QOS AWARE ROUTING WITH ADMISSION CONTROL FOR VIDEO SENSOR NETWORKS

SONG GUO

Dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

BOSTON

UNIVERSITY

BOSTON UNIVERSITY

COLLEGE OF ENGINEERING

Dissertation

QOS AWARE ROUTING WITH ADMISSION CONTROL FOR VIDEO SENSOR NETWORKS

by

SONG GUO

B.E., University of Electronic Science and Technology of China, 2005
M.S., Boston University, 2007

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
2011

Approved by

First Reader

Thomas D.C. Little, Ph.D.
Professor of Electrical and Computer Engineering

Second Reader

David Starobinski, Ph.D.
Associate Professor of Electrical and Computer Engineering

Third Reader

Murat Alanyali, Ph.D.
Associate Professor of Electrical and Computer Engineering

Fourth Reader

Prakash Ishwar, Ph.D.
Associate Professor of Electrical and Computer Engineering

The only laws of matter are those that our minds must fabricate
and the only laws of mind are fabricated for it by matter.

Logic will get you from A to B.
Imagination will take you everywhere.

To Father, Mother, and Yaya.

Acknowledgments

This work would not have been possible without my advisor, Prof. Thomas D. Little who not only encouraged me to give my best to research but also motivated me to excel at academics. He gave me complete freedom to choose my own research topics. Throughout the past five years, he has taught me how to think of intuitive and simple solutions to complex problems. Professor Thomas Little always set high standards for the work which has helped me shape up as a quality-driven professional. The guidance, advice and help I have received from him is going to assist me for the time to come.

I am also deeply grateful to Boston University and all my committee members. Prof. Starobinski and Prof. Ishwar had devoted a lot of their time to review my thesis and provide me precious feedbacks and research insights. I am very impressed with their serious research attitude. I am indebted to Prof. Alanyali who closely involved himself in my research. His results-driven approach and straightforward feedback always helped me keep the focus and target perfection.

I feel fortunate to have met sharp minds and turn them into lifelong friends, during my research experiences at Boston University. Yingwei Lin who was always there to help me out and keep up my enthusiasm level, we enjoyed the time with Thai food together. Research could not have been fun without Zeyu, Zhengsu, Jimmy, Aron, and Yuting, I could always count on these people for thoughtful suggestions and constructive inputs.

Last but not the least, my gratitude goes to family for their love and support, gentle prodding and solid support for my endeavor. Any number of words cannot sufficiently express the love and encouragement I received from my wife and my parents. Their patience and tolerance are unparalleled.

QOS AWARE ROUTING WITH ADMISSION CONTROL FOR VIDEO SENSOR NETWORKS

(Order No.)

SONG GUO

Boston University, College of Engineering, 2011

Major Professor: Thomas D.C. Little, PhD., Professor, Associate Chair
Electrical and Computer Engineering

ABSTRACT

Wireless sensor networks (WSNs) have secured a role in many emerging applications spanning environmental, ecological, industrial, and commercial domains in which data are periodically sourced from sensors and used in monitoring and control. The ubiquity of network-based video streaming, largely in the form of monitoring equipment installed in administrative domains (cities, campuses, offices, highway systems) has been possible by increases in capability of low-cost cameras and computational devices. However the streaming of video data over wireless sensor networks has the nature of heavy data transmission load, relatively long data transmission period, severe inter- and intra-path interference, and unbalanced network energy consumption under severe energy constraints. These characteristics make it very difficult to achieve meaningful throughput performance for video data delivery over WSN. In this dissertation we focus on the development of practical throughput aware video data routing techniques optimized to data delivery cost to address the challenge of QoS support in end-to-end throughput achievement for bandwidth demanding video delivery in a large-scale, battery-operated sensor network.

We utilize video data streaming throughput as the main metric to quantify the video delivery performance. We present a benchmark data routing algorithm to assure the throughput performance (QoS) of a data path via eliminating the impact of inter-path interference through path isolation. Extending the benchmark algorithm, we develop an interference-tolerant data routing algorithm to improve the general data egress rate of the entire network with some tolerant sacrifices of the throughput performance on each stream. The routing algorithms construct throughput-guaranteed end-to-end video delivery data paths based on accurate video delivery performance estimation. We present two analytical models in conjunction with simple proactive admission and congestion control strategies to enable accurate end-to-end data delivery throughput estimation with considerations of data stream's transmission rate, location, and relative positions. Simulations demonstrate that our estimation models are highly consistent with the measurement of the real data transmission scenario and our proposed data delivery scheme adapts to the optimal data rate about three times faster than a traditional CTS/RTS scheme without generating any jitter as observed in CTS/RTS. In addition, we exploit the option of deploying mobile base stations to support improved video delivery QoS. The proposed mobile base station deployment strategy improves the video delivery performances with the best effort to minimize the data delivery cost. As a complementary solution for the deployment of our routing scheme over WSNs, we develop a new code dissemination framework for generic WSN applications. Simulation and prototype implementation illustrate that our framework is almost five times faster for application code dissemination than traditional over the air programming strategy. Another contribution of this dissertation is the constellation graph of routing algorithms. It presents a clear way of selecting the most appropriate routing algorithm for a specific WSN application.

Contents

List of Figures

List of Abbreviations

CDN	....	Content Delivery Network
CMOS	....	Complementary Metal-Oxide-Semiconductor
EHM	....	Error Hanlder Module
FEC	....	Forward Error Correction
MCU	....	Micro-Controller Unit
MAC	....	Media Access Control
OTA	....	Over the Air Programming
P2P	....	Peer-to-Peer
QoS	....	Quality of Service
UAV	....	Unmaned Aerial Vehicle
VGA	....	Video Graphics Array
WSN	....	Wireless Sensor Network
WVSN	....	Wireless Video Sensor Network

Chapter 1

Introduction

Advances in computer and network technology have led to wireless sensor networks (WSN) which is comprised of many small, low-power embedded processors capable of sensing and communicating using short-range networking. Today, sensor networking has emerged as a frontier interconnecting the Internet to the physical world. Typically, a wireless sensor node is an integration of basic components: sensing, computing, communication, and power supply. These components are integrated on a single or multiple boards, and packaged in a dimension of a few inches. These intelligent small devices make our computing in daily life more distributed and pervasive. From this perspective, the emergence and development of the wireless sensor network are essentially the newest trend of Moore's Law [Moo05] toward the miniaturization and ubiquity of computing devices.

The variety of sensing modalities of WSNs embodies the idea of ubiquitous computing at all scales throughout everyday life in environmental monitoring and habitat study, over a battlefield for military surveillance and reconnaissance, in disaster environments for search and rescue, in factories for condition based maintenance, in buildings for infrastructure health monitoring, in homes to realize smart homes, or even on bodies for patient monitoring [LG09, MPSC02, BAB$^+$07]. Among the many sensor modalities supported by the sensor devices (or motes-sensor nodes), in this dissertation we concentrate our focus on ones that produce single or multiple images in a video stream. With the development of low power, low cost CMOS imaging sensors, scientists visualize a great potential of multimedia streaming application for wireless sensor network in the area of homeland security, habitat monitoring and image-based monitoring and control. In these applications a sensor node can capture images, audio and/or video information, and send them in a compressed form to a consumer elsewhere on the network. The user does not have to wait for the download of the entire video content but can play back the content immediately once a small amount of buffered data are received. The flexibility of wireless sensor networks coupled with this sensing modality makes video observation very promising to enable humans to observe phenomena or locations that are otherwise difficult or dangerous to access. This kind of network also shows an incredible value in the areas of military detection and security surveillance applications, and it is identified as a new branch of sensor network named wireless video sensor network (WVSN).

1.1 Wireless Sensor Network

Wireless Sensor Network is in general a collection of tiny and inexpensive devices named sensor node as shown in Fig. 1·1 with the capability of sensing, computing and communication. A sensor node is usually comprised of a series of standard modules, a sensing module to detect and monitor the change of the environment, a micro-controller to convert the signal of the sensing module to some digital readings which can be understood by the users, a wireless communication interface to exchange data or information among neighbor nodes or relay data to the destination which may be out of communication range of the data origin, a small memory to cache data on board, a power supply module to guarantee the execution of the normal node functionalities for computing, transmitting and receiving, etc. As the manufacturing cost of these devices drops dramatically with the increase of their computing capabilities, the large scale deployment of such devices becomes practical and

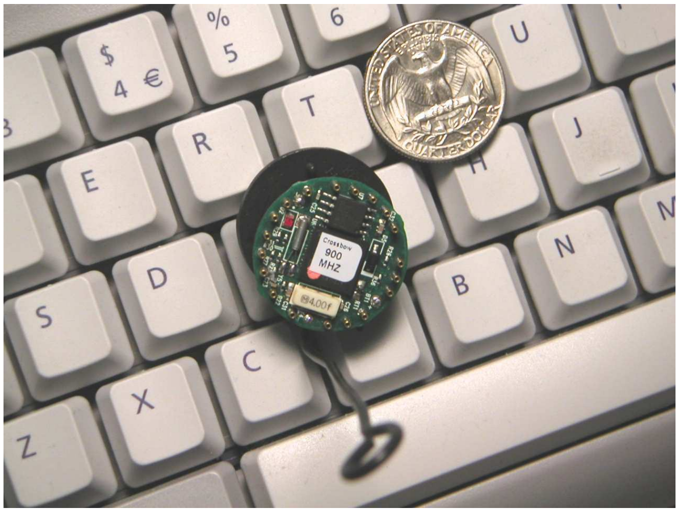


Figure 1·1: Mica2dot sensor board from Crossbow Technology Inc.

affordable. When deployed in large numbers, the redundant sensor node coverage of the interested field is intended to provide reliable and accurate data retrieval services for the network users. A typical WSN deployment scenario is illustrated in Fig. 1·2. In wireless sensor network application scenarios [ASSC02], a large group of sensor nodes are deployed to a place of interest to perform certain desired duties. Without the installation of any network infrastructure, these autonomous sensor nodes can establish connections with each other to form an ad hoc sensor network. This network is basically a peer to peer multi-hop wireless network where information packets are transmitted in a receiving-and-forwarding manner towards the destination node which is usually the gateway via multiple intermediate nodes. The gateway node serves as the information bridge between the sensor networks and the outside networks as the Internet for instance. The end users can leverage the gateway node to send commands or information query to the sensor network and the sensor network will deliver the interested information feedback to the end users via the gateways. The next section describes the basic data delivery scenarios for WSNs.

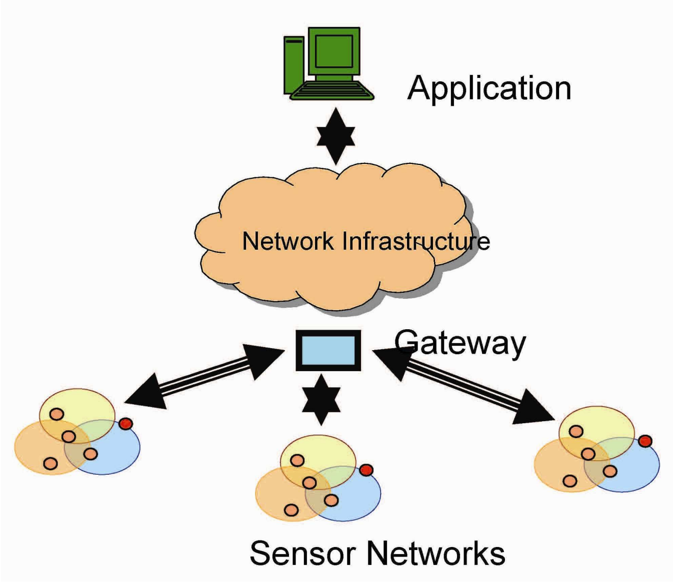


Figure 1·2: Typical wireless sensor network scenario

1.2 Data Delivery over Wireless Sensor Networks

Data delivery over WSNs can be classified into three categories: proactive data delivery, reactive data delivery and hybrid data delivery. In proactive data delivery scenario, the sensor nodes periodically exchange some basic information, such as "link state" information with their neighbors to construct a backbone network [MCdS+05] with the propose of obtaining available data paths from the local area to the destination. The proactive data delivery scheme always attempts to maintain consistent, up-to-date routing information. Therefore, a source node can obtain a routing path immediately when the data delivery is on demand. Using proactive routing strategies, the network users can extract the information from the sensor network in a timely manner. But due to the fact that each sensor node in the network is required to update its state and maintain a route regardless of the traffic existence, proactive schemes introduce relatively high overhead to maintain up-to-date network topology information as well compared to the reactive and hybrid data delivery [ABJ+06].

In reactive data delivery scenario, the sensor nodes do not maintain available data paths in advance. The routing paths are searched only when data delivery is on demand [KK00]. A route discovery procedure is invoked to identify available data paths before the beginning of data sending. The path discovery procedure terminates when either a route has been found or no route is available after the examination for all route permutations. Compared to proactive schemes [KS07], reactive schemes are more adaptive to the network environment alterations. In wireless sensor networks, the network topology and node connectivity may vary over time due to node failures, battery starvation, signal loss etc. Reactive schemes can adapt to these abnormal environment changes with the less control overhead [ABJ+06]. Thus, reactive routing schemes have better scalability than proactive ones in data delivery for WSNs. However, when using reactive routing protocols, source nodes may suffer from long delays for route searching before they can forward data packets [EFK06]. In order to find a balance between proactive and reactive data delivery schemes, hybrid data delivery strategies are proposed to combine their merits to overcome their disadvantages. Usually, the hybrid schemes form the network as zone based structures [SPH04]. It deploys proactive routing protocols for inner-zone data delivery while applies reactive routing schemes to facilitate inter-zone data routing.

The data delivery inside a zone generally employs the proactive routing protocol to seek quick route to send data from one node to another. Reactive routing protocols are exploited in acquiring routes to destinations beyond the routing zone, battling with the network environment change and suppress the control overhead.

1.3 Video Data Delivery Problem Specification

WSNs are usually employed to deliver scaler information such as the temperature, moisture, light intensity, and etc. However, the recent advances in wireless com-

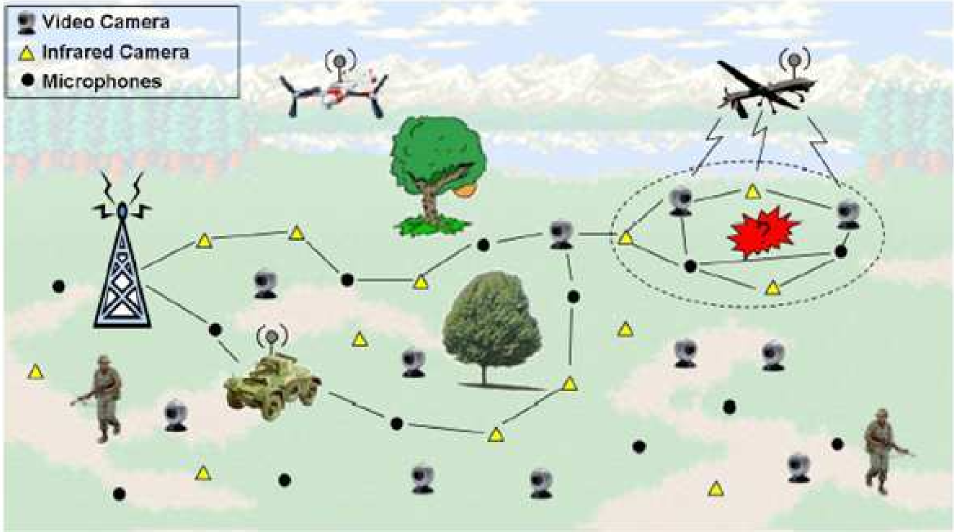


Figure 1·3: WVSN Multimedia Application [Eki11]

munication technology and low-power, low-cost CMOS imaging sensors inspire the development of a new branch of WSN, Wireless Video Sensor Networks (WVSN). WVSN is capable of gathering more information from the physical world and perform versatile environment monitoring tasks. However, to delivery such rich content video data over resource constrained wireless sensor network with limited bandwidth is challenging. In this section, we identify these challenges for the video delivery over WVSN and discuss the requirements of the routing protocol design.

1.3.1 Video Delivery Scenario

In WVSN applications a sensor node can capture images, audio and/or video information, and send them in a compressed form to a consumer elsewhere on the network. The user does not have to wait for the download of the entire video content but can play back the content immediately once a small amount of buffered data is received. Fig. 1·3 illustrates a typical scenario for multimedia applications over WVSN.

The multimedia application above contains two distinct video delivery scenarios over WSNs. The first scenario is centralized video delivery, which is similar to traditional data delivery over WSN. The video sensor node sends the video data towards the base station. The data flows from different video sources will eventually converge at the base station. Unlike traditional WSN which only carries very limited data load, video data delivery will quickly overwhelm the nodes' link capacity near the base station. These sensor nodes near the base station will quickly become bottlenecks for the data extraction of the network due to unbalanced heavy transmission load and severe transmission interferences. The heavy data transmission load on these sensor nodes will also quickly exhaust their batteries and disconnect the base station from the sensor networks. In order to relieve these issues, a second video data delivery scenario, distributed video delivery is illustrated in Fig. 1·3. Instead of converging all the data flows to the centralized gateway node, the second data delivery scenario explores the use of distributed mobile base stations as an option to improve the load balancing and information egress of the network. The primary base station assigns mobile base stations like UAVs (Unmanned Aerial Vehicle) to specific areas to assist video data delivery. As opposed to sending all video data to the centralized base station in "centralized" scenario, video sensor nodes have an option of streaming their data to the mobile base stations nearby. The localized video streaming not only provides better video data transmission load balancing but also improves the throughput performance of the video streaming applications. However, the distributed video delivery scenario requires the mobile base station to retain a higher link capacity and the deployment of these mobile base stations will always result in extra cost for the data delivery. The balance between the network performance and the video data delivery cost in the second scenario is still an open question today.

1.3.2 Application Requirements

Today, many of the existing WVSN applications [HSOH03, TLC$^+$06, CRZ04, KGSL05] are an integration of Internet video streaming solutions to the domain of WSNs. Some applications rely on conventional wired video cameras. Others assume wireless communications but do not address scale-up to large numbers of video cameras nor the support of multiple streams. In this dissertation, we expect a wireless video sensor network to be fully functional with implications of multi-hop grids, battery-powered video sensor nodes, and wide area coverage of order kilometers. Each node is equipped with a low power video camera with the capability of observing the physical world which largely increase the information amount that need to be wirelessly relayed toward the base station. Such fully functional mesh wireless video sensor networks are capable of simultaneously supporting multiple video targets observation and multiple base station routing, in other words, it would support concurrent wireless video streams from different sources to different destinations.

Fig. 1·4 shows the general problem we are going to address. A series of wireless video sensor nodes are scattered in the order of 1000m field and form a connected video sensor network. There is only one static primary base station. Instead of merging all the data to the gateway station, we allow data to be streamed to some other mobile sinks to satisfy certain throughput requirement with certain amount of mobile sink deployment cost. When multiple streams are active, we must ensure that the communication constraints along a path from source to sink are not violated. If there are n camera nodes, m sinks, and r targets, the problem generalizes to one of providing multiple streams from n sources to m possibly mobile sinks. Thus path formation will be triggered on demand by a user (e.g., “look at camera”) or by a network event (e.g., “some suspect appears”). We seek to design a dynamic path formation algorithm for video delivery applications over WSN which will satisfy the

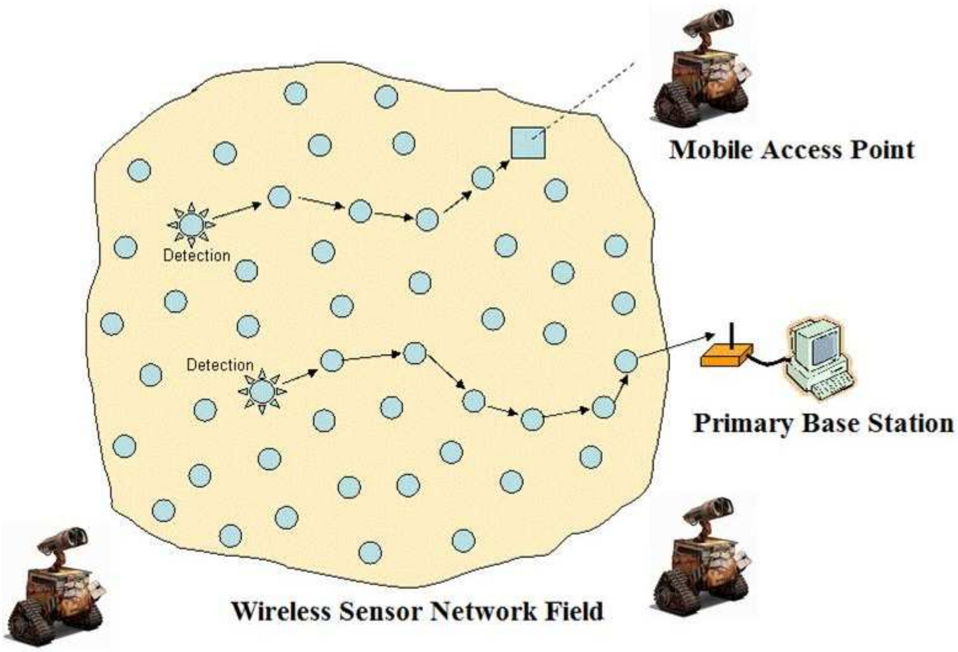


Figure 1·4: Application scenario

following requirements: (1) Sufficient network capacity is allocated on the end-to-end path from source to sink; (2) The path is not created in conflict with other paths - it is non-intersecting under operating assumptions; (3) The path minimizes costs locally and globally in the context of preexisting paths; (4) The path's duration, and anticipated energy consumption is mindful of residual energy and the replenishment behavior of participating nodes.

An extreme example is to form isolated video streams within the network as shown in Fig. 1·5. In this case, the data paths are geographically isolated from each other, the data transmission on one of the concurrent paths will not interfere with the data delivery on the others. In order to achieve path isolation, we can suppress data transmission on all the nodes lying inside the interference range of the nodes along the active data paths. This data path isolation scheme can apparently avoid transmission interferences, however, on the other side, it decreases the number of

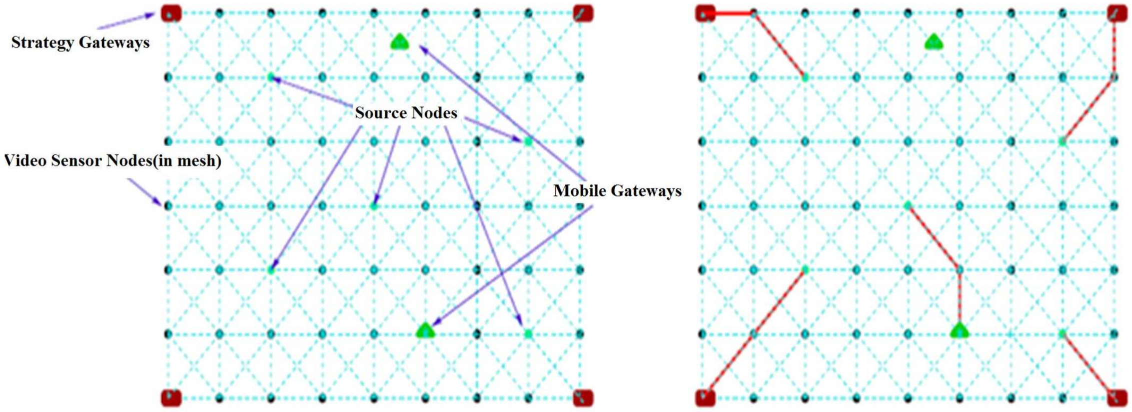


Figure 1·5: Example of path formation for isolated streams

streams that a network can carry and meanwhile the deployment of many scattered mobile base stations increase the cost of the overall data extraction for the entire WSN. Supporting video delivery applications with desired throughput performance (QoS) while achieving economical data extraction is still an open question.

1.3.3 Quality of Service (QoS) for WVSN

It is well known that QoS is an overused term with various meanings and perspectives [GGW04]. Different applications and different researchers may perceive and interpret QoS in different ways. However in general, QoS for data delivery over a WSN can be classified into two different levels: network level and application level. On the network level, QoS is accepted as a measurement of the service quality that the network offers to the applications while on the application level, QoS generally refers to the quality as perceived by the users of the Applications. One simple QoS model [GGW04] is presented in Fig. 1·6. The application level QoS is beyond the interest of this dissertation, we focus our research on the improvement of the network level QoS for video streaming applications over WSNs. From network level, researchers usually define the QoS as some assurance from the network to guaran-

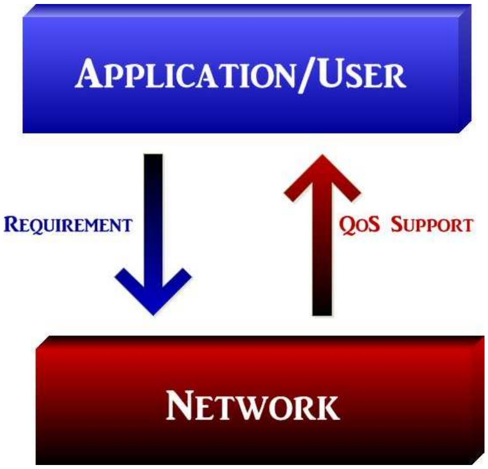


Figure 1·6: QoS Model

tee a set of measurable attributes for end-to-end data delivery. For instance, RFC 2386 [CNRS98] characterizes QoS as a set of service requirements to be met when transporting a packet stream from the source to its destination in terms of delay jitter, available bandwidth, and packet loss. In this dissertation, we take end-to-end throughput performance as the main metric to measure the data delivery QoS for WVSN.

1.3.4 Synopsis

Video streaming applications over WSNs are diverse and have specific requirements for the data delivery. Four basic challenges for video data delivery are identified as sensor node addressing, cost efficient routing, QoS support data delivery and admission control for resource allocation. A summary of the questions regarding the video data delivery challenges that we have addressed in this dissertation include:

- What are the characteristics of the data transmission in WVSN? Given some wireless communication technology and the parameters of the radio range, how the data will be delivered to the destination along a data path?

- What is the QoS support for video streaming applications over WSN?
- How to allocate resource among multiple data flows via admission control?
- What metrics are appropriate to characterize the performance of the application?
- How to form data paths which can deliver video data with satisfied QoS requirements? What are the requirements/constraints of the data path setup?
- How to estimate the performance of the QoS? Given a potential data path, is there a way to recognize the QoS performance of the video delivery along this path? The estimation could serves as the criteria to evaluate the quality of data path so as to assist the routing decision for data delivery and improve the overall performance.
- How the existing data delivery along active data paths will impact on the potential data delivery? Explore the solution to form QoS supported data path in the context of pre-existing video delivery paths.
- How to balance the cost and the QoS performance of video delivery over WSNs? What are the appropriate metrics to characterize the data delivery cost in WVSN?
- What are the criteria to evaluate a routing protocol for video delivery applications? Is there an unified criterion which could be applied to compare different protocols for the same video data delivery application for WVSN?
- How to efficiently disseminate the data delivery programs on the WSNs? Is there an efficient way to spread the programs across the network which is optimized for code generation, propagation and updating.

1.4 Dissertation Outline

1.4.1 Contributions

The main contribution of this dissertation is to develop an analytical model to analyze the QoS in throughput performance of end-to-end video data delivery over WSN. This model quantifies the QoS performance in terms of the data path throughput and provides accurate end-to-end throughput estimation under various situations. This estimation model takes the negative impact of the concurrent video data transmission interferences into consideration, and links the throughput performance with comprehensive network parameters in terms of data transmission rate, data source locations and data path relative positions. This model is employed by the routing protocol to construct dynamic throughput performance aware video delivery paths. Specific contributions include:

- A data transmission model to simplify the mathematical analysis of the end-to-end video data delivery.
- An accurate end-to-end throughput performance estimation model for single path and multi-path transmission scenarios.
- Simple admission control strategies to provide improved and more predictable data delivery performance for wireless video sensor networks.
- A benchmark routing protocol incorporated with the throughput estimation model to achieve throughput-aware video data delivery.
- An improved interference-tolerant routing protocol incorporated with multi-path throughput estimation model to achieve cost efficient throughput-aware video data delivery.

- A mobile base station deployment strategy that improves the network QoS performance based on observed network parameters of pre-existing video streams.
- A routing protocol performance constellation graph to evaluate different routing protocols under unified criteria for specific video delivery applications.
- An efficient "over the air" WSN reprogramming framework to wirelessly disseminate the programs across the network with optimizations in code generation, propagation and updating.

1.4.2 Organization

The remainder of this dissertation is structured as follows:

Chapter 2 describes the background of the video delivery over WSN. It provides an introduction of the wireless video sensor network (WVSN) regarding the concept, importance and challenges. It explores the existing efforts of enabling WVSN applications and provides motivations and guidance to develop the techniques in this dissertation that will satisfy video delivery network platform design requirements.

Chapter 3 focuses on the related work of state-of-art data delivery and throughput estimation techniques that accomplish high quality video streaming applications over WVSN.

Chapter 4 is the detailed description of the development of the analytical end to end throughput estimation models with simple admission control strategies.

Chapter 5 demonstrates the dynamic video data routing algorithms with analytical throughput estimation model discussed in Chapter 4. It also introduces a network routing protocol evaluation framework that compare different routing protocols under unified criteria for specific video delivery applications.

Chapter 6 presents an efficient ”over the air” WSN reprogramming framework to disseminate the programs across the network with optimizations in code generation, propagation and updating.

Chapter 7 presents the performance results based on the analytical throughput estimation model and the program dissemination framework.

Chapter 8 summaries the contributions of the dissertation and layouts an overview of the future research direction in the area of video delivery for WSNs.

Chapter 2

Video Delivery in Wireless Sensor Network

This chapter presents a broad area view of the wireless video sensor networks (WVSN). We first layout an introduction of the techniques enabling video delivery over WSN from hardware devices of those tiny video sensor nodes. Second we provide a broad view of techniques for different design aspects of supporting video delivery over WSN. We highlight challenges of enabling WVSN applications. Last, we concentrate our exploration on data routing techniques that we can leverage to resolve the challenges of facilitating video delivery over WSN.

2.1 Video Sensor Node

The development of CMOS technology enables today's image sensors to capture and process optical image on a single chip. Compared to previous CCD image sensors, CMOS image sensor is less expensive and more energy efficient. The availability of the CMOS image sensors drive the massive deployment of digital video cameras on resource constrained embedded wireless sensor nodes, leveraging the video sensing capabilities for WSN. At the present time, the hardware platforms for video sensor nodes are mainly built to provide imaging resolutions from low range to medium resolution range (100k-300k pixels). The following introduction about video sensor nodes hardware platforms is partially based on [AMC07].

2.1.1 Low Imaging Resolution Motes

Although CMOS image sensors have become the ideal candidate to interface with computational-rich embedded devices such as PDAs, smart phones for a relatively long history, it is not until recent years that this technology has been introduced to computationally-constrained embedded devices to integrate with off-the-shelf wireless sensor nodes. The Cyclops module [RBI$^+$05] is one of the examples that bridges the gap between CMOS cameras and Mica2, MicaZ motes for instance. Cyclops contains programmable logic and memory for high-speed data communication. Cyclops consists of an image sensor (CMOS Agilent ADCM-1700 CIF camera), an 8-bit ATMEL ATmega128L micro-controller (MCU), a complex programmable logic device (CPLD), an external SRAM and an external Flash. The MCU controls the image sensor, configures its parameters, and performs local processing on the image to produce an inference. Since image capture requires faster data transfer and address generation than the 4 MHz MCU used, a CPLD is used to provide access to the high-speed clock. The Cyclops firmware is written in the nesC language, based on the TinyOS libraries. The module is connected to a host mote to which it provides a high level interface that hides the complexity of the imaging device to the host mote. Moreover, it can perform simple inference on the image data and present it to the host.

Another example is CMUcam3 [RAGN07] from Carnegie Mellon University. CMUcam3 is equipped with a CIF resolution (352x288) RGB color image sensor with loading speed of 26 frames per second and can be interfaced with an IEEE802.15.4 [Erg04] compliant Crossbow TelosB mote [Cro05]. It implements software JPEG compression and provides a basic image manipulation library.

The earlier CMOS image sensors are mainly designed to interact with sensor nodes equipped with 8-bit microprocessors. However, with the continued research efforts in multimedia wireless sensor networks and hardware technology improvement in embedded microprocessors, Downes *et al.* [DRA] presents an interesting finding that to perform operations such as 2-D convolution on an 8-bit processor such as the ATMEL ATmega128 clocked at 4 MHz is 16 times slower than with a 32-bit ARM7 device clocked at 48 MHz, while the power consumption of the 32-bit processor is only six times higher. Hence, the researchers argue that 32-bit processors are better fit for image processing than their 8-bit counterparts. And this argument inspired the use of motes with 32-bit microprocessors to support WVSN applications. The more powerful 32-bit motes are capable of providing medium resolution video sensing for the WVSN applications. For example, in [DRA]a new image mote is developed based on an ARM7 32-bit CPU clocked at 48 MHz, with external FRAM or Flash memory, 802.15.4 compliant Chipcon CC2420 [Chi] radio, that is interfaced with mid-resolution (352x288) ADCM-1670 CIF CMOS sensors and low-resolution 30x30 pixel optical sensors.

2.1.2 Medium Resolution Mote

The most well known 32-bit mote platform to be able to support medium resolution video sensing is Intel's Stargate board [Int] and its successor. The Stargate board is a high-performance processing platform designed for sensor, signal processing, control, robotics, and sensor network applications. It is designed by Intel and produced by Crossbow Technology Inc. Stargate is based on Intel's PXA-255 XScale 400 MHz RISC processor, which is the same processor found in many handheld computers including Compaq IPAQ and Dell Axim. Stargate has 32M byte of Flash memory, 64M byte of SDRAM, and an on-board connector for Crossbow's MICA2 or MICAz motes as well as PCMCIA Bluetooth or IEEE 802.11 cards. Hence, it can work as a wireless gateway and as a computational hub for in-network processing algorithms. When connected with a webcam or other capturing device, it can function as a medium-resolution multimedia sensor. Its energy consumption is still high, as documented in [MPOM06]. Moreover, although efficient software implementations exist, XScale processors do not have hardware support for floating point operations, which may be needed to perform efficient multimedia processing algorithms. Following Stargate, Intel has also developed the second generation wireless motes, known as Stargate2 and Imote2 platforms. Intel Imote2 platform has a common core to the next generation Stargate2 platform, and is built around a new low-power 32-bit PXA271 XScale processor at 320/416/520 MHz, which enables performing DSP operations for storage or compression, and an IEEE 802.15.4 Chipcon CC2420 [Chi] radio. It is equipped with large on-board RAM and Flash memories (32M byte), alternate radios, and a variety of high-speed I/O to connect digital sensors or cameras. Its size is also very compact, 48x33 mm, and it can run the Linux operating system and Java applications. The researchers from Enalab of Yale University built a very compact video camera board [TLC$^+$06] to interface with the Imote2 mote (Fig. 2·1). The camera-board packs an OmniVision OV7649 camera, which can capture color images at 30 fps VGA (640x480) and 60 fps QVGA (320x240). Currently, there are two different lens configurations: standard and 162 degree wide-angle. The power consumption of the active camera-board is 44 mW. In fully-active mode, at 104 MHz and 8 fps, the entire system consumes 322 mW (of which the iMote2 is responsible for 279 mW).

2.1.3 High Resolution Mote

One of the major challenges of supporting high resolution video sensing for wireless sensor network is to reduce energy consumption of the video sensor nodes to prolong the network life. A variety of optimization techniques [GHPB02] have been proposed to extend the lifetime of the sensor network including dynamic optimization of voltage and clock rate, wake-up procedures to keep electronics inactive most of the time, and

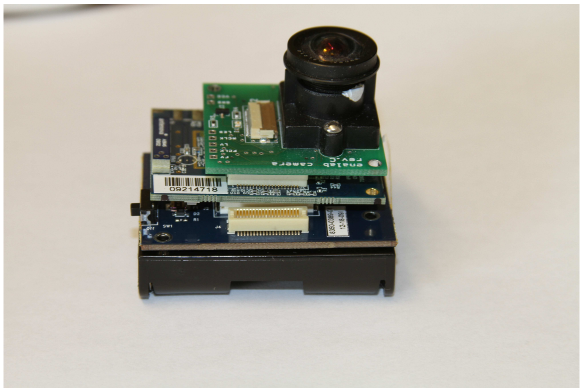


Figure 2·1: Enalab camera board on Imote2

energy-aware protocol development for sensor communications [CT00]. Energy constraint for sensor nodes prevents the installation of energy consuming high resolution cameras on the mote. However, the improvement of energy harvesting techniques inspires the integration of energy harvesting modules on video sensor node platforms to relax the energy consumption constraints. These compact energy harvesting modules operate on the video sensor nodes to extract energy from the environment where the sensor node itself lies, offering another important means to prolong the lifetime of the video sensor devices.

Among the many energy harvesting techniques, solar based energy harvesting technology is in favored for video streaming applications. Platforms being able to continuously power sensors based on simple COTS photovoltaic cells coupled with supercapacitors or rechargeable batteries are demonstrated in [JPC05]. Little *et al.* [LIK07] implements a genuine video senor node system prototype (Fig. 2·2) with solar powered video sensing capabilities.

The video node (VN) prototype in [LIK07] is built based upon Axis wireless camera platforms to support high resolution, high speed video streaming applications for WSN. The VN is equipped with a progressive scan RGB 1.3 Megapixel CMOS image

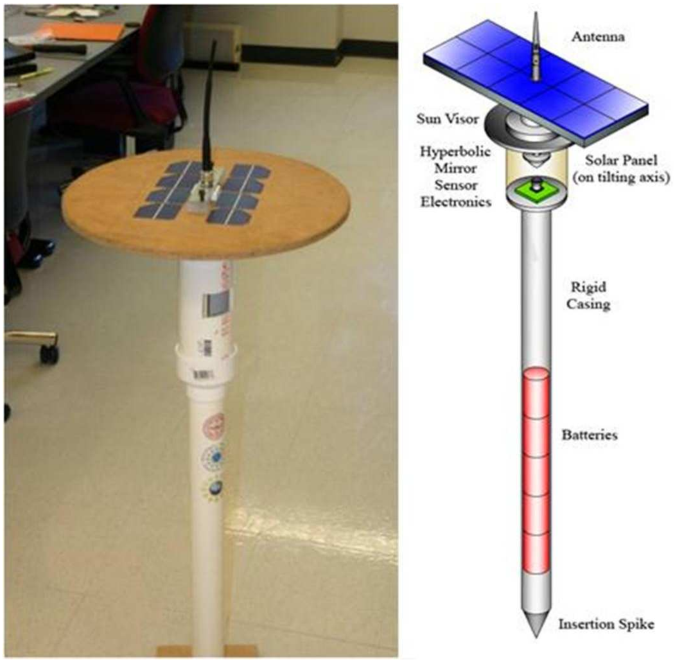


Figure 2·2: Solar powered video sensor node from [LIK07]

sensor and a IEEE 802.11b complied wireless communication module. This VN possesses a flexible directionality and zoom. It uses a catadioptric configuration with an omni-directional mirror fabricated to yield a panoramic image to achieve a wide field of view. This approach eliminates the moving parts of most pan-tilt-zoom configurations, reduces corresponding energy use, and directionality challenges. The solar power supply module of the VN is targeting to provide 500mW per video sensor node in the active (streaming) state and less than 5mW when asleep. Power consumption at an intermediate video sensor node serving to relay data to a gateway will be approximately 250mW. Longevity on stored energy of the VN is highly dependent on the duty cycle of these states. The energy harvesting module of the prototype, a solar panel and supercapacitor bank are sized to only encompass a nominal load profile/duty cycle of 5% over a 24-hour period. The design is one of daily energy input to supercapacitors with conventional disposable batteries to bridge periods of

Figure 2·3: WVSN components

cloud cover.

2.2 Video Delivery Techniques for WVSN

Video streaming in a WSN can be conceptualized as three cascaded components: video generation, video delivery, and video playback as shown in Fig. 2·3. In the remainder of this section we summarize the various techniques utilized in different components above to support video streaming applications over resource constrained wireless sensor networks.

2.2.1 Video Generation Component

Video generation component is the source of the video delivery applications. It performs the role of transforming the raw video sensing data to certain desired format. The transformed video content is usually smaller in size and less error prone to data transmissions. There are three major techniques adopted in video generation component, in-network processing, video data compression/processing and video data encoding.

- *In-Network Processing*: Closely-positioned video cameras record highly correlated video content. In-network processing acts as a filter at the beginning of video recording to remove content redundancy. Localized information is shared by neighboring video nodes to coordinate video recording for a targeted area. One example of in-network processing technique is data aggregation scheme performed in cluster based networks where the cluster head nodes dynamically merge data from all their cluster members before forwarding the content to the base station with the propose of extending the sensor node lifetime by reducing the overall data transmission volume.

- *Video Processing/Data Compression*: To save energy for data transmission, video processing is used to extract relevant features of interest prior to transmission to minimize the total data transmission energy cost. For example some video processing algorithms are proposed to condense the total video size by removing redundant information from sequential video frames.
- *Video Coding*: Video coding is critical for success of video streaming. Many coding techniques have been developed to support a variety of goals. For example, network coding is introduced to achieve high data transmission efficiency, multiple description coding (MDC) is used to achieve reliable data transmission and provide QoS guarantees. A coding technique map is illustrated in 2·4.

2.2.2 Video Delivery Component

Video delivery component carries the video content from source node to the destination node. In video streaming applications, this component is responsible to deliver the video data across the network satisfying the application requirements of delay, throughput, packet loss, and etc. This component encompasses distinct techniques for different network architectures. Video delivery component for WVSN is operating on two major network architectures, content delivery network (CDN) architecture and peer-to-peer (P2P) architecture. Here we summarize the important techniques applied in this component.

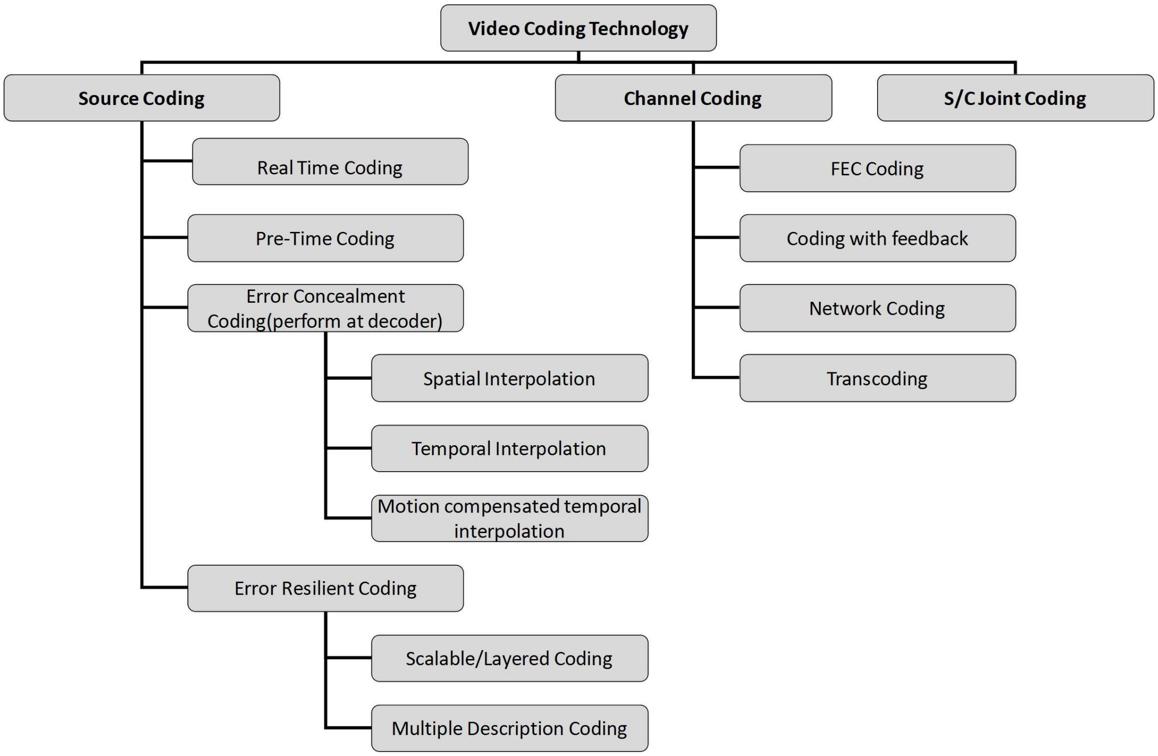


Figure 2·4: Coding technique map

Content Delivery Network (CDN)

A content delivery network or content distribution network (CDN) is a network of sensor nodes containing copies of data, placed at various points in a network so as to maximize the bandwidth for access to the data from clients throughout the network. In other words, the video content is cached at various positions in the sensor network, and the users can access the video data copy near its location as opposed to access the same video node to retrieve information. The CDN architecture can avoid the bottleneck problems near the gateway and the video sources. This architecture provides good data delivery performance for video streaming applications of recorded videos. However, it is difficult to adapt this architecture to support real time video streaming applications under some situations. For example, this network architecture is not appropriate to support live video streaming applications such as live sports broadcasting. One cannot pre-cache the live content on distributed servers before the happening of the event. Fig. 2·5 illustrates this network architecture.

The data caching techniques applied in CDN have direct impact on the data delivery performance of the application. Current data caching techniques include monolithic caching, prefix caching and intelligent frame based caching. Routing techniques are another important set of means to influence the performance of video streaming applications over CDN. Multi-path routing is the most famous one among the many strategies. It leverages multiple available data paths between the source and destination pair to deliver video data. Sending data through multiple data paths instead of one modestly increase the end-to-end data transmission capacity. The degree of capacity increasing is closely related to the management of data congestion and data interferences at the source/destination and in the middle of data transition.

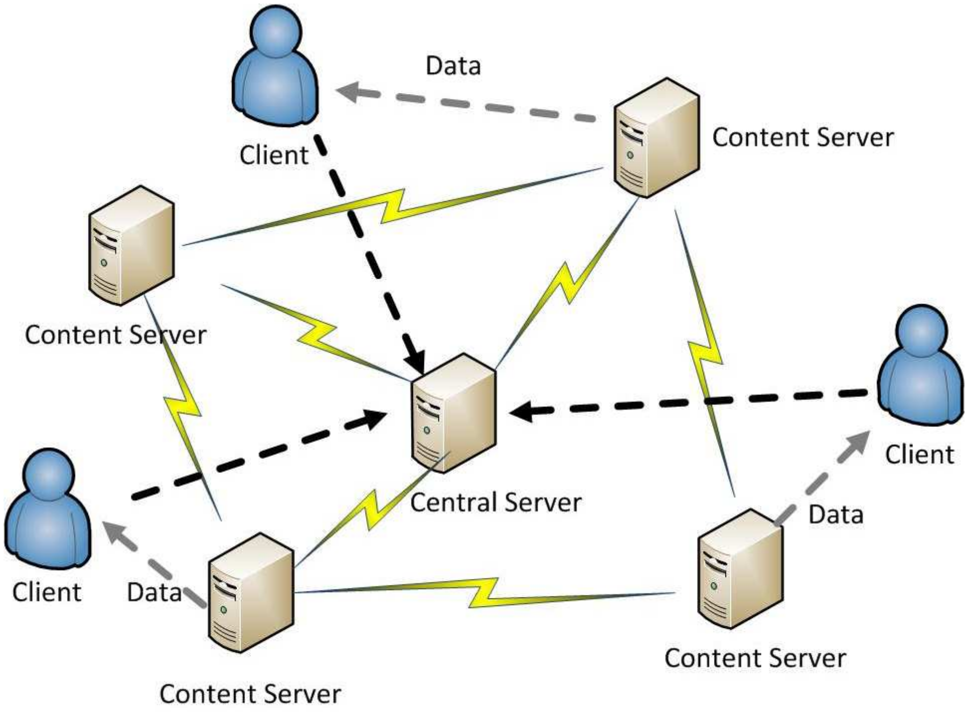


Figure 2·5: CDN architecture

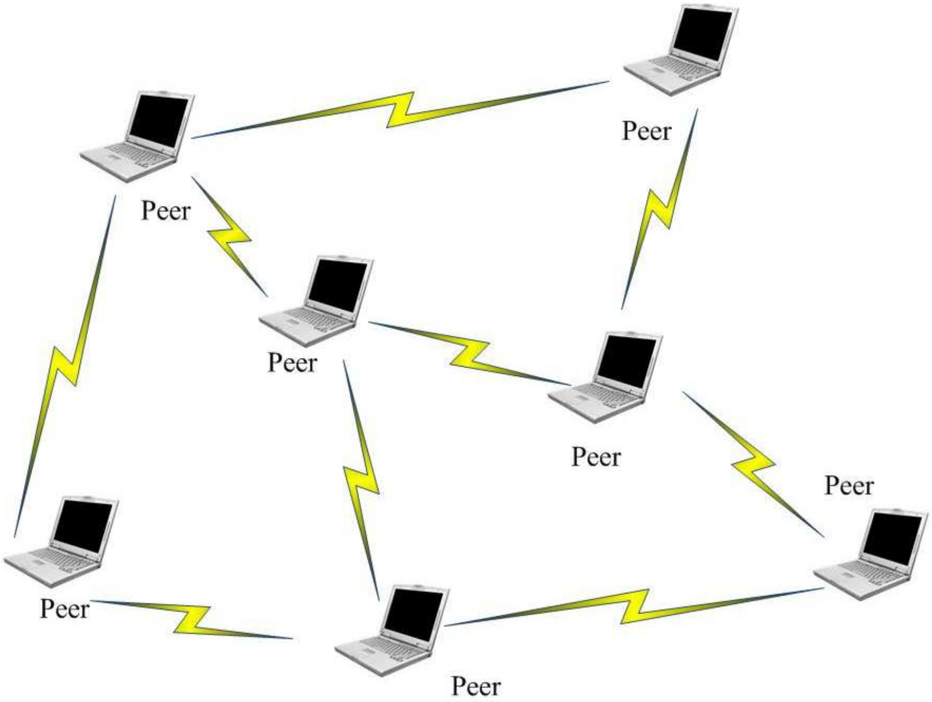


Figure 2·6: Peer to peer architecture

Peer-to-Peer Network

As we have mentioned above, CDN architecture is not appropriate for real time video streaming applications. Real time streaming applications allow the users to query existing video sensor nodes to get live videos. Such application requires information packets to be transmitted in a receiving-and-forwarding manner towards the destination node which is usually the gateway via multiple intermediate nodes. A peer-to-peer network (P2P) architecture is an candidate in this situation. Fig. 2·6 demonstrates an example of P2P network architecture.

The mainstream techniques serve in this network architecture to improve video delivery performances include:

- *Video Distortion Optimization*: Video distortion defines the visual quality degradation due to the data loss. This technique forwards video data packet to the destination under certain constraints with the minimum distortion.

- *Packet Scheduling*: A scheme usually performs on each sensor node to decide whether or when to transmit the received packet. Some approach like only forwarding the most important packet is widely accepted as a classic solution to manage network congestions.

- *Data Buffering*: A scheme to cache data on certain node. Buffering video data at the destination node is an practical way to improve video playback experiences. With a little delay at the beginning of the playback, video buffering can efficiently overcome transmission delay jitter.

- *Tree Based Data Routing*: This technique usually constructs an overlay like a spanning tree topology from the user to the interested video sensor nodes. The video sensor nodes are the leaves of the tree topology and the video data is forwarded from the leaf node back to the root node. Tree based routing techniques achieve quick response to the video queries, however, they lack the ability to adapt to environment variations. The tree topology construction overhead for frequent changing environment is usually not affordable.

- *Directed Mesh Routing*: This routing technique forms a mesh overlay on the WSN and usually employs a distributed routing algorithm to forward data from the source to the destination.

- *Coordinates Based Routing*: This technique is usually employed in conjunction with directed mesh routing technique to provide simple, fast dynamic data routing. Each sensor node has coordinates to be identified in the network. Every node performs a greedy algorithm to forward the data to the neighbor

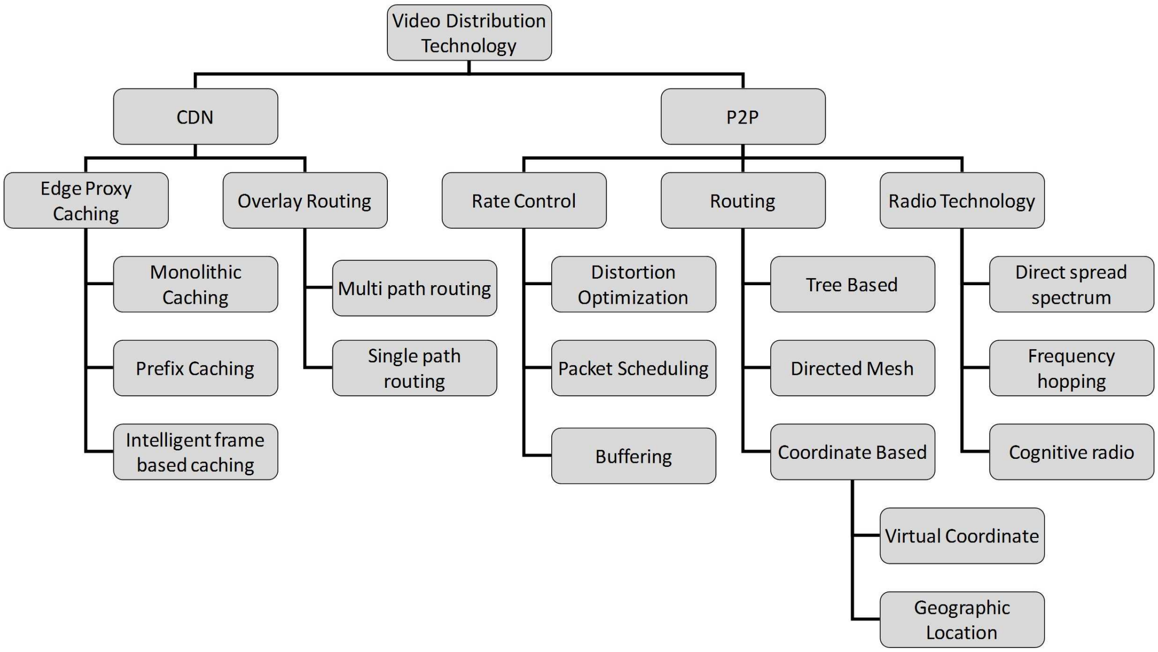


Figure 2·7: Map of video delivery techniques

node which is nearest to the location of destination.

- *Spread Spectrum*: This is a radio technology that a signal generated in a particular bandwidth is deliberately spread in the frequency domain, resulting in a signal with a wide bandwidth. There are two different ways to achieve this technique, frequency hopping and directed sequence spread. It can be used in sensor network to share a single channel among multiple users.

We plot a map of the overall video delivery techniques in Fig. 2·7.

2.2.3 Video Play Back Component

The video play back component enables users to download video data and play back video content at the same time. A local buffer is used to cache downloaded video, and successive frames are fetched from the buffer during playback. The larger the buffer, the more likely the playback can overcome delay variations (jitter) in the video transmission.

2.3 Challenge of WVSN and Solution Proposal

By analyzing the techniques we have mentioned above, we summarize four basic video delivery challenges for WVSN applications, sensor node addressing challenge, resource constraint challenge, QoS supporting challenge and admission control challenge. In this dissertation, we focus our research on the routing technique improvement to facilitate video streaming applications over WSN, we highlight the impact of these four challenges in the domain of video data routing as following:

Addressing Challenges : Unlike IP (Internet Protocol) addressing which is hierarchical, sensor networks are typically configured with flat addressing that leverages locality. Assigning global IDs for each sensor node is not always a requirement. Moreover, a hierarchical model can introduce excessive overhead when most communications are neighbor-based. And in some cases, based on the characteristic and function of wireless video sensor network, it is not necessary to distinguish one node from another. Thus, the development of routing protocols based on IP address is not a requirement for a WVSN and the addressing scheme for WSN needs to be flexible.

Resource Constraint Challenge : A typical wireless sensor network consists of a large number of autonomous, inexpensive and simple sensor nodes that are powered by batteries. Although advances of circuit design have made substantial improvements for battery life in recent years, compared to desktops in wired networks, sensor nodes are still strictly constrained by energy. Such limitations have a great impact on communication ability and routing protocol design. Path formation algorithms of the routing technique should take energy as a dimension for protocol design.

QoS Challenge: QoS is an important design factor for video routing over wireless sensor network. The design in this space focuses on transmitting data on time using multi-path and packet scheduling scheme to avoid network congestion and data loss. However, most of works in this field only focus on simple scenarios; a comprehensive study of multi-source multi-destination with unsynchronized scenario needs to be conducted to exam the efficiency of the protocol. Since in wireless sensor network bandwidth and channel resources are limited, a simple addition of bandwidth of multiple paths is only a relaxed upper bound of the end-to-end throughput that can achieve. Due to the severe intra-path and inter-path interference and congestion at the destination, the actual delay and throughput is quite different. So far, the path interference and packet collision during transmission are not carefully studied. Many existing works [CLMY07, SZZ$^+$08, ZHS$^+$08, LNLL08] simply suggest to setup disjoint paths to ease this issue. An analytical model and an efficient congestion control model that could be applied to quantify and manage the data delivery performance is more appreciate for WVSN applications. Some existing works [LNLL08, LGZ07, CPCT06] take the path length as the metric for data delivery delay; however, it is not always the case. End-to-end data delivery throughput as a QoS metric that combines the path length, delay, traffic load, and queuing time can be a better reflection of the real data delivery scenario.

Admission Control Challenge: Admission control on wireless sensor networks is another challenge. Recently, admission control algorithms have been suggested to allocate sufficient bandwidth for individual data flows in order to provide better data delivery performance for admitted data flows [YK05]. Since sensor networks share a single channel for communication, data transmission from one node may consume the bandwidth of neighboring nodes. Therefore, the bandwidth consumption of flows and the available resources to a node are no longer local concepts, but related to the neighboring nodes [YK05]. Admission control algorithms need to address bandwidth allocation in such environment of WVSN and guarantee that the admitted data flow will not downgrade the performance of existing data delivery and meanwhile not exceed the network capacity.

Based on the suggestions above, we expect to demonstrate new routing algorithms to satisfy the outlined design requirements. We believe the new routing algorithm should have a practical dynamic source-to-sink path formation scheme for video stream delivery; simple admission control strategies could be applied to allocate sufficient bandwidth to admit video streams incrementally and avoid collision of resource use for multiple streams based on the modeling of intra- and inter-path interferences. Mathematical models to measure intra/inter-path data delivery interferences are necessary to be developed and the details of these models are discussed in Chapter 4. These models will benefit the implementation of video streaming applications over the WSN while balancing the energy consumption through load balancing, in-network localized computation and path selection collaboration. With the help of these models, we develop our solutions to the video delivery challenges in two distinct data routing algorithms demonstrated in Chapter 5.

Chapter 3

Related Works

In this section we review the recent data delivery techniques to support video transmissions over WVSN. We conduct an in-depth study of these related data delivery techniques based upon their handling of the basic challenges we have mentioned in Chapter 2, addressing challenge, resource constraint challenge, QoS supporting challenge and admission challenge.

3.1 Addressing Techniques

3.1.1 Content Based Addressing

As previously indicated in Chapter 2, for many wireless sensor network applications, it is not practical nor a requirement to assign global IDs such as an IP address to each sensor node. Many investigators have recognized that the data flow within a sensor network can be characterized by the content of the data itself. A mapping between the sensor node's functionality and data content can be established during data transmission. By summarizing the data interests of participating sensor nodes, content-based routing is achieved [IGE00, CRW04]. In such a routing scheme, data-receiving nodes propagate data interests across the network in a publish-subscribe or push-pull model. A node address is represented by a set of node attributes defining its function in the application (e.g., 'with a light sensor,' 'measure rate,' etc.). Each data stream is prefixed by a structured description using typed language which defines a set of criteria for the destination node of the stream called predicates, such as 'nodes inside certain area,' 'nodes with certain brand.' The protocol establishes hierarchical property-interest tree rooted at source node. The source node pushes the data stream onto the tree and thus the stream will flow to the node with the corresponding interests. Directed diffusion [IGE00] is an example of content-based routing using the 'pull' method. Directed diffusion aims at diffusing data through sensor nodes by using a naming scheme for the data. A significant advantage of directed diffusion is the simplification of the programming abstraction and routing mechanism for propagating data to a data sink. Direct Diffusion suggests the use of attribute-value pairs for the data content and queries the sensors in an on demand basis with those pairs. In order to create a query, an interest is defined using a list of attribute-value pairs such as name, interval, duration, geographical area, and etc. A sink broadcasts an interest through its neighbors. Each node receiving the interest broadcast caches the interest for later use. The nodes also have the ability to perform in-network data aggregation to merge similar interests. The cached interests are then used to compare the received data with the values in the interests. The interest entry also contains several gradient fields. A gradient is a reply link to a neighbor from which the interest was received and is characterized by a data rate, duration and expiration time derived from the received interest fields. Hence, by utilizing interest and gradients, paths are established between sink and sources. Several paths can be established so that one of them is selected by reinforcement. The sink resends the original interest message through the selected path with a smaller interval hence reinforces the source node on that path to send data more frequently. There are many variations proposed for efficient wireless sensor network data routing based on Directed Diffusion.

GEBR [LGZ07] expands the concept of Directed Diffusion to provide global energy balancing and real-time routing for video data transmission. Their path formation process is nearly identical to that of directed diffusion; however, they introduce node energy as a criterion for data transmission and path reinforcement.

Wang *et al.* [WML07] propose synchronized pipelined transmission for video data streaming. Unlike Directed Diffusion which floods interest message to explore the optimal path, the route discovery process uses a probabilistic method. The source node periodically sends out route probing packets. The probing packets are randomly relayed to a neighbor of the current hop until they reach the subscriber node. When the predefined route-probing timer expires, the subscriber node calculates the optimal path based on all received probing packets. Although this process can largely eliminate the data load created by the path exploration stage, it performs poorly when the network is relatively large and the source and destination nodes are scarce and far from each other.

Li *et al.* [LNLL08] provide a multi-path data delivery solution to deal with the challenge of delay control in video transmission applications with another expansion of directed diffusion. Instead of using the metric of transmission time, the scheme uses a weighted metric that captures delay, interference and throughput. A time stamp is given to both interest message and exploration data. The sink station chooses exploration data whose times tamp is within a predefined threshold to satisfy the delay constraint as compared to the interest message's times tamp. The sink tags the senders of these qualified exploration messages and put them into the reinforcement path candidate pool. The reinforcement scheme in this solution supports the setup of multiple disjoint paths. However, the length of the path setup period is largely dependent on the compound metric it utilized and subject to variable delay. Furthermore, the data metric is difficult to obtain. For example, the measurement of SNR needs the power level of three terms: noise, interference, and signal strength. The authors do not provide the method to measure such terms. A simpler metric that can be directly obtained or estimated by the network layer is more preferable and flexible.

From the above exploration, we find that there are three major issues existing in the current content-based video routing technique for wireless sensor networks. First, the current techniques employ application-dependent addressing schemes. It is difficult to port data interest from one application to another unless the two applications are similar. Second, the content-based routing schemes utilize a route exploration stage using route exploration data. The motivation for this approach is to simulate data transmission and thus select what might be an optimal route. However, in video streaming applications, the prevailing conditions during route exploration can be substantially different from when video data are in transmission. In other words, a good route obtained during a data exploration stage will not necessarily be the good route during a data transmission stage. Thus, a specially designed exploration stage is required to find a good video streaming route for WVSNs. Third, content-based schemes lack the control of path selection involving overused nodes. It is very likely that multiple data paths will share common nodes since these paths become reinforced. These nodes soon become overloaded and compromise the video data transmission.

3.1.2 Location Based Addressing

The second group of addressing schemes is location-based. A number of proposed routing schemes fall into this category. Since most WSNs are comprised of nodes deployed in a known area, they have proper coordinates established that can be used to assist routing. The distance between the source node and sink node in real world can be used as a proxy for the energy cost for data delivery. There are two kinds of location-based routing strategies. The first one we call real coordinate routing. GPSR [KK00] is one such example. In real coordinate routing, each node uses geometric distance as the routing metric. This strategy establishes coordinates for each node based on its absolute (Cartesian) location. By obtaining the location information of a destination node and its neighbor nodes, senders always forward a packet to a neighbor with a shorter distance to the destination (a greedy technique). Real coordinate routing suffers from well-known dead-end problem especially in a sparse network or one with physical obstructions. The second strategy is called virtual coordinate routing [RRP$^+$03, NS03, ZCLZ04, ZLZ$^+$05]. This routing strategy applies routing metrics to reflect the relative location of the sensor nodes within the network instead of using absolute coordinates. Zhao *et al.* [ZCLZ04] propose a method in which individual node constructs a vector with elements corresponding to the hop distance to the set of pre-established landmarks. This vector is exactly the virtual coordinate of the node. The routing process is identical to GPSR (a greedy formula) except that it uses a more elaborate distance function instead of the geographic distance. Virtual coordinate routing performs well for the dead-end problem in sparse scenarios and it reduces the hardware requirements of sensor nodes. However, the virtual coordinate setup process is not easy and is energy consuming [ZCLZ04]. Thus most recent works in location-based routing continue to use real coordinates.

Cosma *et al.* [CPCT06] have an interesting application of location-based routing for video streaming. This work is not a complete solution for video transmission over wireless sensor networks but introduces a topology extraction protocol using video cameras equipped on each sensor node. There are two steps to achieve the topology extraction. First, a central node/server or gateway floods routing messages over the network and every node records routing information. After a path setup phase, every node in the network captures an image using its video camera, and passes the image through to the central node/server. This node then performs image registration to extract the topology and location of each sensor node. The result is analogous to a bird's-eye view of the global topology of the system. The authors further suggest that the global topology can be optimized for path routing and energy conservation. This scheme is creative but impractical at present. Image registration of a large number of disparate images is complex, time consuming, and potentially performed with sparseness of view. It is also likely that the extracted topology has significant error due to the limited camera resolutions, focal lengths, and fields of view.

DGR [CLMY07] is a mechanism proposed to transmit real-time video. The idea of DGR is to construct an application-specific number of multiple disjointed paths for a video node to transmit parallel FEC-protected H.26L real-time video streams over a bandwidth-limited, unreliable networking environment. Unlike traditional location-based routing algorithms using greedy routing schemes [KK00, YGE01] this work introduces a concept of "deviation angle" to spread the paths in all directions by the side of the line-proximity of the source and sink pairs. It implies that packets along some paths are likely to be forwarded to a neighbor with a greater distance from the sink. To deal with route coupling [PHST00] issues caused by interference between packets transmitted over different paths, the authors separate physical paths as far as possible. When a node receives a path setup packet, it calculates its virtual coordinates based on the location of the upstream message sender, destination, and itself. The origin of the virtual coordinates is at the upstream node's location and a reference line (x-axis) is fixed by upstream node and sink node. The angle between the x-axis and the line segment of the receiving node and the upstream node is then obtained. The upstream node chooses the node whose angle is least different from the "deviation angle" as the next hop. Deviation angle is controlled by a function with respect to hop count to ensure that the path will eventually converge to the sink. DGR uses the node's location to identify different sensor nodes. Instead of using pure location for routing decision, DGR introduces deviation angle-controlled routing to find detours. This idea is efficient for establishing multiply separated paths from source to sink. The video data can then be subdivided into multiple streams and transmitted through multiple disjoint paths to the sink.

The TPGF routing protocol [SZZ$^+$08] is another example of a greedy location-based scheme. In order to solve the hole-bypassing problem [FGG04, YLC$^+$07, JWWG07], TPGR proposes "step back and mark" process to explore possible paths to the base station and guarantees to find a route to the destination as if one exists. This protocol is designed to execute multiple times to find multiple disjoint paths from source node to sink node. However, unlike the scheme utilized by Chen *et al.* [CLMY07] which introduces a way to separate paths as far as possible, on the contrary, this scheme put these paths as close as possible to the centerline which can cause very severe path coupling problems.

According to the above analysis, location-based routing schemes are application independent. Moreover, the information required for data routing is simple and localized without necessary knowledge of the global network topology. But some variants are not entirely practical. For example, most location-based routing exploits GPS data that is often unavailable due to cost or indoor locations. Another problem for location-based routing is to deal with network holes. An efficient hole-bypassing algorithm is very important for location-based data routing over WVSNs.

3.1.3 Hierarchical Addressing Techniques

Yet another routing scheme is based on hierarchical addressing. The basic idea of hierarchical addressing and routing is to group sensor nodes into multiple clusters based on some assignment criteria. A cluster "head" is selected to coordinate communications inside a cluster and amongst different clusters. LEACH [HCB00] is a milestone protocol in this area and it inspires a large number of hierarchical routing protocols for wireless sensor network. The idea of LEACH is to form clusters based on radio signal strength and to use local cluster heads as routers to the sink. This scheme saves energy by simplifying routing in a locality and managing the propagation of data that must traverse multiple clusters. The optimal number of cluster heads is estimated to be 5% of the total number of nodes. All the data processing such as data fusion and aggregation is local to the cluster. The assignment of cluster head is rotated in order to share the energy burden of this function.

Akkaya and Younis [AY03] provide a three tier network architecture to route data as illustrated in Fig. 3·1. Before network operation is established, sensor nodes are grouped into clusters. Each cluster has a gateway node. Sensor nodes only route data to the gateway nodes, and gateway nodes are responsible for routing data to the central command node. The sensor nodes do not require globally unique IDs. The path setup process is a centralized scheme. The cluster's gateway node is assumed to know the cluster's topology and link state between any two nodes inside a cluster. The idea is to find a detour path to the gateway instead of transmitting data directly. However, the authors neglect the fact that all the sensor nodes of a cluster are inside the radio range of the cluster head and can thus potentially cause severe interferences for concurrent data transmissions toward the gateway.

Politis *et al.* [PTDK08] describe another hierarchical video data routing scheme. The network architecture setup is a slight modification of architecture of LEACH. Instead of using a direct link between a cluster head and base station for data collection, cluster heads are allowed to establish links to each other. Hence, a video sensor node can select a number of available paths through other cluster heads in order to transmit its data to the base station. This modification decreases the transmission power of a cluster head for shorter-range communication and saves energy. This paper can be recognized as a complementary work of Akkaya and Younis [AY03]. Instead of

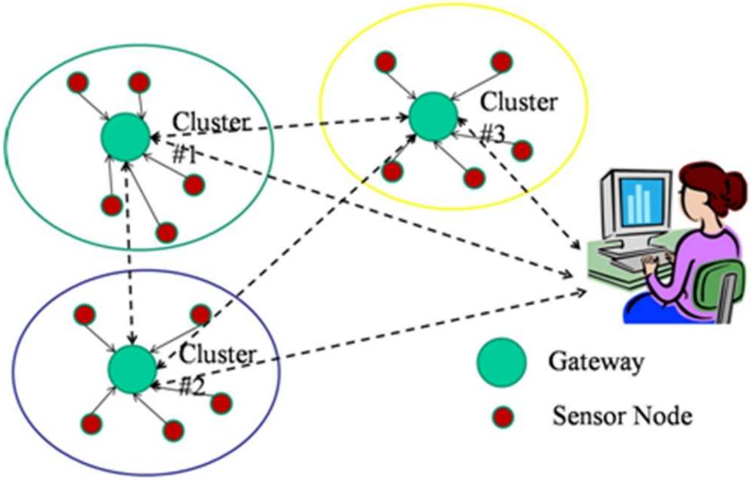


Figure 3·1: Three tier network (Adapted from [AY03])

addressing routing inside a cluster, Politis *et al.* propose a scheme to address routing from cluster head to a base station. The algorithm is adapted from the work of Chen *et al.* [CCL04] and it uses a centralized algorithm based on the knowledge of the network topology, link capacity and link delay. Unfortunately, the scalability of such routing algorithm is not sufficiently addressed.

Besides the communication management benefit, another inspiration for video delivery with hierarchical network architecture is efficient for redundant data removal. High data rates of a video stream inevitably will cause rapid energy consumption by sensor nodes. To avoid node failure and network disruption due to battery depletion, an aggregation-driven routing scheme is proposed. In a hierarchical network, nodes from lower level send their data to higher-level nodes for data aggregation. Nodes at a higher level are then responsible for comparing and removing redundant data from each stream and compressing the data before sending them to their upper-level controllers. Navda *et al.* [NKGI06] propose one such routing scheme. The basic idea is to merge multiple flows at early stage of data transmission and form spatially separated paths to minimize inter-path radio interference. The root node first floods a message to setup a spanning tree. Source nodes attempt to send data to the neighbors of the root through nodes which are not carrying any flows and do not have contending transmitters in their radio range. If the node cannot find such path, it will route to the nearest node that carries the fewest flows. Unlike traditional schemes which route data through disjoint paths, this scheme tries to merge data at an early stage of the data transmission and form spatially independent routing paths.

From the above analysis, the main advantage for building a hierarchical network topology is for data fusion and data transmission management. However, the data fusion costs for video are not very carefully studied in these prior works. According to Liu and Das [LD06], the fusion cost of video can be equivalent to that of transmission. The shortcoming of hierarchical network organization is the unbalanced network load distribution across participating nodes. Such a network would result in unfair resource consumption for different nodes in different layers.

3.1.4 Global ID Addressing Attempts

The absence of globally unique IDs prevents the integration of WSNs and IP-based networks. In order to solve this issue, schemes have been investigated to assign unique network-wide IDs [DBHR05]. However, these ideas face the risk of incompatibility with the established standards of the Internet communication protocols. Another viable approach is leverage IPv6 to identify a sensor node; a sensor node can concatenate its cluster ID with its own MAC address to create a full IPv6 address. However the 16 byte address field of IPv6 potentially introduces excessive overhead in each data packet, but in this way, the existing Internet solution for video transmission can be adapted to video transmission over a wireless sensor network.

3.2 Energy Efficient Routing Techniques

As stated at the beginning of this chapter, energy consumption is always the primary concern of wireless sensor network application design. This section considers recent efforts in energy-efficient routing.

The energy saving idea from GEBR [LGZ07] is to send data through the path with the shortest length and most longevity. The longevity of a path is measured with Minimum-Path-Energy (MPE), which is the minimum energy of all the nodes along a path. The interest message generated by a sink destination contains hop count requirement, MPE value, and path length. When a source node receives a set of interest messages, it calculates the maximum value of the MPE from different paths whose path length is smaller than the hop count requirement. The hop count requirement is a proxy for a real-time streaming requirement. The authors assume that a path with fewer hops will yield the lowest data transmission delay. Afterwards, the source station sends exploration data that contains the maximum MPE value (BMPE value). The node only forwards the exploration data to its upper stream node if the neighbor's energy is larger than the BMPE value. This process ensures that exploration data only goes through the most survivable path and balances the traffic load. However, the pure BMPE-based routing does not consider problems with packet collision. If multiple source and sink nodes exist, the BMPE path will carry a heavy load and interfered by other BMPE paths causing significant packet losses, delay, and energy waste. Fig. 3·2 and Fig. 3·3 illustrate examples of BMPE generation and optimal path selection:

Wang *et al.* [WML07] consider energy conservation via the reduction of packet retransmissions in the presence of node failures. A synchronized and pipelined transmission scheme is proposed with flow control. They use a secondary buffer to ensure the maximum retransmission distance is not more than twice the size of the failed

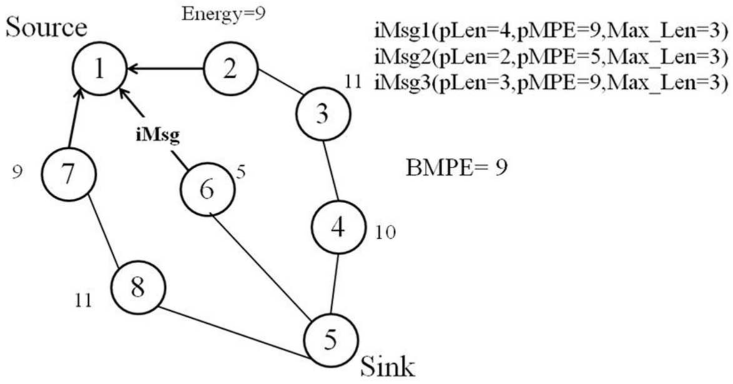


Figure 3·2: BMPE generation (Adapted from [LGZ07])

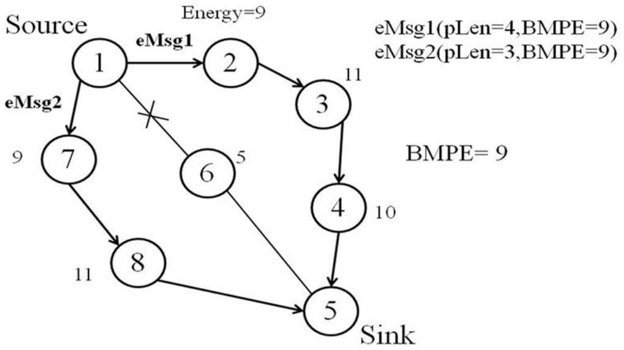


Figure 3·3: Optimal path selection (Adapted from [LGZ07]) node's buffer. However, it is not clear for the scalability of their synchronization requirement. The flooding of synchronization messages will not be efficient for large networks. Instead, synchronizing node for a particular route seems more feasible and efficient. Buffering packets at the neighboring nodes is a good idea; however, the means to update the data in secondary buffer according to the data update of the primary buffer is not demonstrated.

Cosma *et al.* [CPCT06] propose that every node maintains a record of their neighbors' energy level and hop count to the server. Any node with a relatively high energy ($> 20\%$) will be in the candidate set for next hop during routing. The candidate with smallest hop count will be chosen as the next hop. This solution, although more conformal to energy fairness on a per hop basis, does not prevent instantaneous overuse of a path. Moreover, for long-lived video streams, there can be significant change in energy levels for nodes participating in multihop routing. How to adapt to short-term energy change and contention for shared paths are not considered.

Based on the first-order radio model [HCB00, SCK06, FGG04], Shu *et al.* [SZZ$^+$08] derive the node energy consumption rate as a function of data rate and radio transmission range. The objective of this work is to manipulate the radio transmission range to satisfy the energy consumption constraint to achieve a target network lifetime. Results indicate that a greater transmission range leads to a lower latency. A node will prefer to use the maximum radio range if the corresponding node energy consumption rate is lower than the expected rate. The radio range and energy consumption rate are computed based on two critical terms: the energy cost to power the transmitter circuitry to send one bit, and the energy cost for transmitter amplifier to send one bit. These two terms are difficult to measure; inaccuracies will interfere with the success of the routing algorithm.

Politis *et al.* [PTDK08] introduce the cluster-head energy consumption model.

The authors propose a packet-scheduling algorithm that allows a source node to drop packets queued for transmission in order to avoid downstream congestion. This congestion control strategy is conducted with considerations of the residual energy of the cluster-heads along the path.

Akkaya and Younis [AY03] address energy efficient routing in the context of a single cluster. They assume that all sensor nodes in a cluster are within the radio range of its associated gateway. A link cost function is defined based on the consideration of delay, residual energy, distance, and other factors. The gateway node is assumed to know all link states; Dijkstra's algorithm is applied to find the least cost path between sensor nodes and gateways. The gateway continuously monitors the available energy level of every sensor node active in data processing, sensing, or relaying. Rerouting is triggered by the depletion of energy for an active node. But the details of the node residue energy monitoring and energy metrics are not disclosed in this paper.

In summary, we find some practical limitations of many of the exiting energy-constrained routing techniques. Either there is no clearly established energy metrics or the existing metrics are difficult to obtain robust measurements. Designing a practical energy consumption model and residual energy monitoring protocol would be very helpful for energy-efficient video data routing.

3.3 QoS Techniques

Video and audio data transmission requires certain quality of service (QoS) achievement in a WVSN, especially when streamed continuously. For example, a streamed video must deliver each frame to the user on-time to achieve continuous playback. Unfortunately, there are many uncertainties in WVSNs that can cause significant delay and compromise video playback quality. In the following we discuss techniques that have been to deal with these challenges.

Multi-Path/Multi-Priority

The first class of QoS provisioning routing techniques improve the end-to-end data delivery delay for video streaming applications via a combination of the path formation and packet scheduling strategies. These works usually explore multiple paths between source and destination and select the best ones to improve video streaming quality. The packet scheduling scheme is based on the importance of the data packet. If a network congestion is expected, the least important packets could be dropped to relief the network load without major data delivery performance downgrade [SZZ$^+$08, PTDK08].

Li *et al.* [LNLL08] propose to deliver MDC-coded data through multiple selected paths to overcome packet loss due to network congestion and transmission delay. Cross-layer design and disjoint routing path selection are combined to provide better QoS. During the path exploration period, the algorithm tags identified paths to place in a pool and sorts them in an ascending order according to the path length. The first N shortest paths that satisfy the data delivery cost constraints are selected to route data. N is a slight larger than the required number of paths for the transmission, since some candidate paths may not be reinforced if disjoint nodes cannot be found or the delay exceeds the playback deadline. During the path reinforcement stage, if two nodes happen to reinforce the same node, the second reinforcement will be nullify to ensure disjoint path selection and avoid routing loops. To deal with bottleneck problem shown in Fig. 3·4, Li *et al.* [LNLL08] eliminate the existence of bottleneck links via a node deployment density control.

DGR [CLMY07] uses multi-path transmission to achieve low latency. A FEC coding scheme is used to recover data due to packet loss or data corruption. Although multi-path transmission can expand available bandwidth, it does not demonstrate how to resolve interference between adjacent data paths that converge at the destination.

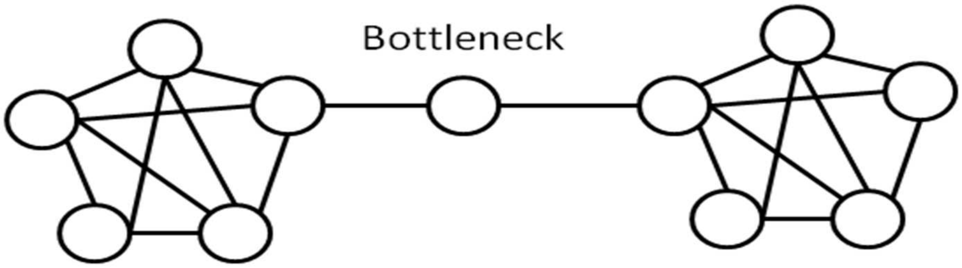


Figure 3·4: Network bottleneck scenario (Adapted from [LNLL08])

Base on TPGR [SZZ$^+$08], L. Zhang *et al.* [ZHS$^+$08] use a multi-priority multipath selection scheme for video streaming for WSNs. In this work, a priority index is assigned to different paths based on end-to-end delay. Streams are identified and prioritized based on their ability to monitor an event. Streams are also split into audio and video components. A mapping scheme is implemented to assign a higher priority path to a higher priority stream. This technique lacks the discussion of the time complexity for its path discovery algorithm and the interference management between adjacent disjoint paths.

Akkaya and Younis [AY03] propose a link cost function that considers link delay. The idea is to adjust the bandwidth split ratio r to let the average end-to-end delay satisfy the delay requirement. The average delay estimation is based on an implied assumption that each packet could be delivered to the destination through multiple hops without any corruption or loss. If packet collision or loss exists, the average delay of each packet will increase beyond their model's prediction and cause inaccurate calculation of the bandwidth split ratio value r .

Politis *et al.* [PTDK08] propose to transmit the most important video packets through multiple paths to the destination to achieve a QoS target. A video distortion model presented by Politis *et al.* [PTP$^+$07] is applied to estimate the importance of different encoded video packets of the H.264/AVC stream. The algorithm developed by Fang *et al.* [FGG04] is used to find multiple paths from source to destination that can satisfy the video bandwidth requirement. A baseline packet-scheduling algorithm is introduced to manage the transmission by dropping excessive video traffic based on the packet importance. For further improvement, the authors developed an packet-scheduling scheme that factors the residual energy of cluster heads in the algorithm. This approach can reduce the distortion of the decoded video sequence by deciding which and how many packets will be dropped according to transmission rate limitations and power failure of the nodes prior to transmission. Politis *et al.* [PTDK08] borrow the multi-path formation scheme developed by Chen *et al.* [CCL04]. The path formation algorithm considers two factors: end-to-end delay and aggregate bandwidth. This scheme guarantees to find multiple paths from source to destination that satisfy the bandwidth and delay requirements. However, due to the uncertainty of wireless channels, the actual throughput of the multi-path is usually less than the aggregated value. This work is also limited to single stream applications that rely on distortion-reduction algorithm for a specific video encoding scheme.

A rate-based flow control algorithm called doubling and reducing-by-half is proposed by Navda *et al.* [NKGI06]. By monitoring the throughput of per-flow at the root, the source is allowed to double the load of the flow until the throughput of other flows is influenced and dropped under a threshold. Thereafter, the source node reduces load in the next step to half of the previous incremental load until throughput per-flow matches again. Packet scheduling based on packet delay requirements and flow priority is proposed to improve data delivery performance which will drop packets in a early stage during data delivery to manage network congestion.

Chai and Ye [CY07] adapt Internet streaming models to the wireless mesh context. Each node is assumed to have a unique IP address. Data consumers contact a central server for authorization and channel setup. The central server coordinates different video servers to stream video data to the corresponding users. A media transfer server working at background covert the various multimedia file formats to the internal streaming file format and upload data to the distributed file systems or data storage center. The authors adapt RTSP/RTP protocols used in Internet video streaming to stream the video data over wireless mesh network. The only difference is that the underlying transmission mode of RTP packet is replaced by TCP connection to overcome the high error rate and bandwidth fluctuation of the wireless channel. Authorization process and packet scheduling algorithm for multi-users are introduced to solve the challenge brought by RTP-on-TCP transmission such as robust and secure connection, delay jitter and system blocking issues. Non-important packets can be dropped before transmission to avoid potential network congestion and improve network performance.

Congestion Control

Congestion control can improves data delivery performance in throughput achievement. Congestion control manages sensor node contentions over a shared media for data transmission in wireless sensor networks to lower the probability of network congestion. In Recent years, a class of works known as back-pressure algorithms [TE92] have received much attentions for jointly routing and scheduling over multi-hop wireless networks to achieve optimal network throughput and load balance for data delivery. This type of algorithms is based on the algorithm introduced in [TE92] to avoid network congestion and achieve network stability.

Jiang and Walrand [JW10] propose a distributed MAC scheduling algorithm to realize the maximum throughput of the network. The algorithm is combined with congestion control to achieve the optimal utility and fairness of competing flows. However the analysis is based on idealized CSMA model and the practice and implementation complexity is not properly addressed.

Wan *et al.* [WEC03] propose a medium sampling algorithm for congestion detection in CODA. In particular, each node periodically samples the medium during a sensing epoch (defined as a multiple of the packet transmission time), and calculates channel loading conditions based on the number of times that the channel state is busy. Once congestion is detected, a node needs to notify its upstream node of the congestion status. Once the congestion happens the congested node will send a message to the upstream node to stop data transmission and the congested node also stop receiving any incoming packets until the transmission load is reduced. A pitfall of this strategy is that the feedback message to the upstream nodes is not guaranteed to be delivered under network congestion. When congestion happens, it is very hard for the congested nodes to grab the medium and send out the message [LPC$^+$09] and this issue can be exaggerated when the carried flow is bandwidth consuming video streams.

Wang *et al.* [WSL05] propose SenTCP to capture early signs of network congestion and avoid congestion collapse. SenTCP requires nodes to periodically issue feedback signal which carries local congestion degree and buffer occupancy ratio. Based on this information, neighboring nodes are able to adjust their sending rates and relieve congestion at its early stage. However, using explicit periodical feedbacks increases network control overhead.

Although back-pressure algorithms can achieve optimal throughput of the network, it is not attractive to apply these algorithms for the network congestion management over WVSNs. Back-pressure algorithms based on [TE92] exploit all feasible paths between each source and destination to maintain the network stability and balance the network load. Such extensive exploration can cause packets to be sent over unnecessarily long routes with unacceptable end-to-end delay for streaming applications [YSR09]. Further, many back-pressure congestion control algorithms involve cross layer optimizations based on simplified media access models [JW10]. It is a challenge to implement back-pressure algorithms on real systems and achieve comparable results declared in the theoretical analysis and simulations [WJHR09]. In addition, back-pressure algorithm will drop packet at random locations on intermediate nodes during the transmission [TE92]. However, video packets usually depend on each other to be able to decode at the receiver. The missing of one important data packet will result in the useless of a group of data packets [LPC$^+$09]. Dropping packets in the middle of data delivery will delay the retransmission of important packets for video decoding. As opposed to back-pressure algorithms, in this dissertation we will present a simple congestion control strategy. It is easy for implementation. It applies a simple distributed TDMA based packet scheduling strategy to yield predictable data delivery throughput performance. This TDMA packet scheduling strategy can be implemented as software on off-the-shelf wireless hardwares as demonstrated in [DM09]. This congestion control strategy limits the occurrence of network congestion only appears at the beginning of the data path for a single stream scenario and at the first mutex pair for concurrent data streaming scenario over adjacent paths. This early stage congestion control strategies facilitate quick retransmission of the important packets from the video source node to improve video delivery performance.

3.4 Admission Control

Recent years, admission control algorithms have received much attention to provide improved data delivery performance via allocating sufficient bandwidth resource to data flows. Admission control algorithm over contention-based access networks such as wireless sensor networks can improve the predictability of the data delivery performance [OL10] and provide end-to-end data delivery QoS guarantees. The resource allocation for admitted data flows in admission control algorithms is based on the estimation of available bandwidth. In this section we introduce the related techniques to achieve admission control.

Lee *et al.* [LAZC00] present the design, implementation, and evaluation of INSIGNIA, an IP-based quality of service framework that supports adaptive services in mobile ad hoc networks. The framework is based on an in-band signaling and soft-state resource management approach that is well suited to supporting mobility and end-to-end quality of service in highly dynamic environments where the network topology, node connectivity, and end-to-end quality of service are time varying. The admission control in this framework allocates resource for an end-to-end data flow based on the measured channel capacity/utilization for all the nodes along the route and requested bandwidth. If the minimum available bandwidth of the nodes along the route is larger than the requested bandwidth, this data flow will be admitted otherwise it will be rejected. But this work focuses on high level issues and do not address the bandwidth allocation and estimation for end-to-end data flows.

Mas *et al.* [MK07] describe a probe-based admission control scheme for differentiated Internet services that offers a reliable upper bound on packet loss, as well as small end-to-end delay and delay jitter. The admission control supports host mobility and multi-cast communications without adding any complexity to the network nodes. The host sends probe packets before starting a new session to measure the data path capacity and decides about the flow admission based on statistics of probe packet loss.

Sun *et al.* [SCY$^+$05] propose a comprehensive solution for the end-to-end data path capacity estimation for WSN named SenProbe. SenProbe relies on the time dispersion between estimation packets to provide data path capacity estimation for WSN. It estimates the maximum achievable rate between two endpoints in wireless sensor networks by injecting packet trains and analyzing the dispersion between the packets. The length of this packet train depends on the interference range and the transmission range of the specific radio technology.

Pathrate [DRM01], Pahtload [JD02] are tools have been developed to facilitate direct measurement of data path capacity. However, according to the study of Gupta *et al.* [GWMC], measurements of available capacity in wireless networks often give inaccurate results especially for multiple hops.

Unlike active measurement strategies that send probe packet to measure data path capacity directly, Ahn *et al.* [ACVS02] propose SWAN, a stateless network model which uses distributed control algorithms to deliver service differentiation in mobile wireless ad hoc network. SWAN estimates the data path capacity based on the local monitor of the traffic rate and employ a sender based admission control strategy. The admission controller located at the source node sends a request packet toward the destination node to estimate the end-to-end bandwidth availability. Available bandwidth is estimated according to the difference between a predefined data path congestion rate and current traffic rate. Once the request packet traversed the entire data path the minimum bandwidth (bottleneck bandwidth) will be obtained by the admission controller to make admission decisions.

Kazantzidis *et al.* [KGL01] measure the available bandwidth in terms of permissible throughput. It provides a mechanism to measure the link utilization within a window of packets, by calculating the idle time and the window duration as time progresses, and shows that it provides a robust and correct permissible throughput measurement. [KGL01] demonstrates a timely propagation mechanism to inform end nodes about the permissible throughput for a data path. Such information can be leveraged by admission controllers to provide data delivery guarantees for admitted data flows.

Although [ACVS02] and [KGL01] provide end-to-end bandwidth estimation for data flows according to their simulation results, the admission controller only makes decisions based on the available resource on the local node along a route. However, in wireless sensor networks, the data transmission on a local node will also consume the bandwidth of its neighbor nodes. Bandwidth allocation for a single flow should not only satisfy the capacity constraint of the local route but also must not exceed the bandwidth capacity of adjacent routes.

Yaling *et al.* [YK05] improve [ACVS02, KGL01] by allowing its admission controller to consider both local resources and resource at neighbor nodes when making admission control decisions. Contention-aware admission control protocol (CACP) estimates the available capacity by measurements from each node of the amount of time that channel is busy. Active and passive strategies are introduced to get the available bandwidth not only for local route but for adjacent ones. Accurate capacity information feedback at the admission controller avoid incremental resource allocations violating the network capacity constraints and maintain a desired quality of service for concurrent data flows

Compared to above techniques, admission control strategy present in this dissertation employ an end-to-end monitoring and measuring mechanism to estimate achievable throughput performance of a potential data path which is more responsive for data delivery QoS over wireless sensor network for the admission decision making. Previous works of admission control [LAZC00, MK07] employ a greedy resource allocation strategy to admit additional data flows as long as the measurement of available bandwidth can satisfy the initialization requirement of a new data flow. However, the measurement of bandwidth is usually not very accurate [GWMC] and the sensor network environment may change greatly during the data delivery session. Moreover, delivery of a single video stream is likely to take up the entire path capacity of a low rate wireless sensor network that employs 802.15.4 communication protocols. Therefore, instead of adopting greedy resource allocation strategies, our admission controller conducts a more conservative scheme that allows at most two video streams to be carried by a single node. Our scheme performs admission control at both node and flow levels with straightforward approach where the decision to either accept or reject an admission request is based on accurate estimation of achievable data flow throughput performance along a route. The throughput estimation in our admission control is not based on the active measurements of a data path that involve injection of extra probing packets [MK07, SCY$^+$05, DRM01]. We exploit off-line throughput computation based on queuing models to minimize the estimation overhead. Simulations demonstrate that the combination of our admission control and congestion control strategies improve the predictability of throughput performance of a route and result in accurate and timely end-to-end throughput estimations to make admission decisions.

Chapter 4

Analytical Model

4.1 Introduction

4.1.1 Motivation

With the development of low-power, low-cost CMOS imaging sensors, there is a great potential for multimedia streaming applications of wireless sensor networks (WSN). Examples include homeland security, ecological habitat monitoring, traffic management, etc. The typical operation of these applications is for a sensor to capture video and audio and to send them in a compressed form to a consumer elsewhere on a wired or wireless network. Data is not necessarily buffered for time-shifted delivery but are instead available for immediate delivery, in real-time, once data begin to arrive at the receiver.

SensEye [KGSL05], and Panoptes [FKFB05] are two early efforts in this research direction. SensEye is a scalable multi-tier sensor network with three different platforms, low power mote, more capable stargates, and embedded computers running Linux to support an object detection application with video cameras from low resolution to high resolution. Panoptes is another example of building scalable video capturing systems over wireless sensor networks using off-the-shelf hardwares. Panoptes employs 802.11 communication protocols to deliver video data and SensEye improves video delivery latency with a mix of 802.15.4, 802.11 and Ethernet communication protocols for different network tiers. Although these system implementations demonstrate the capability of a wireless video sensor network, the data transmission challenges for video delivery especially over bandwidth constrained wireless sensor networks are not substantially investigated.

4.1.2 Challenges

Bandwidth Constraint

A typical wireless video sensor network is comprised of tens of hundreds of sensor nodes which are small in size with limited computation capability and equipped with low power communication radios. Usually, there is only one channel with limited bandwidth shared by all the sensor nodes for communication. However, the bit rate of a video stream is usually several orders of magnitude higher than scaler data such as temperature, light intensity delivered over traditional wireless sensor networks. Transmission of a single video stream can easily consume the entire capacity of the data links along the data path. The conflicting constraints of the wireless channel and the data rate requirements of video are the primary challenge here, but are exacerbated by simultaneous streams in the same delivery infrastructure. As a consequence multiplexing a sensor node to deliver the data for multiple concurrent video flows will not improve data delivery performance. On the contrary, it increase the probability of network congestion. Additional base stations can be leveraged to support multiple concurrent data streams. Mobile base station deployment converts the traditional WSN data deliver model from many-to-one to one-to-one model that maximize the available bandwidth for each stream.

Congestion Control

The high bit rate of video data delivery and multi-hop transmission make the WVSN prone to congestion due to both intra-flow and inter-flow interference. The congestion gets more serious when there are multiple flows and traffic. Network congestions in sensor network are usually resolved via congestion control algorithms. Traditional congestion control algorithms based on back-pressure [WEC03, BJB04] are reactive algorithms that achieve hop by hop on/off flow control. Congested nodes send a transmission-off signal to neighboring nodes and stop accepting further packets. When the load reduces, a transmission-on signal is sent and packet flow resumes [Jai92]. Such control strategy is not appropriate for bandwidth consuming video data delivery. First given that all the sensor nodes only share one radio channel to communicate, once the congestion situation happens, it is very difficult for the congested sensor node to access the media and send the signal message to the upstream nodes. Second, back-pressure algorithm is intended to achieve the optimal throughput and network load balance for the entire network which compromises the data delivery delay for local streams [YSR09]. The random packet forwarding strategy based on sensor node buffer differences makes it difficult to provide QoS guarantee for individual streams. The extra storage space required to buffer video packet on sensor node increases the cost of network operation and the random location for packet drop increases the challenge of data delivery management. Moreover, back-pressure algorithms involve cross layer optimizations that is complicated for implementation on real systems to realize its unrealistic assumptions [WJHR09]. As a consequence a simpler and easy to be implemented congestion control scheme over real systems is preferred for video streaming applications over WSN to improve throughput performance for individual video streams.

Cost-Efficient QoS Provisioning with Admission Control

In this dissertation, our objective is to provide throughput guaranteed data delivery for individual video streams via admission control strategy with optimizations for data delivery cost. However, guaranteed throughput performance of video data delivery is difficult to achieve due to the lack of management of intra-flow and inter-flow interferences. To support QoS guarantees for end-to-end flows, network congestion control approaches need to be combined with admission control which supports end-to-end resource allocation along the route of a flow [YK05]. Moreover, we explore the option of deploying mobile base stations to improve the video delivery performance. Mobile base stations are employed in many previous works [WBMP05, GDP04, RDPV] to optimize the data transmission efficiency but the price of mobile base station deployment in these applications are ignored. In this dissertation, our path formation algorithms cooperate with simple admission and congestion control strategies to form throughput aware data paths that are constrained by the cost of data delivery considering both nodes' energy consumption and mobile base station deployment expense.

Contributions

In this chapter we first propose a single path throughput estimation model that suppresses intra-path interference with a distributed TDMA packet scheduling scheme and a simple congestion control strategy at the beginning of the path. This simple congestion control strategy avoids data congestion at the intermediate nodes by dropping packets only at the beginning of the path and does not require extra storage space on the sensor nodes to buffer congested packets. Video packets usually depend on each other to be able to decode at the receiver. The missing of one important data packet will result in the useless of a group of data packets. Compared to random packet drop locations for back-pressure algorithms, dropping packet at the beginning of the path as in our strategy improves the data delivery performance by allowing quick retransmission of the important packets from the source. A simple admission control strategy is applied to prevent sensor nodes adjacent to an active data path from concurrently transmitting video data. It allocates entire data path capacity to carry a single video stream with the propose of providing the best end-to-end data delivery throughput performance for individual video streams. This simple admission control strategy eliminates the inter-flow interferences and enables single path throughput estimation model to make accurate end-to-end throughput performance estimation. This estimation model can be employed by path formation algorithm to construct isolated throughput-aware data delivery paths for video streaming applications.

Extending this throughput estimation model for isolated data paths we present a complimentary data transmission interference paradigm to estimate data delivery throughput via handling more complicated video streaming scenarios. This paradigm quantifies end-to-end video streaming throughput degradation under transmission contention over adjacent data paths. Similar to congestion control in a single path estimation model, a simple mutex system based congestion control scheme is proposed to avoid network congestion at an early stage of data delivery. This congestion control strategy guarantees that the packet drop only happens at the first pair of mutex systems along the path. An admission control scheme is combined to allow at most two adjacent data paths to have tolerant inter-flow interferences and avoid network congestion on the intermediate nodes. It enables the model to precisely estimate throughput performance of a route according to factors such as path positions, data transmission rate, and length of concurrent data paths.

4.2 Network Model and Assumptions

This section introduces the basic elements of the network model that form the fundamentals of the analytical throughput estimation paradigms. The network structure, the mobile gateway deployment considerations, the radio model assumptions and evaluation metrics are demonstrated in details in this section.

4.2.1 Network Structure

In this dissertation, we assume that the video sensor network is a homogeneous fully functional network. It implies that the sensor nodes in the network are identical to each other, and are capable of video sensing, computing and communication through radios with low data rate. The network has a flat architecture form a mesh network. Each node in the network can only communicate with its neighbors. A multi-hop data path is on demand to delivery data packets from source to the destination if the destination is out of communication range of the source node. Each sensor node plays the role of both a video source and a data relay node and multiple video sources may concurrently send packets to the gateway through the same infrastructure. However, the data produced by these video sources has the potential to overwhelm existing wireless infrastructure. This scenario arises and is particularly severe when multiple sources attempt to share a common path in a multi-hop mesh network. As a result, an exclusive multi-hop data path is dedicated to each video stream intended to satisfy bandwidth requirement of the video streaming application. However, in this network, traditional sensor network data extraction paradigm with single network gateway is not appropriate to support multi-stream, multi-source video delivery applications. We explore the option of using distributed mobile base stations to enhance the data egress of the network and load balancing. Fig. 4·1 demonstrates the data delivery scenario in this network architecture.

In Fig. 4·1, node S is a sensor node equipped with a video sensor. Node D is the primary mobile station to extract data from the WVSN. Node D is out of communication range of Node S, therefore, video streaming data from node S will travel through a multi hop path before reaching node D. The routing algorithm is required to identify a data path that will satisfy the throughput requirement of the streaming application. Unfortunately, if multiple data sources are concurrently

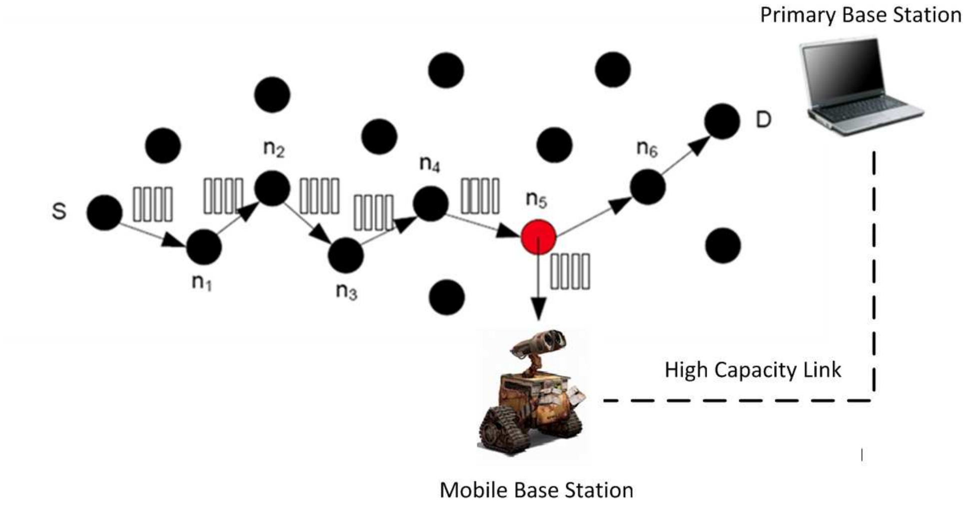


Figure 4·1: Data Delivery Scenario in WVSN

streaming video data to the same base station, the node near the primary base station will become heavily loaded and a bottleneck of the network. As a result, we explore the opportunity of sending more capable mobile base stations to take over the data forwarding near the primary base station.

4.2.2 Mobile Base Station Placement

Mobile base stations have been leveraged to improve the energy efficiency in previous works [VVVV05, WBMP05, YLC$^+$02, KAK03, RDPV, GDP04]. These works solve the network life time optimization problems under the constraints of the mobile base stations' positions. Some works as in [YLC$^+$02, KAK03] assume random work of the base stations, some others [WBMP05, GDP04, RDPV] assume prior knowledge of the total number of the mobile base stations or their locations. These works only focus on the energy optimization on the data transmission side. They ignore the cost of the mobile base station deployment that can increase the total cost of the network operation. In this dissertation, we have no constraint on the positioning and the total number of mobile base stations. We explore the option of deploying mobile base stations to improve the data delivery performance under the constraint of data delivery cost. The mobile base stations have direct high speed link with primary base station. There is only one primary base station in the network. We measure the cost of the mobile base stations' deployment with their distances from primary base station. In this dissertation, we identify positions of the deployment of mobile base stations that can minimize the cost of video data delivery combined with both data transmission energy consumption and mobile base station deployment expenses while achieve the required end-to-end throughput performance.

4.2.3 Radio Model and Basic Assumptions

Sensor nodes of the same network are likely to be equipped with the same radio module. In this dissertation, we assume a round radio communication range with radius R for all video sensor nodes in the network. For simplicity, we assume that the interference range of each sensor node is the same as its communication range. Although this assumption is a bit strong it can be relaxed in our routing solution in Chapter 5. We assume that all the sensor nodes within the network share the same channel for communication. Sensor nodes can not transmit and receive data at the same time. These assumptions are complied with the communication characteristics of wireless sensor networks constructed by commercial sensor platforms such as Mica2, MicaZ, and etc. We use classical free space model to approximate the radio propagation:

$$P \propto \frac{1}{r^2} \tag{4.1}$$

The power density P of an electromagnetic wave is proportional to the inverse of the square of the distance r from a point source. Based on this radio propagation model, we classify two radio interference scenarios the heavy interference and the light interference demonstrated in Fig. 4·2. In the heavy interference scenario, signal of

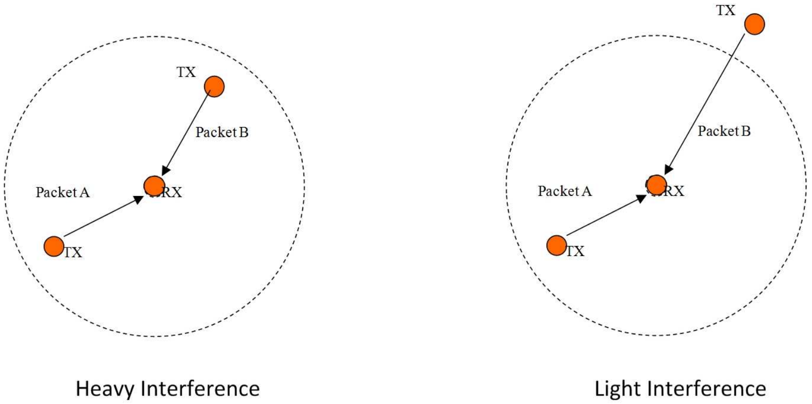


Figure 4·2: Interference Scenarios

packet B is treated as a noise at the receiver for packet A receiving. Since B is generated near the receiver (inside the receiver's communication range), the SNR (Signal to Noise Ratio) of at the receiver is low, so that packet A is likely to be corrupted by the interference from B. In the light interference scenario, signal of packet B is still treated as a noise at the receiver for packet A receiving. However, packet B is generated far away from the receiver (outside the receiver's communication range), the SNR at the receiver is high, so that packet A can be correctly received. The interference from packet B is negligible. As a result, in this dissertation, we assume that the simultaneous data transmission within the communication range (interference range) will cause packet collision at the receiver. Concurrent data transmission from outside of the receiver's communication range will not interfere with the data receiving process.

Some other basic assumptions of WVSN include:

- There are m source nodes and one primary base station.
- The location of the primary base station is known in advance to all the sensor nodes.

- There are m throughput requirements (QoS) with respect to m video sources.
- In network video data processing has already been done and all the content of m source nodes wants to be delivered to primary base station.
- It is an option to dynamically deploy mobile base stations to collect information with a deployment cost.
- Mobile base stations have direct high speed data links to the primary base station.
- Each sensor node can be awakened while receiving a wake up packet similar to [KGSL05]. So that sensor nodes can be put on sleep mode to save energy and awakened to form video streaming data paths on demand.

4.3 General Data Transmission Model and Metrics

Our objective is to develop a dynamic path formation algorithm to satisfy video data delivery requirements per-stream and meanwhile lower the cost of video data egress of the entire network. In the following we propose the metrics to evaluate the data transmission model and quantify the data delivery cost.

- *Throughput*: Throughput is the main metrics to quantify the performance of the data delivery per stream. The degradation of throughput is also a good reflection and measurement of the radio interferences.
- *Mobile Base Station Deployment Cost*: Since previous works [YLC$^+$02, KAK03, GDP04] do not consider the expenses of mobile base station deployment for their data transmission optimizations. In this dissertation, we estimate the mobile base station deployment cost with the distance between the mobile base station deployed location and the location of the primary base station. Mobile base

stations are usually dispatched from primary base station to collect data, the further the mobile base station deployed the higher the cost will be.

- *Data Transmission Cost*: This metric is used to measure the energy cost due to data transmission. In wireless senor networks, most sensor nodes are operating at a very low duty cycle to save energy. Data transmission is the major consumer of the sensor node's battery life. We estimate the energy cost of data delivery for a sensor node with its activation duration for video data transmission.

Before we introduce the general data transmission model, we first list the notations adopted in this section as follows:

k: Number of active video source nodes in the network

np: Number of active paths within the network

C_i: Quantity of video content to be delivered along path i

TP_i: Throughput of path i

D_i: The i^{th} mobile base station deployment cost measured with distance between primary base station and mobile base station

a_i: The factor for throughput of source node i

μ_i: The throughput requirement/threshold for source node i

n_i: Number of nodes along path i

T_i: Data transmission cost for path i measured with the summation of all node's activation time along a path

Based on previous mentioned three metrics, we further present a derived metric:

- *Path Transmission Cost*: This cost is a measurement of energy consumption of the entire data path during the data transmission period. We employ the summation of average node activation time along the path to indicate the energy consumption as shown in Eq. (4.2):

$$T_i = C_i \cdot n_i / (TP_i) \tag{4.2}$$

The objective of the path formation algorithm is to find a series of data paths in the network to facilitate the streaming application of the k source nodes' video content to the primary base station or mobile base stations. The data delivery via these data paths must satisfy the throughput requirements and meanwhile achieve cost-efficient data delivery. The data delivery cost is quantified by the average packet delivery cost including transmission cost and mobile base station deployment cost per packet.

We can represent the general data delivery model as two interactive models, interference model and data transmission model as shown in Fig. 4·3. The data transmission model is the main system we are going to build to characterize the end-to-end throughput performance of the entire data path. The interference model will quantify the negative consequences on the throughput performances of the data transmission model caused by radio interferences.

As a consequence, our goal can be expressed mathematically as to develop a packet forwarding and mobile base station deployment strategy that achieve the objective function given in Eq. (4.3) to obtain minimum value of the weighted sum of average transmission cost per packet and the average mobile station deployment cost for each packet.

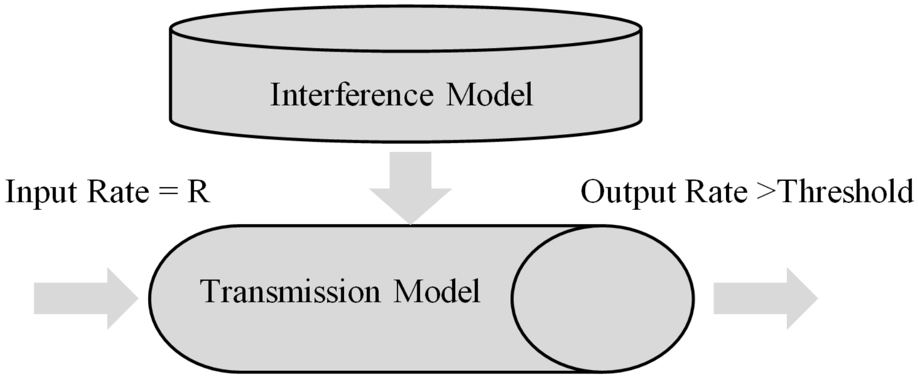


Figure 4·3: General end-to-end data transmission model

$$min\{W \cdot \sum_{i=1}^{np}(C_i n_i/TP_i)/\sum_{i=1}^{np} C_i + (1-W) \cdot (\sum_{i=1}^{m} D_i)/\sum_{i=1}^{np} C_i\} \tag{4.3}$$

$$subject\ \ to\ \ TP_i \geq \mu_i$$

$$W \in (0,1)\ \ Weight\ \ Factor$$

The first term on the right side of the objective function is the average transmission cost per packet and the second term is the average mobile base station deployment cost per packet.

In order to find the strategy to solve this objective function, it is necessary to derive the mathematical data transmission model to accurately characterize the end-to-end throughput performance of the entire data path. We discuss the details of the development of the data transmission model in Section 4.4 and 4.5

4.4 Data Path Transmission Model without Interference

Developing a data transmission model under radio interferences is challenging. According to our analysis, the radio interferences of the data transmission can be classified into two distinct interference types, intra-path transmission interference and inter-path data transmission interference with the following definitions.

- *Intra-Path Data Transmission Interference*: This kind of radio interference is originated from the data transmission on the sensor nodes along the data path. Since all the sensor nodes share the same radio channel the concurrent data transmission and receiving caused by inappropriate packet scheduling scheme on the sensor nodes along the data path will result in packet collision during the packet forwarding.
- *Inter-Path Data Transmission Interference*: This kind of radio interference is caused by the concurrent packet transmission on the neighbors of the senor nodes along the data path. The concurrent packet transmission is likely to be heard by the sensor nodes due to our radio interference model in Section 4.2 along the data path and thus corrupt the forwarding packet.

It is not straightforward to find the solution of building transmission model under the above radio interferences. As a consequence, in this section, we simplify the modeling of data transmission regardless of the inter-path radio interference. This model is appropriate for the singleton video streaming application over WSN where only one data streaming path exists in the network. This model can also be applied to multiple video streaming application scenario where each video stream is physically positioned far away from each other and the interference among different streams is insignificant. In this case, the data transmission we can achieve along a data path is to form a pipelined data forwarding under a interference constraint illustrated in Fig. 4·4:

Since all the sensor nodes share the same radio channel and without losing any generality we assume that the interference range of each sensor node is only one hop. Node A is the source of the video deliver path. At time T_0, Node A begins streaming

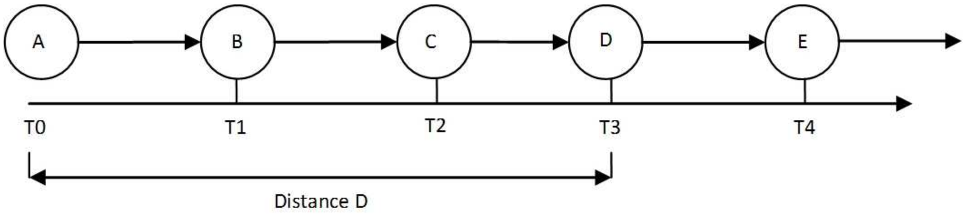


Figure 4·4: Data transmission model

video to Node B until T_1 and then Node B needs to relay the data to Node C. When Node B is sending data during the interval from T_1 to T_2, Node A has to stop sending data to Node B since Node B cannot receive any data during its transmission period. At time T_2, Node C successfully receives the relayed data and starts sending it to Node D. Given the one-hop interference range assumption, although Node C is out of interference range of Node A, it is still in interference range of Node B. In this case, Node A still needs to keep silence during Node C's transmission stage. Otherwise, the simultaneous transmission of Node A and Node C invokes mask node problem at Node B so that Node B cannot correctly decode Node A's data. At time T_3, when Node D starts relaying data to Node E, since Node D is out of communication/interference range of both Node A and Node B, from time T_3 to T_4 both Node A and Node D could simultaneously transmit data.

This scenario implies that only nodes which are three hops away from each other can be simultaneously active to send data. The best transmission scheme is to synchronize the nodes that can concurrent transmit/receive data with minimum distances in between.

As a result if we assume that each link has the same data transmission delay, then the maximum throughput of the path is equal to the throughput of the system consisting of nodes A, B, C. However, the maximum throughput achievement requires high level of sensor node synchronization that is expensive for ad hoc network environment [YK05]. In this dissertation generalize our analysis by introducing some probability models.

In the following we define some terms will be used in our analysis:

- *Independent Nodes*: Nodes in a path which can transmit data simultaneously
- *Channel Reuse Space (CRS)*: the minimum distance between two adjacent independent nodes along the path. In the above optimum scenario the distance D is the CRS.
- *Packet Loss*: A packet is considered lost if it is not received, not correctly received or received beyond the deadline.

In order to analyze the throughput of the entire data path, the straightforward approach is to study the packet transmission on the sensor nodes along the path. However, this approach is very tricky. The challenge of the analysis is caused by the dependency of the data transmissions on adjacent nodes along the path. Take Fig. 4·4 as an example, if Node A is sending data, Node B can only receive data, and moreover Node C is suppressed from transmission as well to avoid packet loss at node B. In order to alleviate the data transmission dependencies, we leverage the property of independent node. We cluster the independent node with its subsequent nodes prior to the next adjacent independent node.

This cluster forms a new data transmission system. Since the independent nodes can transmit at the same time, in other words, the data transmission in one system will have no impact on the data transmission in others. This new system can be employed to simulate the behavior of the data transmission from one independent node to the next independent node. Therefore the path can be modeled as a tandem queue of the systems demonstrated in Fig. 4·5.

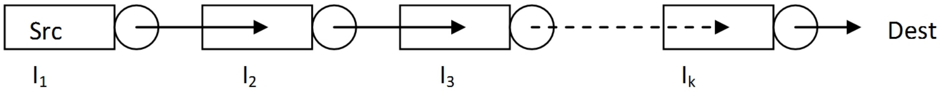


Figure 4·5: Tandem queue model of data path

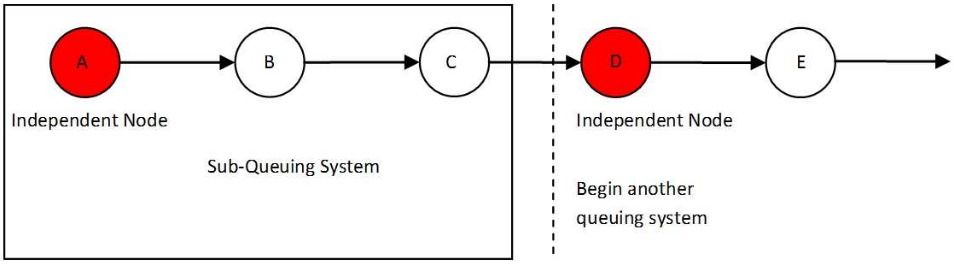


Figure 4·6: Illustration of queuing systems

The systems I_1, I_2, I_3... represent the systems beginning with the first, second, third independent node. The processing time of each system equals to the packet transmission time from current independent node to the next independent node.

According to the model above, a packet collision can happen in two places, inside the system and at the edge of the system. However, since the system is small in size, the independent node can employ an packet scheduling scheme to manage the data transmission inside the system. The independent node is the head node inside the small system as shown in Fig. 4·6. If the packet scheduling scheme can eliminate the packet collision inside the system, the packet collision only happens at the edge of the systems. This approach enables us to model every system along the path as a queuing system, and use queuing systems and tools to conduct the analysis.

We observe that the packet collision probability at the system edge is closely related to the states of the independent nodes. Finding the probability of those states which would cause packet collision at the independent nodes will guide our way to find the packet collision probability at the system edge, in general we call these states collision states. If the packet arrival process to the system is a Poisson process and is independent of the packet scheduling of the system, the packet collision probability can be equivalent to the equilibrium distribution of the collision states according to PASTA [Ber92].

There are four states of the independent node:

- *Idle State*: The independent node can receive packets and forward packets immediately after correctly receiving required number of packets
- *Mask State*: The independent node can not correctly receive packets due to a neighbor's transmission
- *Wait State*: The independent node may receive packets but cannot forward these packets immediately since these packets can not be correctly received at the receiver. Since we do not assume extra buffer space on sensor nodes to cache data, we disable packet receiving in Wait State.
- *Busy State*: The independent node is busy sending data and can not receive packets.

We assume the average packet arrival rate of the system is λ. The state transition diagram of the system is demonstrated in Fig. 4·7.

Where λ is the arrival rate of the current queuing system which equals to the departure rate of the previous queuing system. For the first queuing system, λ is equivalent to the delivery rate of the source node. We also assume that the packet loss is mainly due to the collision and interference, thus the departure rate of the queuing system is $\lambda(1 - p(collision))$. μ_1 is the transmission rate of the independent node, μ_2 is the rate that the data is forwarded to the third node of the queuing system. μ_3 is the rate that the data is forwarded outside the queuing system.

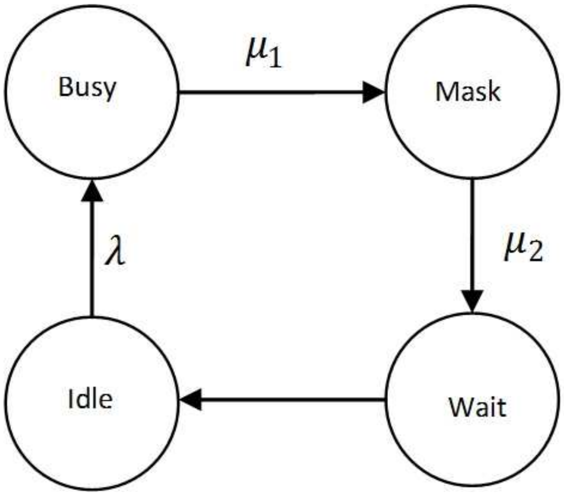


Figure 4·7: State transition diagram

However the challenge is to calculate the collision probability based on the equilibrium distribution of the independent node's states. Generally speaking, the collision probability is not equal to the equilibrium distribution of these collision states unless the arrival is Poisson arrival [Ber92]. Even if the initial arrival is exactly Poisson process for the first system along the path, the departure process of the fist system will likely lose the Poisson property unless the system service time is independent exponential random variable and we have unlimited queuing buffer according to the Burke's theory [Ber92]. The calculation would be complicated and impractical. Thus we turn to a new practical queuing model approach.

We notice that among the states of the independent node, only idle state can receive incoming packet. If we consider the state transition at a system level, the idle state is equivalent to the situation that there is no packet inside the queuing system, the remaining states are equivalent to the situation that there is one packet inside the system. We can characterize our system model as follows: an incoming packet will be served only if the system is empty otherwise the packet will be dropped. If we assume the arrival process is a Markov renewal process and the system service

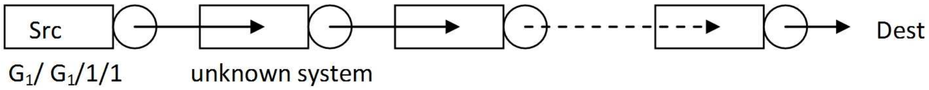


Figure 4·8: Queueing models of data path

time of each packet is iid random variable, then we could leverage G/G/1/1 [Ber92] queuing model to characterize our first queuing system along the path. And our path transmission model is concatenated queuing systems as shown in Fig. 4·8.

To characterize the rest of the concatenated queuing systems and apply the appropriate queuing models to analyze the path transmission, we develop our first theorem.

Theorem 1 *If the Input process of the G/G/1/1 queuing system is a renewal process the output process of the G/G/1/1 queuing system is still a renewal process given that the service time of the system is constant.*

Proof: Let us denote some random variables used in our analysis as follows:

S_k: Service time of the kth accepted packet in the system.

A_i: Inter-arrival time between the i^{th} and $(i+1)^{th}$ packet arrival

D_k: Inter-departure time between the k^{th} and $(k+1)^{th}$ accepted packet by system

Without loss of any generality, we assume that A_n is the inter-arrival time between the k^{th} accepted packet by the queuing system and the next packet arriving the system. We derive D_k in Eq. (4.4):

$$D_k = \begin{cases} A_n - S_k + S_{k+1}, & \text{if } (S_k < A_n) \\ A_n + A_{n+1} - S_k + S_{k+1}, & \text{if } (A_n < S_k < A_n + A_{n+1}) \\ ... & \\ \sum_{i=n}^{n+l} A_i - S_k + S_{k+1}, & \text{if } (\sum_{i=n}^{n+l-1} A_i < S_k < \sum_{i=n}^{n+l} A_i) \\ ... & \end{cases} \tag{4.4}$$

If we assume that the service time of each system is a constant S, then D_k can be simplified in Eq. (4.5)

$$D_k = \begin{cases} A_n, & \text{if } (S_k < A_n) \\ A_n + A_{n+1}, & \text{if } (A_n < S_k < A_n + A_{n+1}) \\ ... & \\ \sum_{i=n}^{n+l} A_i, & \text{if } (\sum_{i=n}^{n+l-1} A_i < S_k < \sum_{i=n}^{n+l} A_i) \\ ... & \end{cases} \tag{4.5}$$

If the arrival process of the queuing system is a renewal process, A_i are i.i.d. random variables and S_k is constant, then D_k is also a series of i.i.d. random variables expressed in Eq. (4.5). As a consequence, the departure process of the queuing system is also a renewal process. This conclusion proves Theorem 1.

In this case, we can model all the rest of concatenated queuing systems to be G/G/1/1 queuing models, and our path transmission model is a cascade of G/G/1/1 queuing systems as shown in Fig. 4·9

Let us denote the CDF of the D_k as $F(t)$, then $F(t)$ can be expressed with Eq. (4.6), where Pr stands for the probability:

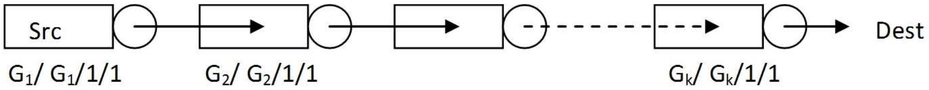


Figure 4·9: G/G/1/1 Tandem queue model of path

$$\begin{aligned} F(t) &= Pr(D_k < t) \\ &= Pr(S < A_n) \cdot Pr(A_n < t) + \\ & \quad Pr(A_n < S < A_n + A_{n+1}) \cdot Pr(A_n + A_{n+1} < t) + ... \\ & \quad + Pr(\sum_{i=n}^{n+l-1} A_i < S_k < \sum_{i=n}^{n+l} A_i) \cdot Pr(\sum_{i=n}^{n+l} A_i < t) + ... \end{aligned} \tag{4.6}$$

We can take the Laplace-transform of $F(t)$ as $L(F, s)$. If $L(F, s)$ is second order differentiable, we can derive the expected inter-packet departure time and the corresponding variance in Eq. (4.7), Eq. (4.8):

$$E(D_k) = -\frac{d[s \cdot L(F, s)]}{ds}|_{s=0} \tag{4.7}$$

$$Var(D_k) = \frac{d^2[s \cdot L(F, s)]}{ds^2}|_{s=0} \tag{4.8}$$

Eq. (4.7) and Eq. (4.8) enable us to estimate the average data throughput and variability of a single queuing system along the path. However, it is still tedious to apply Eq. (4.7) and Eq. (4.8) on every queuing systems along the path in order to characterize the throughput performance of the entire data path. Fortunately, the constant service time of every queuing system along the path is a nice property that can be leveraged to develop our second theorem to simplify our analysis.

Theorem 2 *The data output process of the first G/G/1/1 queuing system along the path is identical to the data output process of the last G/G/1/1 queuing system along* *the path if all the queuing systems along the path have identical constant service time.*

Proof: We denote the following variables:

- S: Service time of each queuing system along the path
- A_k^i: Inter-arrival time between the k^{th} and $(k+1)^{th}$ packet arrival of the i^{th} subsequent system along the path
- D_k^i: Inter departure time between the k^{th} and $(k+1)^{th}$ accepted packet of the i^th sub system along the path

Given that the departure process of the i^{th} queuing system along the path is exactly the arrival process of the $(i+1)^{th}$ system, we have

$$A_k^{i+1} = D_k^i$$

As each subsystem will take a constant time S to forward the packet then:

$$D_k^i > S \Rightarrow A_k^{i+1} > S \Rightarrow A_k^{i+1} = D_k^{i+1} \Rightarrow D_k^1 = D_k^2 = ... = D_k^i = D_k^{i+1}$$

Hence the inter-departure time of the first queuing system along the path is exactly the inter-departure time of the last queuing system along the path. Therefore it is also the output process of the entire path.

According to the above two theorems, to characterize the throughput of the entire data path is equivalent to characterize the throughput of the first subsystem along the data path. Therefore, Eq. (4.6), Eq. (4.7), Eq. (4.8), derived for a single queuing system are also applicable for the throughput analysis of the entire data path. Both the expectation and variance of the inter-departure time of the first queuing system along the path are exactly the same as that of the entire data path. The throughput performance can be accurately estimated using Eq. (4.7) and Eq. (4.8) and it is independent of the path length. This property enables routing algorithm to find a detour when there is a network congestion or form isolated concurrent streaming paths to avoid inter-path interference and meanwhile provides guaranteed end-to-end data path throughput performance.

Both above theorems are based upon an important requirement of constant service time for each queuing system along the path. We propose a practical TDMA-based distributed packet scheduling scheme to satisfy this critical. The details of this packet scheduling scheme are discussed in our routing algorithms in Chapter 5.

Admission Control to Achieve Single Path Transmission: To achieve the single data path transmission scenario that can apply Eq. (4.6), Eq. (4.7), Eq. (4.8) to estimate the throughput performance, we introduce an admission control strategy. Admission control for the single path transmission model consists of two levels of admission, per-hop admission and source node admission. Per-hop admission is first performed on intermediate node of a potential route. It allows an intermediate node to admit a new video stream if and only if the node is not carrying any data stream and the neighbor nodes under the interference range of the current node is also not serving any active streams. The intermediate sensor nodes only need to monitor local information (including the states of neighbor nodes) to make per-hop admission decisions. If all the intermediate sensor nodes along a route pass the per-hop admission criteria, the source node admission control will be performed. Source node computes the throughput performance of the potential data path according to Eq. (4.6), Eq. (4.7), Eq. (4.8) and admits this new video stream for the route if the throughput estimation satisfies the application requirements. Per-hop admission control eliminates the interference among concurrent streams and enables the incremental data flow admission without any negative impact on existing data deliver performance of video streams. Source node admission guarantees that the sufficient end-to-end bandwidth resource will be allocated and provides data deliver QoS assurance in throughput achievement for admitted data flows. Our source node admission exploits off-line throughput computation based on our single path transmission model to minimize the estimation overhead.

4.5 Data Transmission Model with Inter-path Interferences

As previously described, the model in Section 4.4 only works for single data path. In this section, we quantify the impact of adjacent data streams in terms of end-to-end path throughput. This is done under an assumption of a Poisson packet arrival from the video source. We assume that (1) the radio range of each sensor node is identical; (2) a single radio channel is shared by all sensor nodes; (3) simultaneous radio transmission within one node's radio range causes packet collisions at the node; (4) concurrent radio transmission outside the node's radio range does not interfere with the packet receiving at the node.

We start analysis by considering only two adjacent data paths. Two adjacent data paths scenario is a basic scenario to deal with inter-path interference measurements. And inter-path interference scenarios involving more than two paths are not likely to provide satisfied end-to-end throughput performance for our WVSN applications. Thus in this dissertation we concentrate our analysis of inter-path interference on only two adjacent data paths. An admission control strategy is also developed to cooperate with path formation algorithms to avoid the occurrence of inter-path interference scenarios involving more than two paths.

In Section 4.4, we model a single data path as concatenated queuing systems; therefore, two disjoint data paths can be represented by two separate series of concatenated systems as shown in Fig. 4·10. Instead of studying the interferences among

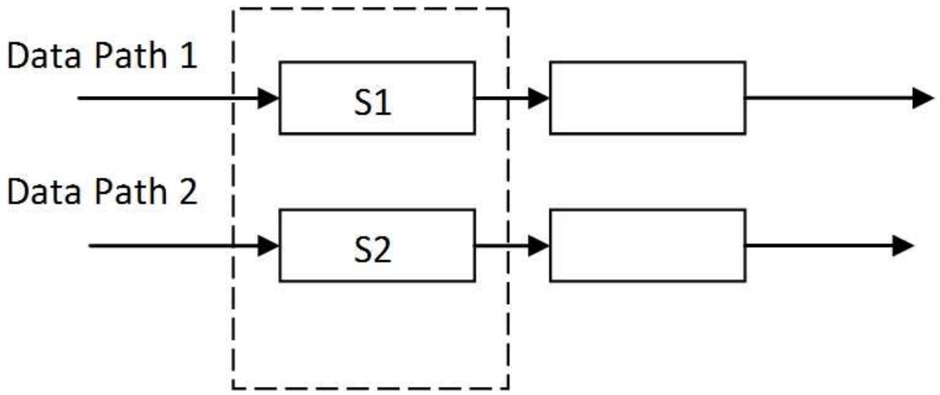


Figure 4·10: Path interference model

sensor nodes from each path, we quantify the interferences among different queuing systems along each path.

In the following we introduce some definitions used in our model:

- *Mutually Interfered Data Paths*: Two data paths are considered mutually interfered if there is at least one node of a data path locating in the interference range of some other node in the other path. Naturally following this definition, we introduce a definition for mutually interfered systems.

- *Mutually Interfered Systems (MIS)*: System A along path 1 and system B on path 2 are MIS if at least one node of system A locates in the interference range of some node in System B.

- *Mutex System*: System A along path 1 and system B on path 2 are mutex systems if system A and system B are mutually interfered systems, arrival packet for either system A or B will not be admitted (will be dropped) unless both system A and B are empty and system A and system B do not belong to the same path.

Due to the fact that our queuing system is relatively small (roughly a size of node's

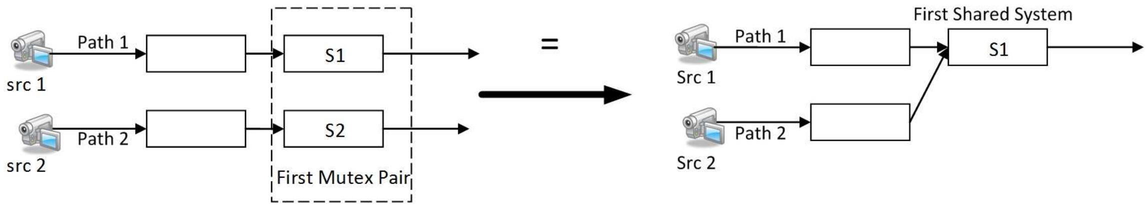


Figure 4·11: Mutex pair equivalance

interference/communication range), the data transmission in one of the mutually interfered systems is likely to corrupt the concurrent data transmission in the other. Moreover, to analyze the worst case scenario of the data path throughput, we consider all the mutually interfered systems in our model as mutex systems.

Note that two disjoint paths can have many mutex system pairs and packet collision can happen at any of the mutex systems. However, since mutex systems can only have one system working at a time, we can equate the first pair of mutex systems as one single system. If we apply a path formation algorithm to avoid establishing two disjoint concurrent data paths and forward data stream from path 2 to the first mutex system along path 1 as shown in Fig. 4·11, we can remove the packet collision issues for the rest of mutex systems according to Theorem 2 and share a common section for both paths to deliver video data.

We assume that the packet arrival of our data path from the video source is a Poisson process. By leveraging our previous transmission model in Section 4.4, we can characterize the inter-packet departure time distribution of the first queuing system along our data path. The expectation of this variable can be expressed in Eq. (4.9):

$$IDT = C + E(\lambda) \tag{4.9}$$

IDT represents the inter-departure time of the first subsystem along the data path. If there is no interference between paths, this departure time is also the end to end inter packet departure time of the entire data path. C is a constant which is the service time of the first subsystem along the path. Due to the distributed TDMA based packet scheduling scheme on each subsystem, the service time of all the subsequent systems along the path are identical and equal to C. $E(\lambda)$ is an exponential random variable with mean $1/\lambda$, where λ is the packet arrival rate of the first subsystem along the data path.

In this section, we link the throughput degradation to the interference between data paths. Moreover, the interference between two data paths is also associated with the relative path positions. If we merge adjacent data paths at the first mutex system of a path as shown in Fig. 4·11, according to Theorem 2, the output process of the first mutex pair is exactly the output process of the entire path.

The output process of the first mutex system pair is related to their relative positions. We classify three basic mutex system positioning scenarios to estimate the throughput degradation on the mutually interfered data paths.

Scenario 1: Both mutex systems are the first systems of their corresponding data path (Fig. 4·12).

In this scenario, the packet arrival process of each system in the mutex pair is a Poisson process. We represent the state of the mutex pair as (N_1, N_2), where N_1 is the number of packet being forwarded in the system S_1 in Fig. 4·12, and N_2 is the number of packet being forwarded in the system S_2. Since S_1 and S_2 are mutex systems, at any time there is at most one packet being forwarded in either of the system. N_1 and N_2 are either 1 or 0 but cannot be 1 simultaneously. As a result we can draw the state transition diagram in Fig.4·13.

Because the arrival process of each system in the mutex pair is a Poisson process,

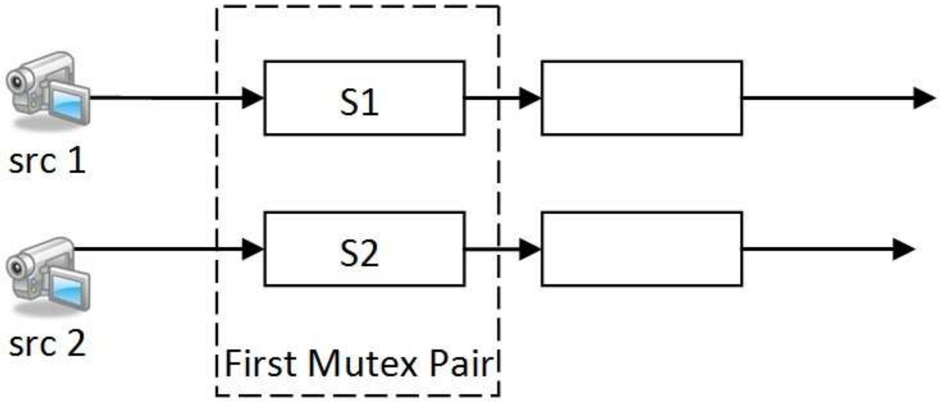


Figure 4·12: Scenario 1 of first pair of mutex systems on interfered paths

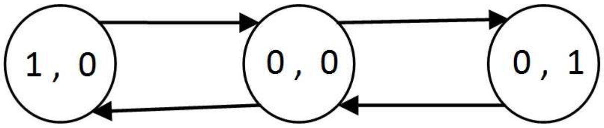


Figure 4·13: State transition diagram of mutex pair in scenario 1

the state transition of the mutex system pair can be identified as an alternating renewal system [Ros95]. We derive the probability that a system is idle at time t in Eq. (4.10) based on the renewal theory [Ros95]:

$$P(t) = \frac{E(Systemisidle)}{E(Systemisbusy) + E(Systemisidle)} \tag{4.10}$$

$E(systemisidle)$ is the expected time for system is idle. If we assume the average packet arrival rate of each system in the mutex pair is λ_1 and λ_2 respectively, then we can derive $P(t)$ in Eq. (4.11):

$$P(t) = \frac{1}{C(\lambda_1 + \lambda_2) + 1} \tag{4.11}$$

C is the constant service time of each queuing system, the average rate (throughput) of each data path can be expressed in Eq. (4.12), Eq. (4.13):

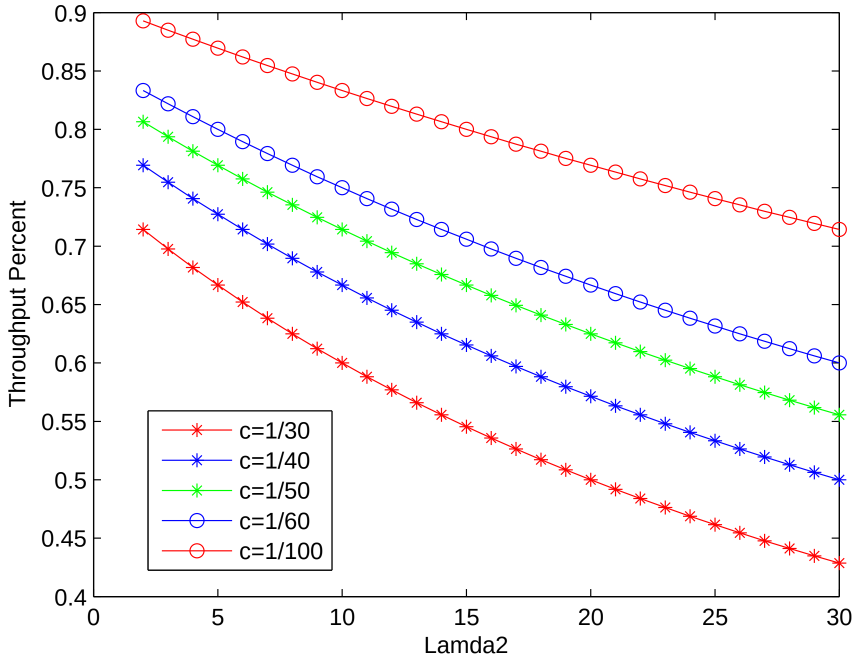


Figure 4·14: Throughput performance for different service rate

$$TP_1 = \lambda_1 \cdot P(t) = \frac{\lambda_1}{C(\lambda_1 + \lambda_2) + 1} \tag{4.12}$$

$$TP_2 = \lambda_2 \cdot P(t) = \frac{\lambda_2}{C(\lambda_1 + \lambda_2) + 1} \tag{4.13}$$

If we fix the arrival rate of one mutex system and change the rate of the other one, the throughput of the path with fixed arrival rate will decrease inversely with respect to the arrival rate of the other system. The Fig. 4·14 shows examples of fixed rate system throughput degradation with respect to different arrival rates at the interfered system and different values of service time. In this graph, λ_1 is fixed λ_2 is varied from 2 to 30 packets per unit time.

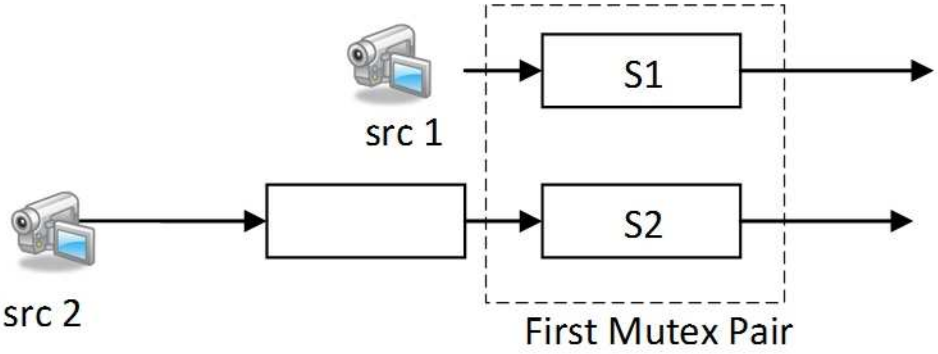


Figure 4·15: Scenario 2 of first pair of mutex systems on interfered paths

Scenario 2: One of the mutex systems is at the beginning of its corresponding data path and the other one is in the middle of the path (Fig. 4·15).

In this scenario, the packet arrival process for the first data path from src_1 is still a Poisson process, however, the packet arrival process of the second data path is not a simple Poisson process. Actually the packet inter-arrival time of the second data path is a random variable. It is a summation of the constant queuing system service time and an exponential random variable. This exponential random variable is equivalent to the inter-arrival time from src_2 due to memory less property of Poisson packet arrival. As a result the expectation of the exponential random variable is an inverse of the packet arrival rate from src_2.

In this scenario, we still represent the state of the mutex pair with (N_1, N_2) as the same as in scenario 1. According to the renewal theory [Ros95], if we prove that the system restart itself during the state transition, we can still leverage the Eq. (4.10), Eq. (4.12), Eq. (4.13) to estimate the throughput of the data path.

In fact there exists a state transition renewal circle of the first pair of mutex systems. The first pair of mutex systems restarts its state transition whenever a packet is accepted at the system S_2 on the second path.

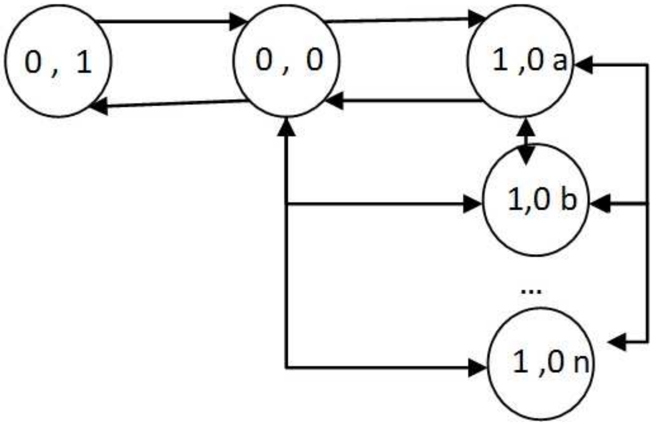


Figure 4·16: State transition diagram of scenario 2

The state transition diagram is expressed in Fig. 4·16:

In this diagram, the reason that we distinguish the state $(1, 0)$ is to consider the different arrival situations at the second path during the packet serving at the first path. The packet arrival at the second data path during this period can affect the next packet arrival interval distribution since the arrival process of the second path is not memory less. It is difficult to compute the average time for the state transition renewal circle. So we decide to find a good approximation to simplify the computation. We carefully study the different situations of the packet arrival on the second data path during the packet serving at the first path in the mutex system pair. If there is no packet arrival at S_2 during the packet serving in S_1, the next packet arrival interval for S_2 is still an exponential random variable with the same mean of the inverse of the packet arrival rate of src_2. Furthermore, the system service time is relative small compared to the expected packet arrival time from the source (the packet processing time should be smaller than the packet arrival time for stable system), the packet arrival interval at S_2 is likely to be dominated by the exponential random variable of the packet inter-arrival time from src_2. As a consequence, we can approximate the packet arrival process of S_2 with the same packet arrival process from src_2 and

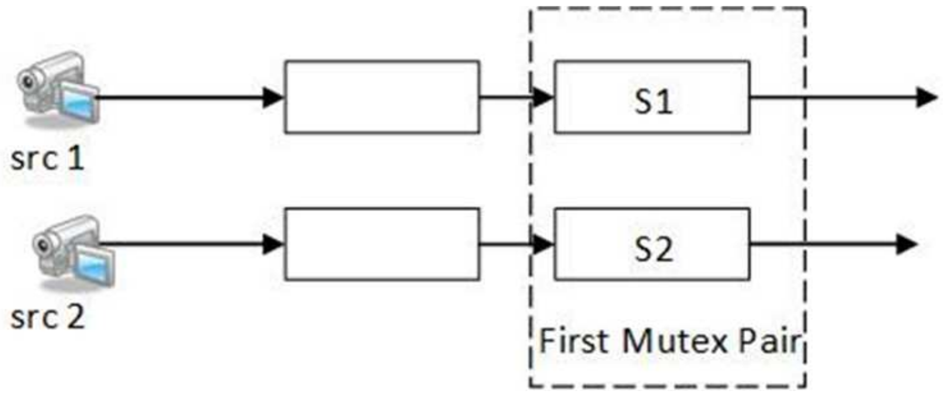


Figure 4·17: Scenario 3 of mutex system pair

Eq. (4.12) and Eq. (4.13) can be employed to deal with the throughput performance estimation for this scenario. Simulation results in Chapter 7 demonstrate that this is a good approximation.

Scenario 3: None of the mutex systems is at the beginning of their corresponding data paths (Fig. 4·17).

In this scenario, neither of the packet arrival process of the mutex system pair S_1 and S_2 is Poisson process. Base on the analysis in Scenario 2, we know that the packet arrival interval for both S_1 and S_2 is a random variable which is a summation of the constant service time and an exponential random variable. The corresponding exponential random variables of S_1 and S_2 are identical to the packet inter-arrival time from src_1 and src_2 respectively.

The state transition of the mutex pair in this scenario is complicated since the packet arrival process to the mutex pair does not possess any memory less property. Due to the same reason we have addressed for the inter-arrival time approximation of S_2 in Scenario 2, we approximate the inter-arrival time for S_1 and S_2 with the corresponding Poisson packet arrival process from src_1 and src_2 respectively and leverage Eq. (4.12), Eq. (4.13) to estimate the throughput of the data paths in this scenario. Simulation results in Chapter 7 also demonstrate that the approximation applied in this scenario can provide accurate end-to-end throughput estimation.

Admission Control to Achieve Adjacent Path Transmission: To achieve the adjacent data path transmission scenario with tolerant inter-flow interference, we apply a extended admission control strategy as introduced in single path transmission scenario. This strategy still consists of two levels of admission, per-hop admission and end node admission. Per-hop admission allows an intermediate node to admit at most two video streams. The sensor nodes along an existing route will be allowed to admit only one additional video stream if the neighbor nodes of the route is not carrying a third active stream. The sensor nodes between the new source node src_2 and the nearest head node in S_1 as shown in Fig. 4·11 will be allowed to admit the new video stream if and only if the node is not carrying any data stream and the neighbor nodes under the interference range of the current node is also not serving any active stream. If all the intermediate sensor nodes pass the per-hop admission criteria, the destination node and new source node src_2 in Fig. 4·11 will perform end node admission. The destination node will allow sensor nodes along the existing route to admit new video stream from src_2 if the throughput estimation derived from Eq. (4.12) satisfies the application requirement. Src_2 will allow sensor nodes of the new route to admit new video stream from src_2 if the throughput estimation derived from Eq. (4.13) satisfy the application requirement. Per-hop admission control in this strategy allows at most two adjacent streams to interfere with each other and end node admission makes sure the sufficient end-to-end bandwidth resource will be allocated and provides data delivery QoS guarantees for admitted data flows. Our end node admission also exploits off-line throughput computation based on the adjacent data transmission model that minimizes the estimation overhead.

Chapter 5

Routing Solution

In this chapter we propose a benchmark isolated data routing algorithm to achieve throughput-guaranteed video data routing for video streaming applications over wireless sensor networks. The benchmark data routing algorithm is developed based on the data path transmission model in Section 4.4. This benchmark routing algorithm constructs a data path with required throughput yield based on end-to-end data path throughput estimation. The benchmark routing algorithm clusters sensor nodes along a data path to form small queuing systems. Inside each queuing system, a distributed TDMA-based packet scheduling scheme is employed to manage the packet forwarding process individually. These small queuing systems are modeled as G/G/1/1 systems as we described in Section 4.4 and the data path turns into a concatenation of these systems. The theoretical analysis derived in Section 4.4 based on G/G/1/1 system enables the benchmark algorithm to estimate the throughput of a data path and make routing decisions. According to the Theorem 2 in Section 4.4 the throughput of the data path constructed by the benchmark routing algorithm is independent of the path length. The Benchmark routing algorithm eliminates interferences between data paths to provide QoS for data path throughput achievement via data path isolation strategy. It can incrementally construct new data paths in the network without impact on the existing data transmission. The second part of this chapter provides an interference-tolerant data routing algorithm by extending the benchmark algorithm. The interference tolerant routing algorithm leverages the data path transmission model with interferences in Section 4.5 to construct data paths with satisfied throughput performance. Interference-tolerant data routing algorithm improves the general data egress rate of the entire network with some tolerance of throughput performance on individual streams. It allows two adjacent data streams to share an existing data path as the data deliver infrastructure as long as the estimated throughput performance per stream still satisfies the requirement of the application. This algorithm also limits the packet collisions and only allow the collisions to happen at the beginning of the first pair of mutex systems.

5.1 Benchmark Isolated Data Routing Algorithm

5.1.1 Benchmark Routing Algorithm

The Benchmark Routing Algorithm is composed of two basic modules: Data Transmission Module (DTM) and Error Handler Module (EHM) as shown in Fig. 5·1. DTM is responsible for establishing a data path for video data delivery and EHM takes care of errors happening during data path construction and data transmission session such as the link failures caused by wireless environment change.

DTM: Data Transmission Module

DTM takes five steps to establish a throughput requirement satisfied data path as shown in Fig. 5·2. Once a request is initiated by a video source node to deliver data to the gateway, DTM of the source node first sends a path setup request to acquire an available data path between the source and the gateway. As soon as a potential data path is obtained, DTM on each node along the path takes the second step to form queuing systems as mentioned in Section 4.4. In the third step, DTM of the source node computes the estimated potential data path throughput performance based on the data estimation model in Section 4.4. If the estimation result satisfies

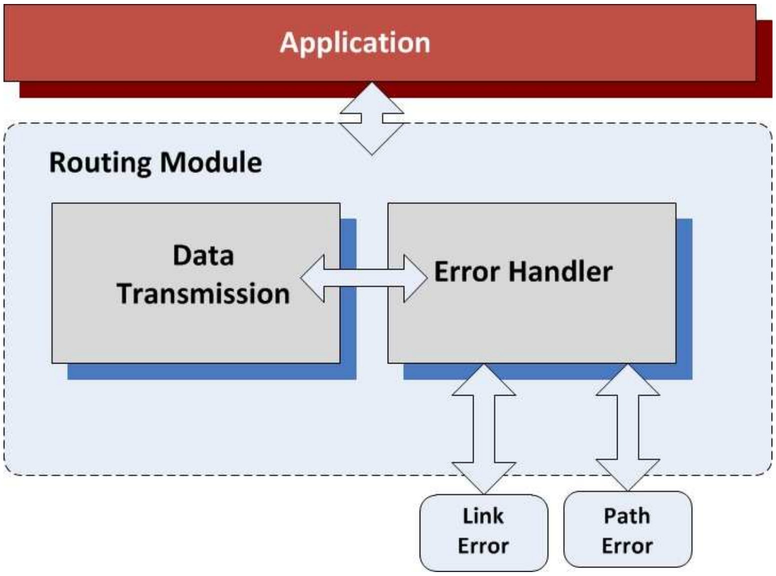


Figure 5·1: Block diagram of benchmark routing

the application requirement, the source node engages in the fourth step to send data. Otherwise DTM of the source node will clear the potential data path and request a new data path after a random period of time. We elaborate the details of each step in Fig. 5·2 as follows:

1. *Step One : Data Path Setup Request*

   The data path setup request step is employed to discover a potential data path between source node and gateway node. A geographic based greedy routing algorithm is employed to obtain a potential data path based on sensor nodes' states and positions. We define two sensor node states for our path set up process to achieve the admission control we have discussed in Section 4.4: if a node is currently delivering the video data for a specific video stream or it is in the interference range of some nodes along one active streaming path, the node holds a state of OCCUPY. The sensors node with OCCUPY states are not allowed to admit new video streams. The names of these streams are stored

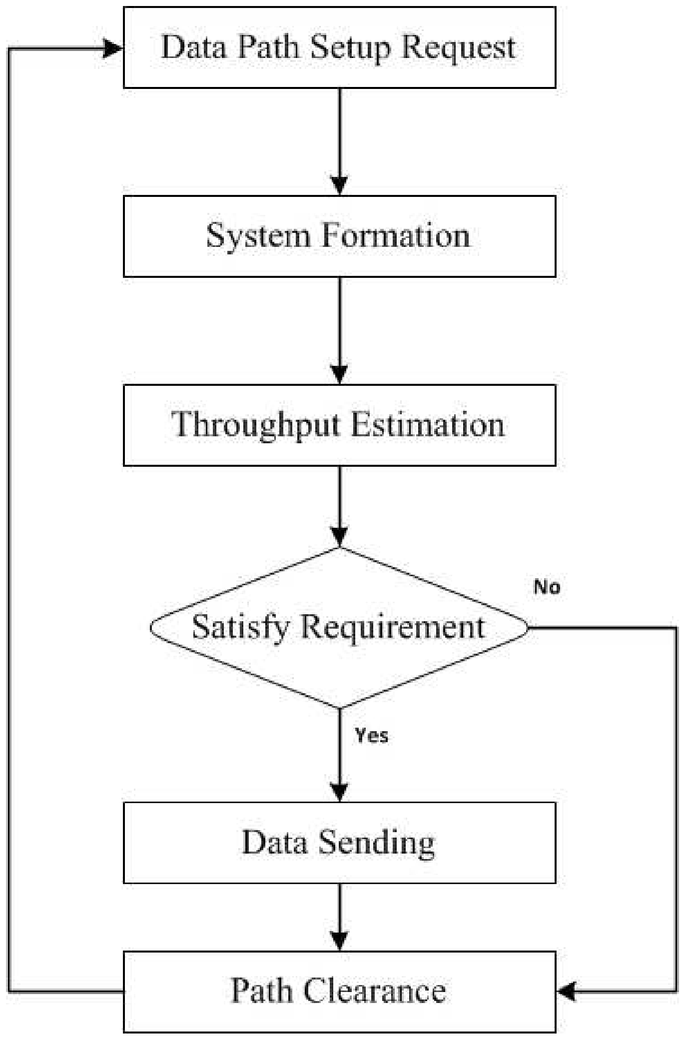


Figure 5·2: Flowchart of path formation in benchmark routing

Stream ID	Property
Stream One	Interference
Stream Two	Interference
Stream Three	Interference

Table 5.1: OCCUPY table example

in OCCUPY table on a node. Since a sensor node can be interfered by multiple concurrent data streams, there may be multiple entries in the OCCUPY table. Table 5.1 is an OCCUPY table example. It indicates that the current sensor node is in the interference range of three concurrent video streams. The 'property' field of the table indicates whether this node is interfered by a active data stream (value = interference) or carry a data stream (value=serve). If the table is empty, the node holds a state of AVAILABLE. Per-hop admission control introduced in Section 4.4 allow a sensor node with the AVAILABLE state to admit a new video stream.

The geographic based routing algorithm is described in Algorithm 1.

A path request query is initiated by the source (n_s) and forwarded to the neighborhood. The query is intended to form a path P which is a set of nodes n_s, n_1, n_2 $\ldots$, n_i, n_{i+1}, $\ldots n_d$ from source to gateway. The intermediate node n_i forwards the query to a node n_{i+1} in its neighborhood satisfying that (1) n_{i+1} is the nearest neighbor node to destination n_d, (2) the state of node n_{i+1} is AVAILABLE and (3) n_{i+1} is out of interference range of all upstream nodes n_s to n_{i-1}. The process is repeated until the path request is received by the destination n_d.

2. *Step Two: System Formation*

   In this step, DTM cluster sensor nodes along the potential path into small systems. Once the gateway node receives the path setup request, it replies

Algorithm 1 Geographic Greedy Routing: GGR(S,P,$Neighbors$,$Pre\_Neighbors$)

```
Input:
    Neighbor nodes states, $S$;
    Neighbor nodes positions, $P$;
    Neighbor nodes, $Neighbors$;
    Neighbor nodes of upstream nodes, $Pre\_Neighbors$;
Output:
    Next Hop, $Next\_Hop$;

$temp\_next\_hop = null$;
$temp\_next\_Position = null$;
$next\_hop\_flag = false$;
$Next\_Hop = null$;
for $i = 1$ to $Neighbors.length$ do
    if $temp\_next\_hop$ is null then
        if $S[i]$ is AVAILABLE, && $Neighbors[i] \notin Pre\_Neighbors$ then
            $temp\_next\_hop = Neighbors[i]$;
            $next\_hop\_flag = true$;
            $temp\_next\_Position = P[i]$;
        end if
    else
        if $P[i]$ closer to gateway than $temp\_next\_Position$ && $S[i]$ is AVAILABLE, && $Neighbors[i] \notin Pre\_Neighbors$ then
            $temp\_next\_hop = Neighbors[i]$;
            $next\_hop\_flag = true$;
            $temp\_next\_Position = P[i]$;
        end if
    end if
end for
if $temp\_next\_flag$ is false then
    invoke EHM(Error Handler Module);
    return null;
else
    $Next\_hop = temp\_next\_hop$
    forward message $PATH\_SETUP\_REQUEST$ to $Next\_hop$;
end if
return $Next\_Hop$;
```

back a path acknowledgment to the source node. Every node consisting of the data path will add an entry in their OCCUPY table. Due to the broadcast nature of the wireless media, each sensor node appearing in the neighborhood of the new path will overhear this acknowledgment and add an entry in their corresponding OCCUPY tables as well. These nodes with OCCUPY states are reserved by the current data path and cannot be employed by other streams to deliver video according to the admission control. Furthermore, given the assumption that interference range is the same as communication range, this sensor nodes reservation strategy eliminates the interferences between different streams. However, this assumption can be relaxed. If we assume the interference range can impact a sensor node that is c hop away, the sensor nodes along the path can send a notification messages to their c-hop neighbor nodes to update their OCCUPY tables and eliminate the inter-path interference as well. DTM of each sensor node along the path will leverage the information of distance to source node (in hop count) and system size to get their corresponding system ID and relative location inside its local system according to Eq. (5.1), Eq. (5.2), Eq. (5.3).

$$System\_Size = Interference\_Range + 2 \tag{5.1}$$

$$System\_ID = \frac{Hop\ \ count\ \ from\ \ source\ \ node}{System\_Size} \tag{5.2}$$

$$System\_Location = (Hop\ \ count\ \ from\ \ source) \mod (System\_Size) \tag{5.3}$$

$System\_Location$ indicates the relative location of the senor node in the local system. For example, the value 1 indicates that this node is the first node of the local system, the value 2 indicates the second node of the local system, and so on. This information is employed to synchronize packet scheduling in our distributed TDMA packet scheduling strategy discussed in Section 5.3.

3. *Step Three: Throughput Computation and Notification*

   In this step, DTM on the source node will valid the data path via throughput performance estimation. Once the source node receives the acknowledgment from the gateway, it estimates the data path throughput performance according to Section 4.4. If the estimation result satisfies the throughput requirement of the stream, the source node will send a 'ready-to-send' notification packet to the gateway, otherwise the source node will engage in 'path clearance' step.

4. *Step Four: Sending Data*

   When the destination receives the 'ready-to-send' notification from the source, it replies a 'ready-to-receive' message to the source. The source then is informed and begins streaming data to the gateway.

5. *Step Five: Path Clearance*

   When the source node finishes data transmission or the data path does not pass the validation in step three, DTM on the source node will send out a *Path_Clear* request to the destination. Due to the broadcast nature of the wireless communications, all the nodes along the path and their neighbors will overhear the *Path_Clear* packet. The nodes overhear this message will remove the path from their OCCUPY tables. If the interference range affects c-hop neighbors, nodes along the path can send active messages to their corresponding c-hop neighbor nodes to update their OCCUPY tables.

EHM: Error Handler Module

EHM is a module that handles errors during the entire data transmission session. So far, the benchmark algorithm can handle two types of errors: path error and link error.

1. *Path Error Handler*

   Since our path formation process employ a location based greedy algorithm similar to [SZY$^+$10] to search the potential path, the process will still face the problem to avoid network holes. Solutions provided in [SZY$^+$10] can be easily incorporated into our protocol. However, our path formation is not purely depended on the location. In addition, it also takes the node's interference range and existing data streams into account. In our protocol, we apply a simple iterative scheme adapt existing "step back and mark" strategy in [SZY$^+$10] to bypass the network hole. If the node faces a network hole during the data path construction period, it sends '*Route_Search*' message to its previous node. This message informs the previous node to iterate all other AVAILABLE nodes in neighborhood to search the destination with regular greedy routing algorithm as described in Algorithm 1. If the previous node receives a path acknowledgment from the destination during the route search, the path set up is successful otherwise the node will recursively inform its previous hop to run this route search algorithm until the destination is identified. Algorithm 2 depicts the iterative route search algorithm of path error handler on the previous node.

2. *Link Error Handler*

   One critical challenge for WSN data delivery protocol design is to battle with the network environment change. For example, the link quality may downgrade sharply due to radio interferences from outside of the network such as start of

---

Algorithm 2 Path Search Algorithm: Path_Search(S,P,$Neighbors$,$Pre\_Neighbors$, $Next\_Holes$)

```
Input:
    Neighbor nodes states, $S$;
    Neighbor nodes positions, $P$;
    Neighbor nodes, $Neighbors$;
    Neighbor nodes of upstream nodes, $Pre\_Neighbors$;
    Senders of '$Route\_Search$' message in neighbors, $Next\_Holes$
Output:
    Next Hop, $Next\_Hop$;
$Next\_Hop = null$;
$Neighbors\_NH$ = null;  // holds neighbors nodes except holes
$Positions\_NH[]$ = null;  // holds positions of neighbor nodes except holes
$States\_NH[]$=null;  //holds states of neighbor nodes except holes
if Received message is $Route\_Search$ then
    $Next\_Holes$.add(sender of $Route\_Search$ message);
end if
for $i = 1$ to $Neighbors.length$ do
    if $Neighbors[i] \notin Next\_holes$ then
        $Neighbors\_NH$.add($Neighbors[i]$);
        $Positions\_NH$.add($P[i]$);
        $States\_NH$.add($S[i]$)
    end if
end for
return GGR($States\_NH$,$Positions\_NH$,$Neighbors\_NH$,$Pre\_Neighbors$);
```

a microwave oven. In our algorithm, we let the sensor nodes to monitor their link quality. If the link quality drops under a threshold, the sensor node will broadcast an '*Edge_Update*' message to its neighbors and previous node. The OCCUPY nodes that contains current path in their OCCUPY table will delete the corresponding entry. The previous node will start a similar route search process as in Path Error Handler to find alternative route to the destination.

5.1.2 Mobile Base Station Deployment

In order to adapt our benchmark routing algorithm to handle evolving video streaming scenarios from single stream application to multi-stream applications, we incorporate a mobile base station deployment scheme in our benchmark algorithm to achieve cost-efficient video delivery.

Since the neighbor nodes of an existing active data path can not be employed to relay video data for other paths because of their OCCUPY states, the neighbor nodes of the gateway will be suppressed from transmitting video data if the gateway is the destination of an active data path. As a result, additional video streams cannot find a feasible route to reach the gateway and the mobile base stations are on demand to bridge the video data delivery.

The mobile base station deployment scheme is a slight modification of the strategy that we have employed in Path Error Handler. In addition to recursively apply Algorithm 2 to explore alternative routes to primary base station, each network hole node encountered during the path search period will send their location and corresponding distance from source node (in hop count) back to the source. The source node calculates the data transmission cost based on Eq. (5.4) for every reported network hole node and selects the least one to send data to. The source node will also inform gateway node to deploy mobile base station to the corresponding location of the least cost network hole node to relay the data.

$$cost = W \cdot (C \cdot L_i/TP_i) + (1 - W) \cdot D_i \tag{5.4}$$

In Eq. (5.4), TP_i is the estimated throughput of the potential data path, C is the total volume of video to be delivered, L_i is the number of nodes from the source node to the corresponding network hole node i. D_i is the distance between the network hole node i and gateway. W is the weight. The first term of Eq. (5.4) on the right hand is the potential data transmission cost measured with the overall node activation time of the entire data path during the data transmission. The second term of Eq. (5.4) on the right hand is the potential mobile base station deployment cost measured with the distance between the deployment location and the primary base station. Fig. 5·3 shows an example of mobile base station deployment. In this example, the network holes H_1, H_2, H_3, H_4, during the path search period will report their location and distance from source to the source node. According to the computation at source node, H_1 is the least cost node to deploy mobile base station. The source node will inform gateway to deploy mobile base station to H1 and send data accordingly.

However, this strategy is equivalent to a brute-force search of all possible routes to the network holes to find the optimal solution. It actually builds a tree from the source node to all the network holes. Network holes are indeed the leaves of this tree. In the worst case this strategy is equivalent to traversing all the paths from the root node to all the leaves of the tree. If we assume the maximum number of neighbor nodes for a single node is N and the maximum data length is K, a node in the tree can have at most N children and the depth of the tree is K. As a consequence, the total number of paths from root to leaves of this tree in the worst case is O(N^K) [CSRL01]. The number of paths increases exponentially with respect to the maximum length of a data path. In order to improve the network hole nodes search efficiency, we propose a new route search strategy as follows:

This new strategy can be achieved in two steps. In the first step, the source node is notified that the destination cannot be reached. To realize this notification, we predefine a data path discovery time and a bound for the length of path construction. If the path discovery period exceeds the time limit or the constructed data path exceeds the given bound, the source node is informed that the destination cannot be reached. Note that the data transmission cost via a multi-hop data path is proportional to the length of the path given a fixed throughput performance and the cost for a multi-hop data path transmission must be lower than the cost to deploy mobile bases station directly to the source node to relay data. There exists an upper bound of the path length that can be derived with Eq. (5.5). TP is the estimated throughput of a potential data path. C is the total volume of video to be delivered. L is the maximum number of nodes along a path. D_s is the cost of mobile base station deployment measured with distance between the primary base station and the source node. W is the weight.

Second, once the path formation algorithm realizes that the destination cannot be reached, the end node of the current path under construction will flood a "No_Route" message including the video source node ID and its location to inform all the sensor nodes in the network that the primary base station cannot be reached. All the network holes overhear this message will follow Algorithm 1 to construct data paths toward the source node with the same parameters of path discovery time and route length limit. The source node calculates the data transmission cost based on Eq. (5.4) for each reported network hole node and selects the least one. The source node also informs the primary base station to deploy mobile base station to the corresponding location of the least cost network hole node to relay data delivery.

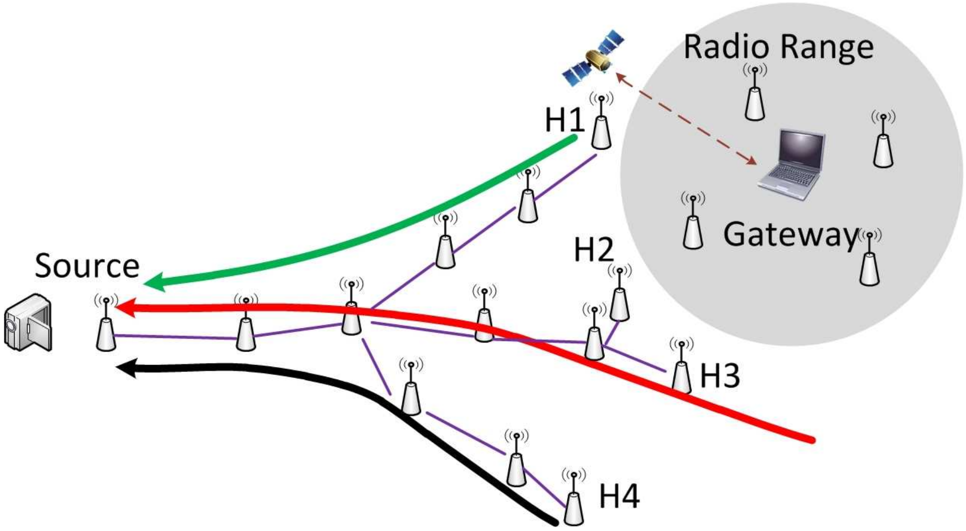


Figure 5·3: Mobile basestation deployment example

Time Complexity Analysis for New Route Search Strategy: With the improved data path search strategy, if the bound of data length is L, the time for broadcasting the notification packet to network holes is T, and the time to construct an one-hop path is constant, the time cost to establish a single data path from a network hole node to the source node is $\mathrm{O}(L+T)$ [CSRL01]. Since all the network hole nodes construct data paths toward the source node in parallel, the total time to construct paths from all network holes to the source is still $\mathrm{O}(L+T)$. Since T is usually proportional to the number of nodes in the network, the time complexity in this improved strategy that search paths in a reversed direction from network hole nodes to the source node is faster than our previous schemes especially in large and dense networks.

$$L < (1-W) \cdot D_s \cdot TP/(W \cdot C) \tag{5.5}$$

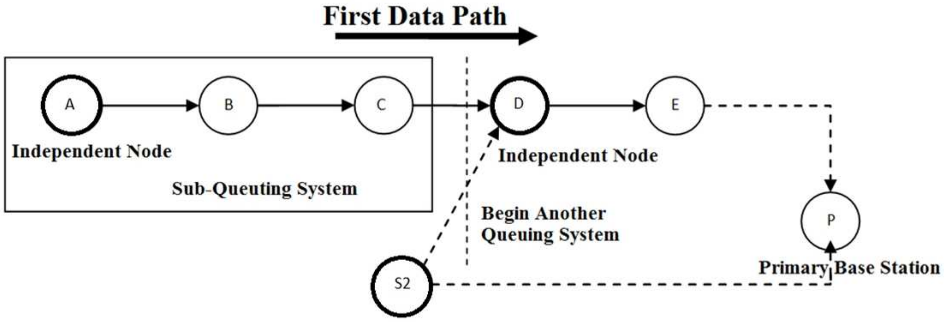


Figure 5·4: Interference tolerant data routing

5.2 Interference-Tolerant Path Formation Algorithm

The benchmark routing algorithm can be extended to support interference-tolerant video streaming applications. In such applications, some interference from adjacent data paths is acceptable as long as the estimated video streaming throughput performance still satisfy the application requirements. Such tolerance allows two video streams to share a section of one existing path to lower data transmission cost rather than construct two disjoint paths as in the benchmark algorithm. Fig. 5·4 demonstrates the path formation scenario of our interference-tolerant routing algorithm.

There are three main steps to set up a data path in the interference-tolerant routing algorithm. In step one, the video source node S_2 finds a temporary path to the primary base station P and sends a request to setup the data path. In step two, the primary base station receives the request from the video source node, it calculates the best potential destination based on throughput and data delivery cost estimation. Then the primary base station informs the video source node of its data delivery destination. Usually the best destination is a head node of a queuing system that is nearest to the new source node along an existing path. And this path is not interfered by other streams. Node D in Fig. 5·4 is an example of such head node. Node D is a good candidate for data routing destination of S_2 attributed to three factors: first, node D is close to S_2; second, the mutex system pair can be equivalent to a single system led by node D to resolve packet collision issues as shown in Fig. 4·11; third, as the first data path is not interfered by other streams, the throughput performance after the admission of new video stream from S_2 in queuing systems led by node D is still likely to satisfy the throughput requirements for both video streams. Therefore, we can share the nodes after node D along the first data path to deliver data with good performance and save energy cost. In step three, the video source node constructs the data path toward that destination and transmit data.

In the following we describe the details of each step in our interference-tolerant algorithm:

- *Step One: Sending Path Setup Request*

  At the first step, the video node dynamically forms a temporary data path to the primary base station using exactly the same greedy routing algorithm described in Algorithm 1. If the route is not available, it runs the Path Search Algorithm (Algorithm 2) to find a temporary route to a network hole node with the least data delivery cost. Once the temporary route is established, the source node sends a request to inform the gateway of the cost estimation of the temporary route. The route cost estimation is calculated according to Eq. (5.4).

- *Step Two: Destination Notification*

  Once the primary base station receives the request, it estimates the throughput performance and data delivery cost for different path construction options. There are two distinct path construction options.

  The first option is that the video source constructs a data path directly to gateway without interfere with other existing paths and mobile base stations can be employed to enhance the throughput performance and bridge the network gap. The cost of the first data path construction option is obtained from the received request message.

The second option is to let the video source construct data path towards a head node of a queuing system along an existing data path. The head node must be close to the source node and the existing data path is not interfered with other paths. This option allows the video source node to share a section of the existing active data path for data delivery and tolerant the interferences between the new video stream as long as the throughput performance estimation still satisfies the application requirements. Since the head node can be close to the video source node, constructing the data path toward the head node and sharing existing active paths for data delivery can achieve better path setup delay and lower data delivery cost.

To estimate the throughput and cost, the primary base station scans its existing path information table, select a head node that is nearest to the video source node and the data path is not interfered by other streams. Once the head node is selected, the primary base station obtains the length of the data path section between the selected head node and the primary base station in hop count denoted as N_h. The primary base station also estimates the necessary number of hops N_e between the new video source node S_2 and the selected head node D according to their distance in between. N_e is estimated with Eq. (5.6).

$$Ne = Distance/Communication\_Range \tag{5.6}$$

We denote the length of the data path that the head node belongs to as L, the residual volume of video content to be delivered of the existing stream through this data path as C_r , the current throughput of the existing data path as TP_c, the throughput estimation of the existing video stream after the admission of new stream at the selected head node as TP_r, the throughput estimation of the new stream as TP_n and the volume of video content on the new source node S_2 as C_n. TP_r and TP_n can be obtained via Eq. (4.12) and Eq. (4.13). The head node is a good data delivery destination candidate if both TP_r and TP_n satisfy the application requirements. Otherwise a second round of head node selection will be performed by excluding the previous candidate data path information. Eq. (5.7) computes the data delivery cost estimation for the second data path construction option.

$$W \cdot [C_r \cdot (\frac{1}{TP_r} - \frac{1}{TP_c}) \cdot L + \frac{C_n}{TP_n} \cdot (N_h + N_e)] \tag{5.7}$$

$C_r \cdot (\frac{1}{TP_r} - \frac{1}{TP_c})$ is the extra data delivery cost per node of the original stream due to the decrease of the resulting throughput measured with extra node activation time. $\frac{C_n}{TP_n} \cdot (N_h + N_e)$ is the estimated data delivery cost for the new video stream. If the cost of the second option is less than the cost of the first data path construction option obtained via received request, the primary base station replies the information of the selected head node to the video source node including the head node location, path ID, node ID. Otherwise it informs the video source node to deliver data via temporary route constructed in step one.

- *Step Three: Data Path Construction*

  When the video source node receives a reply from the primary base station with information of the head node, it constructs data path toward the head node with the benchmark algorithm. All the available neighbor nodes and nodes that only under interference of the selected head node can be employed to construct the path. If the path construction toward the head node fails, the video source node will fall back to the first data construction option and deliver data via the temporary route constructed in step one and can leverage mobile base stations to assist the video data delivery.

5.3 Distributed TDMA Packet Scheduling with Congestion Control

As we have mentioned in Chapter 4, the success of our data transmission model is depended on the service time variability of each subsystem. If we can construct a packet scheduling scheme which guarantees the constant service time, then such scheme is perfect for our throughput estimation model. Inspired by this motivation, we propose the following TDMA-based packet scheduling scheme with a simple congestion control scheme.

1. *Media Access Control*

   We assume that each packet has the same size, thus the slot length for transmitting one packet can be calculated according to the channel bandwidth or manually assigned by the source node. The size (number of nodes in the queuing system) of each system can be calculated according to Eq. (5.1). Since our subsystem is only comprised of three sensor nodes, by assigning each node a transmitting slot, three time slots will be enough for packet transmitting in the subsystem.

2. *Congestion Control*

   The first node of each queuing system along the path is the head node of the system. The head node performs a simple congestion control strategy to decide whether the received packet is dropped or forwarded. If the system is busy serving one packet, all the packets received by the head node are dropped due to packet corruption. No extra buffer is needed on sensor nodes to cache packets. In order to implement this congestion control, we start a count-down timer on the head node every time it begins forwarding a packet. The initial value of the timer is three times of the slot length. The head node only accepts packets when the timer reaches 0. This property is complied with our G/G/1/1 queuing system. The reason that we drop the received packets when system is busy because these received packets are likely to be corrupted by the forwarding packet inside the system given our shared media assumption. Simulations show that this congestion control scheme does not have any performance degradation. Furthermore if the source node sends packets with high rate, this simple proactive congestion control at the head node drops packet at the beginning of the data path and avoids network congestion in the middle of the path.

3. *Time Synchronization*

   We employ the similar time synchronization strategy as described in [DM09] to synchronize sensor nodes within a local system. Pairs of nodes achieve mutual synchronization by exchanging clock offsets in the network beacon messages, similar to the Network time Protocol (NTP) [GKS03]. Unlike [DM09] that builds a network wide synchronization tree, we only build the synchronization structure inside the local system. Each node in the local system is only synchronized to the first node (head node) of the system and there is no synchronization requirements amongst different systems.

Previous TDMA packet scheduling strategies in [HTLG97, CTG97] require an effective network-wide synchronization among all nodes in the network and applying such highly synchronized solutions is expensive for ad hoc networks [YK05]. Our

TDMA scheme employs a distributed synchronization strategy to lower the synchronization requirements. We only perform effective synchronizations within the local systems and there is no synchronization requirements amongst different systems. Soft-TDMAC proposed in [DM09] is a good candidate for the implementation of time synchronization inside our local systems. Soft-TDMAC has a synchronization mechanism, which can synchronizes all nodes in our system to within microseconds of each other and it is a software based TDMA and can be implemented over off-the-shelf hardware.

Chapter 6

Network Reprogramming Framework for Wireless Sensor Networks

Another challenge of WVSN applications is how to efficiently disseminate the program code in the network which can be adapt to the improvement of the sensing algorithm, environment change, application adjustment, and etc. In this section, we propose network reprogramming framework to implement flexible and efficient code dissemination across the WSNs.

6.1 Introduction

Wireless sensor networks (WSNs) are expected to encompass very large numbers of devices. For example, in a building automation scenario, it is not unreasonable to expect many sensors and actuators to exist in each room. These can be smoke detectors, light switches, lights, HVAC valves and thermostats. History shows us that these devices continue to reduce in size, energy consumption, and improve in performance. however, replenishment of these devices presents heterogeneity challenges as newer devices often rely on more recent code releases or features. Thus efforts have been applied to create code dissemination schemes that permit updates to be propagated wirelessly, "over the air"(OTA) to all devices in the system.

However, many existing OTA schemes [HST08, SCC$^+$06] are usually an all-or-nothing activity for various reasons, primarily because the devices are resource limited

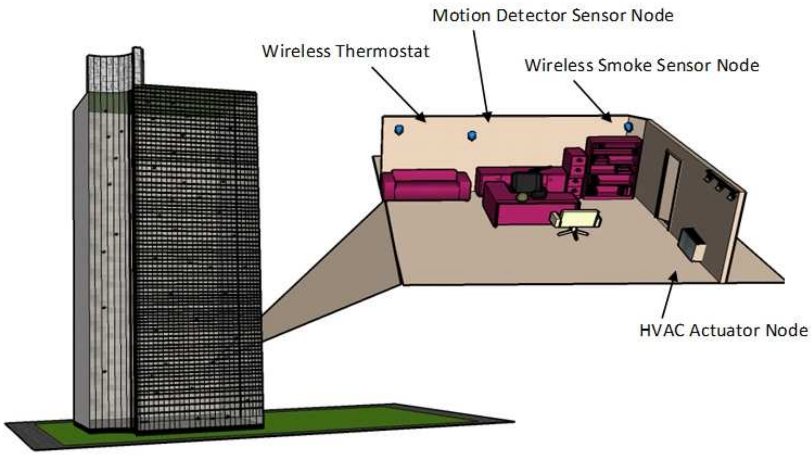


Figure 6·1: WSN in building automation example

and are challenged to support large or complex process management, and because of the goal to have predictable behavior by limiting the system to a single application. We challenge these assumptions by targeting specific objectives:

1. Allow the WSN to support multiple coexisting missions
2. Allow the missions to be dynamically created, injected, and retired
3. Be energy efficient as a goal but not a primary motivation

A mission is defined as a WSN function, such as an HVAC-thermostat-actuator control loop. The function is implemented by code blocks rendered and configured on the motes involved in the control loop. These small code blocks called tasks. Thus there can be n task instances that exist for this function on a subset of the WSN. In order to implement this model we must achieve certain behaviors that emerge:

1. Concurrent functions: multiple control loops in the wireless sensor networks that are non-intersecting (e.g., multiple rooms)

2. Concurrent tasks on a device: multiple functions that leverage the same sensor
3. Baseline propagation of tasks for installation and propagation in the system
4. Injection of missions, defined by tasks, into the system for dissemination, adaptation, and execution
5. Limiting of task propagation and instantiation to the devices that need the tasks
6. Mobile code translation to accommodate functions not known in advance

With this model, we design general paradigm for programming subsets of a WSN. An artifact of this model is the ability, by leveraging attributes associated with devices and their data, to form specialized communication abstractions [WM04].

6.2 Related Work

Wireless sensor networks are characterized by low cost, low power embedded sensor devices that are network-enabled. For most applications, sensor nodes in a WSN are initialized once and intended to operate for the duration of the application or mission. However, shortcomings of this approach include the costs associated with implementing code updates. Thus there has been considerable interest in alternative models to (a) reprogram, (b) design APIs for programming, (c) better map interactions among nodes to the WSN via programming techniques. Hood [WSCB04] is a neighbor-based programming framework addressing the latter topic. This scheme parameterizes the network application as different attributes and provided filtering mechanism to select and share data based on data attribute. Abstract Regions [WM04] generalize and improve the idea of Hood via abstractions above required inter-device communication. It also considers mechanisms for data aggregation. Mate [LC02], TinyDB [MFHH05], Trickle [LPCS04], and Squawk [SCC$^+$06] address code dissemination with a focus on efficiency. Mate and TinyDB use high-level virtual code representations to reduce the code transmission cost. Mate is quite functional but is limited in the scope of instructions that can be achieved in its virtual machine and is best suited for simple applications. Trickle improves Mate, with a remedy for simple flooding for dissemination by limiting code propagation to subsets of network nodes. Squawk introduces Java as a WSN programming tool and uses Squawk bytecodes to transmit a program.

Unfortunately these code dissemination schemes do not support incremental application updating. [JC04] and [RL03] provide some solutions. Reference [RL03] develops an algorithm that can efficiently encode program updates. An edit script is generated by host program and corresponding operations are cast on each device at the instruction level. However, this work is essentially a coding scheme with strong processor dependencies. Reference [JC04] describes a generalized program update scheme. Without prior knowledge of the program code structure, this scheme is designed to distribute key changes of new version of a program. However, a shortcoming is the inability to distribute different application code to different subsets of nodes. MOAP [SHE03] and Deluge [HC04] are two multi-hop network programming implementations. MOAP succeeds in propagating program code to a selective number of nodes without saturating the whole network. Deluge disseminates the program in an epidemic fashion and improves the throughput using optimization techniques like adjusting transmission rate. Unfortunately, the required rebooting process of new program loading degrades the performance of network application.

6.3 Task Based Reprogramming

Our proposed tasking scheme relies on three key components: (1) a decomposition of tasks, (2) installation or injection into the system, and (3) message processing based on attributed data at each node. These are each discussed in detail below.

6.3.1 Task Decomposition

We seek to enable incremental, functional updates, and the ability to support concurrent missions. In our scheme we model a sensor net application as a mission supported by a set of independent and identical tasks that are disseminated on a subset of the WSN devices (a mission is achieved by identical tasks disseminated on a subset of all nodes). However, this subset is guided by attributes associated with the devices and predicates that aid in their instantiation. For example, one can envision a temperature monitoring application that is instantiated on nodes that possess temperature sensors with two independent task modules: a routing task and a temperature measuring task as shown in Table 6.1. Other nodes in the network need only to support the routing task. Multiple missions are achieved by the injection of multiple tasks that are disseminated to nodes with matching attributes. The decomposition process is based on the analysis of the running application. Right now we do not have a general algorithm for the decomposition but we require the decomposition to achieve the following goal. We denote the function of the running application as $F(A)$, the function of the decomposed task as $F(T_i)$, the decomposition requires that:

1. $F(A) = \sum_{i=1}^{N} F(T_i)$
2. All of the decomposed tasks T_i form a partition of the general application A, a partition of a set A is a set of disjoint subsets of A, whose union is A.

$$T_i \in A, \quad \bigcup_i T_i = A, \quad T_i \bigcap T_j = \emptyset, \quad if \quad (i \neq j)$$

The independence of the different tasks in the WSN benefits the task installation since it ensures no interference with currently running tasks. Every node contains a

Task Table	
Application: Temperature Monitoring	
Task One: Measure Temperature	Task Two: Routing Temperature Data

Table 6.1: Task table sample of sensor network temperature monitoreing applciation

task table and tasks run in parallel on the node. Furthermore, the process of new task installation is eased by only adding a new entry to the node's existing task table.

6.3.2 Task Installation

Unlike existing over-the-air programming (Deluge [HC04], Squawk [SCC$^+$06]) which feeds all the nodes with the same code, we install different tasks onto different nodes based on the predicates used in dissemination that are applied on node attributes.

Our task installation and forwarding scheme are inspired by Content Based Routing protocol proposed in [CW03]. In this scheme, a set of predefined predicates on each node can be used to delineate a subset of nodes to target our tasks. These predicates per node are called the Installation Predicate (IP). The Installation Predicate is defined as follows:

$$IP = P_1, \;\; P_2, \;\; P_3, \;\; \ldots \;\; where \;\; P_i \;\; is \;\; individual \;\; predicate \;\; on \;\; each \;\; node$$

The Installation Predicate (IP) over the received task code defines a task addressing scheme.

$$IP(T_i) \longrightarrow \{true, \;\; false\}$$

If the result is true, the node will install the received task module, otherwise the node ignores the task and forwards it to the proper neighbors. In the forwarding process, each node forms a *Forwarding Predicate* (FP) related to each neighbor using the algorithm proposed in [CW03]. We denote the set of all neighbors of each node as N and give a *Task module Forwarding Function* (TFF) to identify the neighbor which

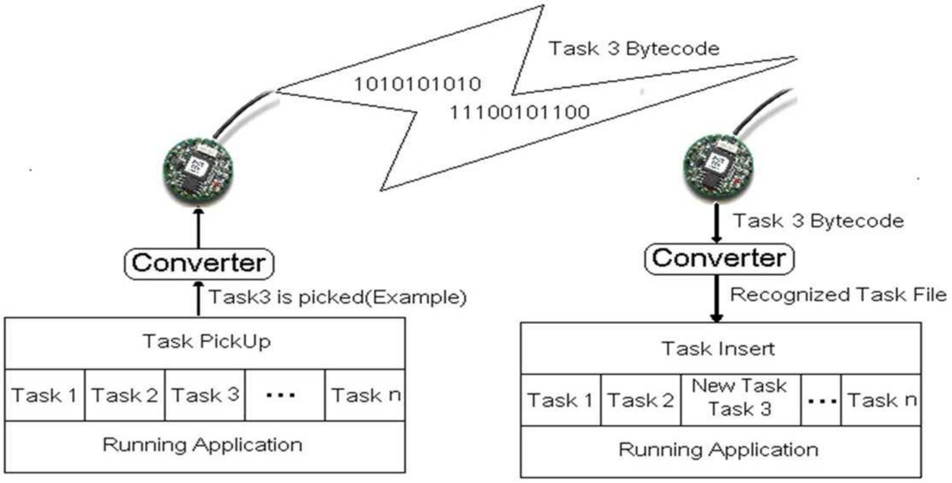


Figure 6·2: Illustration of wireless task installation

needs to receive the task module code as shown in Eq. (6.1).

$$TFF(T_i, \quad FP) = \{i : \quad i \in N \bigcap FP(T_i) = True\} \tag{6.1}$$

During the task installation process, there is no reboot or reset operation as associated with other types of over-the-air programming. Every application can be modified and updated by stopping, retiring, and adding different tasks. The task installation and deletion process is intended to leverage common data structures on each node (e.g., neighbor connectivity information) yet not interfere with concurrent tasks. Optimization of inter-task cooperation is beyond the scope of our current scheme. The installation process is illustrated in Fig. 6·2. Here a set of running applications is defined by multiple tasks. The sender converts the updated task (Task 3 in Fig. 6·2) into byte code and transmits the code over the network. The receiver converts the byte code back to a task module and inserts the task module into its own task table to accomplish the application updating without interfering the other running task modules.

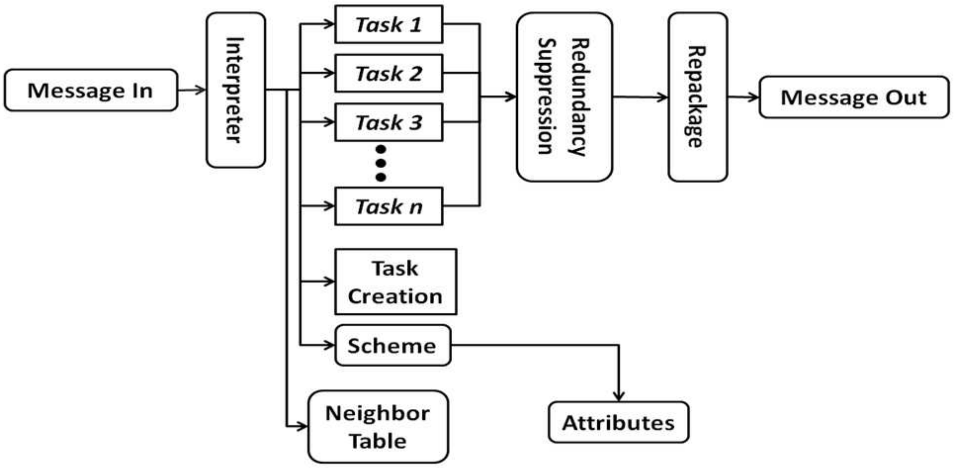


Figure 6·3: Message processing within a node

6.3.3 Message Processing

In our framework, the node behaves as a message forwarder. Fig. 6·3 shows the message-processing framework within a node. Every incoming message is first parsed by an interpreter according to the predefined message format. The Interpreter will then forward the message to a proper task module for further processing based on the attribute forwarding scheme. The processed messages are all sent to a package module for duplicate removal. This module will repackage the received messages and forward them to other nodes.

6.3.4 Main Features of the Task Based Reprogramming

Independent tasks serve as the logical programming units mapped to nodes in the network. The application can be easily modified or updated by installing, updating and deleting individual task modules.

We use predicates, which are predefined installation rules, to guide the installation of the task module. Predicates enable us to install different tasks on different nodes based on the nodes' property. This feature prevents the node from installing the code unrelated to its function and thus saves resources.

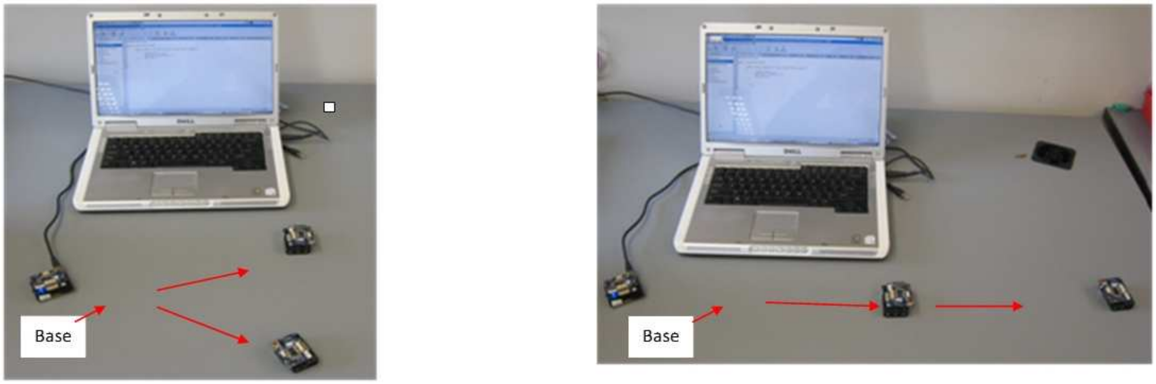
There is no reboot or reset operation on the node. The node's function is achieved by combining different independent and parallel running task modules together. Task independence along with the message processing scheme guarantees that the update of one particular task module will not interfere with other running tasks. Thus there is no direct impact on the network's performance.

Finally, the support for independent tasks allows the injection of multiple missions represented by independent tasks. Different missions are achieved by injecting unique tasks, or identical tasks with different injection predicates.

6.4 Proof of Concept Implementation of Task Based Reprogramming

We implement our framework on the Crossbow Imote2 platform with attached sensor boards. The motes were installed with embedded Linux v.2.6.14 including Java v.1.3.0. Java is selected due to its popularity, convenient APIs, and the potential to support code mobility. Support for handling multiple tasks is achieved via Java multi-threading operations. Code injection is achieved with Java's mobile code transmission architecture. However, some effort is required to adapt the Java VM to support instantiations from foreign nodes that receive injected code due to the security compliance for Java's dynamic class loading.

In our demonstration of the framework we program a base station with two parallel tasks that are injected into a set of autonomous WSN nodes. The first task enables each mote to toggle its blue LED once per second. The second task leverage the Imote2 sensor board to periodically sample the temperature and collect temperature reads from other nodes in the network. The node with the highest value illuminates its red LED. We successfully inject this two concurrent tasks in both single-hop (star)



Single-hop Scenario

Multi-hop Scenario

Figure 6·4: Prototype Setup

and multi-hop scenarios in our experiments as shown in Fig. 6·4. For the single hop scenario (Fig. 6·4), the two satellite motes are within one hop range of the base station. We observe that the two satellite motes would install new task simultaneously which is a gain from the code broadcast of the base station. For the multi-hop scenario (Fig. 6·4), one of the satellite motes is out of range of the base station; the three motes consist of a two hop network. Periodically broadcasting of task list guarantees the more distant mote from base station can correctly install the new tasks. In our demonstration we implement some basic Java APIs for Imote2 programming, such as *Task* interface, *Sensing* interface, *Listening_Message* interface, *Data_Sending* interface. These APIs could be easily ported to any other sensor nodes which are equipped with a standard Java virtual machine. We hope in the future we can provide additional standard Java APIs of our scheme for the users to make the programming of WSN simpler and thus encourage more effort to be given to investigate the potential of WSN application instead of worrying about the implementation complexity.

Chapter 7

Performance Results

In this chapter, we present the performance results obtained from simulation for our analytical models and proposed network code dissemination framework. We demonstrate the advantage of our distributed TDMA-based packet scheduling scheme over traditional widely accepted CTS/RTS scheme for video data transmission by presenting resulting throughput performance comparisons in different scenarios with OPNET simulations. The distributed TDMA packet scheduling scheme is the guarantee of the accuracy of our end-to-end throughput estimation model. We evaluate the accuracy of our data path throughput estimation analytical models via the comparison of the analytical results with the simulation results. These comparisons are conducted under two different application scenarios; isolated video stream scenario and interference-tolerant video stream scenario. Under the isolated video stream scenario, we form a single video data delivery path from the video source node to the destination node according to the benchmark routing strategy in Chapter 5. Under the interference-tolerant scenario, instead of forming separate data paths to support distinct video sources, at most two video sources are allowed to share a common section of an existing path as described in interference-tolerant routing algorithm of Chapter 5 to resolve packet collision and throughput estimation issues. The results of comparison demonstrate that our estimation maths the throughput measurements with high accuracy.

In addition, we illustrate the simulation results for the code dissemination efficiency of our network reprogramming framework in this section. The results indicate that our task based reprogramming framework outperforms traditional ‘over the air code’ code dissemination strategy. At last we present a framework “Routing Algorithm Constellation Graph” to compare the performance of different video data routing techniques. This graph makes a easy way to select the most appropriate routing technique for a specific video streaming application over WSNs..

7.1 Performance Evaluation of TDMA Scheduling Scheme

We implement our TDMA-based media access scheme and a simplified CTS/RTS scheme which is widely adopted for IEEE802.11 based protocols in the simulation. We compare the data path throughput performance of the two schemes, and the results indicate that TDMA based scheme outperforms CTS/RTS when the data rate is relatively high. Although our model is suitable for general packet arrival distribution, given the fact that most video streaming applications have the constant bit rate property, our experiments focus on the data source with constant transmitting rate.

We compare the end-to-end data path throughput performance of our distributed TDMA scheme and widely adopted CTS/RTS scheme in two different scenarios. In Scenario 1, the link utilization ratio is very low. The data source generates packets with a rate of 1024 bit every 5 units time. The link capacity is 1024 bit per unit time. The end-to-end data path capacity is 1024 bit every 3 units time. The results are shown in Fig. 7·1. In Scenario 2, with the same link and end-to-end path capacity, we increase the data arrival rate to the data path capacity of 1024 bit every 3 units time. The results are shown in Fig. 7·2.

From the results, we conclude that in the low link utilization scenario (Scenario 1), both TDMA and CTS/RTS schemes can eventually achieve the same performance,

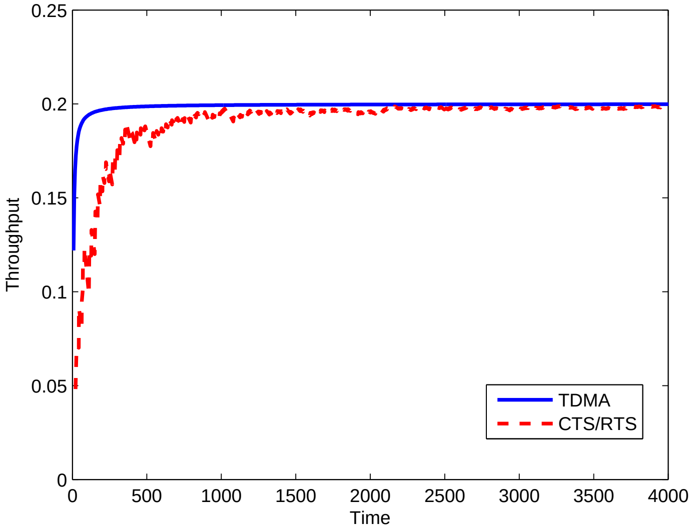


Figure 7·1: TDMA vs. CTS/RTS in low data rate scenario (1024 bit every 5 units time)

however, the TDMA scheme adapts to the path capacity faster than the CTS/RTS scheme. In contrast, the CTS/RTS scheme performs poorly in substantial high link utilization scenario (Scenario 2). It only achieves 75% of the path capacity and also generates jitter in the data delivery. Our TDMA scheme outperforms the CTS/RTS scheme in both scenarios. More over the implementation of our TDMA scheme is not difficult as discussed in Chapter 5. Our TDMA scheme avoids extra energy consumption over handshake packets that are required for the implementation of CTS/RTS schemes.

7.2 Accuracy Analysis for Throughput Estimation Model

In this section, we analyze the accuracy of our throughput estimation model. We compare the analytical results with the simulation measurements under two different scenarios: isolated video streaming scenario and interference-tolerant video streaming scenario.

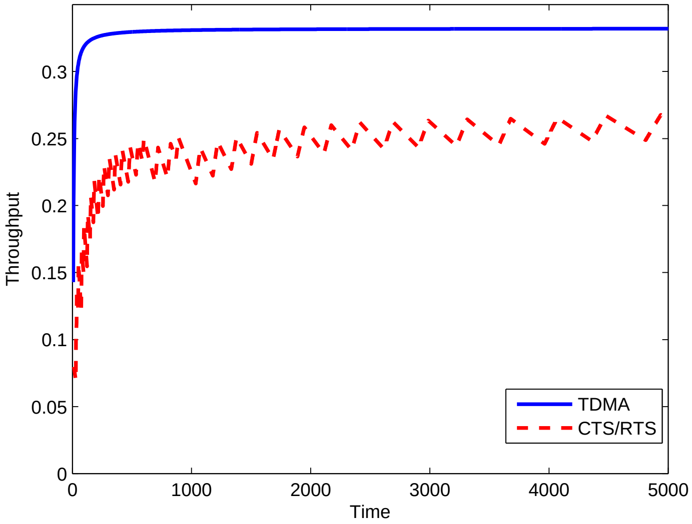


Figure 7·2: TDMA vs. CTS/RTS in high data rate scenario (1024 bit every 3 units time)

7.2.1 Throughput Estimation for Isolated Video Streaming Scenario

In this scenario, we form a data path with 3 concatenated queuing systems. Each subsystem consists of three sensor nodes. According to Theorem 2 of Chapter 4, we can derive the throughput performance of the entire path via just calculating the throughput of the first subsystem along the path. For simplicity, we assume the packet arrival interval of the first queuing system along the path is exponentially distributed with mean of 10 units time, and the link transmission delay of each packet is one unit time. Consequently, the queuing system delay by applying our TDMA scheme is 3 units time. By taking expectation on both sides of Eq. (4.5), we derive the theoretical expected throughput of the queuing system to be approximately 0.079 packets per second which is very close to the data path simulation measurement of 0.0784. Fig. 7·3 demonstrates the simulation results.

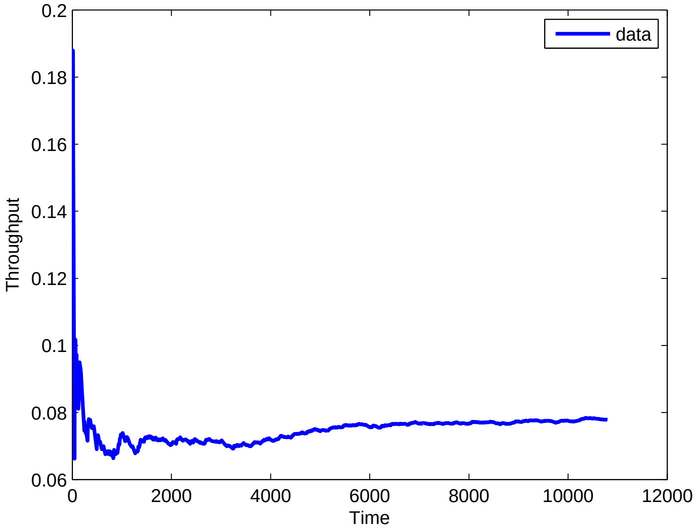


Figure 7·3: Data path throughput measurement in simulation

7.2.2 Throughput Estimation for Interference-Tolerant Video Streaming Scenario

As we have mentioned in Section 4.5, the output throughput performance of the entire data path is equivalent to the throughput performance of the first mutex system pair if adjacent streams can share an existing data path for video data delivery and satisfy certain requirements. We can achieve this adjacent stream data delivery scenario with our Interference-Tolerant Path Formation Algorithm in conjunction with simple admission and congestion control strategies discussed in Chapter 4 and Chapter 5. An example of the achieved data delivery topology is shown in Fig. 7·10. Instead of forming two disjoint data paths to the sink, Src_2 leverages the path formation algorithm to construct path toward the nearest head node of the existing path of Src_1 and share a section of the existing path to deliver video data to the sink. We use this topology for our simulations. The throughput performance estimation is related to the location of the mutex pairs along the direction of data delivery. Approximations are conducted to simplify the mathematical deduction.

We compare our estimation results with the simulation measurements under the three scenarios we mentioned in Section 4.5. We assume a Poisson packet arrival process from both video source nodes. The simulation experiments are conducted under two different environments. In the first environment the packet arrival rate for both video source nodes are identical. In the second environment we use different arrival rates for different video source node.

Fig. 7·4, Fig. 7·6, Fig. 7·8 are results for identical arrival rate situation corresponding to different mutex system pair positions elaborated in Section 4.5 where the constant service time for each subsystem along the path is set to 3 units time, and the average interval of packet arrival from video source nodes is 10 units time. Fig. 7·5, Fig. 7·7, Fig. 7·9 present the results of distinct packet arrival rate situation under different mutext system positioning scenarios mentioned in Section 4.5 where the constant service time for each subsystem along the path is set to 3 units time as well and the average intervals of packet arrival are 10 units time and 20 units time for data paths 1 and 2 respectively. Our simulations are also conducted against different data path length. The horizontal axis represents how many queuing systems the data paths contains: a larger number indicates a longer data path. The vertical axis represents the average packet arrival interval at the destination. Each subsystem is comprised of three sensor nodes as described in Chapter 4.

The simulations demonstrate that our estimation model is accurate. The relatively flat line of packet inter arrival time at the sink is also a reflection of our Theorem 2 in Section 4.4. It indicates that the throughput of the entire data path is independent of the path length and is only associated to the output process of the first mutex system along the path.

In order to demonstrate that our throughput estimation for adjacent data paths

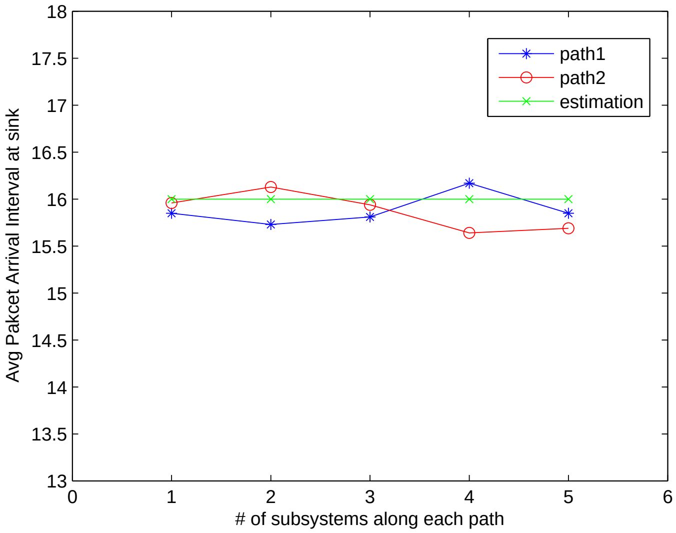


Figure 7·4: Scenario 1 estimation results for identical data arrival

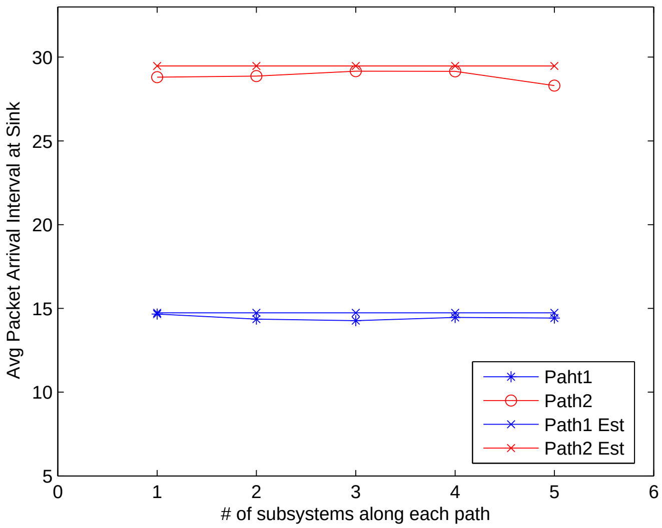


Figure 7·5: Scenario 1 estimation results for distinct data arrival

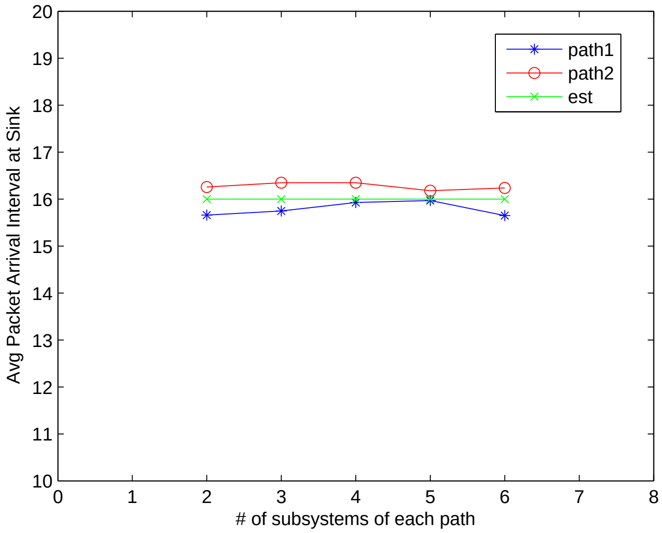


Figure 7·6: Scenario 2 estimation results for identical data arrival

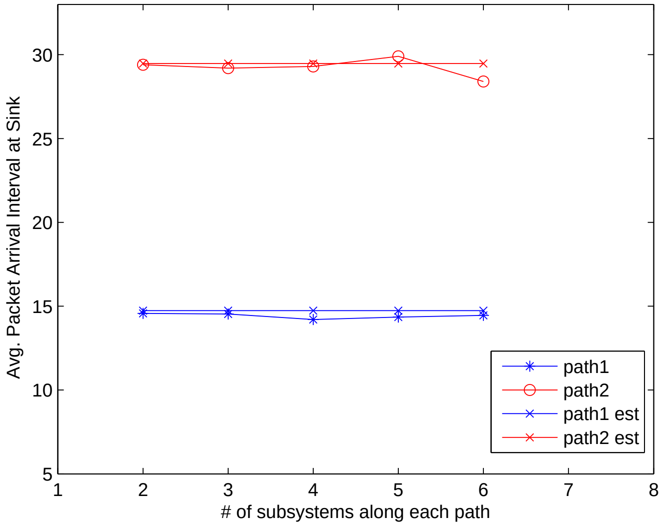


Figure 7·7: Scenario 2 estimation results for distinct data arrival

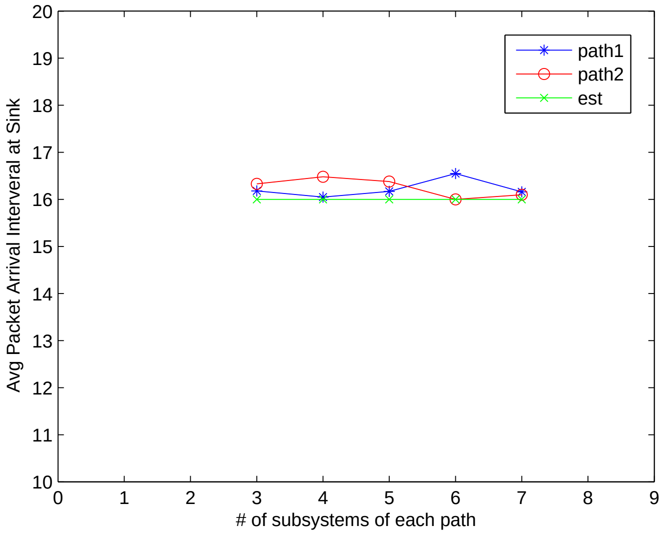


Figure 7·8: Scenario 3 estimation results for identical data arrival

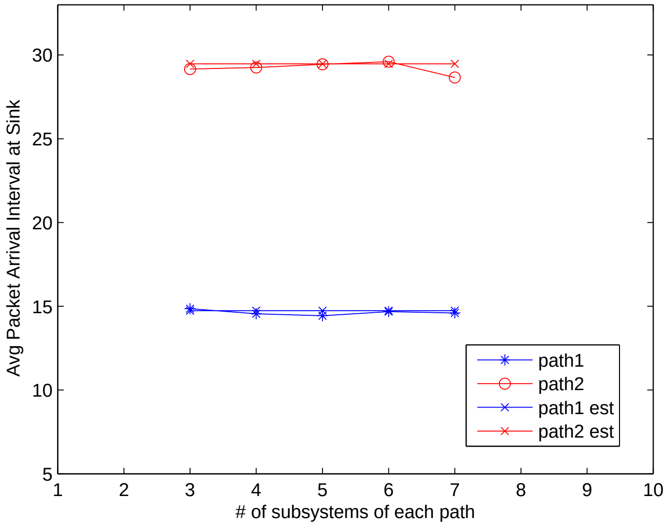


Figure 7·9: Scenario 3 estimation results for distinct data arrival

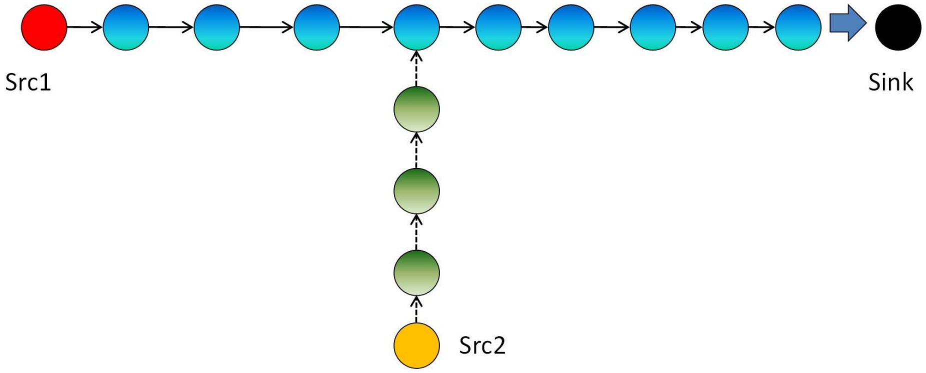


Figure 7·10: Topology of simulation for 9 nodes on each path

is not sensitive to the constant service time that we ignore for our Scenario 2 and Scenario 3 to simplify the analysis. We conduct our simulation in Scenario 3 against different service time ranging from 3 units time to 9 units time and the average interval of packet arrival from both video source nodes are configured to 10 units time. The results are shown in Fig. 7·11. From the simulation result, we can see that a 300% increase of service time from 3 units time to 9 units time only results in a limited (approximated 5%) inaccuracy increase for our estimation. This result suggests that the approximation we have conducted to simplify our analysis is tolerant and acceptable.

7.3 Network Reprogramming Framework Efficiency Analysis

We build a simulation environment for our proposed network reprogramming framework of Chapter 6 using OPNET based on grid-shaped network topology shown in Fig. 7·12: (*node*_0 is the base station).

Radio transceivers in the node model have the following parameters: a 2.4GHz center frequency, a single physical channel with a bandwidth of 5MHz, and the raw bit rate of 250kbps. These parameters are consistent with the standard physical channel

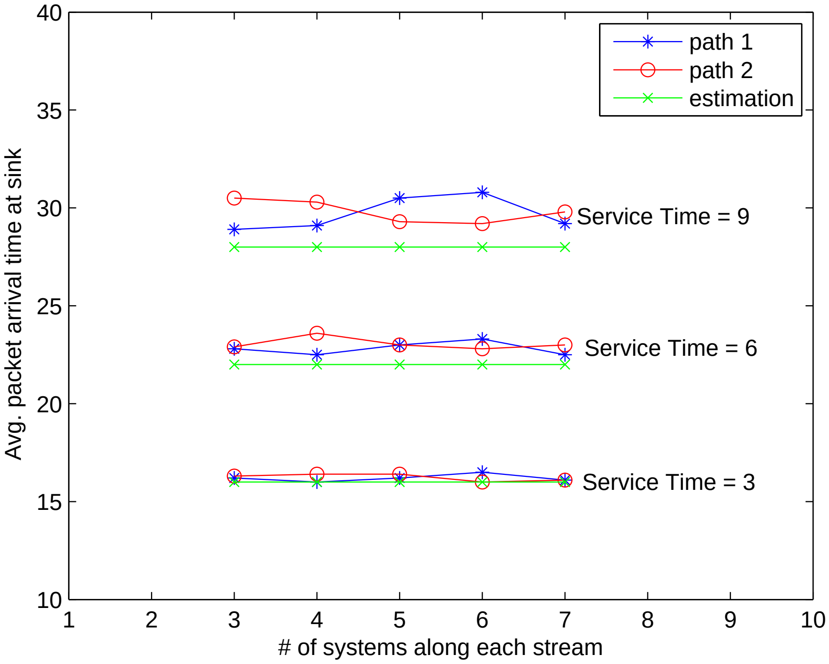


Figure 7·11: Throughput estimation related to different constant service time

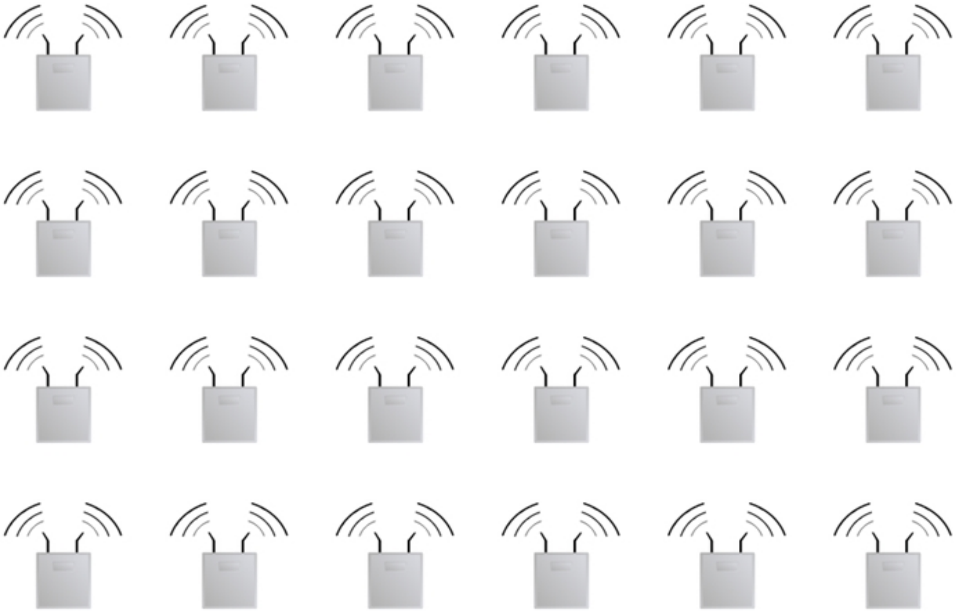


Figure 7·12: Grid topology for simulation

setup used in the IEEE 802.15.4 used by the Imote2s in our testbed in Chapter 6. We also customized the receiver group pipeline stage function such that the transmission range of a node is set to 10 meters for simulation

We conduct OPNET simulation to exam the efficiency of our proposed network reprogramming framework. For simplicity in the simulation we do not consider radio collision problems and assume that they are addressed by an underlying MAC layer. We regulate the size of each task to be identical and equal to 1KB. In order to avoid the collision, we set the task advertisement period TAD_TIMER expiration to be a random time uniformly distributed between $[5, 10]$ seconds. When the node receives the task advertisement, it follows the rule we introduced in Chapter 6 to decide whether to install the task based on the computation results of task installation predicator. The node will wait a random back-off time to reply back task advertisement to avoid packet collision at the origin of the advertisement. The back-off time

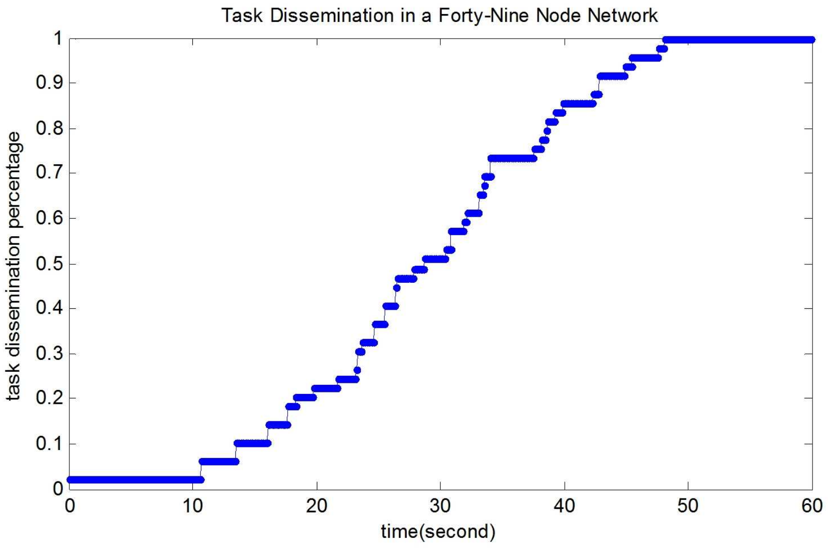


Figure 7·13: Task dissemination time (network size = 49)

is uniformly distributed between $[0.2, 0.4]$ seconds.

We first analyze the single task dissemination time over different sized network. We apply our task injection scheme on a square mesh network with the size range from $4(2 \times 2)$ nodes to $49(7 \times 7)$ nodes. Fig. 7·13 shows the results for the largest network size (7×7). From the results, we observe that a task can be completely installed on all nodes in a $49 - node$ network spanning an area of 360 square meters within one minute. This time is negligible compared to the lifetime of a typical WSN and little impact on the overall performance of the network.

We further measure the task dissemination speed on different sized networks. The definition of average task dissemination speed is as follows: The results are provided in Fig. 7·14.

$$AVGSpeed = \frac{TotalnumberofTaskstoInstall}{ApplicationDisseminationTime}$$

Fig. 7·14, illustrates an advantage of our scheme; the task dissemination speed increases with the size of the network. By further analyzing the relationship be-

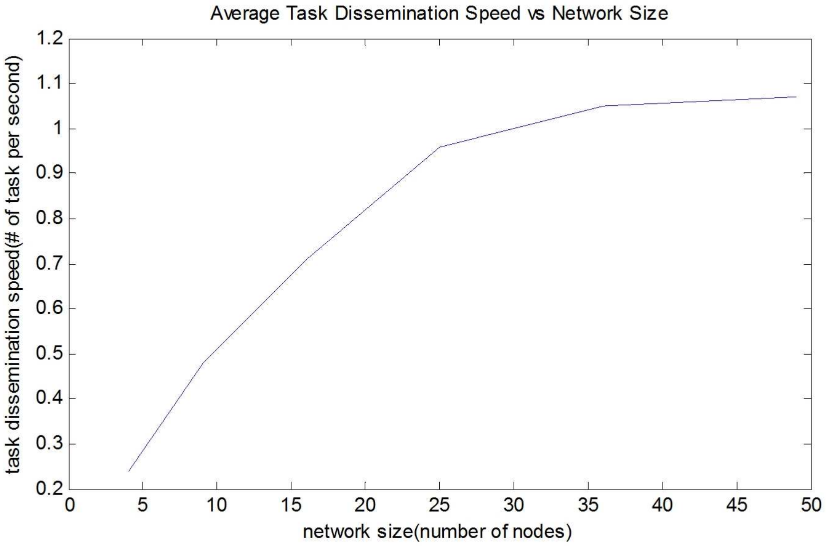


Figure 7·14: Task dissemination speed w.r.t. network size

tween Task Injection Completion Time and Network size we get the results shown in Fig. 7·15:

Fig. 7·15 demonstrates that task dissemination time complexity in our scheme grows almost linearly with respect to the size of the network. This result indicates good scalability of our scheme.

To demonstrate that injecting wireless sensor network applications as a composition of smaller, autonomous tasks is more efficient than injecting large monolithic applications as a complicated heavily coupling single program we compare a sample wireless sensor network application injection results in Fig. 7·16. This sample application is first implemented following the philosophy of task decomposition according to our proposed framework with five independent tasks and deployed according to our task installation scheme described in Chapter 6. The second approach installs the entire sample application at once on each sensor node according to traditional "over-the-air" programming scheme similar to Deluge [HST08]. Fig. 7·16 represents the network application installation percentage against installation time for both

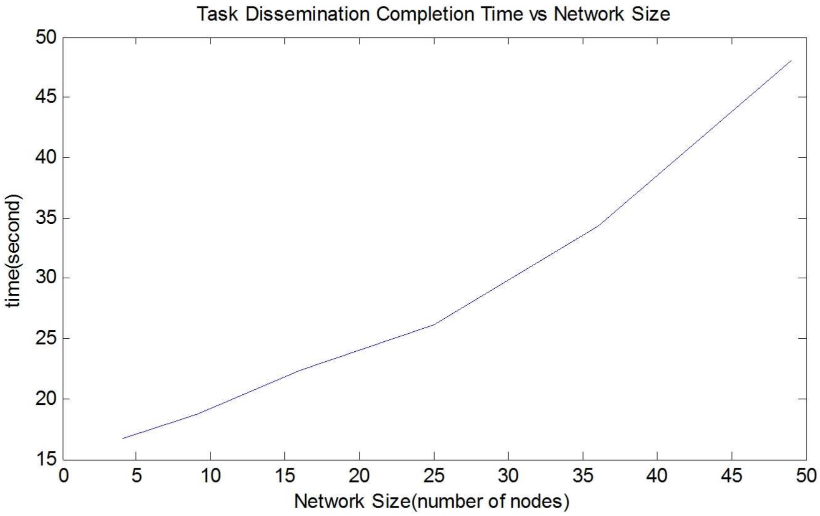


Figure 7·15: Task dissemination time vs. network size

schemes across a 7 × 7 square mesh network (Fig. 7·17). The horizontal axis is the time to install the sample application across the network and the vertical axis is the percentage of nodes that have acquired the entire sample application. The results favor our tasking dissemination model in terms of application installation time.

Based on the simulation results we claim that compared to the traditional program dissemination scheme, our tasking approach has beneficial functional and performance characteristics. Additionally, by eliminating mote rebooting processes as required for other schemes, we can reduce the time for code installation.

7.4 Routing Protocol Evaluation Methodology

We have proposed two routing algorithm in this dissertation to enable video delivery over wireless sensor networks. There are also plenty of other algorithms as we mentioned in related works to facilitate video streaming applications over wireless sensor networks. However, different routing algorithms are developed under different assumptions and targeting for different applications. It is very difficult to compare

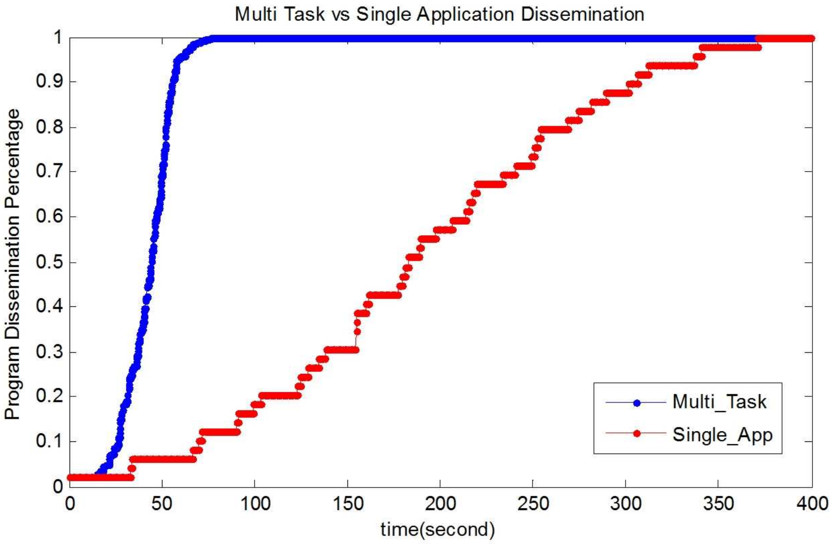


Figure 7·16: Task based network reprogramming vs. traditional OTA

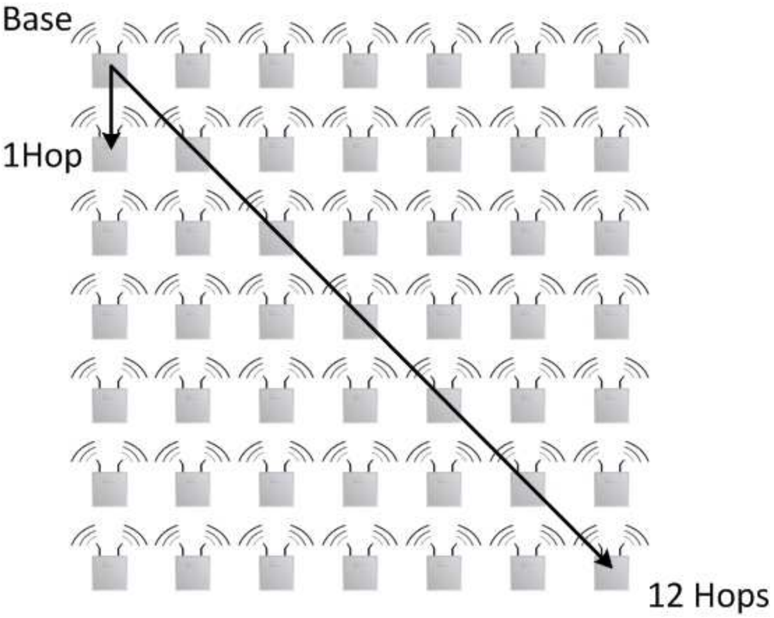


Figure 7·17: Topology of 49 nodes

their performances for the same application. In this section We develop a framework to evaluate the performance of video delivery in WVSN. We measure the performance of the video delivery based upon the network operation cost and the achieved QoS. Network operation cost is an abstract term that captures the cost of data transmission and data gathering. Designers using our performance evaluation framework can adapt this metric to local conditions. We select throughput as our QoS metric. We select the throughput as our QoS metric since many existing video delivery QoS requirements can be projected to this metric. Moreover, there are two basic QoS requirements for video delivery applications, timeliness and reliability. Timeliness requires that packets be delivered as promptly as possible. This requirement can be implemented by designing a routing algorithm with prioritized packet scheduling scheme to provide a lower bound of throughput satisfying the delay constraint of the application. The higher the video delivery throughput the more likely the application will deliver data on time. As a result, a timeliness requirement can be projected to our throughput requirement. A reliability algorithm leverages channel diversity to overcome packet loss while increasing the data redundancy by exploiting multi-path transmission. The data loss constraint in this kind of application can also be translated to throughput constraint. We can maximize the data redundancy to recover lost packets by designing a routing algorithm that achieves the maximum aggregated throughput at the destination. Such conversion enables us to measure the performance of these applications with our QoS metric.

Although achieving high QoS with less cost is an implied objective for many video delivery applications, many existing works focus on the optimization of either the cost or the QoS. In this framework, we present a balanced consideration on both sides. We find that the two design factors are indeed closely related to each other. The network operation cost unit can be expressed as “cost/second.” We rewrite this unit

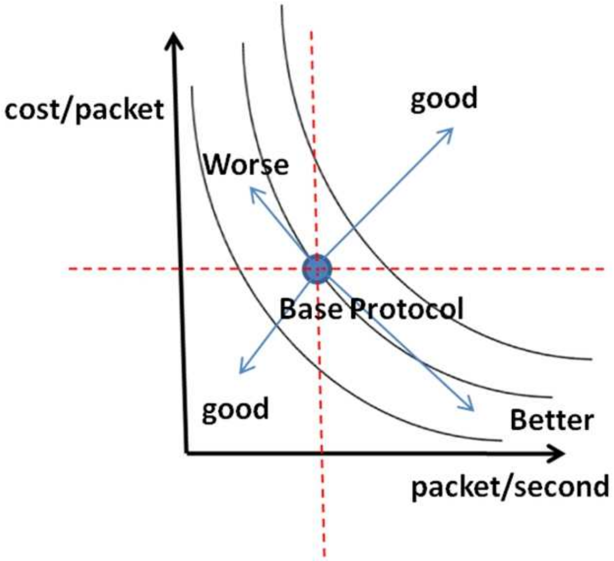


Figure 7·18: WSN routing evaluation graph

in Eq. (7.1).

$$\frac{cost}{second} = \frac{cost}{packet} \cdot \frac{packet}{second} \tag{7.1}$$

The first term on the right side is the average data delivery cost for each packet, the second term on the right side is the throughput of the entire network, which can be used as a QoS indicator. If we fix the network operation cost, we can plot the data delivery cost vs. network throughput as shown in Fig. 7·18.

The different routing algorithms will achieve different performance nodes along the black curves in Fig. 7·18. These curves are corresponding to different network operation costs. Given a location of the performance node for a base routing algorithm as shown in Fig. 7·18, we can divide the graph into four quadrants. If a new algorithm performance node lies in the upper left quadrant, we can immediately conclude that such algorithm is even worse than the base algorithm since the performance node in this quadrant indicates higher packet delivery cost with lower network throughput.

If the algorithm performance node is in the upper right or lower left quadrant, we say that the new algorithm is as good as the base algorithm since these algorithms achieve high throughput at high cost or low throughput at low cost. If the algorithm performance node is in the lower right quadrant, we can tell that this algorithm is better than the base algorithm. The new algorithm achieves higher throughput performance with lower packet delivery cost. Fig. 7·18 is called constellation graph of video data routing algorithm. This graph provides a easy way to select the best routing algorithm for a specific WVSN application. As long as you have the values of the two metrics of each routing technique and plot its constellation, a simple glance of this graph can get the technique with the best performance.

7.5 Summary

The results presented in this chapter demonstrate the advantage of our provided solutions to enable video delivery over wireless sensor networks. The distributed TDMA-based packet scheduling scheme proposed in this dissertation provides better video streaming performance compared to contention based scheduling scheme such as CTS/RTS in terms of throughput achievement. It simplifies the theoretical analysis of the throughput estimation and achieves smoother throughput curve by eliminating the delay jitter from media access contention. The simulation results demonstrate the accuracy of our throughput estimation model. In addition, the results also reflect that it is possible to combine path formation, congestion control and admission control strategies to construct data paths with throughput performance independent of the path length.

In order to deploy the routing algorithm to the network, we propose a code dissemination framework (network reprogramming framework) in Chapter 6. Compared to traditional "over-the-air" network reprogramming framework (e.g. deluge [HC04]), this new framework is faster, more flexible. The performance evaluation results indicate that the speed of the application dissemination of our network reprogramming framework is proportional to the size of the network. The speed of the application deployment of our framework is 4 to 5 times faster than traditional “over-the-air” framework.

A routing algorithm evaluation tool is also illustrated in this chapter. We present a metric to quantify different routing algorithm performances under the same application requirements. In addition, “routing algorithm constellation graph” is introduced to compare the performance achieved by different routing algorithms. This evaluation tool is intended to help us identify and leverage a more appropriate routing algorithm to support data delivery over wireless sensor networks.

Chapter 8

Conclusion

8.1 Summary

In this dissertation, we consider the problem of supporting video streaming application over wireless sensor networks. Wireless sensor networks are comprised of resource constrained devices. They usually share the same channel with limited transmission speed. However, the video delivery on the network requires relatively high data rate, therefore the data sending for one video stream may overwhelm the capacity of the wireless links and concurrent transmission of multiple such video streams will intensively interfere with each other and largely downgrade the performance of the video streaming application. Our focus is to develop a dynamic video data routing scheme for P2P video delivery with guaranteed throughput performance in wireless sensor networks.

In order to achieve this objective, we first propose a data path throughput estimation model for a single stream application, where and admission control strategy is applied to ensure only one video stream will be admitted along a data path instantiated between source and destination. In this model we group sensor nodes along a data path into small cascaded clusters and apply queuing theory to analyze the output process of the entire data path. We develop a distributed TDMA-based packet scheduling scheme to eliminate the media access contention inside each cluster and simplify the mathematical analysis. This path throughput estimation model is employed by path formation algorithms to construct data paths from source to destination to yield throughput-aware video delivery. Simulation results indicate that our model is accurate and our proposed TDMA-based packet scheduling scheme is preferred for video streaming applications. This model can also be applied to wireless sensor network applications where video streams are isolated from each other.

Extending the throughput estimation model for singleton video streaming applications, we measure the negative impact of the adjacent disjoint data path transmission in terms of the end-to-end data path throughput degradation. We apply an extended admission control strategy from single path transmission scenario to setup more comprehensive interference-tolerant throughput estimation model that associate the end-to-end data path throughput performance with the positioning of adjacent data paths, and the transmission rate based on a Poisson packet arrival from the video sources.

Based on the throughput estimation models, we present two routing algorithms for video delivery over wireless sensor networks. The first benchmark routing algorithm isolates each data path to maximize the throughput performance per steam via physically positioning each path far away from each other to eliminate path coupling interference. The second interference-tolerant data routing algorithm allows at most two video sources to share a common section of a data path as long as the throughput estimation satisfies the throughput requirements of the application. This strategy intends to allow two data streams to share an existing path for the data delivery. It is expected to lower the data delivery cost by leveraging existing data paths with affordable throughput performance sacrifices of each stream. In addition, we explore the option of deploying mobile base stations to elevate the performance of the video streaming over WSNs. A new mobile base station deployment scheme is suggested to balance the path formation delay and video delivery cost.

Finally we present an efficient and flexible network programming framework to install the application code across the network. Following the philosophy of application decomposition, this task-based network reprogramming framework is more adaptive to WSN application evolution and environment change.

8.2 Contributions

1. **Throughput Estimation Model for Isolated Data Path**

   A new mathematical throughput estimation model is proposed for single path data transmission scenario in this dissertation. By clustering sensor nodes along a data path into small systems, we decoupled the dependency of data transmission between adjacent sensor nodes. We transform the small systems to well known queuing models and apply the queuing theory to accurately estimate the throughput performance of the entire data path. In addition with the aid of TDMA-based packet scheduling scheme on each sensor node, two important theorems are derived as well to simplify our throughput estimation analysis and obtain the accurate estimation.

2. **Throughput Estimation Model for adjacent video streams**

   We present a throughput estimation model for adjacent video streams to quantify the negative impact of the inter-stream interference. This model associates the resulting data path throughput performance to more comprehensive factors such as the data rate of the adjacent data path, the relative locations of each path, and etc. This model applies approximations to simplify the mathematical analysis. The simulation results demonstrate that this model is accurate and the approximation is reasonable. This model can be further incorporated with our path formation algorithm to construct throughout-aware multi-hop data paths.

3. **Benchmark Data Routing Algorithm**

   To guarantee the video streaming performance for each data path, based on the throughput estimation model for isolated data path, we propose a benchmark data routing algorithm to deliver video data over WSNs. This algorithm form video data paths dynamically with a greedy path formation algorithm and a path search scheme. By applying simple admission control strategy to disable the sensor nodes adjacent to a data path under construction, this algorithm separates each data path from each other and ensures the throughput performance of the data path.

4. **Interference Tolerant Data Routing Protocol**

   Extending the benchmark routing algorithm, we present an interference-tolerant data routing algorithm to allow two video streams to share an existing data path. It is built based upon the throughput estimation model for adjacent video streams. It is expected to lower the data delivery cost by leveraging existing data paths with affordable throughput sacrifices of individual streams.

5. **Simple Admission Control Strategy**

   We apply an simple admission control strategy for our video data delivery applications to allocate sufficient bandwidth resource for end-to-end data flows and provide QoS guarantees. The admission control strategy enables the data flows to be delivered incrementally without impact on the existing data flows' delivery performance and the simulation results indicate that introduction of this admission control strategy improves the quality and predictability of the data delivery performance along a route. Our admission control algorithms also employ an off-line throughput computation based on queuing models to minimize the resource estimation overhead.

6. **Mobile Base Station Deployment Strategy**

   In this dissertation, we explore the option of deploying mobile base stations to improve the video streaming performance. We measure the mobile base station deployment cost to the distance between the mobile base station and the primary base station. We propose a mobile base station deployment strategy to facilitate cost efficient data delivery from the source node to the primary base station.

7. **Routing Algorithm Evaluation Tool**

   To evaluate the performance of different routing algorithms, we quantify the routing algorithm performance with a specific metric, average data delivery cost per unit time. Based upon a proper decomposition of this metric we develop the routing algorithm constellation graph as a tool to evaluate different routing algorithms for the same application requirements. The routing algorithm constellation graph presents a clear way of selecting the most appropriate routing algorithm for a specific WSN application.

8. **Efficient Network Reprogramming Framework**

   Another important contribution in this dissertation is the development of an efficient and flexible WSN application code dissemination framework. Due to this framework, network updating becomes an easier job with the help of proper application decomposition and individual updating of the task module on each mote; run-time updating is achieved and new task installation can be executed in parallel with other running applications; network programming and reprogramming are simplified by using Java and its standard APIs.

8.3 Future Work

The following section summarizes some of the future works that are related to the work in this dissertation. They are organized as extensions of work presented in this dissertation.

1. **Comprehensive modeling of data transmission interference**

   The interference-tolerant throughput estimation model only allows at most two disjoint data paths to interfere to each other. The accuracy of the estimation is largely based on the data packet arrival approximation and only three simple data path position scenarios are evaluated in this dissertation. More comprehensive throughput estimation model is on demand to deal with data transmission interferences involving any number of data paths and arbitrary corresponding locations. Comprehensive interference modeling can be leveraged to develop more general throughput-aware data routing algorithms.

2. **Testbed implementation**

   It is necessary to develop a real testbed to make our works become practical. This testbed can serve as a flexible platform to facilitate routing algorithm replacement. Researchers can leverage this platform to customize their routing algorithm for various WSN applications.

3. **Comparison with real video stream traces**

   The analytical model and simulation is only appropriate for certain arrival stochastic processes and some video data arrival simplifications. To compare the real video throughput performance with the analytical results can further provide insights for appropriate mathematical modeling improvements to the current model. Video streaming applications always related to different coding schemes, many existing routing protocols are targeting for specific coding strategy. A comprehensive video streaming study with different coding schemes is on demand to test the generality of the model.

4. **Evaluation of the state-of-art routing algorithms for video streaming applications over WSN**

   With the tool of "routing algorithm constellation graph," one can easily compare different routing algorithms for a specific WSN application. Evaluation of the state-of-art routing algorithms under different requirements with the constellation graph will enable us to build a useful table to facilitate the fast selection of the most appropriate routing algorithm for a given application.

5. **Extending work to support multi-channel environments**

   Although our model is based on a single channel communication environment, we can extend this work to support multi-channel environments. We can build multiple network topologies corresponding to different network communication channels and apply our model on each topology separately.

References

[ABJ+06] D. P. Agrawal, R. Biswas, N. Jain, A. Mukherjee, S. Sekhar, and A. Gupta. *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad Hoc Networks, and Peer-to-Peer Networks*, chapter Sensor Systems: State of the Art and Future Challenges. Auerbach, 2006.

[ACVS02] G. Ahn, A. T. Campbell, A. Veres, and L. Sun. SWAN: Service Differentiation in Stateless Wireless Ad Hoc Networks. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '02)*, pages 457–466, 2002.

[AMC07] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury. A Survey on Wireless Multimedia Sensor Networks. *Computer Networks*, 51:921–960, 2007.

[ASSC02] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless Sensor Networks: A Survey. *Computer Networks*, 38:393–422, 2002.

[AY03] K. Akkaya and M. Younis. An Energy-aware QoS Routing Protocol for Wireless Sensor Networks. In *Proceedings of the IEEE Workshop on Mobile and Wireless Networks (MWN '03)*, pages 710–715, 2003.

[BAB+07] C. R. Baker, K. Armijo, S. Belka, M. Benhabib, A. D. Minassians, V. Bhargava, N. Burkhart, G. Dervisoglu, L. Gutnik, M. Brenthaick, C. Ho, M. Koplow, J. Mangold, S. Robinson, M. Schwartz, C. Sims, H. Stoffregen, A. Waterbury, E. S. Lel, T. Pering, and P. K. Wright. Wireless Sensor Networks for Home Health Care. In *Proceedings of International Conference on Advanced Information Networking and Applications Workshops*, pages 832–837, 2007.

[Ber92] D. Bertsekas. *Data Networks*. Prentice Hall, 1992.

[BJB04] B.Hull, K. Jamieson, and H. Balakrishnan. Mitigating congestion in wireless sensor networks. In *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys '04)*, pages 134–147, 2004.

[CCL04] J. Chen, S. G. Chan, and V. O. K. Li. Multipath Routing for Video Delivery over Bandwidth-limited Networks. *IEEE Journal on Selected Areas in Communications*, 22(10):1920–1932, 2004.

[Chi] Chipcon. 2.4 GHz IEEE 802.15.4 / ZigBee-Ready RF Transceiver . Available at `http://focus.ti.com/docs/prod/folders/print/cc2420.html`.

[CLMY07] M. Chen, V. C. M. Leung, S. Mao, and Y. Yuan. Directional Geographical Routing for Real-time Video Communications in Wireless Sensor Networks. *Computer Communications*, 30(17):3368–3383, 2007.

[CNRS98] E. Crawley, R. Nair, B. Rajagopalan, and H. Sandick. A Framework for QoS-based Routing in the Internet. Technical report, RFC 2386, 1998.

[CPCT06] M. Cosma, D. Pescaru, B. Ciubotaru, and D. Todinca. Routing and Topology Extraction Protocol for a Wireless Sensor Network Using Video Information. In *3rd Romanian-Hungarian Joint Symposium on Applied Computational Intelligence*, 2006.

[Cro05] Crossbow. Crossbow Technology Releases TelosB Mote Platform. May 2005.

[CRW04] A. Carzaniga, M. J. Rutherford, and A. L. Wolf. A Routing Scheme for Content-based Networking. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '04)*, Hong Kong, 2004.

[CRZ04] M. Chu, J. Reich, and F. Zhao. Distributed Attention in Large Scale Video Sensor Networks. *IEEE Seminar Digests*, pages 61–65, 2004.

[CSRL01] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition, 2001.

[CT00] J. Chang and R. Tassiulas. Energy Conserving Routing in Wireless Ad-hoc Networks. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '00)*, pages 22–31, March 2000.

[CTG97] T. Chen, J. T. Tsai, and M. Gerla. QoS Routing Performance in Multihop, Multimedia, Wireless Networks. In *Proceedings of IEEE International Conference on Universal Personal Communications (ICUPC '97)*, pages 557–561, 1997.

[CW03] A. Carzaniga and A. L. Wolf. Forwarding in a Content-based Network. In *Proceedings of Annual Conference of the ACM Special Interest Group on Data Communication (SIGCOMM '03)*, pages 163–174, 2003.

[CY07] Y. Chai and D. Ye. The Design and Implementation of a Scalable Wireless Video Streaming System Adopting TCP Transmission Mode. In *Proceedings of the 7th IEEE International Conference on Computer and Information Technology*, pages 534–538, Fukushima, Japan, 2007.

[DBHR05] E. O. Douglas, D. M. Blough, B. S. Heck, and G. F. Riley. Distributed Global Identification for Sensor Networks. In *Proceedings of IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS '05)*, 2005.

[DM09] P. Djukic and P. Mohapatra. Soft-TDMAC: A Software TDMA-Based MAC over Commodity 802.11 Hardware. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '09)*, pages 1836–1844, 2009.

[DRA] I. Downes, L. B. Rad, and H. Aghajan. Development of a Mote for Wireless Image Sensor Networks. In *Proceedings of Cognitive Systems with Interactive Sensors (COGIS '06)*, Paris, France.

[DRM01] C. Dovrolis, P. Ramanathan, and D. Moore. What Do Packet Dispersion Techniques Measure? In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '01)*, pages 905–914, 2001.

[EFK06] Jakob Eriksson, Michalis Faloutsos, and Srikanth Krishnamurthy. *Routing Scalability in MANETs*. Auerbach, 2006.

[Eki11] E. Ekici. Wireless Multimedia Sensor Networks. http://www2.ece.ohio-state.edu/`~ekici`/res_wmsn.html, March 2011.

[Erg04] S. C. Ergen. ZigBee/IEEE 802.15.4 Summary. `http://pages.cs.wisc.edu/~suman/courses/838/papers/zigbee.pdf`, September 2004.

[FGG04] Q. Fang, J. Gao, and L. Guibas. Locating and Bypassing Routing Holes in Sensor Networks. In *Proceedings of the 23rd Conference of the IEEE Communications Society*, pages 2458–2468, 2004.

[FKFB05] W. Feng, E. C. Kaiser, W. Feng, and M. L. Baillif. Panoptes: Scalable Low-power Video Sensor Networking Technologies. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMCCAP '05)*, 1(2):151–167, 2005.

[GDP04] S. Gandham, M. Dawande, and R. Prakash. An Integral Flow-based Energy-efficient Routing Algorithm for Wireless Sensor Networks. In *Proceedings of IEEE Wireless Communications and Networking Conference*, pages 2341–2346, 2004.

[GGW04] A. Ganz, Z. Ganz, and K. Wongthavarawat. *Multimedia Wireless Networks: Technologies, Standards, and QoS*. Prentice Hall, 2004.

[GHPB02] Q. Gao, D. J. Holding, Y. Peng, and K.J. Blow. Energy Efficiency Design Challenge in Sensor Networks. In *London Communications Symposium*, 2002.

[GKS03] S. Ganeriwal, R. Kumar, and M. B. Srivastava. Timing-sync Protocol for Sensor Networks. In *ACM Conference on Embedded Networked Sensor Systems (SenSys '03)*, pages 138–149. ACM Press, 2003.

[GWMC] D. Gupta, D. Wu, P. Mohapatra, and C. N. Chuah. Experimental Comparison of Bandwidth Estimation Tools for Wireless Mesh Networks. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '09)*, pages 2891–2895.

[HC04] Jonathan W. Hui and David Culler. The dynamic behavior of a data dissemination protocol for network programming at scale. In *In Proceedings of the 2nd international*, pages 81–94. ACM Press, 2004.

[HCB00] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient Communication Protocol for Wireless Microsensor Networks. In *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences (HICSS '00)*, Jan 2000.

[HSOH03] R. Holman, J. Stanley, and T. Ozkan-Haller. Applying Video Sensor Networks to Nearshore Environment Monitoring. *IEEE Pervasive Computing*, 2:14–21, 2003.

[HST08] Andrew Hagedorn, David Starobinski, and Ari Trachtenberg. Rateless deluge: Over-the-air programming of wireless sensor networks using random linear codes. In *Proceedings of the 7th international conference on Information processing in sensor networks (IPSN)*, Saint Louis, MO, USA, Apr. 2008.

[HTLG97] Y. Hsu, T. Tsai, Y. Lin, and M. Gerla. Bandwidth Routing in Multihop Packet Radio Environment. In *Proceedings of the 3rd International Mobile Computing Workshop*, 1997.

[IGE00] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking (MOBICOM '00)*, pages 56–67. ACM, 2000.

[Int] Intel. Stargate Platform. `http://platformx.sourceforge.net/Links/resource.html`.

[Jai92] R. Jain. Myths about Congestion Management in High-Speed Networks. *Internetworking: Research and Experience*, 3:101–113, 1992.

[JC04] J. Jeong and D. E. Culler. Incremental Network Programming for Wireless Sensors. In *Proceedings of the First IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON '04)*, pages 25–33, 2004.

[JD02] M. Jain and C Dovrolis. Pathload: A Measurement Tool for End-to-End Available Bandwidth. In *Proceedings of Passive and Active Measurements Workshop*, 2002.

[JPC05] X. Jiang, J. Polastre, and D. Culler. Perpetual Environmentally Powered Sensor Networks. In *Proceedings of IEEE Workshop on Sensor Platform, Tools and Design Methods for Networked Embedded Systems*, UCLA, Los Angeles, California, USA, April 2005.

[JW10] L. Jiang and J. Walrand. A Distributed CSMA Algorithm for Throughput and Utility Maximization in Wireless Networks. *IEEE/ACM Transactions on Networking (TON '10)*, 18:960–972, June 2010.

[JWWG07] W. Jia, T. Wang, G. Wang, and M. Guo. Hole Avoiding in Advance Routing in Wireless Sensor Networks. In *Proceedings of the IEEE Wireless Communication and Networking Conference (WCNC '07)*, 2007.

[KAK03] H. Kim, T. F. Abdelzaher, and W. H. Kwon. Minimum-energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pages 193–204. ACM Press, 2003.

[KGL01] M. Kazantzidis, M. Gerla, and S. Lee. Permissible Throughput Network Feedback for Adaptive Multimedia in AODV MANETs. In *IEEE International Conference of Communications (ICC '01)*, pages 1352–1356, 2001.

[KGSL05] P. Kulkarni, D. Ganesan, P. J. Shenoy, and Q. Lu. SensEye: A Multi-tier Camera Sensor Network. In *ACM Multimedia*, pages 229–238, 2005.

[KK00] B. Karp and H. T. Kung. GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In *The 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pages 243–254, 2000.

[KS07] A. Koliousis and J. Sventek. Proactive vs Reactive Routing for Wireless Sensor Networks. Technical report, University of Glasgow, Department of Computing Science, 2007.

[LAZC00] S. Lee, G. Ahn, X. Zhang, and A. T. Campbell. INSIGNIA: An IP-based Quality of Service Framework for Mobile Ad Hoc Networks. *Journal of Parallel and Distributed Computing*, 60:374–406, 2000.

[LC02] P. Levis and D. E. Culler. Maté: A Tiny Virtual Machine for Sensor Networks. In *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS '02)*, pages 85–95, 2002.

[LD06] Y. Liu and S. Das. Information-intensive Wireless Sensor Networks: Potential and Challenges. *IEEE Communications Magazine*, 44(11):142–147, 2006.

[LG09] T. Little and A. Ganick. Remote Wildlife Observation with IP Cameras: The Great Point Lighthouse Camera, September 2009. MCL Technical Report No. 09-01-2009, Department of ECE, Boston University.

[LGZ07] P. Li, Y. Gu, and B. Zhao. A Global-energy-balancing Real-time Routing in Wireless Sensor Networks. *Asia-Pacific Conference on Services Computing. 2006 IEEE*, 0:89–93, 2007.

[LIK07] T. Little, P. Ishwar, and J. Konrad. A Wireless Video Sensor Network for Autonomous Coastal Sensing. In *Proceedings of Conference on Coastal Environmental Sensing Networks*, Boston, MA, USA, April 2007.

[LNLL08] S. Li, R. Neelisetti, C. Liu, and A. Lim. Delay-constrained High Throughput Protocol for Multi-path Transmission over Wireless Multimedia Sensor Networks. In *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM '08)*, pages 1–8, 2008.

[LPC$^+$09] C. Luo, W. Pu, C. W. Chen, J. Sun, and F. Wu. Forepressure Transmission Control for Wireless Video Sensor Networks. In *Proceedings of the 6th Annual IEEE communications society conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON '09)*, pages 46–54. IEEE Press, 2009.

[LPCS04] P. Levis, N. Patel, D. E. Culler, and S. Shenker. Trickle: A Self-Regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks. In *Proceedings of the First USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI '04)*, pages 15–28, 2004.

[MCdS$^+$05] D. F. Macedo, L. H. A. Correia, A. L. dos Santos, A. A. F. Loureiro, and J. M. S. Nogueira. A Pro-active Routing Protocol for Continuous

Data Dissemination in Wireless Sensor Networks. *Proceedings of thProceeding ISCC '05 Proceedings of the 10th IEEE Symposium on Computers and Communicationse 10th IEEE Symposium on Computers and Communications (ISCC '05)*, pages 361–366, 2005.

[MFHH05] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. TinyDB: an Acquisitional Query Processing System for Sensor Networks. *ACM Transactions on Database Systems (TODS '05)*, 30(1):122–173, 2005.

[MK07] I. Mas and G. Karlsson. Probe Based Admission Control for a Differentiated Services Internet. *Computer Networks*, 51:3902–3918, September 2007.

[Moo05] G. Moore. Excerpts from a Conversation with Gordon Moore: Moore's Law, November 2005. Video Transcript.

[MPOM06] C. B. Margi, V. Petkov, K. Obraczka, and R. Manduchi. Characterizing Energy Consumption in a Visual Sensor Network Testbed. In *Proceedings of IEEE/Create-Net International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom '06)*, Barcelona, Spain, March 2006.

[MPSC02] A. Mainwaring, J. Polastre, R. Szewczyk, and D. Culler. Wireless Sensor Networks for Habitat Monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA '02)*, September 2002.

[NKGI06] V. Navda, A. Kashyap, S. Ganguly, and R. Izmailov. Real-time Video Stream Aggregation in Wireless Mesh Network. In *Proceedings of 17th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '06)*, pages 1–7, 2006.

[NS03] J. Newsome and D. Song. GEM: Graph EMbedding for Routing and Data-Centric Storage in Sensor Networks without Geographic Information. In *Proceedings of the 1st international conference on Embedded networked sensor systems (SenSys '03)*, pages 76–88. ACM Press, 2003.

[OL10] I. Orhan and T. Lindh. Measurement-based Admission Control in Wireless Sensor Networks. In *Proceedings of the 2010 Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM '10)*, pages 447–452, Washington, DC, USA, 2010. IEEE Computer Society.

[PHST00] M. R. Pearlman, Z. J. Haas, P. Sholander, and S. S. Tabrizi. On the impact of alternate path routing for load balancing in mobile ad hoc networks. In *Proceedings of the 1st ACM International Symposium on Mobile Ad hoc Networking and Computing (MOBIHOC '00)*, pages 3–10, 2000.

[PTDK08] I. Politis, M. Tsagkaropoulos, T. Dagiuklas, and S. Kotsopoulos. Power Efficient Video Multipath Transmission over Wireless Multimedia Sensor Networks. *Mobile Networks and Applications*, 13(3-4):274–284, 2008.

[PTP$^+$07] I. Politis, M. Tsagkaropoulos, T. Pliakas, T. Dagiuklas, and S. Kotsopoulos. Intelligence Packet Scheduling for Optimized Video Transmission over Wireless Networks. In *Proceedings of the 3rd international conference on Mobile multimedia communications (MobiMedia '07)*, page 38, Nafpaktos, Greece, 2007.

[RAGN07] A. Rowe, Goode A, D. Goel, and I. Nourbakhsh. CMUcam3: An Open Programmable Embedded Vision Sensor, 2007. Carnegie Mellon Robotics Institute Technical Report, RI-TR-07-13.

[RBI$^+$05] M. Rahimi, 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. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems (SenSys '05)*, pages 192–204. ACM Press, 2005.

[RDPV] S. Rao, M. Daw, R. Prakash, and S. Venkatesan.

[RL03] N. Reijers and K. Langendoen. Efficient code distribution in wireless sensor networks. In *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications (WSNA '03)*, pages 60–67, 2003.

[Ros95] S. M. Ross. *Stochastic Processes*. Wiley, 1995.

[RRP$^+$03] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. Geographic Routing without Location Information. In *Proceedings of the 9th Annual International Conference on Mobile Computing and Networking (MobiCom '03)*, pages 96–108. ACM Press, 2003.

[SCC$^+$06] D. Simon, C. Cifuentes, D. Cleal, J. Daniels, and D. White. Java$^{\mathrm{TM}}$ on the Bare Metal of Wireless Sensor Devices: the Squawk Java Virtual Machine. In *Proceedings of the 2nd international conference on Virtual execution environments (VEE '06)*, pages 78–88, 2006.

[SCK06] J. Shin, M. Chin, and C. Kim. Optimal Transmission Range for Topology Management in Wireless Sensor Networks. In *Proceedings of the International Conference on Information Networking (ICOIN) 2006*, pages 16–19, Japan, 2006.

[SCY$^+$05] T. Sun, L. Chen, G. Yang, M. Y. Sanadidi, and M. Gerla. SenProbe: Path Capacity Estimation in Wireless Sensor Networks. In *Proceedings* *of Third International Workshop on Measurement, Modeling, and Performance Analysis of Wireless Sensor Networks (SenMetrics '05)*, 2005.

[SHE03] Thanos Stathopoulos, John Heidemann, and Deborah Estrin. A remote code update mechanism for wireless sensor networks. Technical report, 2003.

[SPH04] P. Samar, M. R. Pearlman, and Z. J. Haas. Independent Zone Routing: An Adaptive Hybrid Routing Framework for Ad Hoc Wireless Networks. *IEEE/ACM Transactions on Networking (TON '04)*, 12:595–608, 2004.

[SZY$^+$10] L. Shu, Y. Zhang, L. T. Yang, Y. Wang, M. Hauswirth, and N. Xiong. TPGF: Geographic Routing in Wireless Multimedia Sensor Networks. In *Telecommunication Systems*, pages 79–95, 2010.

[SZZ$^+$08] L. Shu, Y. Zhang, Z. Zhou, M. Hauswirth, Z. Yu, and G. Hynes. Transmitting and Gathering Streaming Data in Wireless Multimedia Sensor Networks Within Expected Network Lifetime. *Mobile Network Application*, 13(3-4):306–322, 2008.

[TE92] L. Tassiulas and A. Ephremides. Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks. *Automatic Control, IEEE Transactions on*, 37(12):1936–1948, 1992.

[TLC$^+$06] T. Teixeira, D. Lymberopoulos, E. Culurciello, Y. Aloimonos, and A. Savvides. A Lightweight Camera Sensor Network Operating on Symbolic Information. In *The First Workshop on Distributed Smart Cameras 2006, held in conjunction with ACM SenSys 2006*, November 2006.

[VVVV05] D. Vass, Z. Vincze, R. Vida, and A. Vidacs. Energy Efficiency in Wireless Sensor Networks Using Mobile Base Station. In *Proceedings of Networked Services and Applications Engineering, Control and Management*, 2005.

[WBMP05] Z. M. Wang, S. Basagni, E. Melachrinoudis, and C. Petrioli. Exploiting Sink Mobility for Maximizing Sensor Networks Lifetime. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS '05)*, 2005.

[WEC03] C. Wan, S. B. Eisenman, and A. T. Campbell. CODA: Congestion Detection and Avoidance in Sensor Networks. In *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys '03)*, pages 266–279, 2003.

[WJHR09] A. Warrier, S. Janakiraman, S. Ha, and I. Rhee. DiffQ: Practical Differential Backlog Congestion Control for Wireless Networks. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '09)*, 2009.

[WM04] M. Welsh and G. Mainland. Programming Sensor Networks Using Abstract Regions. In *Proceedings of the 1st Conference on Symposium on Networked Systems Design and Implementation (NSDI '04)*, pages 29–42, 2004.

[WML07] J. Wang, M. Masilela, and J. C. L. Liu. Supporting Video Data in Wireless Sensor Networks. *Proceedings of the Ninth IEEE International Symposium on Multimedia (ISM '07)*, 0:310–317, 2007.

[WSCB04] K. Whitehouse, C. Sharp, D. E. Culler, and E. A. Brewer. Hood: A Neighborhood Abstraction for Sensor Networks. In *Proceedings of ACM International Conference on Mobile Systems, Applications, and Services (MobiSys '04)*, 2004.

[WSL05] C. Wang, K. Sohraby, and B. Li. SenTCP: A Hop-by-Hop Congestion Control Protocol for Wireless Sensor Networks. In *Proceedings of IEEE International Conference on Computer Communications (INFOCOM '05)*, 2005.

[YGE01] Y. Yu, R. Govindan, and D. Estrin. Geographical and Energy Aware Routing: A Recursive Data Dissemination Protocol for Wireless Sensor Networks. Technical report, UCLA, 2001. UCLA Computer Science Department Technical Report UCLA/CSD-TR-01-0023.

[YK05] Y. Yang and R. Kravets. Contention-aware Admission Control for Ad Hoc Networks. *IEEE Transactions on Mobile Computing*, 4(4):363–377, 2005.

[YLC$^+$02] F. Ye, H. Luo, J. Cheng, S. Lu, and L. Zhang. A Two-tier Data Dissemination Model for Large-scale Wireless Sensor Networks. In *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking (MobiCom '02)*, pages 148–159, 2002.

[YLC$^+$07] F. Yu, E. Lee, Y. Choi, S. Park, D. Lee, Y. Tian, and S. Kim. A Modeling for Hole Problem in Wireless Sensor Networks. In *Proceedings of the 2007 International Conference on Wireless Communications and Mobile Computing (IWCMC '07)*, pages 370–375, 2007.

[YSR09] L. Ying, S. Shakkottai, and A. Reddy. On Combining Shortest-Path and Back-Pressure Routing over Multihop Wireless Networks. In *Proceedings of*

IEEE International Conference on Computer Communications (INFOCOM '09), pages 1674–1682, 2009.

[ZCLZ04] Y. Zhao, Y. Chen, B. Li, and Q. Zhang. Hop ID: A Virtual Coordinate-based Routing for Sparse Mobile Ad Hoc Networks. In *IEEE Transactions on Mobile Computing*, 2004.

[ZHS+08] L. Zhang, M. Hauswirth, L. Shu, Z. Zhou, V. Reynolds, and G. Han. Multi-priority Multi-path Selection for Video Streaming in Wireless Multimedia Sensor Networks. In *Proceedings of the 5th International Conference on Ubiquitous Intelligence and Computing (UIC '08)*, pages 439–452, 2008.

[ZLZ+05] Y. Zhao, B. Li, Q. Zhang, Y. Chen, and W. Zhu. Efficient Hop ID Based Routing for Sparse Ad Hoc Networks. In *Proceedings of the 13TH IEEE International Conference on Network Protocols (ICNP '05)*, pages 179–190, 2005.

CURRICULUM VITAE

Song Guo

Song Guo was born in 1982 in Chengdu, China. He received the Bachelor of Engineering degree in Communication Engineering from University of Electronic Science and Technology of China, Chengdu, China in 2005 and Master of Science degree in Computer Engineering from Boston University in 2007. At Boston University, he was awarded Graduate Research Assistantship from 2006 to 2010.

For industry experience, in fall 2010 Song worked at IBM, Littleton, MA. to develop an automation testing report system for IBM Lotus Notes team. In summer 2007, he worked at Softartisans Inc, Watertown, MA. to develop Microsoft officewriter 2007. For academic experience, Song is an IEEE student member since 2006. He has authored several papers and served as reviewer for journals and conferences. He can be reached at the following address:

Song Guo
ECE Department
Boston University
8 Saint Mary's Street
Boston, MA 02215
Email: guosong@bu.edu