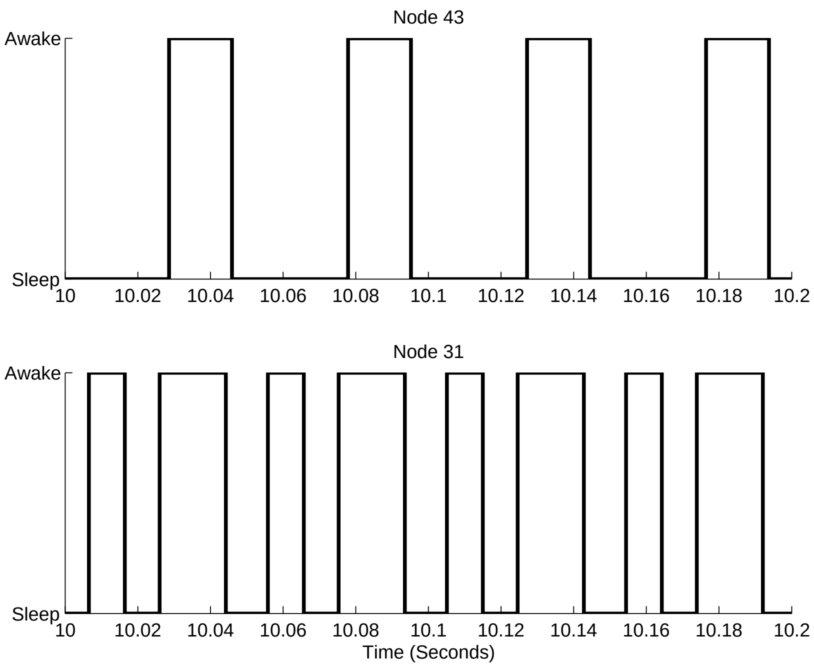


Figure 6.8. The sleep schedules on two different nodes for the no traffic scenario

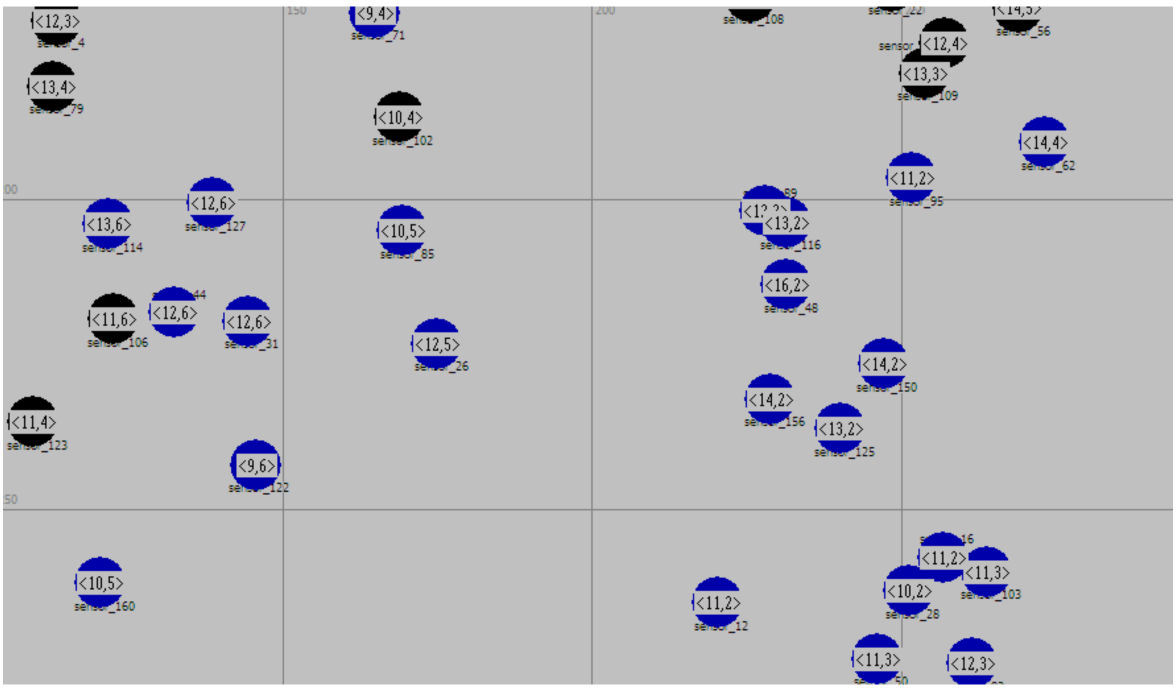


Figure 6.9. A sectional view of an OPNET animation where labels $< i, j >$ denote the node degree and the number of sleep schedules, respectively

listen period may suppress wakeups at the next listen periods due to the NAV (Network Allocation Vector) being announced. Therefore, the traffic carried over a node directly affects the energy consumption rate, hence the lifetime, of its neighbors. The node energy level graph presented in Figure 6.10 depicts a node's behavior under different traffic loads. Compared with the *no traffic* case, the node has a lower consumption rate due to the change in its effective duty cycle. The changing effective duty cycle of the nodes under different traffic loads is the basic mechanism behind the TTLE phenomena.

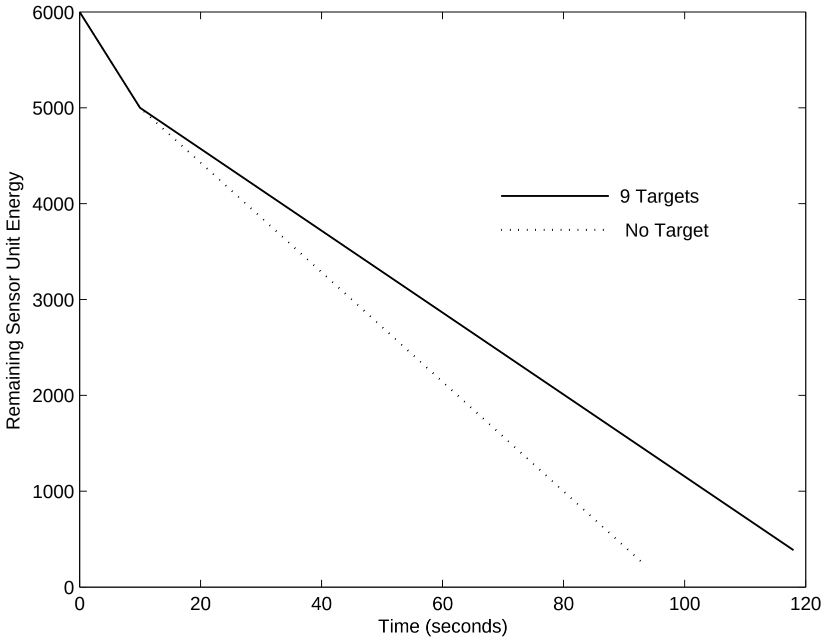


Figure 6.10. Energy consumption rate of a specific node under two different traffic loads

6.4. Event Delivery Based Lifetime Assessment for Video Surveillance Sensor Networks

6.4.1. Event Delivery Model

To assess the event delivery performance of a VSSN, we need a criteria about when to assume an event created in monitored geographical area is successfully delivered at the sink node. The criteria can be expressed in terms of the image frames received for a specific event and may depend on the performance of the recognition engine as well as the level of detection and recognition accuracy required for the application. Accordingly, we classify the possible application types and their minimum required frame deliveries in Table 6.2. The reader should note that the minimum frame requirements presented are generic values and further tests with real recognition algorithms should be carried out for more accurate requirement sets. The real scope of this document is on how to perform the lifetime assessment given such application level requirements. Nevertheless, it is apparent that the presented application category types require increasing minimum number of frames as their type number (e.g., TYPE 0, 1, 2) increases which is captured in Table 6.2.

Table 6.2. VSSN application categories and their minimum frame requirements.

Application Type	**Category**	**Description**	**Frames Required**
TYPE 0	Basic Presence Detection	Deducing the presence of an alien in the area	1
TYPE 1	Crude Classification	Classification among already defined sets	2
TYPE 2	Target Identification	Specify the specific instance of the target	5

6.4.2. Instantaneous Event Delivery as a Network Utility Indicator

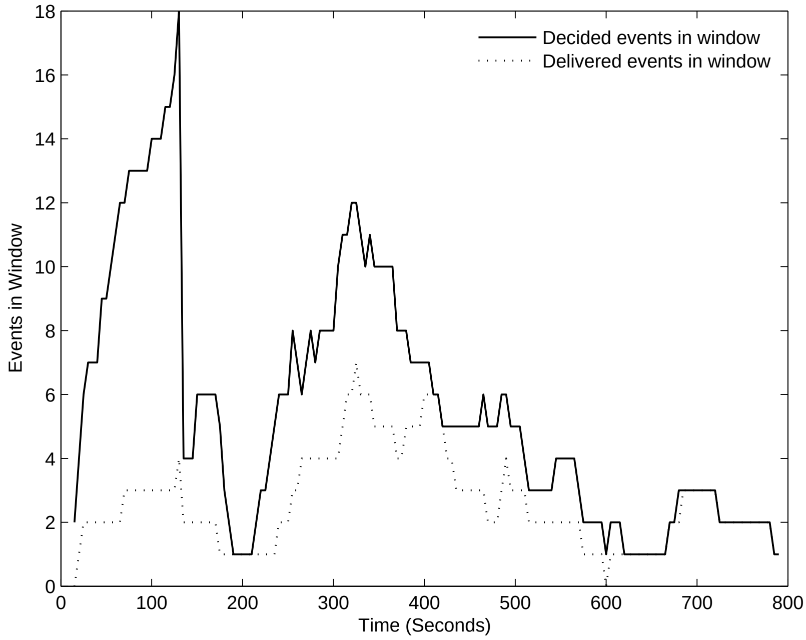
For the VSSN case, we show that the sensing coverage based utility does not reflect the actual utility of the network. Event delivery is a good candidate but, as presented in Figure 6.6, the average event delivery ratio gives information about the whole network operation duration. However, to be able to employ event delivery figures in WCOT, we need the instantaneous event delivery performance of the network so that we can incorporate it as the weight for the the operational time. The working principal behind WCOT for converting utility into a lifetime figure is via the weighted summation where the utility values are the weights. Section 2.4.2 presents the mechanical procedure on how WCOT employs utility values. To be able to obtain the event delivery performance of the network for smaller durations as opposed to the whole lifetime, we define a time window and periodically check the events delivered in the window to obtain the IEDR of the network. Average event delivery information is given in Figure 6.6.

6.4.3. A Realistic Utility Function for VSSNs

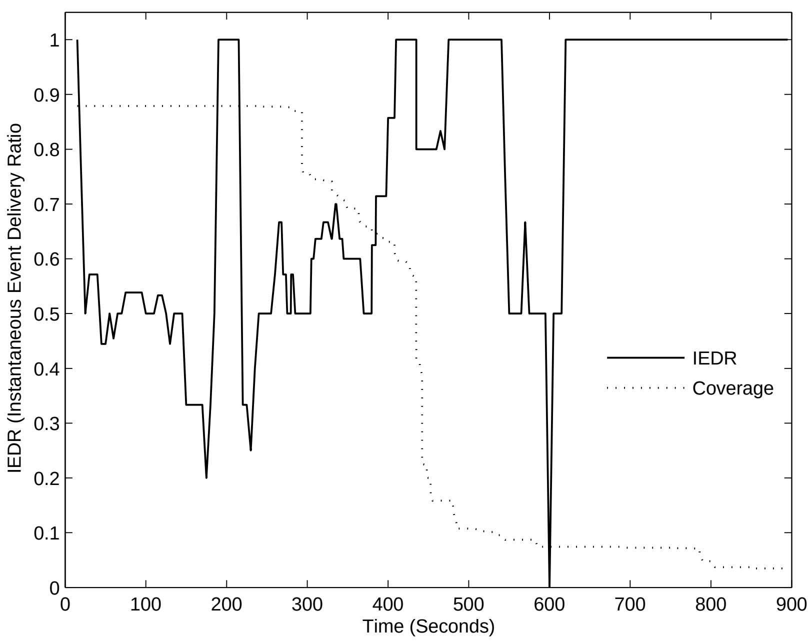
IEDR is appropriate for assessing the utility of small periods of the network operation in terms of event delivery. However, IEDR, by itself does not capture the actual utility of the network as it focuses on the readily created events in the network not on the possible events that could be created in the whole network. To exemplify, we co-present the IEDR and the sensing coverage values in Figure 6.11(b). Here, the coverage abruptly drops below 0.5 around t=450 seconds which indicates the death of the network. However, IEDR values get even better after this point. The reason for this behavior is that IEDR presents the ratio between delivered and created events in a specified time window which is equal to 120 seconds for the graphs depicted. Therefore, it is possible to observe high IEDR values in a poorly covered VSSN. To reflect the actual utility of the network, we need to express it as a function of the two independent variables being the IEDR and the sensing coverage. Figure 6.13 shows the utility function that is employed for the performance evaluation carried out in this section. It is obtained by the multiplication of the individual utility functions given in Figure 6.12.

6.4.4. Lifetime Results

Figure 6.14 shows the cumulative lifetime results for the target population sizes and the mobility values tested. Unlike FND and coverage based WCOT, IEDR and coverage based WCOT indicate a decreasing network lifetime for increasing traffic load in the network. This is in accordance with the average event delivery figures reported in Figure 6.6. Increased target population triggers more traffic which further causes more packet drops to be experienced in the network. For any given target population size, increased mobility also deteriorates the network performance and results in less events to be received at the sink. Mobility spreads the created traffic around the network keeping the nodes in constant contention mode and causing accumulations in the buffers

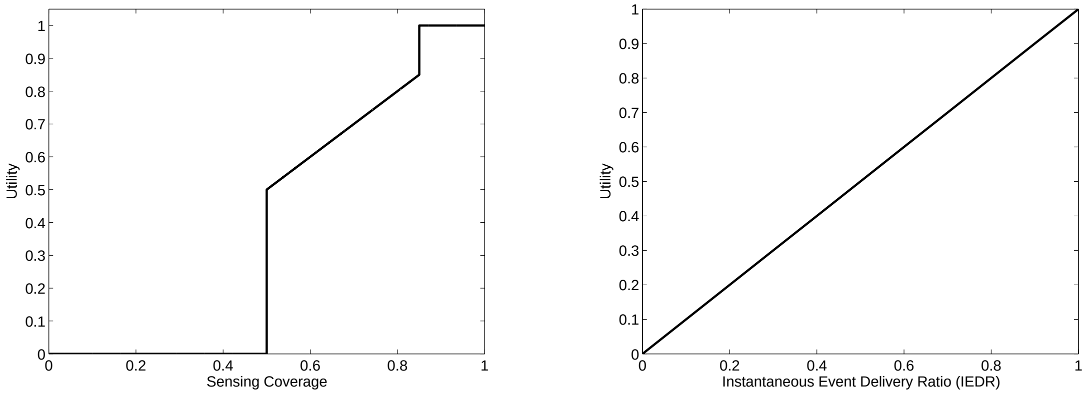


(a)



(b)

Figure 6.11. Dynamic network behavior in terms of (a) decided and delivered events in window (b) IEDR and sensing coverage



(a) (b)

Figure 6.12. Utility functions in terms of (a) the sensing coverage (b) IEDR

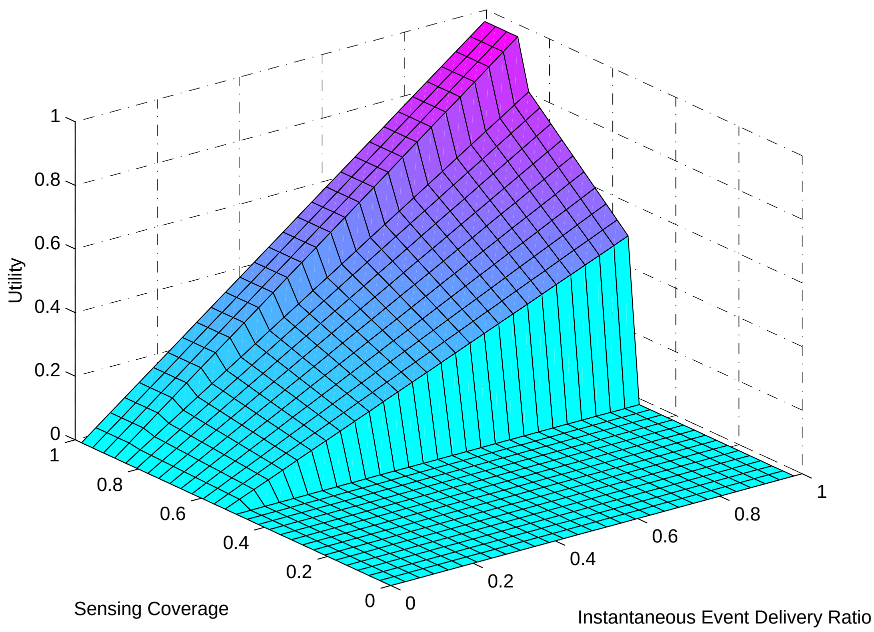
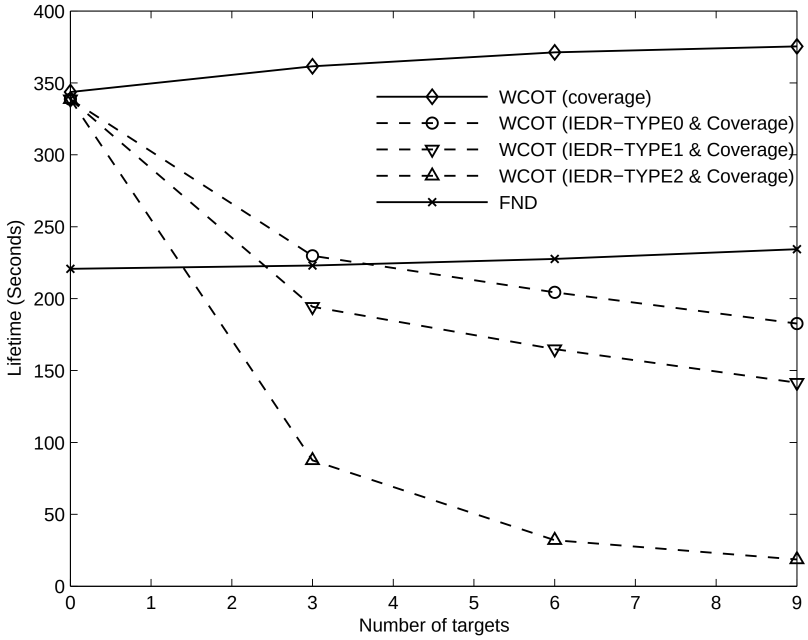
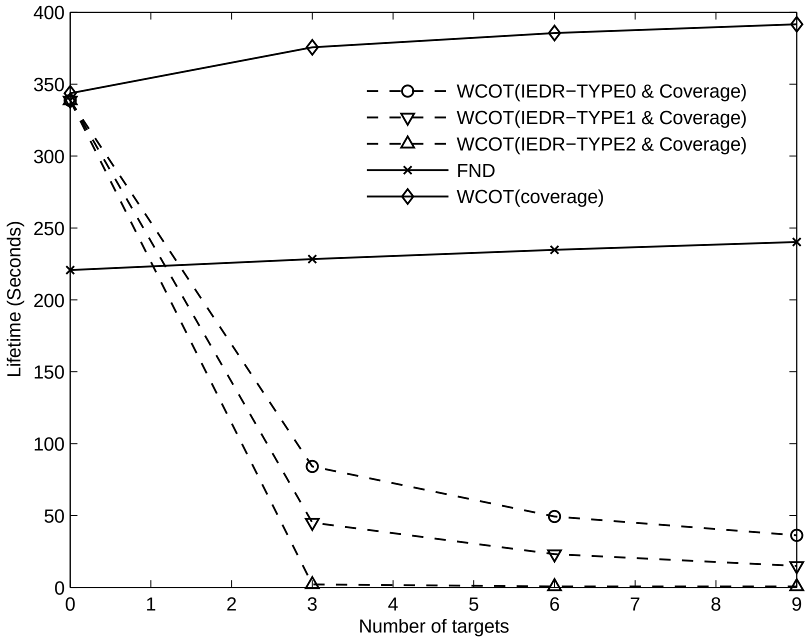


Figure 6.13. The utility as a function of the sensing coverage and IEDR

on the paths to the sink. Note that TYPE 2 and TYPE 1 applications are more drastically affected from the increased traffic and mobility due to the higher minimum frame requirements they have for the event delivery. Figure 6.15 further contrasts the effect of the mobility on the obtained results. The coverage based approach report higher lifetime values for increased traffic which is due to the TTLE as previously discussed.



(a)



(b)

Figure 6.14. Lifetime based performance evaluation (a) Target Speed = 0.33 m/s (b) Target Speed = 3 m/s

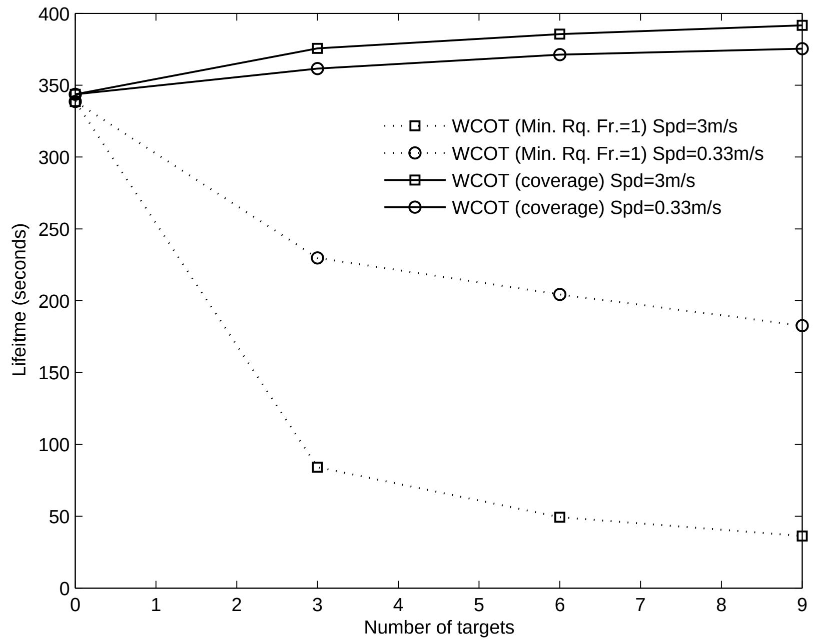


Figure 6.15. The effect of mobility and lifetime metric on the network lifetime

7. CONCLUSIONS

Network lifetime assessment problem in wireless sensor networks is examined in this work. We introduced a novel, utility based framework, called WCOT, for the realistic lifetime quantification of the sensor networks. Application dependence observed in the WSNs complicates quantifying the lifetime and in the literature rudimentary metrics such as the time till the first node death (FND) are generally employed which, according to our view, jeopardize the dependability of the results as a whole. We show via extensive simulation experiments based on realistic settings that the lifetime metrics that are not compatible with the application scenario give misleading results about the actual operational duration of the network.

To assess the proposed scheme WCOT, firstly we considered the scalar type of sensor networks that run environmental monitoring scenarios. We classify the environmental monitoring scenarios as periodic and event triggered and showed the effect of two routing algorithms, namely the Minimum Energy Routing (MER) and the Probabilistic Energy Driven Routing (PEDER), on the network lifetime for the both cases. We demonstrated that the frequently used lifetime metric FND is incapable of correctly quantifying the lifetime performances.

Apart from the scalar sensor applications, we focused on the video based WSN applications. Current sensor network products use low profile hardware and employ sleep schedule to lower the communication duty cycle. Firstly, we run tests to assess the quality of video transmission over realistic assumptions that reflect current constraints. Our observations dictate that low throughput observed is due to the sleep schedule as well as the hardware constraints. To enhance video transmission, we implemented an event based buffer management scheme which is shown to reduce the initial reporting delay and increased the total number of reported events. The buffer management alone does not increase the packet level throughput of the system. However, our implementation by introducing fairness lets the events get their fair share of network resources. Therefore, both sent and dropped frames are more equally distributed over events. LAS implementation further gives priority to the initial frames of the events to reduce the initial event reporting delay.

The lifetime behavior of the video based sensor networks is examined for a specific application scenario, namely the video surveillance. We showed that the lifetime metrics solely based on the number of sensor nodes or on the sensing coverage do not qualify as the appropriate lifetime metrics for the video surveillance sensor networks (VSSNs). The reason behind this is that the performance of VSSNs is based on the event reporting level of the network and legacy metrics ignore event delivery. As a remedy, we come up with a realistic lifetime utility indicator which is a function of both the sensing coverage and the instantaneous event delivery ratio (IEDR) which quantified the lifetime trends correctly when the target population size and mobility is varied. During the tests we discovered that increased traffic load positively affected the lifetime of the nodes in the network. The dynamics and the underlying mechanisms of this counter intuitive phenomena that we call *Traffic Triggered Lifetime Extension* is studied in detail.

Our future plans involve different work items for the various chapters of the thesis. Related to WCOT, we will further demonstrate the effect of the application dependence for other application types by representing the utility function in terms of appropriate utility indicators, some candidates being the packet latency, event reporting delay, reliability and breach probability. We will concentrate on the sensor networks with hybrid sensing modalities in which the scalar data and the visual data simultaneously flow in the network. We will implement prioritization schemes to effectively relay the aggregate data to the sink and define the utility function in terms of the both type of data received where their relative importance is expectedly application dependent. Event based queueing enhanced the VSSN performance without actually altering the packet level delivery statistics of the network. To be able to further increase application level performance we plan to implement complementary solutions that are implemented in the routing and the transport layer to enhance the VSSN functionality.

We consider *Traffic Triggered Lifetime Extension* to be a crucial phenomenon that deserves further study. Many MAC proposals for sensor networks make heavy use of the communication duty cycle to decrease the energy consumption rate of the nodes. However, the outcome of this effort may be more than the mere scaling down of the energy consumption of the network, as we discovered in this work. Related to this issue, we plan to test various listen offset dissemination models for SMAC. For low duty cycle values, we foresee that decreasing the number of virtual clusters and forcing nodes to have less number of sleep schedules may result in enhanced lifetime figures at the cost of decreased network performance in terms of throughput and delay. In this trade-off, we hope to obtain the optimum number of virtual clusters and their positional distribution in the monitored area of interest.

REFERENCES

1. Gershenfeld, N., R. Krikorian, and D. Cohen, "The Internet of Things.", *Scientific American*, Vol. 291, No. 4, pp. 76–81, 2004.

2. "XBOW Technologies, EKO product http://www.xbow.com/Eko/", 2008.

3. "The Arch Rock Company, http://www.archrock.com/", 2009.

4. "The Sensicast Company, http://www.archrock.com/", 2009.

5. Chang, J.-H. and L. Tassiulas, "Maximum Lifetime Routing in Wireless Sensor Networks.", *IEEE/ACM Transactions on Networking*, Vol. 12, No. 4, pp. 609–619, 2004.

6. Chang, J.-H. and L. Tassiulas, "Energy Conserving Routing in Wireless Ad-hoc Networks", *INFOCOM*, pp. 22–31, 2000.

7. Heinzelman, W. R., A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks", *HICSS*, 2000.

8. Yu, M., A. Malvankar, and S. Y. Foo, "An Energy-Efficient Path Availability Routing Algorithm For Mobile Ad Hoc Sensor Networks", *IEEE International Conference on Communications*, 2006.

9. Shah, R. and J. Rabaey, "Energy aware routing for low energy ad hoc sensor networks", *Wireless Communications and Networking Conference (WCNC) 2002.*, Vol. 1, pp. 350 – 355, March 2002.

10. Mahapatra, A., K. Anand, and D. P. Agrawal, "QoS and Energy Aware Routing for Real-time Traffic in Wireless Sensor Networks", *Computer Communications*, Vol. 29, No. 4, pp. 437–445, 2006.

11. Jain, N., D. K. Madathil, and D. P. Agrawal, "EnergyAware Multi-path Routing for Uniform Resource Utilization in Sensor Networks", *Information Processing in Sensor Networks (IPSN)*, LNCS, pp. 473–487, 2003.

12. Cheng, Z., M. Perillo, and W. B. Heinzelman, "General Network Lifetime and Cost Models for Evaluating Sensor Network Deployment Strategies", *IEEE Transactions on Mobile Computing*, Vol. 7, No. 4, pp. 484–497, April 2008.

13. Ahn, S., Y. Lim, and H. Yu, "Energy-Efficient Flooding Mechanisms for the Wireless Sensor Networks", *International Conference on Information Networking (ICOIN) 2008.*, pp. 1–5, 23-25 Jan. 2008.

14. Shu, H., Q. Liang, and J. Gao, "Wireless Sensor Network Lifetime Analysis Using Interval Type-2 Fuzzy Logic Systems", *IEEE Transactions on Fuzzy Systems*, Vol. 16, No. 2, pp. 416–427, April 2008.

15. van Hoesel, L., T. Nieberg, J. Wu, and P. Havinga, "Prolonging the Lifetime of Wireless Sensor Networks by Cross-layer Interaction", *IEEE Wireless Communications*, Vol. 11, No. 6, pp. 78–86, December 2004.

16. Deng, J., Y. Han, W. Heinzelman, and P. Varshney, "Scheduling Sleeping Nodes in High Density Cluster-based Sensor Networks", *Mobile Networks and Applications*, Vol. 10, No. 6, pp. 825–835, 2005.

17. Bhardwaj, M., A. Chandrakasan, and T. Garnett, "Upper Bounds on the Lifetime of Sensor Networks", *IEEE International Conference on Communications*, pp. 785–790, 2001.

18. Tian, D. and N. D. Georganas, "A Coverage-Preserving Node Scheduling Scheme for Large Wireless Sensor Networks", Raghavendra, C. S. and K. M. Sivalingam (editors), *WSNA*, pp. 32–41, ACM, 2002.

19. Mhatre, V., C. Rosenberg, D. Kofman, R. Mazumdar, and N. Shroff, "A Minimum

Cost Heterogeneous Sensor Network with a Lifetime Constraint", *IEEE Transactions on Mobile Computing*, pp. 4–15, 2005.

20. Zhang, H. and J. C. Hou, "Maintaining Sensing Coverage and Connectivity in Large Sensor Networks", *Wireless Ad Hoc and Sensor Networks: An International Journal*, Vol. 1, No. 1-2, pp. 89–123, January 2005.

21. Wang, X., G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks", *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pp. 28–39, ACM New York, NY, USA, 2003.

22. Maleki, M. and M. Pedram, "QoM and Lifetime-constrained Random Deployment of Sensor Networks for Minimum Energy Consumption", *IPSN*, pp. 293–300, IEEE, 2005.

23. Pan, J., Y. T. Hou, L. Cai, Y. Shi, and S. X. Shen, "Topology control for wireless sensor networks.", *MOBICOM*, pp. 286–299, 2003.

24. Pan, J., L. Cai, Y. Hou, Y. Shi, and S. Shen, "Optimal Base Station Locations in Two-tiered Wireless Sensor Networks", *IEEE Transactions on Mobile Computing*, Vol. 4, No. 5, pp. 458–473, 2005.

25. Brown, T. X., H. N. Gabow, and Q. Zhang, "Maximum Flow-life Curve for a Wireless Ad hoc Network", *MobiHoc*, pp. 128–136, ACM, 2001.

26. Hou, Y., Y. Shi, and H. Sherali, "On Lexicographic Max-Min Node Lifetime for Wireless Sensor Networks", *IEEE International Conference on Communications,*, Vol. 7, pp. 3790–3796 Vol.7, 20-24 June 2004.

27. Blough, D. M. and P. Santi, "Investigating Upper Bounds on Network Lifetime Extension for Cell-based Energy Conservation Techniques in Stationary Ad hoc Networks", *Proceedings of the 8th annual international conference on Mobile com-*

puting and networking (MOBICOM), pp. 183–192, ACM, New York, NY, USA, 2002.

28. Pandana, C. and K. Liu, "Maximum Connectivity and Maximum Lifetime Energy-aware Routing for Wireless Sensor Networks", *IEEE Global Telecommunications Conference (GLOBECOM)*, Vol. 2, pp. 1034–1038, 28 Nov.-2 Dec. 2005.

29. Madan, R. and S. Lall, "Distributed algorithms for maximum lifetime routing in wireless sensor networks", *Wireless Communications, IEEE Transactions on*, Vol. 5, No. 8, pp. 2185–2193, Aug. 2006.

30. Youssef, M., M. Younis, and K. Arisha, "A constrained shortest-path energy-aware routing algorithm for wireless sensor networks", *Wireless Communications and Networking Conference*, Vol. 2, pp. 794–799, 2002.

31. Adlakha, S., S. Ganeriwal, C. Schurgers, and M. B. Srivastava, "Density, Accuracy, Delay and Lifetime Tradeoffs in Wireless Sensor Networks - A Multidimensional Design Perspective", Akyildiz, I. F., D. Estrin, D. E. Culler, and M. B. Srivastava (editors), *SenSys*, pp. 296–297, ACM, 2003.

32. Mainwaring, A., J. Polastre, R. Szewczyk, D. Culler, and J. Anderson., "Wireless Sensor Networks for Habitat Monitoring", *ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, 2002.

33. Chiti, F., M. Ciabatti, G. Collodi, D. D. Palma, R. Fantacci, and A. Manes, "Design and Application of Enhanced Communication Protocols for Wireless Sensor Networks operating in Environmental Monitoring", *IEEE International Conference on Communications*, 2006.

34. Akyildiz, I. F., W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey.", *Computer Networks*, Vol. 38, No. 4, pp. 393–422, 2002.

35. Bertsekas, D. and R. Gallager, *Data networks*, Prentice-Hall, Inc. Upper Saddle

River, NJ, USA, 1987.

36. Baggio, A., "Wireless Sensor Networks in Precision Agriculture", *ACM Workshop on Real-World Wireless Sensor Networks (REALWSN)*, June 2005.

37. Langendoen, K., A. Baggio, and O. Visser, "Murphy Loves Potatoes: Experiences from a Pilot Sensor Network Deployment in Precision Agriculture", *20th International Parallel and Distributed Processing Symposium (IPDPS)*, p. 8, 2006.

38. Manjeshwar, A. and D. P. Agrawal, "TEEN: ARouting Protocol for Enhanced Efficiency in Wireless Sensor Networks.", *IPDPS*, 2001.

39. "OPNET Technologies Inc. www.opnet.com", 2009.

40. Ye, W., J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks", *INFOCOM*, 2002.

41. Xiong, Z., A. Liveris, and S. Cheng, "Distributed source coding for sensor networks", *Signal Processing Magazine, IEEE*, Vol. 21, No. 5, pp. 80–94, 2004.

42. Ye, W., J. S. Heidemann, and D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks", *IEEE/ACM Transactions on Networking*, Vol. 12, No. 3, pp. 493–506, 2004.

43. Downes, I., L. Rad, and H. Aghajan, "Development of a Mote for Wireless Image Sensor Networks", *Cognitive Systems with Interactive Sensors (COGIS)*, 2006.

44. Kulkarni, P., D. Ganesan, P. J. Shenoy, and Q. Lu, "*SensEye*: A Multi-tier Camera Sensor Network.", *13th ACM International Conference on Multimedia, Singapore*, pp. 229–238, ACM, 2005.

45. Akyildiz, I., T. Melodia, and K. Chowdhury, "A survey on Wireless Multimedia Sensor Networks", *Computer Networks*, Vol. 51, No. 4, pp. 921–960, 2007.

46. Karp, B. and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks", *MOBICOM*, pp. 243–254, 2000.

47. "XBow, http://www.xbow.com", 2009.

48. Rahimi, M., R. Baer, O. I. Iroezi, J. C. Garcia, J. Warrior, D. Estrin, and M. Srivastava, "Cyclops: in situ image sensing and interpretation in wireless sensor networks", *SenSys '05: Proceedings of the 3rd international conference on Embedded networked sensor systems*, pp. 192–204, ACM Press, New York, NY, USA, 2005.

49. Ye, W., J. Heidemann, and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks", *IEEE/ACM Trans. Netw.*, Vol. 12, No. 3, pp. 493–506, 2004.

50. Demers, A., S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm", *SIGCOMM '89: Symposium proceedings on Communications architectures & protocols*, pp. 1–12, ACM, New York, NY, USA, 1989.

51. Shreedhar, M. and G. Varghese, "Efficient fair queueing using deficit round robin", *SIGCOMM Comput. Commun. Rev.*, Vol. 25, No. 4, pp. 231–242, 1995.

52. Jun, J. and M. Sichitiu, "Fairness and QoS in multihop wireless networks", *Vehicular Technology Conference, VTC. IEEE 58th*, Vol. 5, pp. 2936–2940, 2003.

53. Raniwala, A., P. De, S. Sharma, R. Krishnan, and T. cker Chiueh, "End-to-End Flow Fairness Over IEEE 802.11-Based Wireless Mesh Networks", *INFOCOM*, pp. 2361–2365, IEEE, 2007.

54. Rangwala, S., R. Gummadi, R. Govindan, and K. Psounis, "Interference-aware fair rate control in wireless sensor networks", Rizzo, L., T. E. Anderson, and N. McKeown (editors), *SIGCOMM*, pp. 63–74, ACM, 2006.

55. Fan, K.-W., Z. Zheng, and P. Sinha, "Steady and Fair Rate Allocation for Rechargeable Sensors in Perpetual Sensor Networks", *Proc. of ACM SENSYS,*

Raleigh, NC, To Appear,, November 2008.

56. Chen, S., Y. Fang, and Y. Xia, "Lexicographic Maxmin Fairness for Data Collection in Wireless Sensor Networks", *IEEE Transactions on Mobile Computing*, pp. 762–776, 2007.

57. Sridharan, A. and B. Krishnamachari, "Max-min fair collision-free scheduling for wireless sensor networks", *In Workshop on Multihop Wireless Networks (MWN '04), IPCCC*, 2004.

58. Özgür B. Akan and I. F. Akyildiz, "Event-to-sink reliable transport in wireless sensor networks", *IEEE/ACM Trans. Netw.*, Vol. 13, No. 5, pp. 1003–1016, 2005.

59. Wan, C.-Y., S. B. Eisenman, and A. T. Campbell, "CODA: congestion detection and avoidance in sensor networks", *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pp. 266–279, ACM, New York, NY, USA, 2003.

60. He, T., J. A. Stankovic, C. Lu, and T. F. Abdelzaher, "SPEED: A Stateless Protocol for Real-Time Communication in Sensor Networks", *ICDCS*, pp. 46–55, 2003.

61. Tassiulas, L. and S. Sarkar, "Maxmin fair scheduling in wireless networks", *INFOCOM*, 2002.

62. Golestani, S. J., "A Self-Clocked Fair Queueing Scheme for Broadband Applications", *INFOCOM*, pp. 636–646, 1994.

63. Vaidya, N., A. Dugar, S. Gupta, and P. Bahl, "Distributed Fair Scheduling in a Wireless LAN", *IEEE Transactions on Mobile Computing*, pp. 616–629, 2005.

64. Nuyens, M. and A. Wierman, "The Foreground-Background queue: A survey", *Performance Evaluation*, Vol. 65, No. 3-4, pp. 286–307, 2008.

65. Schrage, L., "A proof of the optimality of the shortest remaining processing time

discipline", *Operations Research*, Vol. 16, No. 3, pp. 687–690, 1968.

66. Kleinrock, L., *Queueing Systems: Volume 2: Computer Applications*, John Wiley & Sons, 1976.

67. Coffman Jr, E. and P. Denning, *Operating Systems Theory*, Prentice Hall Professional Technical Reference, 1973.

68. Rai, I., G. Urvoy-Keller, and E. Biersack, "Analysis of LAS scheduling for job size distributions with high variance", *Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and modeling of computer systems*, pp. 218–228, ACM Press New York, NY, USA, 2003.

69. Wierman, A., N. Bansal, and M. Harchol-Balter, "A note on comparing response times in the M/GI/1/FB and M/GI/1/PS queues", *Operations Research Letters*, Vol. 32, No. 1, pp. 73–76, 2004.

70. Wierman, A. and M. Harchol-Balter, "Classifying scheduling policies with respect to unfairness in an M/GI/1", *ACM SIGMETRICS Performance Evaluation Review*, Vol. 31, No. 1, pp. 238–249, 2003.

71. Shao, Z. and U. Madhow, "Scheduling heavy-tailed data traffic over the wireless Internet", *Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall. 2002 IEEE 56th*, Vol. 2, 2002.

72. Komar, C. and C. Ersoy, "Optimization of Power Consumption Using Trespassers' Favorite Path and Variable Sensing Range Integrated Sleep Schedule in Surveillance Wireless Sensor Networks", *ISCIS '08*, 2008.