Challenges and Design Principles of Large Scale Tactical Network Architecture

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Andy Shih-Che Peng

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

December, 2010

Acknowledgements

Upon completion of this Ph.D. dissertation, I found myself profoundly indebted to many people who had provided tremendous support and unconditional assistance along the way. First and foremost, I would like to express my deepest appreciation to both of my academic advisers, Professor David J. Lilja and Professor Tian He, for their guidance, encouragement, support, and patience throughout years of my graduate school career. I would like to express my gratitude to Professor Lilja for providing assistance ranging from fulfilling a simple administrative request to providing in-depth research advice throughout years in graduate school. My gratitude also goes to Professor He for accepting me into his research group and his willingness to collaborate independent research work with me. His continuous research feedback and insightful criticism had helped me to excel academically. I would like to thank both academic advisors for providing continuous research motivation and their flexibility to accommodate my full-time working schedule when meetings were requested. I truly enjoyed countless meetings with both advisors for fruitful research discussions as well as sharing precious life experiences. Both academic guidance and real life advice provided by my academic advisors truly inspired me as well as nourished my professional career.

Next, I would like to thank Professor Nihar Jindal and Professor Zhi-Li Zhang, for their willingness to serve on my Ph.D. thesis committee and for providing valuable research comments when requested.

Furthermore, I would like to acknowledge Lockheed Martin Corporation for

providing education assistance in supporting my Ph.D. program as well as accommodating my academic career with a flexible working schedule. Special thanks to my colleagues, Dr. Dennis M. Moen and Joseph A. (Tony) Spinks, who provided invaluable support throughout the years. Denny has been a competent research mentor for providing countless technical discussions to improve the quality of the research and for tirelessly encouraging me to complete this Ph.D. dissertation. Tony has been a great colleague and a true friend in providing much needed friendship and moral support throughout the years.

Finally, and perhaps most importantly, my love and gratitude toward each member of my family. To my father, Dr. Shih-Sen Peng, to my mother, Mei-Chu Lan, to my brother, Shih-Yen Peng, and last, but not least, my dear wife, Ting-Wen Chen, Ph.D.-to-be, for their continuous unconditional love, patience, and understanding throughout this challenging and memorable journey of my life.

Dedication

To

My Family

and

My Professors

with

Love and Gratitude.

Abstract

Modern tactical communications systems are moving towards Internet-style system architectures to support information sharing for improving overall mission effectiveness. Development of such large scale communications systems presents various system design challenges. Research works discussed in this thesis are motivated by the technical challenges commonly encountered during the development of several large scale communications systems and proposes system-level design principles in overcoming these technical challenges. The first part of this thesis addresses system-level quality of service issues by modeling and simulation of an afloat wide area network system architecture. This simulation study investigates the system performance of real-time applications and provides quality of service design recommendations. The second part of this thesis proposes a consolidated network architecture for designing an afloat local area network system. A simulated prototype system is developed to investigate the system performance trade-off in the proposed consolidated network architecture. The third part of this thesis proposes an automatic dynamic resource management system architecture to efficiently manage shared computing resources in resource-constrained network environments without any human operator intervention. Test results in this experimental study demonstrates improved network performance when a communications system employs the automatic dynamic resource management software. Finally, the last part of this thesis proposes a reliable data aggregation and dissemination framework for tactical communications systems operating in disruptive networking environments with intermittent network connectivity. A prototype system is developed and implemented to demonstrate that the proposed framework can ensure reliable data delivery which is beneficial to the current and future development of tactical communications system architectures.

This thesis makes several significant research contributions in designing a large

scale communications system. First of all, the thesis suggests a simulation methodology for developing simulation models to study the performance of a large scale communications system and makes recommendations on system-level quality of service design. Secondly, the thesis reduces the complexity of future communications system design by proposing a consolidated system architecture. Thirdly, an automatic dynamic resource management software prototype is developed to alleviate resource contention issues commonly found in the tactical networking environments. Fourthly, a reliable data aggregation and dissemination framework is proposed and its function is demonstrated. The proposed framework can accurately infer meaningful messages from a large sensor data set and can reliably deliver the messages to the appropriate network destinations. Finally, the thesis organizes all of these relevant system-level design experiences and recommends system design principles for developing future large scale communications systems.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

The concept of the Global Information Grid (GIG) was originally developed by the U.S. Department of Defense (DoD) to support ever-increasing information demands and collaborative decision making [1]. The GIG is the foundation for enabling current and future network-centric (or net-centric) operations and warfare [2] [3]. Network-centric operations rely on information sharing through heterogeneous computing resources connected via the GIG backbone network. The GIG is a globally connected network, which consists of information capabilities including Information Technology (IT) infrastructures, all associated personnel and services, and processes that support all DoD organizations in accomplishing their tasks and missions [4]. As depicted in Figure 1.1 [5], the heterogeneous GIG includes all communications, computing systems, policies, and other resources separately deployed at different geographical locations which form a global network infrastructure providing all necessary data and services to achieve Information Superiority [2]. Due to the demand for rapidly making collaborative decisions, these systems comprise hardware equipments and software tools to deliver data on-demand as well as visually presenting the data to end-users. The vision of GIG connotes a fundamental shift in system design paradigm from developing

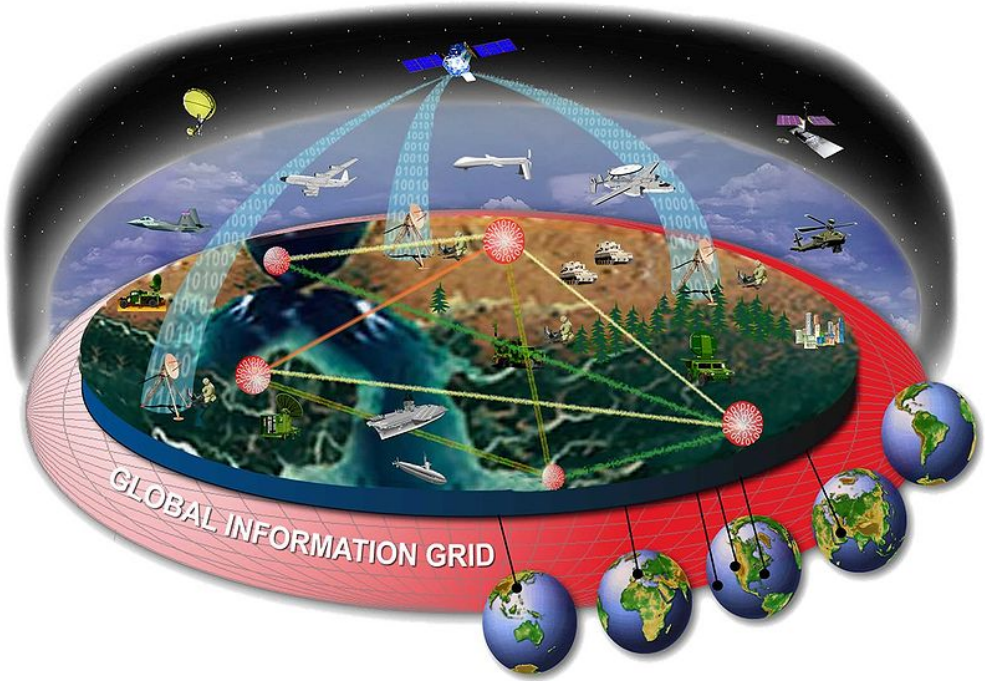


Figure 1.1: Concept of the Global Information Grid

many stovepipe systems serving stand-alone functions to designing highly interconnected systems which operate cohesively to satisfy mission objectives. In order to accommodate such a shift in system design paradigm, modern military communications systems are moving toward an Internet-style system architecture for improved mission effectiveness.

Current and future military operations will increasingly capitalize on the advantages and innovations of advanced information technology [3]. In supporting the network-centric operations, modern military communications systems must be architecturally designed to be interoperable with each other by exploiting advanced ad hoc networking techniques as well as reaching back to higher-level systems through either land-based or radio frequency (RF) based network connections within the GIG backbone network. A military communications system is typically a large scale system of systems in terms of its design complexity, required computing resources, and the network size. A military communications

system is a type of tactical communications system which is commonly defined as a communications system that is "*used within*" or "*in support of*" tactical forces to provide secure communications such as data, video, and voice, and to facilitate command and control (C2) functions [6]. Such specialized communications system is designed to operate under rapidly changing tactical conditions and harsh physical environments. A tactical communications system commonly consists of radio frequency equipments, computing resources, networking devices, and software applications highly integrated within a common network infrastructure to support diverse net-centric operations. Coupling with a variety of radio frequency links, the system provides network connectivity with others when deployed in all types of mobile units such as airborne, shipboard, undersea, and terrestrial.

The network architecture of the tactical communications system is generally structured hierarchically such that it is either directly connected with other peering systems or indirectly connected to a higher-level system through the GIG for providing information exchange capabilities. The system is specifically designed so that it can be accessed and managed by both local authorities (*e.g.* local tactical user communities) as well as higher level authorities (*e.g.* network operations centers). Due to the sensitive nature of the data in the network, the security of information is generally maintained throughout the tactical communications system.

A tactical communications system is typically deployed across diverse types of mobile platforms such as afloat, airborne, undersea, and terrestrial units as well as at the network operations centers for supporting a variety of tactical missions. Development of a large scale communications system is a technically challenging task which generally involves many system-level design options and considerations. The complexity of the system design is mostly driven by the user's requirements defined for specific operational environments.

1.2 Motivations

DoD Open Architecture (OA) system design initiatives enable current and future military communications systems to adopt "*open*" (*i.e.* "*non-proprietary*") and publicly available standards from mainstream commercial-off-the-shelf (COTS) computing technologies in its system architecture [7]. The open architecture initiatives also align military technology with commercial products to provide more opportunities for competition and innovation [7]. An open architecture system provides several key benefits such as reduced system development and life cycle cost, rapid system upgrades and technology insertion, improved interoperability, and optimized system performance. Since COTS components are employed in building a communications system, a rapid product development cycle can be achieved which translates into faster time to market (TTM) and a reduction in the overall system maintenance cost. This also reduces the engineering time to develop a new component for rapid system upgrades and reuses software tools. Integration of publicly available open standards improves the system's interoperability, which increases the opportunities to further optimize the system's performance.

In order to apply open architecture principles in designing a more efficient military communications system, selection of COTS computing technologies to meet specific system requirements and targeted cost objectives, integration of multiple open standards, management of COTS computing resources, system performance evaluations, and other system-level design issues become a much needed research area. This thesis is motivated by this challenging and interesting research area in designing various large scale communications systems using COTS computing technologies.

The first part of this thesis is motivated by the modernization of a communications system which provides wide area network (WAN) interface for afloat platforms. The second part of this thesis proposes a consolidated network architecture hosting local area network (LAN) services which is motivated by a fundamental design shift from a stovepipe system to an integrated system. The

third part of this thesis is motivated by the system requirements to address the resource management problem at a tactical edge network. Finally, the last part of this thesis is motivated by the need to reliably deliver data in the tactical environment by integrating several network protocols for data aggregation and dissemination.

1.3 Thesis Contributions

The goal of this thesis is to focus on design issues, principles, and solutions for system of systems such as a tactical communications system using COTS products as the fundamental building blocks. The research contributions of this thesis listed by each chapter are as follows:

- *Chapter 2: System-Level Quality of Service Design Considerations* [8]
  - This thesis develops a detailed OPNET simulation model to investigate the performance characteristics of a tactical network architecture which provides WAN services.
  - This thesis proposes a discrete event simulation methodology to model each system component of the WAN services.
  - This thesis develops relevant WAN test scenarios for investigating QoS performance trade-offs.
  - This thesis presents simulation results which suggest that careful QoS planning is required for delivering data across a high-latency satellite communications (SATCOM) link.
  - This thesis presents simulation results which suggest that QoS can be effectively implemented at the congested link to achieve sufficient system performance while minimizing system integration cost.
- *Chapter 3: Toward Consolidated Network Architecture* [9]

- This thesis proposes a system approach to consolidate a network architecture which provides LAN services. In this proposed consolidated network architecture, multiple security network domains are joined into a common network infrastructure to reduce the overall system components which reduces system integration efforts and translates into system cost savings.
- This thesis leverages previous system design experiences from Chapter 2 and separately develops another OPNET simulation model for investigating the performance characteristics of the proposed consolidated network architecture.
- This thesis presents simulation results that characterize fail-over and traffic scalability scenarios as well as the application-level performance in the tactical edge LAN environment.

- *Chapter 4: Automatic Dynamic Resource Management System* [10] [11]
  - This thesis develops a large scale software architecture known as the Automatic Dynamic Resource Management (AutoDRM) system. AutoDRM system architecture is an attempt to efficiently manage computing and networking resources at a tactical edge network.
  - This thesis develops and implements a large scale prototype test bed mimicking a realistic tactical network environment in order to validate the effectiveness of AutoDRM system. The test bed includes computing resources (*i.e.* laptops and workstations), networking resources (*i.e.* switches and routers), SATCOM simulator, and OPNET System-In-The-Loop (SITL) scenario for evaluating the AutoDRM system and collecting experimental results.
  - This thesis presents experimental results that demonstrate improved network performance when AutoDRM system is deployed at a tactical edge network. Test results further re-confirm the need for an advanced resource management function at the tactical edge network.

- *Chapter 5: Reliable Data Aggregation and Dissemination Framework* [12]

- This thesis proposes a reliable data aggregation and dissemination framework in the context of tactical networks. The framework takes a hybrid approach of combining disruption tolerant networking advantages and an adaptive sensor data aggregation method to ensure reliable data delivery.
- This thesis develops and implements an integrated prototype system architecture to demonstrate the system capabilities of the proposed data aggregation and dissemination framework.
- This thesis provides a demonstration scenario to validate that the proposed framework accurately inferred meaningful messages from raw sensor data and reliably delivered messages to the appropriate destinations.

1.4 Thesis Organization

This thesis is based on a series of large scale tactical communications system design challenges and recommendations of prudent approaches to meet these technical challenges. The remainder of this thesis is organized as follows. Chapter 2 investigates system-level quality of service design considerations based on a tactical network architecture which provides wide area network services. Chapter 3 discusses a new system design approach to consolidate the tactical local area network architecture as well as characterizing the performance of system architecture. Chapter 4 proposes a software architecture to dynamically manage computing resources at a tactical edge network as an attempt to improve the end-to-end network performance. Chapter 5 presents a reliable data aggregation and data dissemination framework that can potentially benefit future tactical system architecture. Finally, chapter 6 concludes this thesis and discusses future research direction in the related area.

Chapter 2

System-Level Quality of Service Design Considerations

This chapter investigates system-level quality of service (QoS) design considerations by modeling and simulation methodology. System-level QoS design greatly impacts the performance of a large scale communications system. Understanding the performance of a system architecture is an initial step to aid the design of a large scale communications system. High-fidelity modeling and simulation of the communications systems play a key role in investigating the system-level performance. This simulation study models a large scale communications system providing a wide area network (WAN) interface to the tactical network over a satellite communication (SATCOM) link. Each system component in the large scale communications system is modeled in details. Several network simulation test scenarios are developed to investigate the network performance of real-time applications.

2.1 Introduction to Automated Digital Network System

Automated Digital Network System (ADNS) was developed by the U.S. Navy with the primary objective of creating a more efficient network-centric communication environment for mobile units at sea [13] [14]. ADNS serves as a primary WAN interface between an afloat platform and the shore-based network operations centers (NOC). The system provides a means for an afloat platform to fully utilize the bandwidth allocated by the onboard radio frequency communications systems [15] [16]. It is essentially a radio-based wide area network which provides transparent end-to-end services to tactical and non-tactical user communities on both afloat units and shore based systems [13] [17] [14]. ADNS is the key enabler for connecting the afloat user communities to the resources and services provided by the GIG. ADNS leverages Commercial Off-The-Shelf hardware and adopts open standards such as TCP/IP network protocols to provide interoperability between Joint, Allied, and Coalition forces. The design of ADNS network architecture is an imperative step towards a joint concept for the Global Information Grid which utilizes IP network to provide secure and reliable network operations [18].

Since the deployment of initial ADNS in 1988 [18], the system has evolved through several generations including *Increment I* and *Increment II*. Due to inadequate bandwidth in the older satellite communication systems, ADNS *Increment I* has very limited WAN capabilities. It primarily provides integrated transport services for multi-level security enclaves over a single satellite communication link. A dynamic fail-over RF links mechanism is supported to avoid service interruption during SATCOM link failure. However, no guaranteed network bandwidth is available since the system merely transports tactical messages based on best-effort delivery. ADNS *Increment II* was deployed in 2005 [18], which provides more capabilities than its predecessor. ADNS *Increment II* fully utilizes the aggregated on-ship RF bandwidth, because it is capable of simultaneously distributing network traffic over multiple SATCOM links. The system provides a guaranteed

flexible bandwidth, application prioritization and some application level monitoring capabilities. To keep up with the growing demand of IP networks that support future tactical missions, the U.S. Navy has a roadmap in place to acquire ADNS *Increment III* in the near future [18].

The proposed ADNS *Increment III* architecture will provide support for higher bandwidth, including 50 Mbps for force level platforms and 25 Mbps for unit level platforms [14]. In order to migrate to an all IPv6 network in the near future, ADNS *Increment III* design requires a IPv4/IPv6 dual-stack backbone. Such system design is capable of supporting services on the existing IPv4 network while transitioning to the IPv6 network in the near future. Since ADNS is a part of mission critical components, it is important to maintain the fail-over RF links feature as well as implementing appropriate quality of service options. QoS design is necessary to ensure that each type of data packet is delivered according to dynamic-assigned or pre-assigned priorities. IP multicast protocol is also required to minimize the bandwidth requirement for broadcasting applications. The protocol conserves the bandwidth requirement by replicating the data packet at the intermediate routers, which reduces the SATCOM bandwidth burden at both sender and receiver nodes. Finally, ADNS *Increment III* aligns the afloat WAN service with the operational concept of GIG [19].

One of the primary objectives for deploying ADNS *Increment III* is to accommodate the increasing demand of running various time-constrained applications across a fully IP-based transport architecture. QoS design in ADNS employs differentiated services (DiffServ) [19] [20] [21] to assure timely delivery of tactical video and voice messages. A detailed network model has been developed using OPNET network simulator [22] to evaluate the performance metrics of video and voice applications with and without QoS implementation.

2.2 Configuration of the Network Model

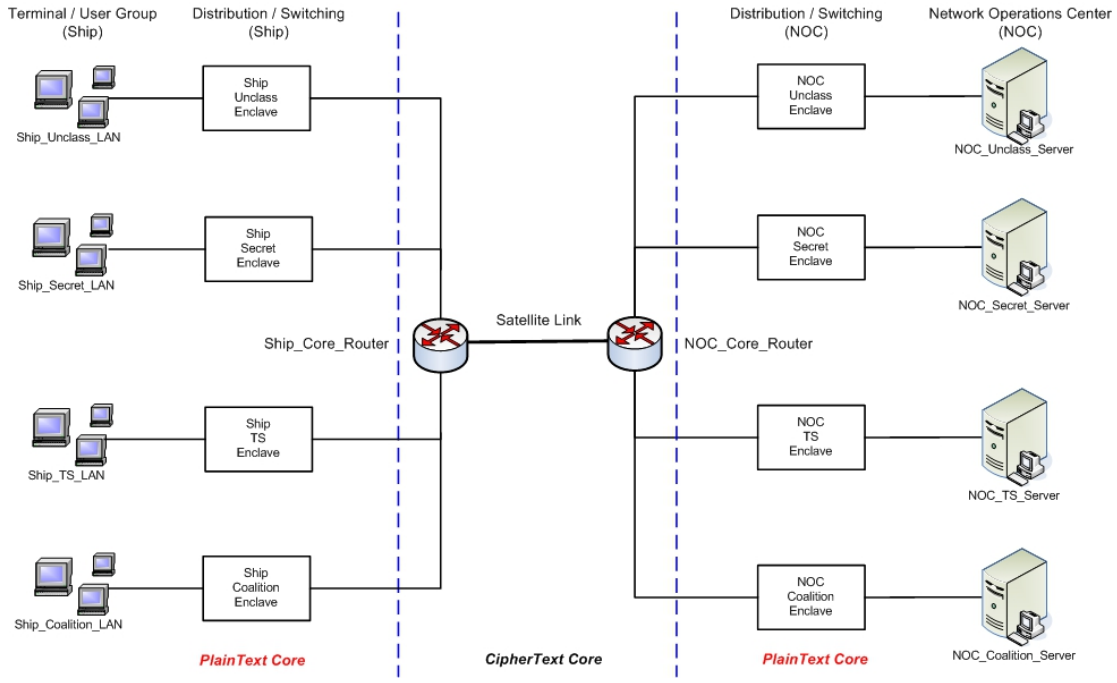
In order to evaluate and predict the system performance of ADNS, this large scale communications system is modeled using OPNET network simulator. In this simulation study, a network model based on COTS computing technologies is developed to represent the network architecture on an afloat platform (*i.e.* a ship) and at the shore (*i.e.* network operations center). These two major network entities, a ship and the network operations center, form a notional tactical network architecture for system performance investigation.

2.2.1 Network Simulation Topology

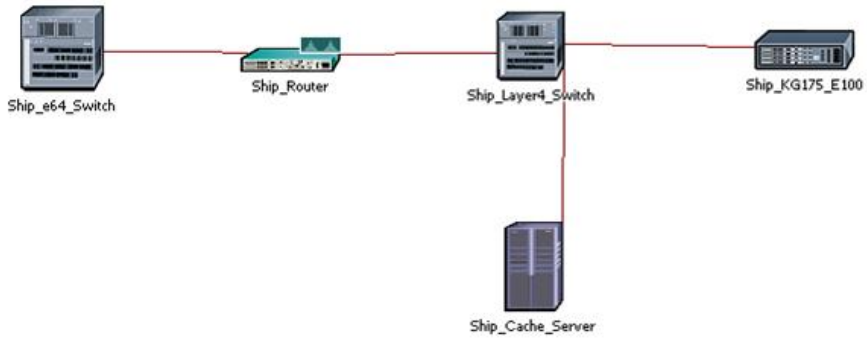
A notional tactical network topology simulating ADNS *Increment III* architecture is illustrated in Figure 2.1(a). In order to represent a typical ship-to-shore network environment, the network model employs a dumbbell topology with a high-latency and high-loss SATCOM link connecting a ship and the network operations center [19] [20] [21]. ADNS *Increment III* architecture supports users from multiple security enclaves, including Unclass, Secret, Top Secret, and Coalition[1] . The network model represents each security enclave by including an enclave router (*i.e.* edge router), a layer 4 switch for redirecting web traffic to an off-line cache server, and an encryption device. Figure 2.1(b) illustrates the network topology of a ship security enclave. Since both ship and the network operations center are functionally symmetrical from network simulation perspective, each security enclave at the network operations center is represented by the mirror image of Figure 2.1(b). All devices in the PlainText[2] core are connected with Ethernet 100BaseT link models. Network traffic generated from each LAN users group is destined for the respective servers resided in network operations center via the ship security enclave network architecture. Since ADNS implements Open

[1] All security classification labels used in this thesis are simulated and for discussion purpose only.

[2] PlainText core refers to a network transporting unencrypted IP data whereas CipherText core refers to a network transporting encrypted IP data such as the GIG.



(a) Notional Tactical Network Topology



(b) Network Topology of a Ship Security Enclave

Figure 2.1: Network Simulation of ADNS Increment III Architecture

Table 2.1: Characteristics of Mixed Application Traffic Profile

Application Type	**Inter-Arrival Time (sec)**	**Object Size (byte)**	**PHB**	**DSCP (Decimal)**
E-Mail	36	5000	BE	0
FTP	60	180000	AF13	14
HTTP	10	1000	AF23	22
Video	15 Frames/sec		AF43	38
Voice	Low Quality Speech		EF	47

Shortest Path First (OSPF) routing protocol [15] [19], the routing algorithm in the network model is configured with OSPF protocol [23].

2.2.2 Network Traffic Generation

Standard applications such as E-mail, File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), Database Query, Telnet, Video, and Voice are provided as standard application models in the OPNET simulation tool [22]. Mixed application traffic profiles are constructed from these standard application models to represent more realistic network traffic loads. This simulation study assumes the mixed applications has the network traffic characteristics shown in Table 2.1. Video traffic is derived from the default Low Resolution Video application and voice traffic is derived from Low Quality Speech application using G.723.1 encoder model [22]. In Table 2.1, assumptions were made on Per-Hop Behavior (PHB) and Differentiated Service Code Points (DSCP), which were derived based on the test results reported in [20] [21].

By adjusting the duration and repeatability of each application in the traffic profile, different average traffic throughput can be constructed. The traffic generation in each performance test is based on this mixed application traffic profile. Although OPNET simulation produces some randomness to the discrete values of the traffic profile, the average traffic throughput remains fairly constant. In order to obtain more insight into the network traffic profile, an offline analysis provides the simulation results shown in Figure 2.2 and Figure 2.3. The combined

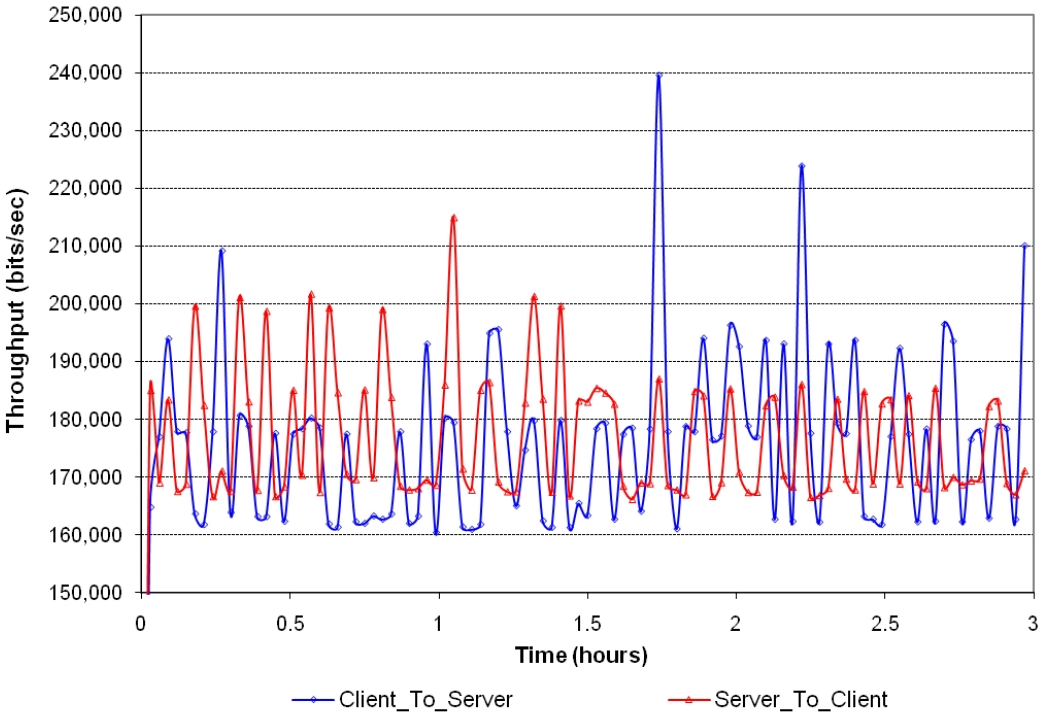


Figure 2.2: Combined Throughput of the Mixed Application Profile

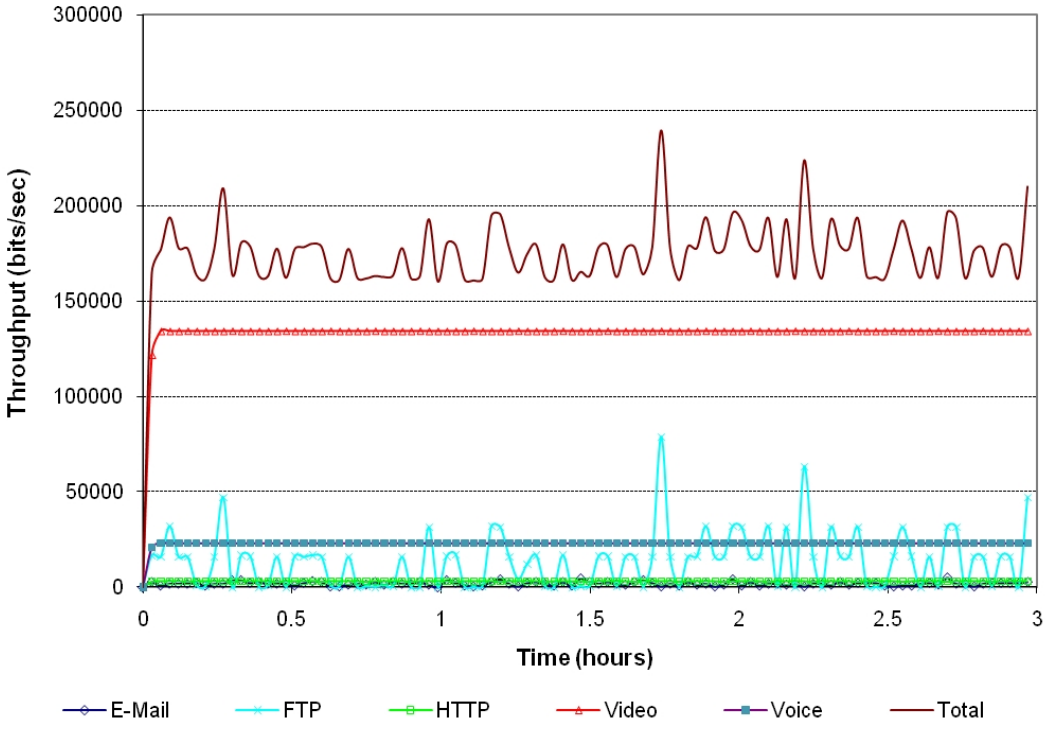


Figure 2.3: Seperated Throughputs of the Mixed Application Profile

throughput of the mixed application profile is shown in Figure 2.2, whereas Figure 2.3 separates the throughput for each application type in the profile.

2.2.3 Satellite Link

A Geosynchronous Earth Orbit (GEO) satellite system is 36,000 kilometers above the earth. The round-trip propagation delay for this path is approximately 270 milliseconds [24]. The drifting motion of a satellite varies this propagation delay. A Doppler buffer is required to restore this periodic time shifting, which results in a more constant delay time. In addition to propagation delay, the electronic equipment converting RF signals to baseband data packets introduce signal processing time delay into the path. Therefore, the end-to-end SATCOM link delay can be estimated to be roughly 300 milliseconds [24]. Another critical parameter to the SATCOM link is the Bit Error Rate (BER). Mathematically, BER is the probability that a bit sent over the communication link will be lost or received incorrectly. This bit probability is an approximate estimate which is only accurate over a long studied time and a large number of bit errors. Generally, a bit error occurs because the communication channel noise corrupts the transmitted signal and the decision circuitry at the receiver cannot identify the signal correctly. In the legacy communication systems, BER is typically 1×10^{-7} on average and 1×10^{-4} worst case [24]. In this simulation study, the SATCOM link is modeled with 300 ms propagation delay and 1×10^{-7} BER. The WAN data rate of the ship varies depending on the aggregate RF communication systems allocated to the specific platform. This simulation study assumes 512 Kbps SATCOM link using GEO satellite system [25].

2.2.4 Data Encryption

Network data encryption is required in order to guarantee that tactical messages are transported in a secure fashion. ADNS requires IP packets to be encrypted before being injected into the CipherText core network. The IP packets are then

decrypted at the destination host before being forwarded to upper network layers. In this simulation study, the node model configuration mimics the in-line KG-175 encryption model from the NetWars network simulator [26]. The node model captures essential parameters such as the processing overhead and synchronization time. Encryption processing generally increases the bulk size of the IP packet whereas decryption decreases the bulk size of the IP packet [26]. Processing overhead is used to determine the increasing or decreasing percentage data amount during the encryption or decryption process respectively. The synchronization time is modeled using failure and recovery effect. In general, these parameters affect the overall data throughput and time delay in the network.

2.2.5 Data Compression

Data compression increases the throughput of network traffic over a bandwidth-limited RF link by reducing the size of each data packet to be transmitted. In general, data compression permits more information to be transported than uncompressed data at the expense of packet processing time. An efficient data compression can effectively increase the data throughput over the RF link. When a compression scheme is deployed in the network, the sender first compresses each data frame before transporting it to the network. At the destination, the receiver then decompresses the data frame and forwards it to the upper layer services. In order for all intermediate nodes to forward each data packet to the appropriate destination, these nodes must also recognize the format of the compression. Since this step may incur more processing delays at intermediate nodes, the general practice is to only employ a data compression scheme between WAN interfaces. The compression efficiency depends on the data traffic dynamics as well as the underlying compression technique. This simulation study assumes that network compression occurs at the IP layer.

In order to determine the effectiveness of the IP layer compression, an offline

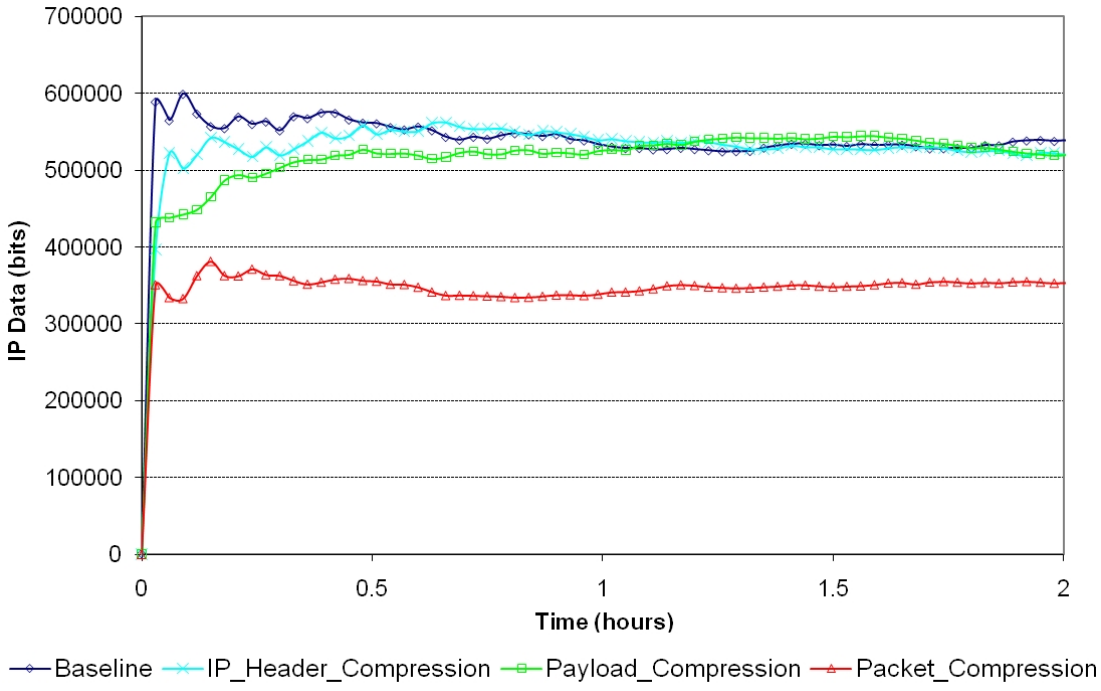


Figure 2.4: Comparison of IP Compression Schemes

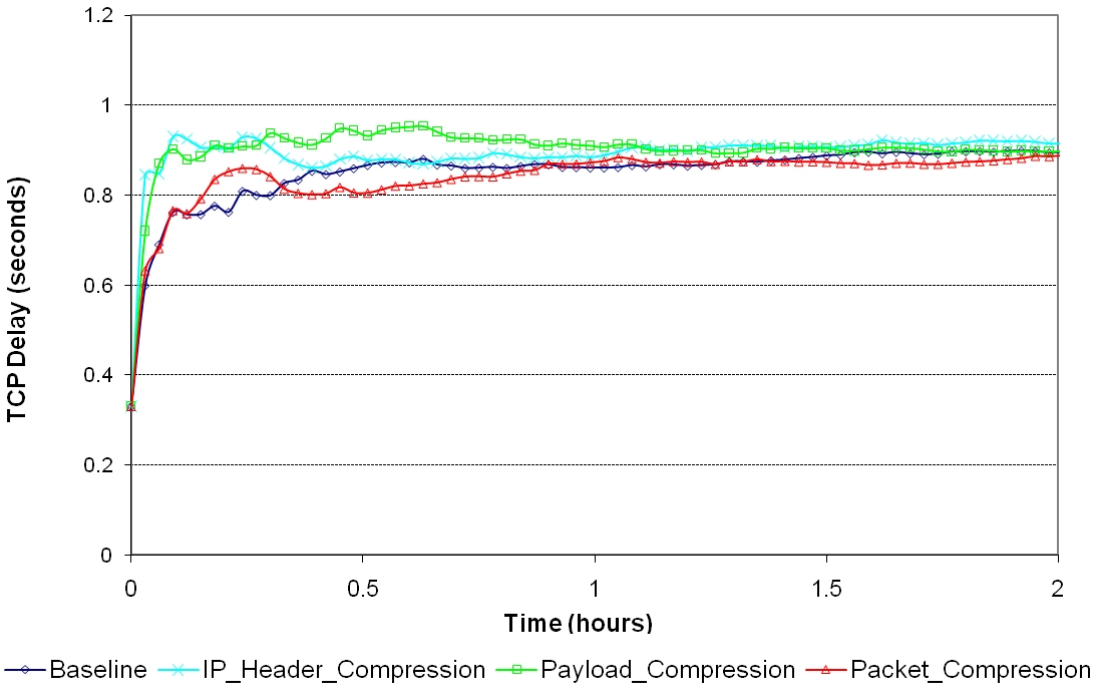


Figure 2.5: Comparison of Server TCP Delay

analysis using a simple client-router-server topology is conducted. The simulation results in Figure 2.4 show the comparison of three IP layer compression models in OPNET: Per-Virtual Circuit (*i.e.* Payload) compression, Per-Interface (*i.e.* Packet) compression, and IP header compression [22]. For a given network traffic profile (Baseline shown in Figure 2.2), the offline comparison shows that Per-Interface compression has lower bandwidth requirement while incurring some network delay due to packet processing time. As shown in Figure 2.5, the difference in time delay between packet compression and baseline is less than 10 milliseconds after 0.85 hours of the simulated time. Since this difference in time delay becomes negligible over a long period of time, the simulation study is configured with Per-Interface compression

2.2.6 Web Caching

Web or HTTP caching mechanism improves the performance of web browsing in the ADNS architecture. Depending on the web caching technology, a typical cache server can be placed either in-line or off-line in the network architecture. As shown in Figure 2.1(b), the simulation study assumes that each ship enclave incorporates an off-line web cache server connected to a layer 4 switch. Web cache reduces latency, conserves bandwidth, and provides information when the hosts become unavailable [27] [28]. Providing cached information is critical to any mobile tactical unit, because it sometimes undergoes communication interrupts due to various unexpected conditions such as adverse weather condition, Line-of-sight (LOS) issues, power failures and others. Web caching not only minimizes network service interrupts but alleviates the bandwidth requirement over the bandwidth-limited SATCOM link. Given that a cache usually exhibits high locality and temporality, these characteristics reduce latency to service client requests by retrieving data from the local storage. However, the freshness of the temporarily stored data depends on the actual implementation of the web cache server. Borrowing the term from computer architecture, Hit Ratio is often used

Table 2.2: Summary of Simulation Test Scenarios

Test Scenario	**Traffic Generation**	**QoS Location**
Baseline_Non_Congested	Secret Enclave ONLY	None
Baseline_Congested	All Enclave Users	None
QoS_Core_Routers	All Enclave Users	Core Routers ONLY
QoS_All_Routers	All Enclave Users	All Routers

as a performance metric in web caching. Past simulation studies have shown that web cache Hit Ratios range from 30% to 50% [29] [30], while the network equipments vendor reported as high as 70% in the mixed application traffic condition [31]. These variations are likely due to differences in user community, geography, or when the traces were obtained [27]. Considering that current ADNS is limited to the Navy community, it is reasonable to assume a higher cache Hit Ratio. The simulation model assumes a 45% cache Hit Ratio in an off-line cache server.

2.3 QoS Simulation Test Scenarios

Based on the simulation assumptions described in section 2.2, four different test scenarios are created in this simulation study. Each application type is marked with corresponding DSCP values shown in Table 2.1. Depending on the test scenario, QoS parameters are configured at different routers in the simulation test bed. Table 2.2 summarizes the configuration of each test scenario.

The first test scenario, *Baseline_Non_Congested*, establishes a performance baseline without any system-level QoS design implementation. In this test scenario, one mixed application traffic profile is loaded to represent LAN users in the secret enclave onboard a ship. Traffic profiles for other security enclaves are disabled in this test scenario. All network traffic originated from the the secret enclave onboard a ship are destined for the secret server at the network operations center. Since the networking resources are under utilized, the network performance statistics collected in this test scenario represent a non-congested network condition. As shown in Figure 2.2, the throughput of the mixed application profile

is much less than the 512 Kbps data rate configured in the SATCOM link model.

The second test scenario, *Baseline_Congested*, serves as another performance baseline to illustrate data delivery in a best-effort network under congested conditions. Similar to the first test scenario, QoS design is not configured in this test scenario. Each security enclave is loaded with one mixed-application traffic profile. All traffic originated from a security enclave of the ship are destined to the respective security enclave at the network operations center. This configuration ensures that there are network activities in every path of the simulation test bed. As shown in Figure 2.2, the lowest data throughput is about 160 Kbps. All four security enclaves will generate more than 640 Kbps network traffic, which effectively exceeds the 512 Kbps bandwidth configured at the SATCOM link model. Thus, a highly congested network environment is simulated in this test scenario.

In the third test scenario, *QoS_Core_Routers*, QoS design is only implemented at the two core routers (*Ship_Core_Router* and *NOC_Core_Router* as shown in Figure 2.1(a)) that are connecting the ship and NOC. It is important to investigate QoS performance at these critical routers, because they serve as primary gateways connecting the ship's network to a bandwidth-limited SATCOM WAN interface. The QoS parameters are configured at all interfaces in these two routers. The fourth test scenario, *QoS_All_Routers*, is similar to the third test scenario. In addition to QoS parameters configured at the core routers, QoS parameters are also configured in all available enclave routers in the simulation test bed for investigating any further performance improvement.

2.4 QoS Simulation Results and Discussion

The simulation results from each test scenario are compared and analyzed in this section. It is valuable to first examine the performance of the SATCOM link under different test scenarios, because this WAN link is essentially the network bottleneck. Figure 2.6 and Figure 2.7, show the average throughput and utilization rate of the SATCOM link respectively. In Figure 2.6, the *Baseline_Non_Congested*

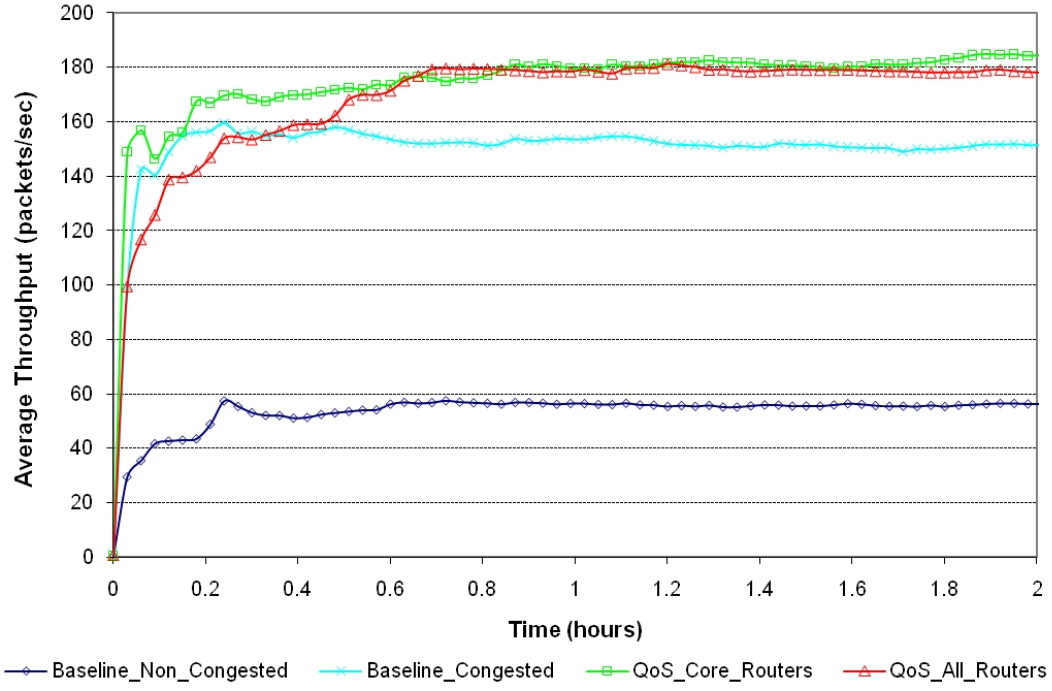


Figure 2.6: Average Throughput of the SATCOM Link

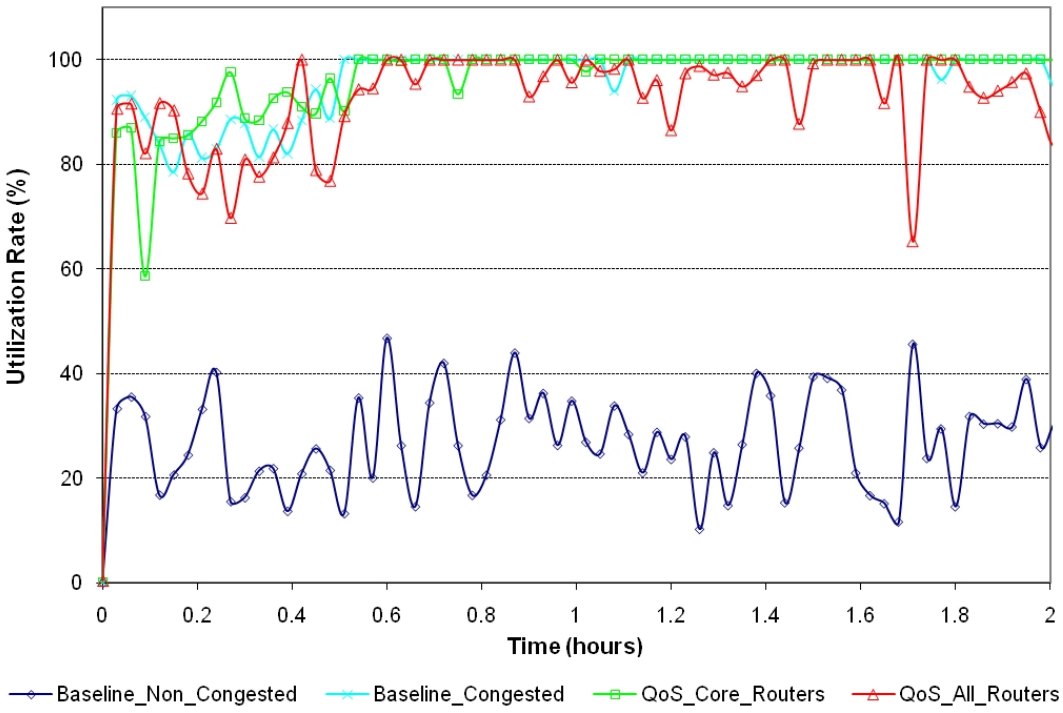


Figure 2.7: Utilization Rate of the SATCOM Link

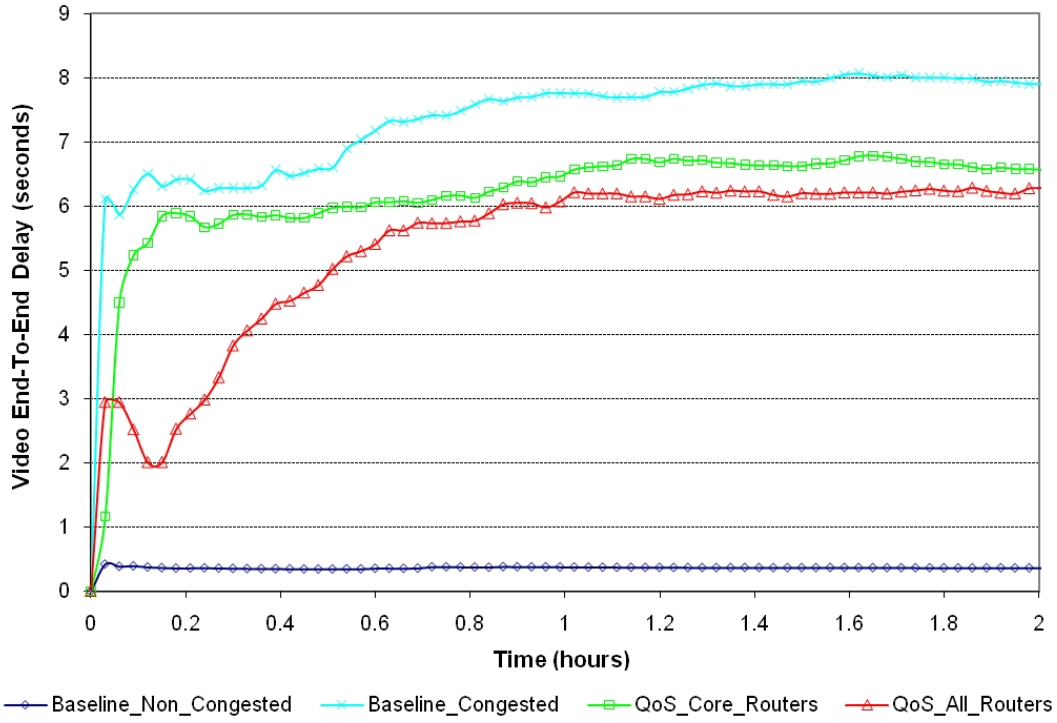


Figure 2.8: Average End-To-End Time Delay of Video Application

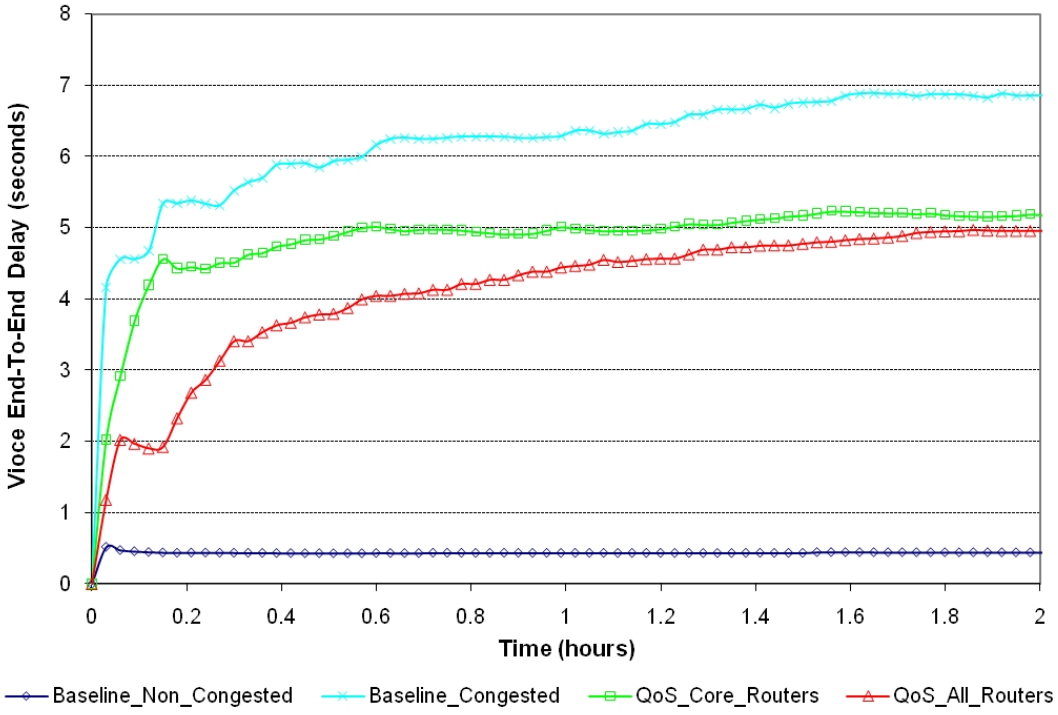


Figure 2.9: Average End-To-End Time Delay of Voice Application

test scenario has much less average throughput than the other three test scenarios, because only one instance of the mixed traffic profile is generated from secret enclave. The utilization rate shown in Figure 2.7 only ranges from 18% to 42%. The SATCOM link is far from being saturated in the *Baseline_Non_Congested* scenario. Figure 2.6 also demonstrates that, in general, QoS implementation can achieve higher average throughput over the SATCOM link. After 0.7 hours, both *QoS_Core_Routers* and *QoS_All_Routers* show nearly the same average throughput which is about 20% higher than *Baseline_Congested*. This may be due to more efficient packet delivery when QoS is properly implemented. It is interesting to point out that, in Figure 2.6, the plot for *QoS_All_Routers* reaches maximum level slower than *QoS_Core_Router* and *Baseline_Congested*. This is likely due to the implementation of QoS to all available routers in the network which incurs longer packet processing time in each router. Except for the *Baseline_Non_Congested* test scenario, Figure 2.7 shows that all other test scenarios eventually achieve nearly 100% utilization after 0.6 hours. However, it can be seen that *QoS_All_Routers* plot does not remain at full 100% utilization rate as much as other test scenarios. As expected, these simulation results confirm that QoS parameters implemented in all routers are more likely to perform better than other test scenarios.

In order to investigate how a bandwidth-limited SATCOM link affects the performance of time-constrained applications, the end-to-end (ETE) time delay for video and audio applications are examined. The ETE time delay parameter is defined as the time it takes for a packet to travel from a source node to a destination node. Figure 2.8 and Figure 2.9 show the performance measurements of ETE time delay for video and audio application respectively. In the non-congested network (*Baseline_Non_Congested*), both video and audio traffic exhibit approximately 400 milliseconds ETE time delay. With the simulation assumption of 300 milliseconds time delay in the SATCOM link, this means the network traffic takes about 100 milliseconds to travel across both the ship enclave and the NOC network architecture. Reducing processing delay in the enclave system architecture can help improve the end-to-end network performance. As shown in Figure 2.8, the video

application can experience high ETE time delay up to 7.8 seconds when the network is heavily congested. The voice application shown in Figure 2.9 experiences slightly less ETE time delay at about 6.2 seconds. In general, it can be seen from Figure 2.8 and Figure 2.9 that QoS implemented at core routers can improve the ETE time delay in both applications. This performance can be further improved if QoS is implemented in all available routers, although the simulation results show that performance gain becomes insignificant after a long period of time. Based on simulation results beyond one hour, the *QoS_Core_Routers* and *QoS_All_Routers* show no significant performance difference.

2.5 System-Level QoS Design Considerations Conclusion

This chapter investigates system-level quality of service design considerations by modeling and simulation of ADNS *Increment III* system which plays a crucial role in providing on-ship WAN services over a bandwidth limited, high latency and high-loss SATCOM link. The simulation results show that a SATCOM link under congested conditions will result in high packet drop and long ETE time delay in both video and audio applications. The results also show that under highly congested network conditions, QoS offers a performance advantage. As a minimum system design requirement, QoS implementations at both ship and shore core routers are recommended. However, depending on the choices of the COTS network equipment, implementing QoS at all routers does not necessarily provide best cost per performance value. Simulation results also confirm that video and voice applications suffer very long time delays due to the physical characteristics of the SATCOM link. While the physical characteristics of the SATCOM link imposes a performance upper bound, these long time delays can be improved under congested traffic conditions if QoS design is integrated in the ADNS *Incrememt III* architecture. Furthermore, optimizing QoS parameters may be required to

improve the system's performance. In addition, faster packet processing in each router minimizes the overhead time in each of the enclaves network architecture. This simulation study provides invaluable insights for refining the current tactical network architecture as well as developing future communications systems.

Chapter 3

Toward Consolidated Network Architecture

This chapter proposes a consolidation network architecture which provides an integrated shipboard local area network infrastructure for disparate afloat platforms operating with multiple security level enclaves. The proposed consolidated network architecture is evaluated using modeling and simulation techniques. System performance trade-offs are investigated in the simulation test scenarios. System performance results of several applications are presented and analyzed to validate the system-level design.

3.1 Introduction to Consolidated Network Architecture

The U.S. Navy has developed a plan to revolutionize the future of its fleets' communications and networking capabilities by consolidating all shipboard network-centric communications systems to a common architecture known as Consolidated Afloat Networks and Enterprise Services (CANES) [32] [33] [34] [35]. CANES

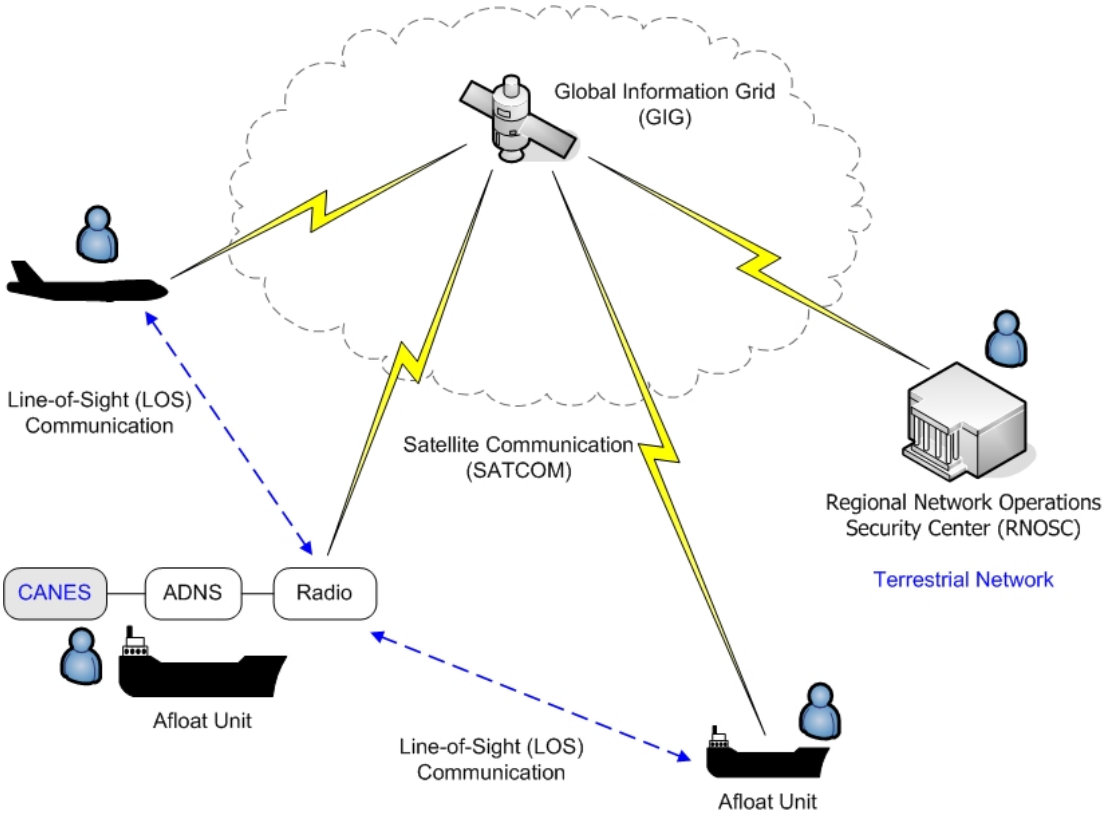


Figure 3.1: Operational View of Afloat Networks

implements a Common Computing Environment (CCE), which includes an integrated local area network infrastructure for different types of afloat platforms operating with multiple security level enclaves [35]. The high-level operational view of the afloat networks in Figure 3.1 illustrates that CANES interfaces with Automated Digital Network System to provide end-to-end network services across the Global Information Grid. Inspired by the concept of Service Oriented Architecture (SOA), CANES consists of loosely coupled hardware devices capable of hosting a wide variety of tactical and non-tactical applications. The design of the hardware architecture utilizes open architecture COTS products which are modular and scalable. The core services are comprised of reusable software applications that can be efficiently adapted to support rapidly changing demands in a wide range of tactical operations.

In the traditional system-centric design paradigm, each type of tactical communications system was designed to support a single warfighting function [35]. This design paradigm often created many stove-piped systems, each developed in isolation and without considering the requirements for interoperability with existing and future technologies. A stove-piped system typically requires a unique set of hardware devices, software applications, and a distinct network infrastructure. Depending on its application, the system is often inefficiently utilizing the available processing resources. The development of the software applications is based on the specific hardware architecture of the host. In addition, as the demands for network capacity increase, the total number of required stove-piped systems can potentially grow into an unmanageable state for both operations and maintenance staff. It also increases the complexity of system integration, training, and supportability [35]. From a security perspective, these individual networks are difficult to certify and defend against potential network attacks [33]. A consolidated tactical network architecture has the advantage of reducing the number of system components, and furthermore has the ability to mitigate the risks incurred with integrating many stove-piped systems. The future shipboard network-centric communications systems developed around the common computing architecture

will require less power, smaller rack space, lighter weight, and lower overall cost in system development and maintenance.

The objective of this study is to develop a simulated network test bed using OPNET Modeler for the performance evaluation of a consolidated tactical network architecture. The performance statistics were collected from the simulated test bed to investigate the performance trade-offs under various operational scenarios in a consolidated network architecture.

3.2 Consolidated Network Configuration

A simulated network test bed based on COTS products is developed to represent a consolidated network architecture. As illustrated in Figure 3.2, the network test bed incorporates multiple security level enclaves to model a consolidated LAN infrastructure on a typical tactical afloat platform. Each security level enclave is modeled with a LAN client entity and a respective service provider. This network test bed serves as the baseline architecture for each test scenario in Section 3.3.

3.2.1 Consolidated Network Topology

The topology of the consolidated network architecture is illustrated in Figure 3.2. The network supports users from multiple security levels, including *Top Secret*, *Secret*, *Unclassified* and *CENTRIXS*[1] enclaves. All security level enclaves are joined together at the core switching network. The core switching network has the provision for network redundancy to prevent service interruption from potential link or node failures. All network components are interconnected with 100BaseT Ethernet links. The links joining the two core switching devices are configured as trunk ports to prevent any service interruption due to failures. Since the High Assurance Internet Protocol Encryptor (HAIPE) model currently only supports Routing Information Protocol (RIP), the nodes connected to the HAIPE models

[1] Combined Enterprise Regional Information Exchange System (CENTRIXS) is a secure network for coalition forces interoperability.

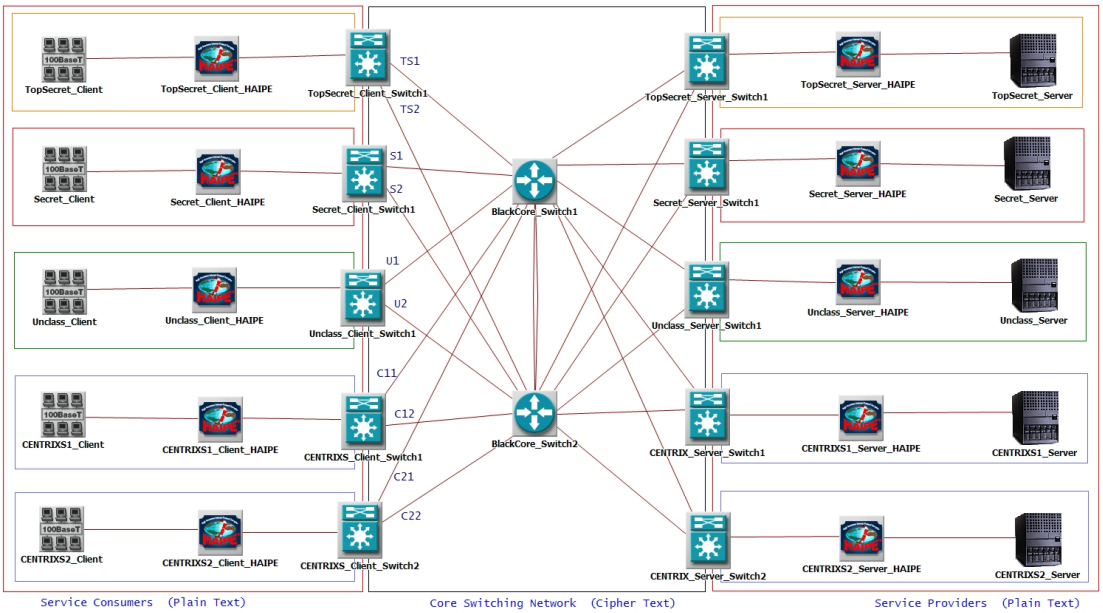


Figure 3.2: Consolidated Network System Architecture

are configured to use RIP protocol [36]. The core network routing is configured with Open Shortest Path First protocol [23].

3.2.2 Network Encryption Device

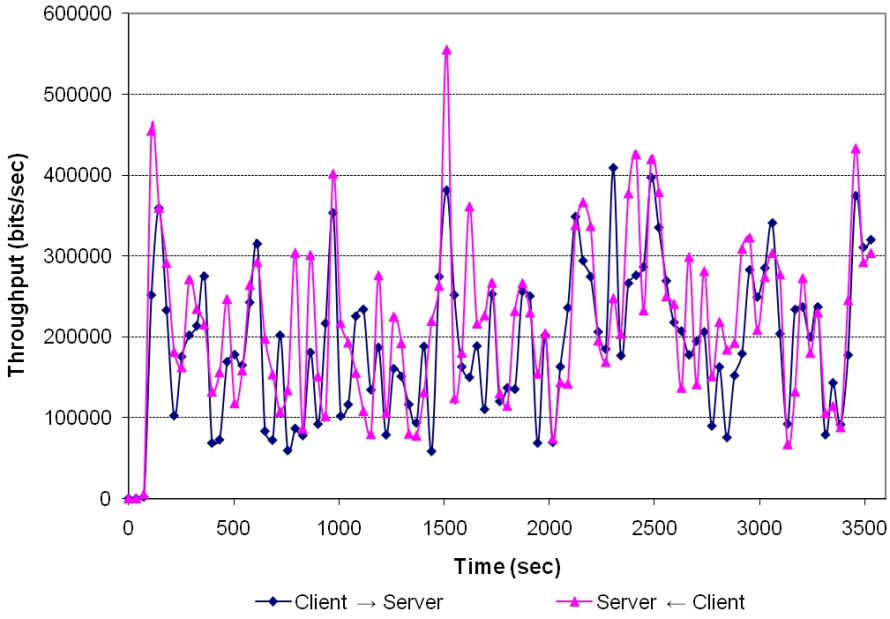
Tactical networks regularly rely on Internet Protocol (IP) encryption devices to ensure that data is securely transported. HAIPE is a Type 1 encryption device that complies with National Security Agency (NSA) security requirements. In a consolidated network architecture with multiple levels of security, each security enclave requires peering HAIPE devices to provide the secure end-to-end data delivery. When the security association policies are configured in the peering HAIPE devices, a secure network tunnel is created in the CipherText network in order to maintain the data separation between security enclaves. The network test bed is configured such that data can only be sent and received within its respective security enclave. Plain-text data packets are encrypted by a HAIPE device before being injected into the CipherText only core switching network. The

data packets are decrypted by the peering HAIPE device before being forwarded to a destination with the same security classification level as the source. Since all data passes through peering encryption devices, the HAIPE model affects the overall throughput and packet latency in the network architecture.

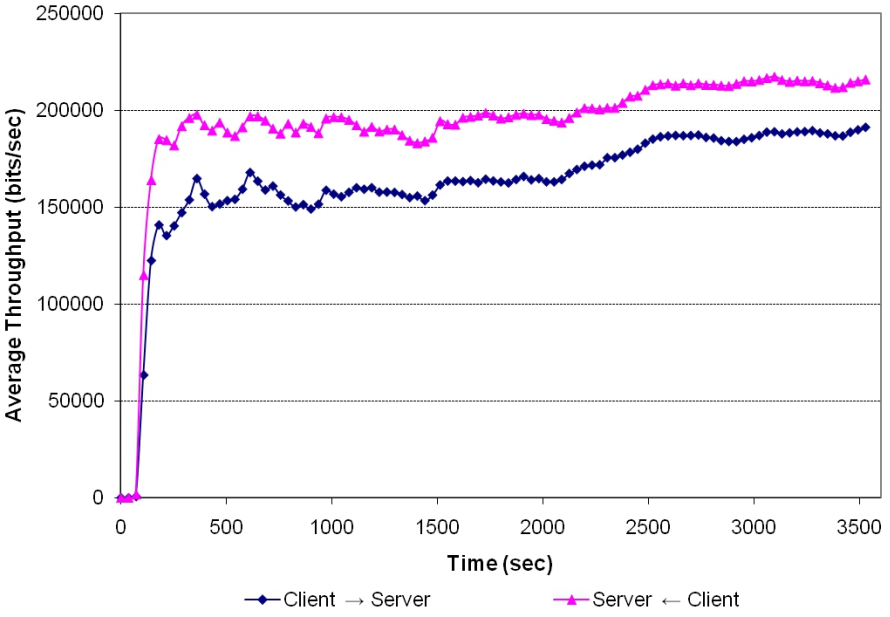
3.2.3 Service Classes

Network traffic models are typically derived from the output of a dynamic stochastic processes (*i.e.,* random) which require challenging modeling techniques. OPNET Modeler provides a convenient solution to represent standard applications such as *Database Query*, *E-mail*, *FTP*, *HTTP*, *Remote Login*, *Telnet*, *Video*, *Voice* and other custom applications in its application configuration model [22]. Various network traffic behaviors can be modeled by adjusting application parameters such as object size, request inter-arrival time, packet distributions, start/end times, duration, repeatability, and others. Before investigating the performance statistics of various simulation scenarios, it is important to construct a common network traffic profile representing the behavior of the actual network applications and to understand its basic characteristics. This study assumes the configuration of mixed applications as shown in Table 3.1 [8] [22] [37] [38], which includes four major service classes commonly used in tactical applications. The control and management class typically consumes very low network bandwidth with near-constant logging or polling of the system status. It is important to ensure the control and management packets are delivered on time and with low packet loss. The inelastic[2] real-time, preferred elastic, and elastic traffic have low, medium, and high tolerance, respectively, in term of packet loss, delay, and jitter parameters. Figure 3.3 and Figure 3.4 illustrate the network traffic throughput of the mixed applications used in this study. In Figure 3.3(b), the network traffic throughputs exhibit asymmetric characteristics in both directions indicate that there is more network traffic from a server to a client due to FTP downloading.

[2] Inelastic traffic require a certain QoS level to function, whereas elastic traffic can continue to function given constrained QoS level.

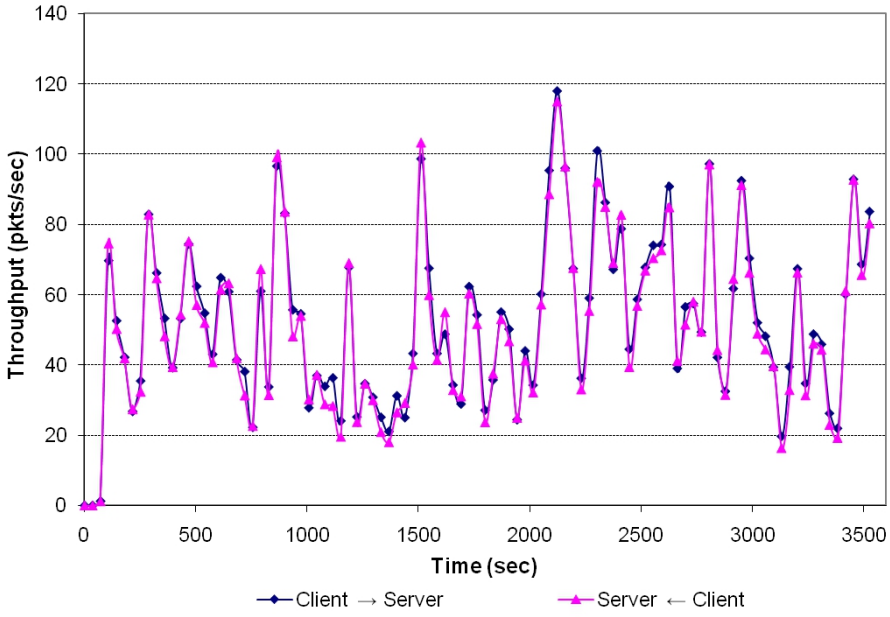


(a) Throughput (bits/sec)

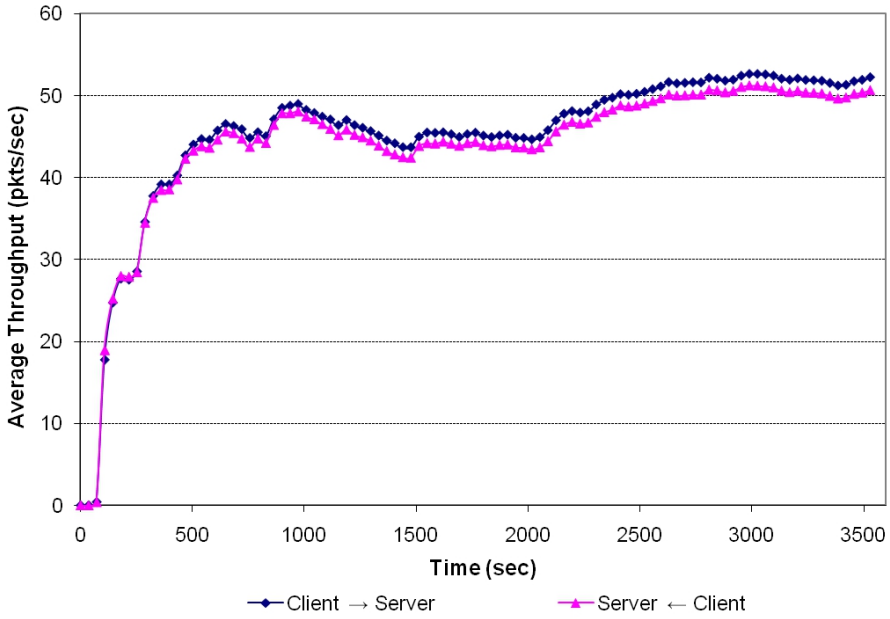


(b) Average Throughput (bits/sec)

Figure 3.3: Characteristics of Mixed Application (bits)



(a) Throughput (pkts/sec)



(b) Average Throughput (pkts/sec)

Figure 3.4: Characteristics of Mixed Application (pkts)

Table 3.1: Configuration of Mixed Applications

Service Classes	Applications	Attributes	Setting	PHB / DSCP
Control & Management	Remote Login	Inter-Command (sec)	normal(60,5)	CS5 / 40
		Terminal Traffic (bytes/command)	normal(20,16)	
		Host Traffic (bytes/command)	normal(10,11.111)	
Inelastic Real-Time	Voice	Encoder Scheme	G.723.1 5.3K	EF / 47
		Voice Frame per Packet	1	
		Compression Delay (sec)	0.02	
		Decompression Delay (sec)	0.02	
		Conversation Environment	Land Phone - Quiet Room	
	Video	Frame Inter-arrival Time Information	15 frames/sec	AF43 / 38
		Frame Size Information (bytes)	Incoming Stream Frame Size = constant(1066) Outgoing Stream Frame Size = constant(1066)	
Preferred Elastic	HTTP	HTTP Specification	HTTP 1.1	AF23 / 22
		Packet Inter-arrival Time (sec)	constant(10)	
		Page Properties	Object size (bytes) = constant(1000) Objects/Page = constant(1) Medium Image = constant(7)	
	FTP	Command Mix (Get/Total)	50%	AF13 / 14
		Inter-Request Time (sec)	exponential(80)	
		File Size	constant(200000)	
Elastic	E-mail	Send Interarrival Time	exponential(24)	BE / 0
		Send Group Size	exponential(24)	
		Received Inter-arrival Time	constant(1)	
		Received Group Size	exponential(24)	
		E-Mail Size (bytes)	constant(6000)	

3.2.4 Network Performance Statistics

OPNET Modeler provides built-in performance statistics collection that supports both global and local statistics [22]. Global statistics are computed from the overall performance characteristics of a network scenario. Local statistics provide performance parameters for particular objects such as nodes, links, and modules. Common network performance measures such as *Throughput*, *Delay*, *Delay variation* (*i.e., Jitter*), and *Packet loss* are included in the performance statistics collection. The mathematical foundation of these performance statistics are briefly discussed in this section.

The channel capacity of a network connection can be characterized by the throughput parameter. Throughput may be unidirectional and is defined as the average rate that data is being successfully sent or received over a fixed time interval. It is typically rated in terms of bits-per-second (*bits/sec*) or packets-per-second (*pkt/sec*).

$$Throughput_{(sent/received)} = \frac{\sum Data_{(sent/received)}}{Time}$$

The latency of an IP network is measured by the delay parameter. Typically, ETE delay is a critical parameter for studying the performance of time-sensitive applications such as video and voice applications. The ETE delay is the elapsed time for a packet traveling from a source to a destination. In an IP network, the ETE delay can be further divided into time intervals taken in various stages such as processing, queuing, transmission, and propagation. Processing delay is the total computing time required to assemble and disassemble a network packet. Queuing delay refers to the waiting and servicing time in a queuing buffer. Transmission delay is the required time interval for pushing all data bits in a packet onto the physical connection (*e.g.,* wired Ethernet [39] or wireless IEEE 802.11 [40]). Propagation delay refers to the elapsed time that a packet is in transit on the physical connection.

$$\begin{aligned} Delay_{(ETE)} &= Time_{(destination)} - Time_{(source)} \\ &= Delay_{(processing)} + Delay_{(queuing)} \\ &+ Delay_{(transmission)} + Delay_{(propagation)} \end{aligned}$$

Packet delay variation (v_k or ${\sigma_k}^2$) is sometimes loosely defined as "*jitter*", which is the difference between an absolute packet delay time (d_k) and a predefined reference packet delay time (d_{ref}) [41]. A predefined reference packet delay time must be a selected value which maintains an acceptable quality of service (QoS) for the particular application. The average packet delay ($\bar{d}$) in the same network connection flow can sometimes be used as a reference packet delay time. This parameter greatly affects the quality of the real-time applications such as video and voice services.

$$v_k = d_k - d_{ref}$$

Or,

$$\sigma = \sqrt{{\sigma_1}^2 + {\sigma_2}^2 + \ldots + {\sigma_n}^2} = \sqrt{v_1 + v_2 + \ldots + v_n}$$

$$\sigma_k = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(d_i - \bar{d})^2} \quad , \quad \bar{d} = \frac{1}{n}\sum_{i=1}^{n} d_i$$

Packet loss is the number of packets that fail to reach their destinations in a timely manner. Packet loss frequently occurs under congested network conditions, as the packets are dropped when the queuing buffers in the network devices are overflowed. Packet loss ratio illustrates the likelihood of packets not being delivered successfully in a network. In order to maintain the desirable efficiency of the applications, packet loss must be minimized in a network architecture.

$$PacketLoss_{ratio} = \frac{\sum Packets_{(lost)}}{\sum Packets_{(transmitted)}}$$

3.3 Performance Evaluation Scenarios

Failover and traffic growth test scenarios were developed based on the previously described network test bed architecture to investigate the behavior of the network traffic and the effect on the mixed applications performance, respectively. Failover minimizes service interruption by switching over to a redundant system upon failure(s) in the network architecture. A sufficient number of redundant components must be implemented in the consolidated network architecture in order to fulfill various failover conditions.

In addition to failover requirements, it is very important to understand the behavior of the mixed applications as the network traffic growth is scaled to utilize more link bandwidth. The results of the traffic growth scenario guide the investigation of application performance under high bandwidth utilization. Thus, appropriate techniques such as QoS policies can be applied to improve the performance of the network architecture.

3.3.1 Failover Test Scenarios

Network failures are commonly caused by link-down or node-down conditions. The failover scenarios in this study were developed as an attempt to understand the behavior of the network traffic under failover conditions. In other words, "How does the consolidated network architecture behave during failover conditions?"

Effects of Failover Links

In this test scenario, each Ethernet link connecting the client-side switches to the core switches are configured to fail for a fixed time interval during the simulation run. The labels of all network links are from the consolidated network test bed architecture shown in Figure 3.2. During the initial five minutes, all network links connecting to both core switches are operational. The network links then undergo failure and recovery modes according to the link failure/recovery timing diagram shown in Figure 3.5. Each network link fails sequentially at a specific time. The duration of each link failure is set to five minutes. Once a network link is recovered, it remains operational until the end of the simulation.

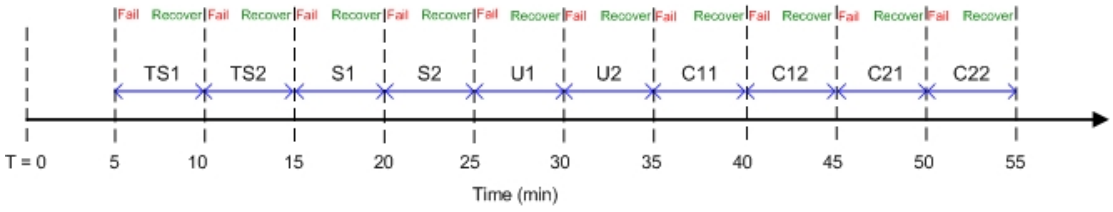


Figure 3.5: Timing Diagram for Failover Links Scenario

Effects of Failover Nodes

Similar to the failover links scenario, the failover nodes scenario follows a pre-defined node failure/recovery timing diagram shown in Figure 3.6. Both core switches are operational during the initial 10 minutes of the simulation time. The *Blackcore_Switch1* fails at $T = 10$ minutes and then recovers at $T = 20$ minutes.

At that time, the *Blackcore_Switch2* enters the failure mode and then recovers later at $T = 30$ minutes. Upon recovery of a core switch, it remains operational until the end of the simulation.

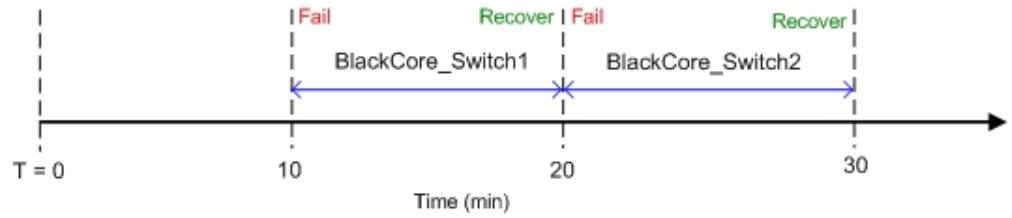


Figure 3.6: Timing Diagram for Failover Nodes Scenario

3.3.2 Traffic Scalability Test Scenario

A network traffic profile of the background load is configured in order to investigate the application performance as a function of the network traffic load scaling up. Figure 3.7 illustrates a predefined network traffic profile for loading the background traffic. This profile gradually increases the background traffic load every five minutes until reaching 98 Mbps, which nearly utilizes most of the 100 BaseT Ethernet link bandwidth. This scenario is an attempt to understand what happens to the performance statistics at the application layer when the background network traffic loads are gradually increased.

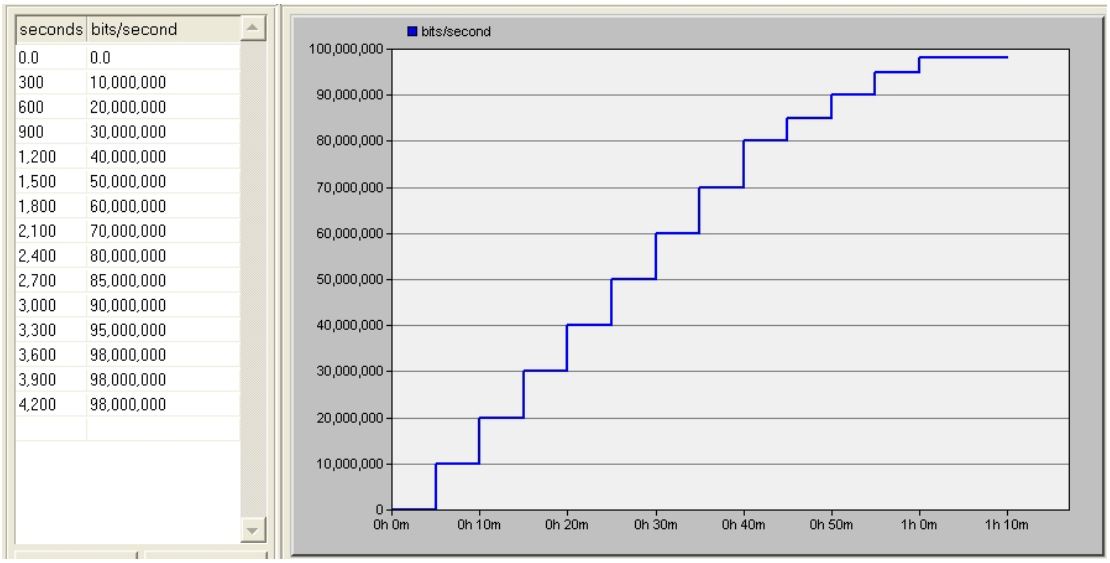


Figure 3.7: Background Traffic Load Profile for Traffic Scalability Scenario

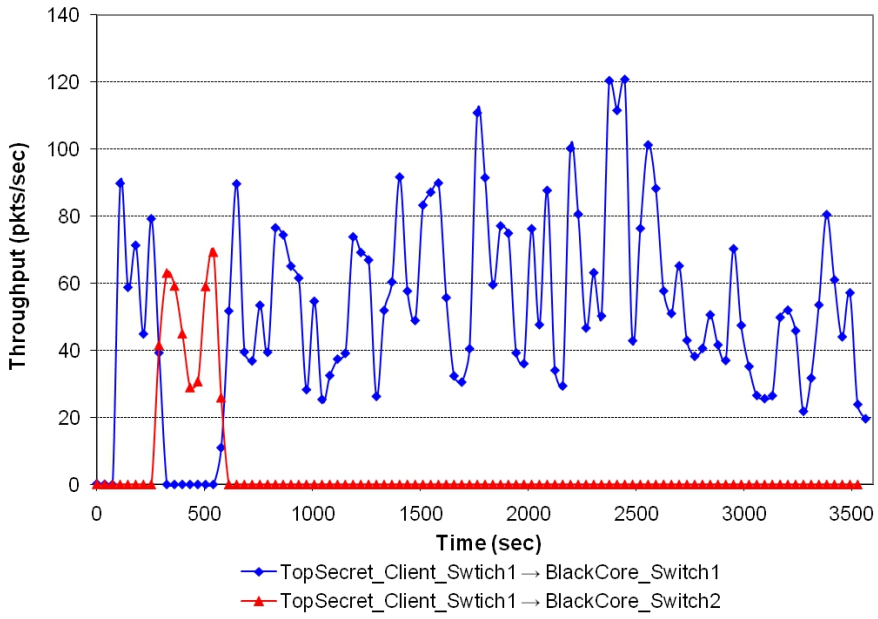
3.4 Performance Simulation Results and Discussion

The test results of the failover scenarios are presented in Figure 3.8 and Figure 3.9. These test results were taken from a single enclave since all security level enclaves are symmetrical in the network test bed architecture. As the expected behavior of the routing protocol, the network traffic is redirected to the appropriate destination through an alternative path. This effect is illustrated in both failover scenarios. The results of the failover links scenario in Figure 3.8 demonstrate that primary link *TS1* transports the data packets in the enclave under normal operating conditions. Upon failure of the *TS1* link, the network traffic is rerouted through the secondary link *TS2*. The test results show that it takes less than one minute for the secondary link *TS2* to take over the network traffic load from the primary link *TS1* and vice versa. Since the packets are not configured to be distributed across both links, both Figure 3.8(a) and 3.8(b) show that failure

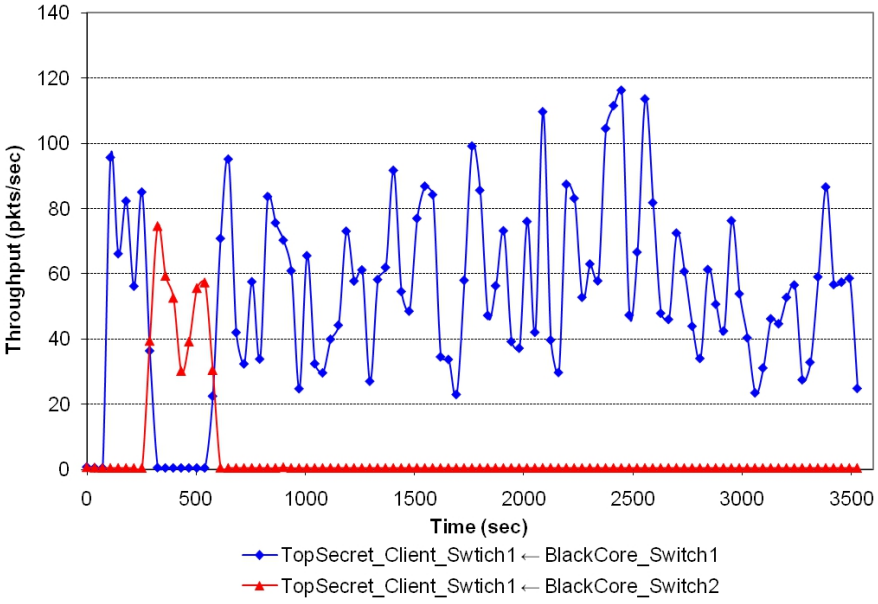
in the secondary link *TS2* has no effect in the network traffic throughput.

The results of the failover nodes scenario are illustrated in Figure 3.9. As expected, the results show similar traffic behavior as in the failover links scenario. In the failover nodes scenario, *Blackcore_Switch1* is the primary node and *Blackcore_Switch2* is the secondary node. The network traffic is redirected through *Blackcore_Switch2* during the failure mode of *Blackcore_Switch1* from $T = 10$ minutes to $T = 20$ minutes. The results show that it takes less than one minute to redirect all network traffic through the secondary core switch.

Figure 3.10 and Figure 3.11 summarize the results of the traffic scalability scenario. Figure 3.10 illustrates the expected traffic behavior; the throughput is increasing according to the predefined background load shown in Figure 3.7. The results of mixed applications show that the response time for E-mail in Figure 3.11, FTP in Figure 3.12, and HTTP applications in Figure 3.13 scaling up as the background load increases in the network. Application response time remains steady until 60 Mbps (or 60%) of the link bandwidth is utilized. Beyond this threshold, the response time and delay time experience greatly increases as a function of the background traffic load. Both video (shown in Figure 3.14 and Figure 3.15)and voice (shown in Figure 3.16 and Figure 3.17)applications also demonstrate simliar performance characteristics. This performance measurements create baselines to determine if QoS policies are required in the consolidated network architecture to maintain service-level requirements.

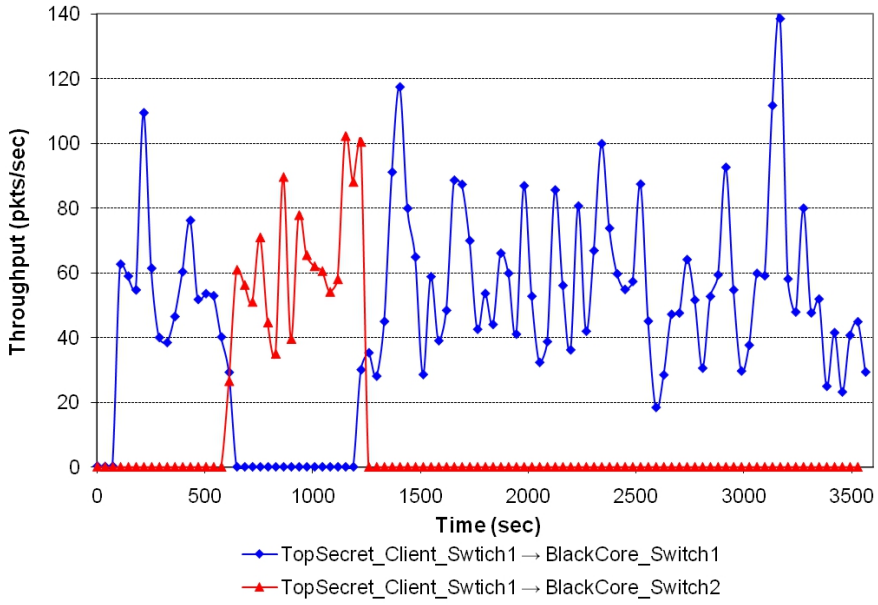


(a) Failover Links (Client → Server)

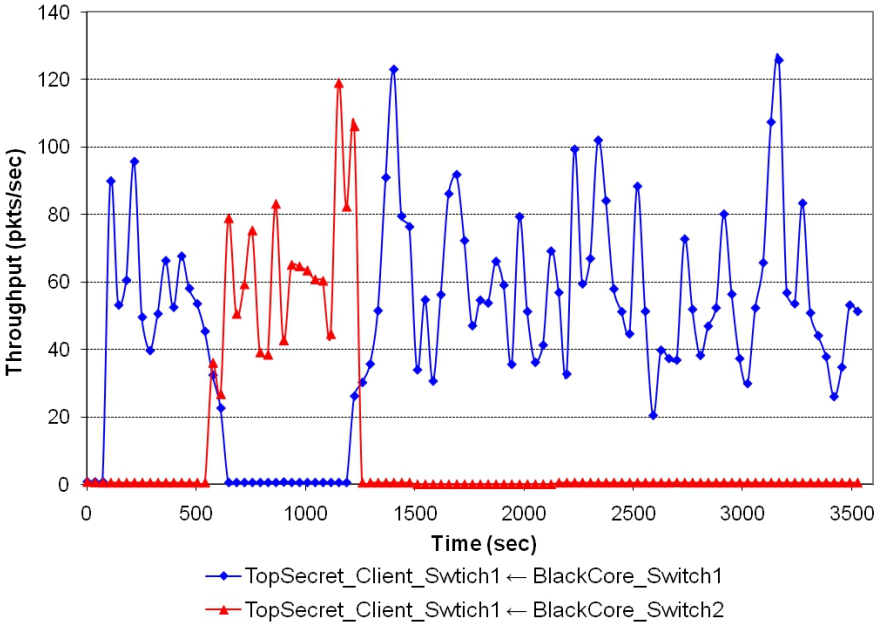


(b) Failover Links (Client ← Server)

Figure 3.8: Failover Links Scenario Results

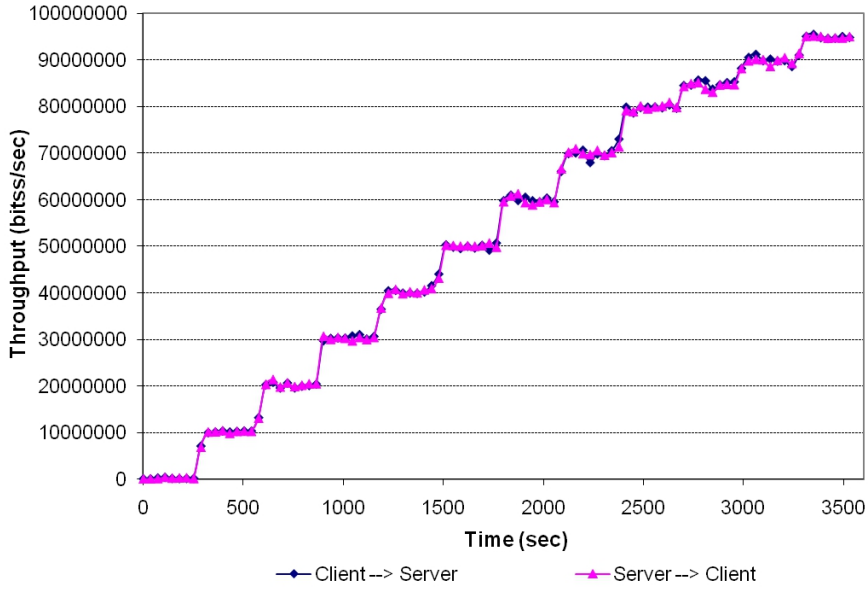


(a) Failover Nodes (Client → Server)



(b) Failover Nodes (Client ← Server)

Figure 3.9: Failover Nodes Scenario Results



(a) Throughput (bits/sec)

(b) Throughput (pkts/sec)

Figure 3.10: Throughput for the Traffic Growth Scenario

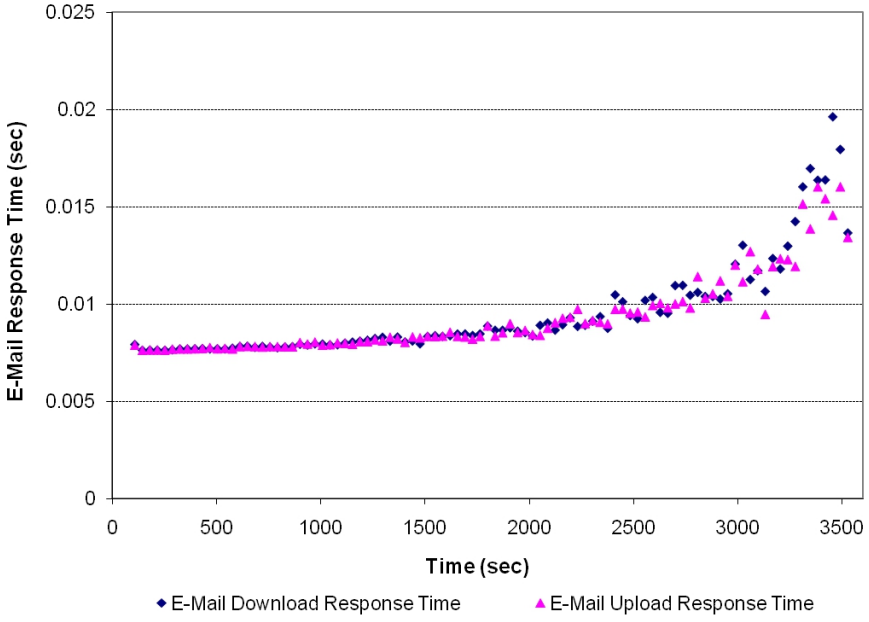


Figure 3.11: Email Response Time

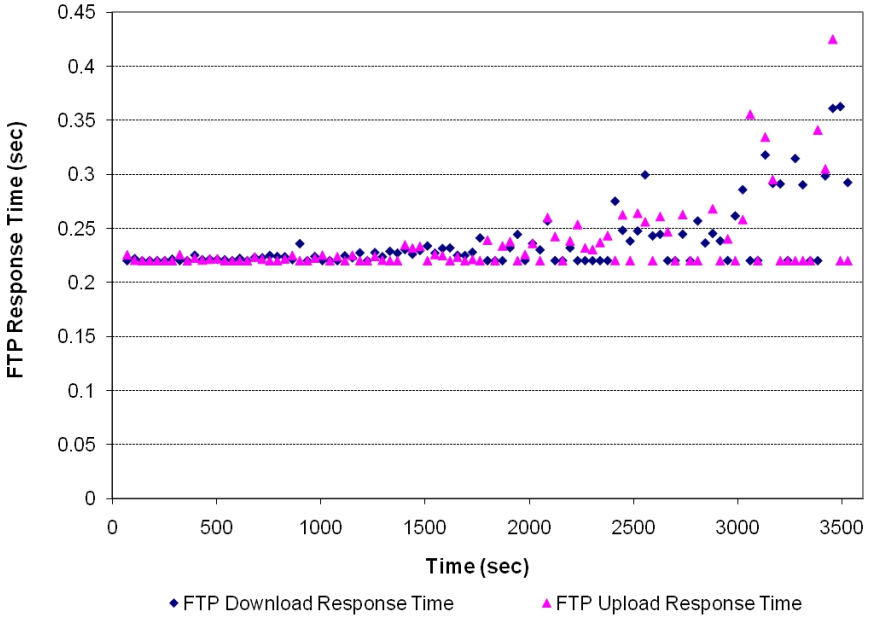


Figure 3.12: FTP Response Time

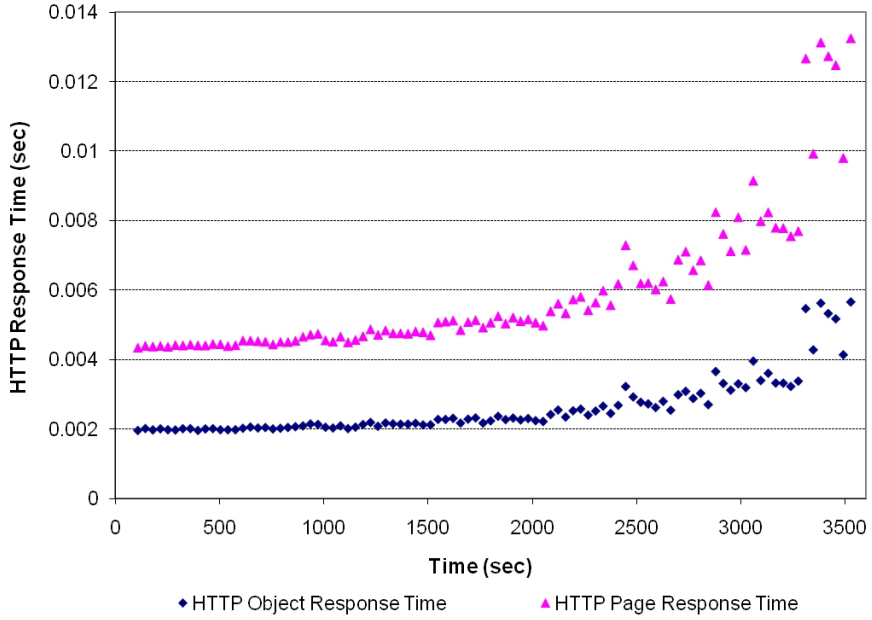


Figure 3.13: HTTP Response Time

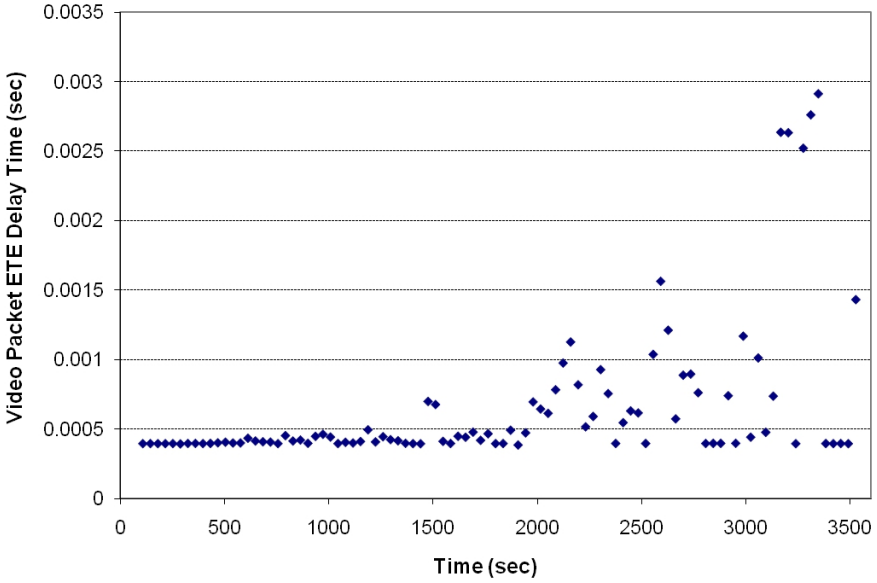


Figure 3.14: Video ETE Delay Time

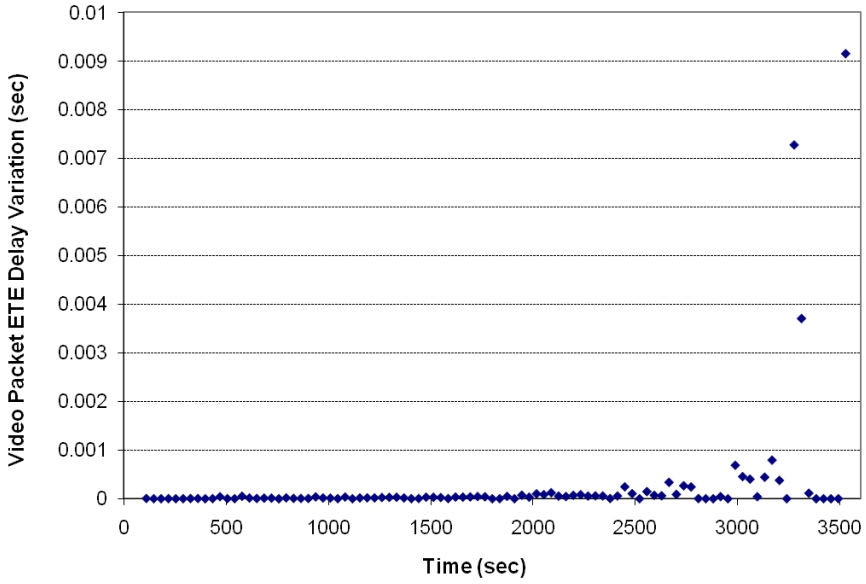


Figure 3.15: Video Packet Delay Variation

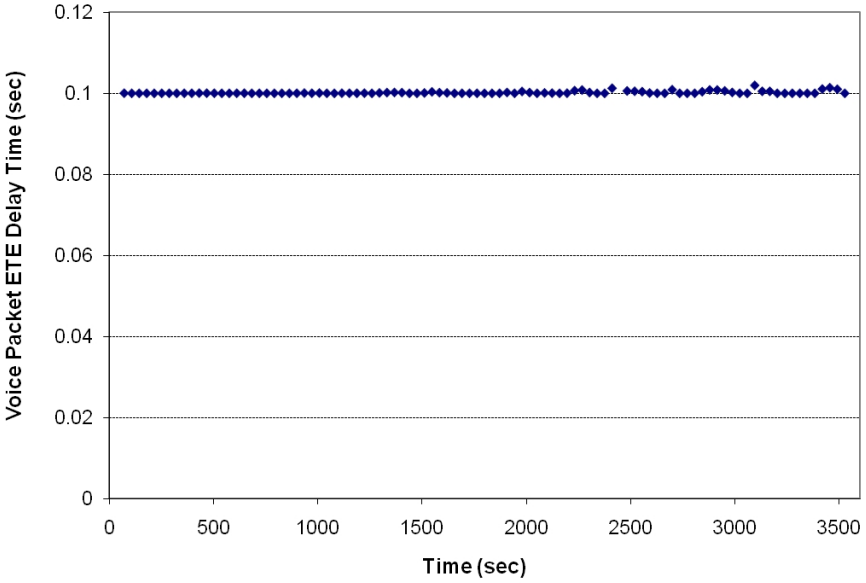


Figure 3.16: Voice Packet ETE Delay Time

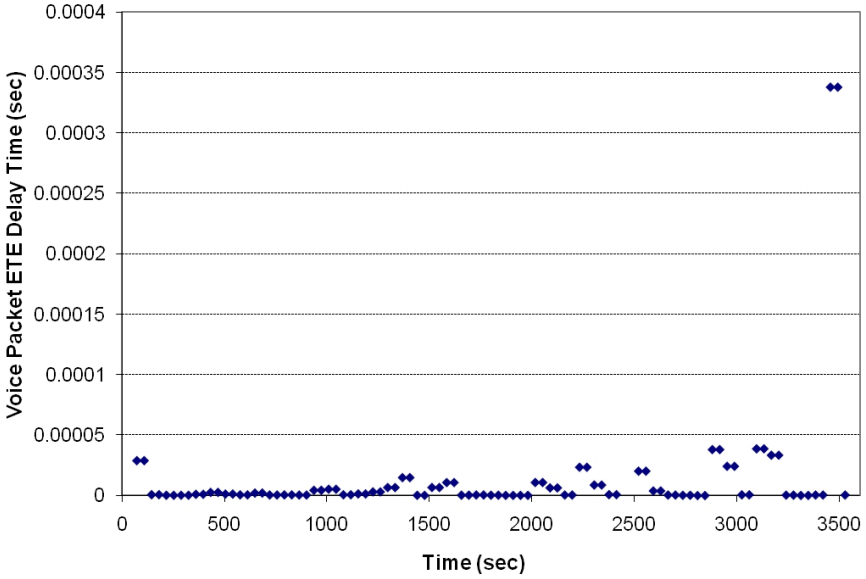


Figure 3.17: Voice Packet Delay Variation

3.5 Consolidated Network Architecture Conclusion

The main purpose of this study was the development of a network test bed architecture using OPNET Modeler to simulate a consolidated network architecture. Test scenarios such as the failover scenario and traffic scalability scenario were developed to baseline the system performance for further investigation in other aspects of the network architecture. The simulation test results show that failover requirements can be fulfilled by implementing redundant systems with a routing protocol enabled. The test results also show that response times for mixed applications degrade quickly after 60% of the link bandwidth has been utilized. These performance statistics were collected to provide valuable insights which will aid the developement of future tactical network architectures.

Future works include QoS study, network topology study, performance study

with other COTS products and cross security domain related scenarios. The network test bed can also be combined with ADNS and RF communications systems to further evaluate ETE performance across the GIG. Furthermore, validation of the simulation results may be required to increase the credibility of the simulation results.

Chapter 4

Automatic Dynamic Resource Management System Architecture

This chapter investigates the automatic dynamic resource management (AutoDRM) system architecture in the context of tactical network environments. AutoDRM system architecture is a framework to efficiently manage shared resources in the tactical network environments without human operator intervention. The system architecture is developed to resolve the resource contention issues and to improve the quality of service in the tactical network environment. An experimental network prototype system consisting of simulated satellite communications and an OPNET system-in-the-loop (SITL) scenario is developed to demonstrate the capability of the AutoDRM system.

4.1 Introduction to AutoDRM

Providing End-To-End (ETE) Quality of Service (QoS) in the Department of Defenses (DoD) Global Information Grid (GIG) network is vital for supporting communication activities in tactical missions [42] [43]. An initial step towards such an

ETE QoS support in the large scale network is to ensure that computing resources in each edge network domain are managed efficiently and in accordance with the GIG architectural framework [38] [44]. Future computing requirement for diverse tactical missions rapidly increases the complexity of the heterogeneous tactical edge networks such as the existing Total Ship Computing Environments (TSCE) [45], upcoming Consolidated Afloat Networks and Enterprise Services (CANES) [9], Command and Control systems (C2), and Command, Control, Communications, Computers, Intelligence, Surveillance and Reconnaissance systems (C4ISR) [46]. Figure 4.1 illustrates an operational view of tactical edge networks1. Tactical systems consist of many computing and networking devices highly integrated within a common network infrastructure. The systems often provide real-time services such as Voice-over-IP (VoIP), streaming video, real-time messaging and other time-sensitive tactical applications that require stringent QoS guarantees with limited computing and networking resources. Each time-constrained application demands resources at different levels in order to achieve various QoS requirements. Without an adequate resource management solution, simultaneous increases in all QoS levels can result in resources contention which inevitably impacts the overall system performance. An automated method to dynamically allocate resources based on the prioritization across multi-dimensions (*e.g.* traffic classes, user precedence..*etc.*) and multi-choices (*e.g.* QoS policies..*etc.*) is needed to address such a technical challenge. The development of the AutoDRM architecture mainly leverages the performance monitoring capability of a network management system (NMS) and policy based QoS capability in the network domain. AutoDRM does not replace the overall network management system for the network domain. It merely serves as a supplemental function to the NMS by providing the capability for efficiently utilizing the resources within the context of a tactical edge network domain.

A significant amount of research in dynamic resource management has been studied in various contexts. Rajkumar et al. [47] [48] developed a QoS-based resource allocation model (QRAM), which establishes an analytical approach to

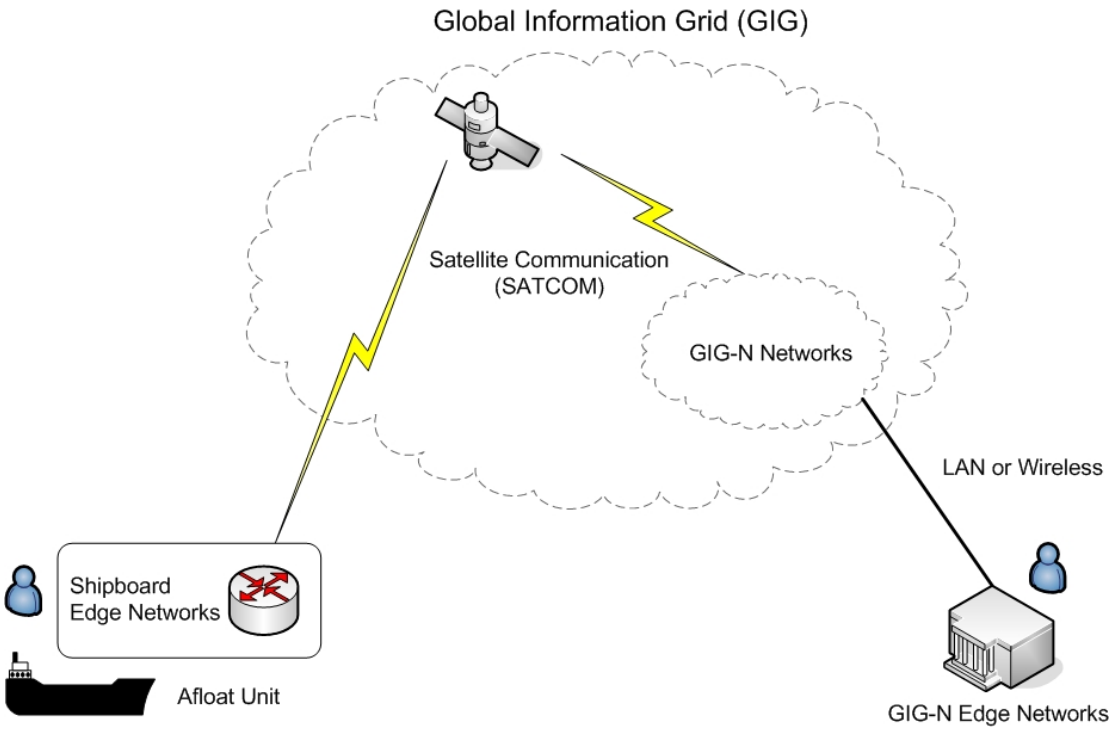


Figure 4.1: Operational View of Tactical Edge Networks

Table 4.1: Definition of the Parameters

Name	**Notation**
Service (or Application)	$\{S_1,S_2,S_3,..,S_n\}$
Shared Resources	$\{r_1,r_2,r_3,..,r_m\}$
Maximum Resources	$\{R_1,R_2,R_3,..,R_m\}$
QoS Requirements	$\{Q_{i1},Q_{i2},Q_{i3},..,Q_{nm}\}$
QoS Achieved	$\{q_{i1},q_{i2},q_{i3},..,q_{nm}\}$

distribute system resources among multiple applications while maximizing the utility function. Harada et al. [49] and Stankovic et al. [50] have proposed adaptive resource allocation methods based on feedback control theories. In the context of shipboard computing environments, Lardieri et al. [46] developed a multi-layered resource management framework in enterprise distributed real-time and embedded (DRE) systems. They primarily focused on managing the dynamics of computing resources in response to mission mode changes and/or resource load changes. Dasarathy et al. [51] developed a CORBAbased multi-layer management framework to manage changes in network resources, work load, and mission requirements at the network layer. Their study described the interactions of four key network QoS components: bandwidth broker, flow provisioner, network performance monitor, and network fault monitor. All of these studies in tactical network environments focused on providing a middleware framework to achieve QoS objectives. This paper focuses on the architectural concept of AutoDRM as well as the development of a network architecture prototype test bed to support the experimental study.

4.2 Dynamic Resource Management Theories

The dynamic resource management problem is generally formulated based on the 0-1 Knapsack problem which is known to be *NP-Hard* [52] [53]. Table 4.1 [47] [48] [52] defines the sets of general parameters in the resource management problem. A system can provide n number of independent services (*e.g.* VoIP, streaming video,

real-time messaging..*etc.*), $n \geq 1$. There are m number of shared resources (*e.g.* processing capacities, queue sizes, network bandwidth..*etc.*), $m \geq 1$. Each service S_i requires a set of shared resources r_j to accomplish its QoS objectives, where $i \in \{1..n\}$ and $j \in \{1..m\}$. A portion of resource r_j allocated to a service S_i is denoted by r_{ij}. Since each service often needs to meet a set of QoS requirements (*e.g.* packet latency, packet loss ratio..*etc.*), Q_{ij} represents a QoS requirement based on a service S_i consuming a shared resource r_j. This is done under the constraint that the total amount of resources is finite such that $\sum_{i=1}^{n} r_{ij} = R_j$ and $\sum_{j=1}^{m} R_j = \mathbf{R}$, where R_j is the maximum amount of each shared resource and $\mathbf{R}$ is the total available resources in the system. Since all of the resource requests may not necessarily be satisfied in a resource constrained environment, an actual achieved QoS level is represented by q_{ij} such that $q_{ij} \in Q_{ij}$. In order to accomplish adequate QoS levels for each service, the following condition must be met.

$$Maximize \ \sum_{j=1}^{m} x_i \cdot q_{ij} \ \ subject\ to \ \ \sum_{j=1}^{m} x_i \cdot r_{ij} \leq R_j$$
$$where \ i = \{1, ..., n\} \quad and \quad x_i = \{0, 1\}$$

Fundamental theories in developing real-time near-optimal heuristics are discussed in [48] [49] [53] [52]. These approaches involve sorting orders in each data set and gradually assigning a portion of each shared resource r_j to each service S_i. When a QoS level q_{ij} satisfies the requirement Q_{ij}, the iterative process to assign resources is halted. Otherwise, it will continue to assign more portion from each resource r_j to each service until the upper limit of that resource R_j has been reached. If a QoS requirement for a service has not been satisfied after a specific resource R_j has been exhausted, a decision to accept the current quality or downgrade resources from other lower priority services is required. The priorities of services are determined by aggregated QoS policies in the network domain. This dynamic process repeats itself until reaching a stable state.

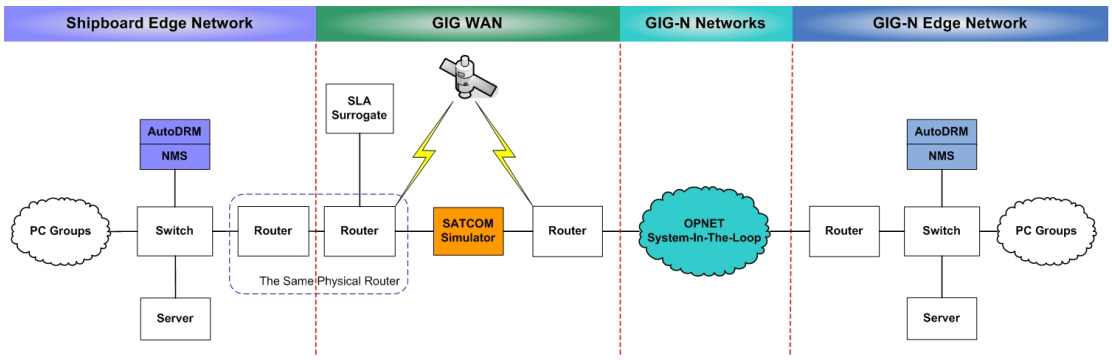


Figure 4.2: End-To-End Network Prototype Test Bed

4.3 Network Architecture

4.3.1 Tactical Edge Network

The current *TSCE* system implements the Navy's open architecture strategy and has achieved a full commercial off-the-shelf (COTS) solution. Inspired by the concept of Service Oriented Architecture (SOA), the upcoming *CANES* system aims to perform more tactical functions while reducing the physical footprint by consolidating the network architecture across multiple security enclaves [9]. *TSCE* and *CANES* are distributed real-time enterprise systems typically deployed on tactical afloat units. Both systems provide various services for the user communities in the tactical edge networks. Since these services are deployed in a common network, they often compete for shared resources. Examples of these shared resources in the systems include processor units, memory devices, storage disks, networking devices, security devices, and others that have finite constraints. To ensure that each service is performed at adequate QoS levels, the AutoDRM function is required in these edge networks.

4.3.2 End-To-End Network Prototype Test Bed

An experimental ETE network prototype test bed as shown in Figure 4.2 was developed in order to host the AutoDRM system. In order to maintain consistency with the operational view in Figure 4.1, the prototype network includes four representative network domains: Shipboard Edge network, GIG Wide Area Network (WAN), GIG-N networks, and GIG-N edge network. The shipboard edge network is simulated by a Local Area Network (LAN) group consisting of several PCs, a network switch, and a gateway router. For the purpose of simplifying the prototype test bed development, the router between the shipboard edge network and the GIG WAN is shared. Two Virtual LANs (VLANs) were configured to represent the respective network domains. In practice, a gateway router from the shipboard edge network is connected to another router residing in the GIG WAN domain. A satellite communications simulator configured with the same settings as in [8], is used to simulate the network characteristics across a long latency SATCOM link. An OPNET System-In-The-Loop [22] scenario was developed to simulate the latency and packet loss rate in the GIG-N networks. The GIG-N edge network is assumed to mirror the shipboard edge network in the prototype test bed.

4.4 AutoDRM Software Architecture

AutoDRM interfaces with a NMS to provide dynamic resource management capability for the shipboard edge networks. Figure 4.3 illustrates the QoS concept in the AutoDRM architecture. In the context of QoS, network devices such as routers and switches can be conceptually represented with a network packet classifier, various queues, and a packet scheduler. Depending on the priority tag marked in each packet header and number of traffic classes, incoming packets are examined and classified into different outgoing queues. Departures of the outgoing packets are scheduled using a queuing technique (*e.g.* priority queuing, weighted

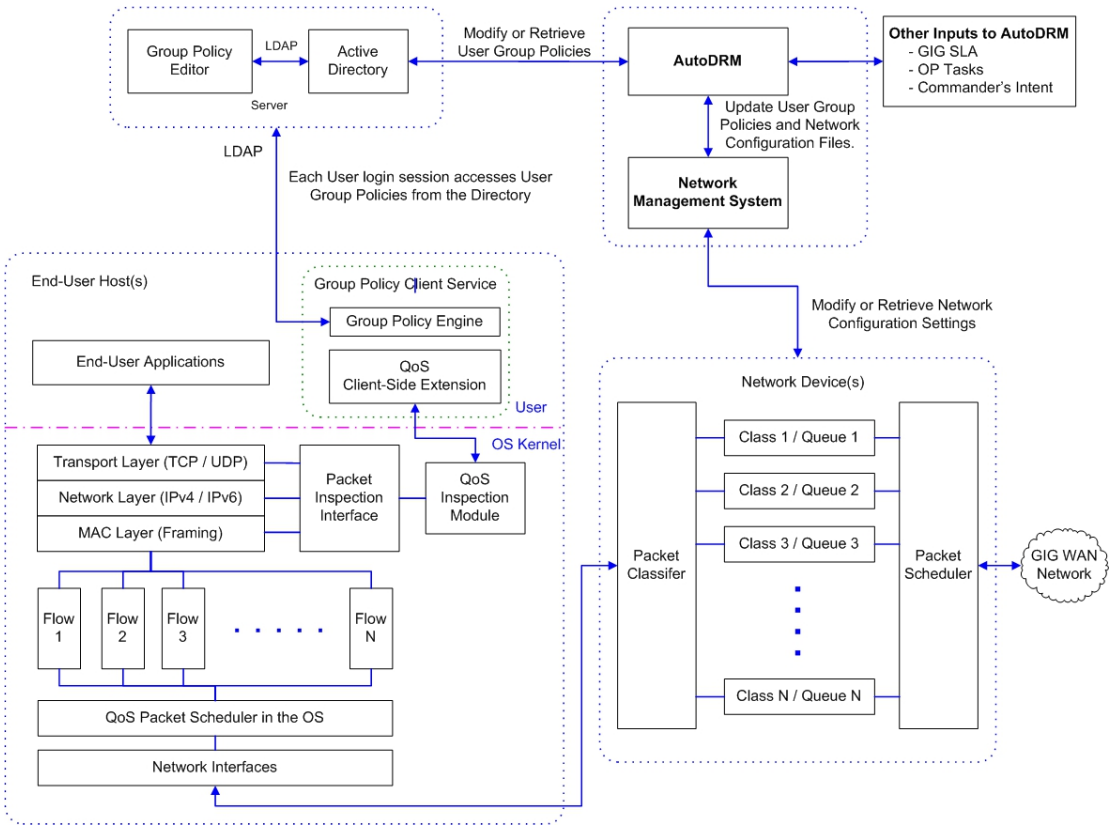


Figure 4.3: AutoDRM QoS Architectural Concept

fair queuing..*etc.*). QoS policies using differentiated service (DiffServ) [54] [55] can be configured in each network device to control the behavior of these QoS related components.

As shown in Figure 4.3, an end-user host system (*i.e.* PC, Server..*etc.*) executes heterogeneous user applications while utilizing services provided by a common set of network protocol stacks (*i.e.* TCP/IP) in the operating system kernel [56]. AutoDRM can retrieve and modify user group policy objects (GPO) which are stored in the Active Directory of the network domain server. Each GPO defines the QoS parameters such as data throttle rate and Differentiated Services Code Point (DSCP) value at the application level on a per-user or per-computer basis. Upon authentication of a user login session, the group policy client service

retrieves user group policies from the Active Directory server. Depending on the user's privilege or the IP address of a host system, GPO enforces the behavior of the network traffic generation from each application. More technical details of this policy-based QoS architecture is discussed in [56]. Based on the dynamics of the network performance measures, AutoDRM utilizes the technology by remotely updating the GPOs via external scripts.

To accomplish the QoS objectives in AutoDRM, Figure 4.4 illustrates functional components of AutoDRM which includes Graphical User Interface (GUI), remote interface, several input translators, resource negotiator, performance monitor, and resource allocator. The standalone GUI provides a user friendly interface for the system administrator and mission planning operator to perform initial setup and any subsequent system-level update. The remote user interface provides a convenient method to remotely control the AutoDRM system. The following subsections describe detailed functions of the translators, resource negotiator, performance monitor, and resource allocator.

4.4.1 Input Translator

Several built-in translators are required for AutoDRM in order to parse and translate structured documents containing Commander's Intent [57] [58], operational task orders (OP Task), and GIG Service Level Agreement (SLA) [59]. Commander's Intent is a concise expression for the purpose of the operation and the desired end state that serves as the initial impetus for the mission planning process [58]. Commander's Intent may also include acceptable levels of risk for the operation. The operational task orders are typically mission specific and are the derivatives of communication plans for tactical units. The input parameters from GIG SLA can be derived from Service Level Specifications (SLS) which is a subset of GIG SLA. The SLS defines the communication parameters for the edge user communities to subscribe to the GIG networks. All of the input parameters are assumed

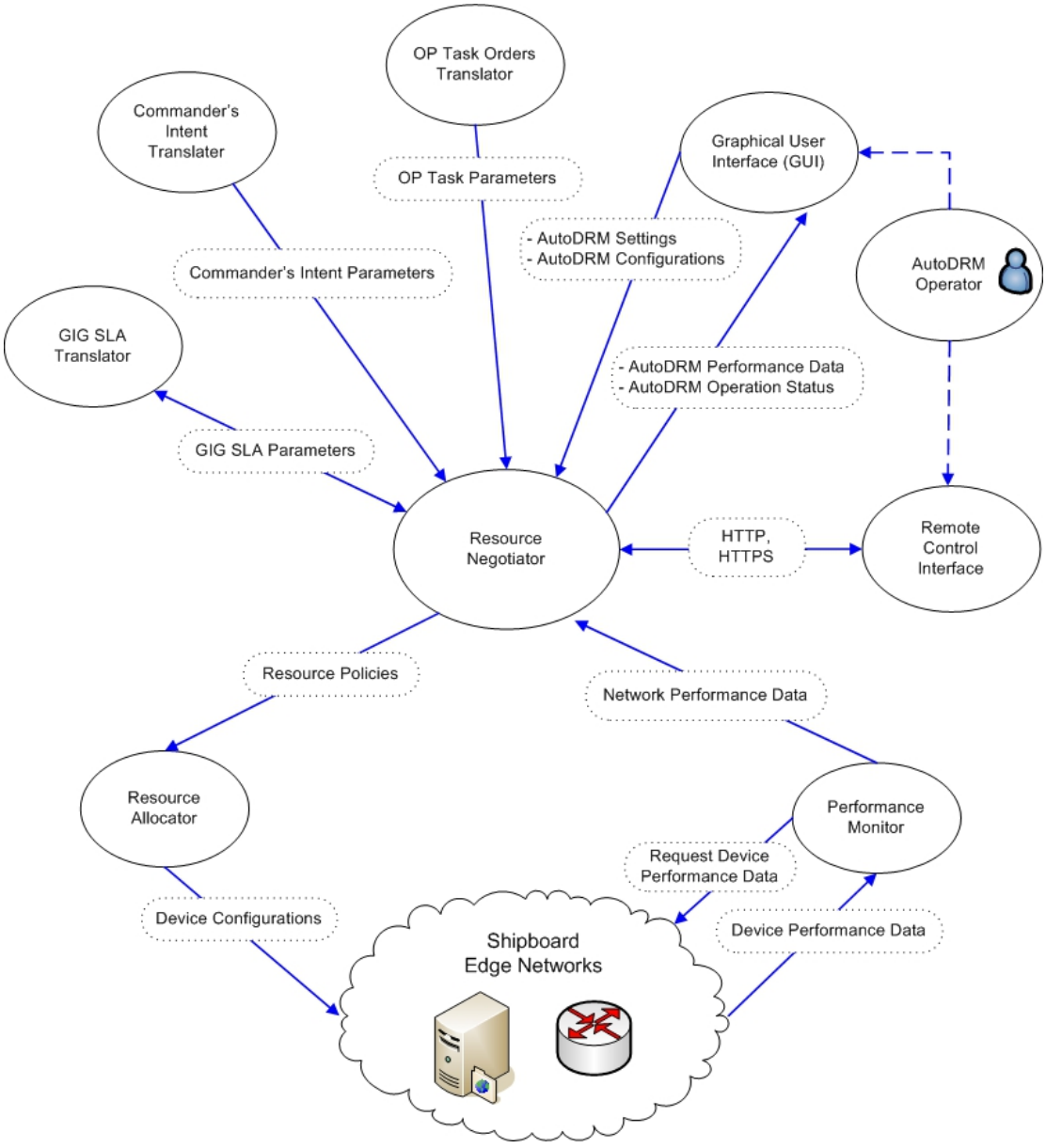


Figure 4.4: AutoDRM Functional Block Diagram

to be contained in structured documents (*e.g.* XML). Figure 4.5 illustrates an example of the parsed and translated Commander's Intent which contains network related parameters for AutoDRM's inputs[1] . The operational task orders and GIG SLA are assumed to be formatted in a similar fashion. A parsing function first parses through the structured documents to extract a set of key communication attributes. A translating function then converts the extracted attributes into measurable parameters and stores the information in a translator database.

4.4.2 Resource Negotiator

The core function of AutoDRM is the resource negotiator, which determines the resource allocation based on real-time network performance measurements and notifications from the network management system. The resource negotiator exploits a real-time near-optimal heuristic algorithm to determine the resource assignments. Figure 4.6 depicts the functional block diagram of the resource negotiator. There are five fundamental functions in the resource negotiator: data fetch function, parameters mapping, sorting based on prioritization, resource assignments, and results transformation.

The data fetch function retrieves translated parameters such as bandwidth, packet delay, packet loss and other measurable network parameters from the translator database and the performance monitor database. The mapping function maps input parameters into a multidimensional array data structure where each array represents a services set, a resources set, a QoS set and other required data sets. The sorting function uses sorting algorithms (*e.g.* quick sort, binary sort..*etc.*) to efficiently sort each data set based on the prioritization of each item within the data set. The real-time near-optimal heuristic algorithm takes the mapped data set and gradually assigns resources to each task in order to minimize the error between the requested QoS levels and the actual QoS levels [49]. The results from the algorithm are transformed in the results transformation

[1] In this research discussion, assumption is made such that network related parameters can be extracted from Commander's Intenet.

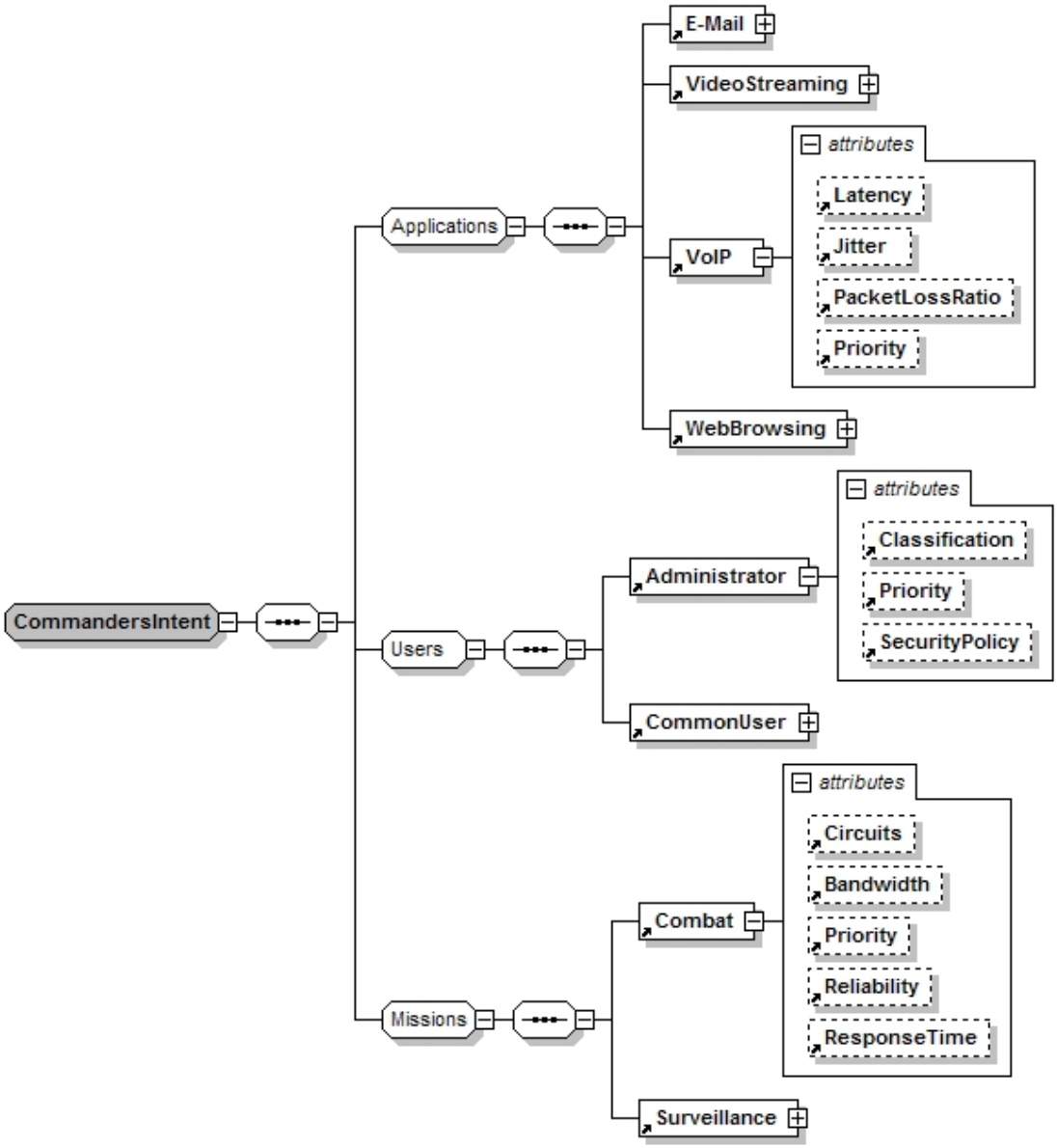


Figure 4.5: An Example of Translated Commander's Intent

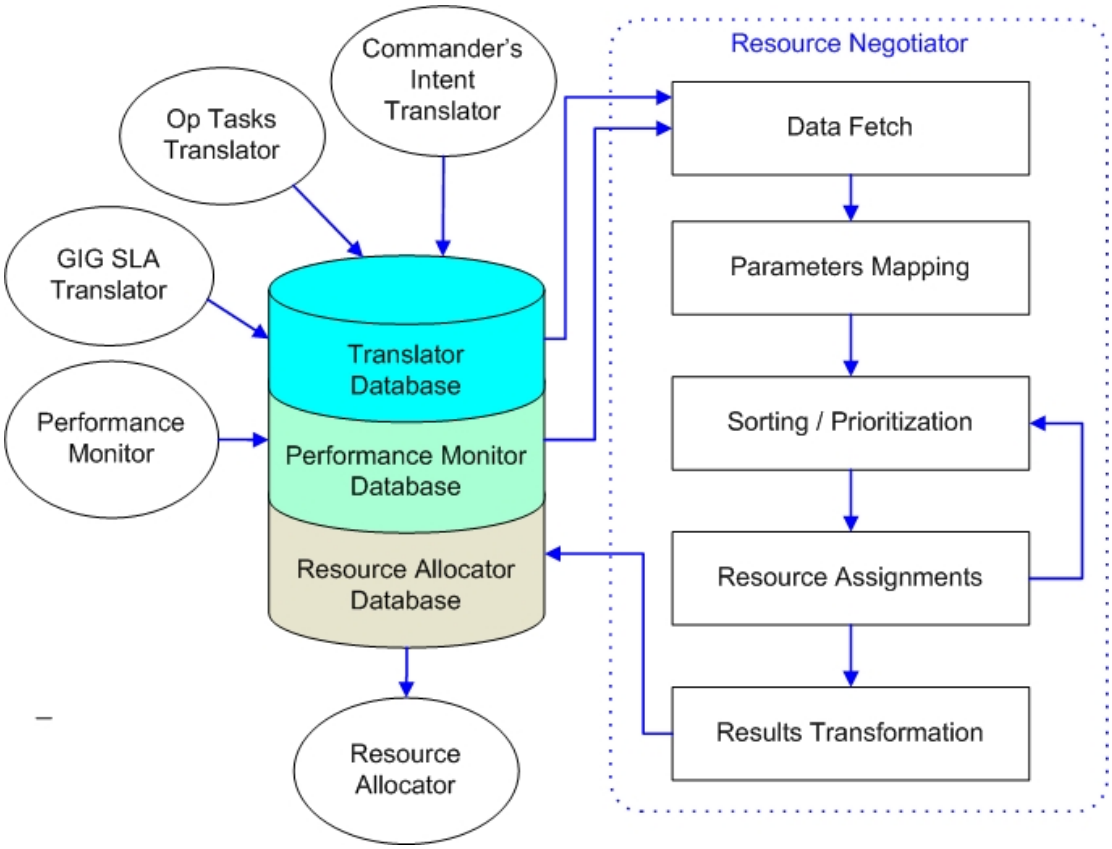


Figure 4.6: AutoDRM Resource Negotiator Functional Block Diagram

function. The main objective of the results transformation function is to convert the acceptable resource assignments into actual user group policies and/or network configuration formats that can be used to regulate the QoS behavior of the devices in the edge network domain. For storage efficiency, a centralized database is shared among translators, a performance monitor and the resource allocator. A scheduler in the resource allocator will then update the GPOs in the network domain using external scripting methods and the network device configurations will be updated via the network management system.

4.4.3 Performance Monitor

The performance monitor in AutoDRM exploits the rich set of network monitoring features in NMS to collect real-time network performance measurements through the Simple Network Management Protocol (SNMP) [60]. The major categories of these features include service polling, data collection, events and notifications [61]. By leveraging the monitoring features in NMS, the performance monitor function constantly retrieves network performance information from the NMS into the performance monitor database in AutoDRM. In addition, the performance monitor also interfaces with the Active Directory server containing the GPOs. Modification of any GPO is updated in the database as well. Database updates are monitored by the resource negotiator. In the event of any network performance change (*e.g.* bandwidth saturation, increase of packet loss ratio, increase in average packet delay..*etc.*), the resource negotiator re-evaluates the QoS requirements and re-computes resource assignments to mitigate the possibility of resource contention in the network domain. The performance monitor makes use of common remote scripting tools and the development toolkits provided by the NMS for developing the required software interfaces.

4.4.4 Resource Allocator

The resource allocator is primarily responsible for scheduling the configuration updates in the network domain. Similar to the performance monitor, the resource allocator in AutoDRM also interfaces with the NMS to modify the network configurations in each individual network device as well as updating user group policies in the network domains by remote scripting methods. In the event of degraded network performance, the resource negotiator responds with an updated resource allocation and stores the necessary changes into the shared resource allocator database. The resource negotiator sends out notifications to the resource allocator which then updates the configurations of affected network devices and the host systems.

4.5 AutoDRM Experimental Setup

To investigate the effectiveness of the AutoDRM system, a relevant use case consisting of three test scenarios was developed in the experimental setup. The use case assumes a tactical community user in the shipboard edge network is receiving a mission critical streaming video service from a video server residing in the GIG-N edge network. The real-time video streaming service is assumed to be using User Datagram Protocol (UDP) as its transport layer protocol. The network traffic flow representing a streaming video service is simulated as an UDP flow using Distributed Internet Traffic Generator (DITG) [62]. A network management system deployed in the shipboard edge network monitors real-time network performance. The network management system is pre-configured to generate asynchronous notifications (*i.e.* SNMP traps) to the AutoDRM system when certain network thresholds are met (*i.e.* packet delay $\geq$ 5 seconds, packet loss $\geq$ 20 pkts). Upon receiving the notifications, AutoDRM determines that the streaming video service has the highest priority among other background traffic flows. Thus, the system provides preferential service to the streaming video flow by allocating more

resources to it. The AutoDRM resource allocator archives this goal by updating the policy-based QoS parameters in the Active Directory as well as updating QoS policies in the configurations of the router and the switch within the edge network domain.

For the purpose of performance metrics comparison, three test scenarios with different configurations were performed. Each test scenario had different network traffic conditions. The first test scenario consists of a single streaming video traffic flow without any background network traffic load. This test scenario established a baseline test result. Two test scenarios consisting of a streaming video traffic flow with mixed background traffic flows were also performed. One test scenario was with AutoDRM enabled and another test scenario was with AutoDRM disabled. The background network traffic contains nine heterogeneous UDP and TCP flows. Since the performance of the mission critical streaming video was under investigation, key QoS performance metrics of interests in this experimental setup include throughput, packet delay, packet loss, and jitter measurements.

4.6 AutoDRM Experimental Results and Discussion

The experimental results are primarily focused on evaluating the functionality and effectiveness of the AutoDRM system by collecting the QoS performance metrics of the streaming video service over a period of five minutes. Several key QoS performance measurements including throughput results shown in Figure 4.7, packet delay results shown in Figure 4.8, packet loss results shown in Figure 4.9, and jitter results shown in Figure 4.10 are collected. The throughput results in Figure 4.7 compare the primary streaming video flow and the background traffic which is the aggregate of other data flows. When AutoDRM is enabled during the test run, the throughput of the primary streaming video flow is improved by

trading the performance of other data flows as their aggregated throughput are shown to be decreased.

Results in Figure 4.8 show that the baseline end-to-end packet delay for the streaming video service without any background network traffic is about 0.8 seconds. When the streaming video service is running with background network traffic and with AutoDRM system disabled, the end-to-end packet delay quickly surges to more than 5 seconds after 20 seconds of run time. In Figure 4.8(a), the packet delay stays at 5 seconds throughout the remaining duration of this test scenario. With AutoDRM system enabled, the network management system generates asynchronous notifications when the packet delay exceeds 5 seconds threshold at 20 seconds of run time. In Figure 4.8(b), the packet delay for the streaming video flow starts to decrease after 130 seconds of run time. The packet delay becomes stable after 250 seconds of run time. The stabilization time for packet delay is approximately 120 seconds in this test.

As illustrated from results in Figure 4.9, baseline packet loss for the streaming video service is less than 10 packets at any given time interval. With AutoDRM system disabled as shown in Figure 4.9(a), the streaming video flow running with background network traffic results in packet loss ranging from 20 to 50 packets after about 10 seconds of run time. With AutoDRM system enabled, the network management system generates asynchronous notifications when the packet loss exceeds 20 packets threshold at 10 seconds of run time. In Figure 4.9(b), the streaming video flow running with background network traffic shows performance improvement in terms of packet loss to less than 10 packets at 130 seconds of run time.

Plots in Figure 4.10 show the jitter measurements. Minimizing jitter is very essential to the performance of the streaming video. Figure 4.10(a) serves as the baseline for the aggregated data flows which has 150 milliseconds to 470 milliseconds jitter measurements when AutoDRM system is disabled. In contrast, Figure 4.10(b) shows improved jitter performance when AutoDRM system is enabled.

From these test results, it can be concluded that AutoDRM system improves

packet delay by about 60%, reduces packet loss by about 66%, increases the throughput by 40%, and improves jitter by about 30% in this experiemental setup.

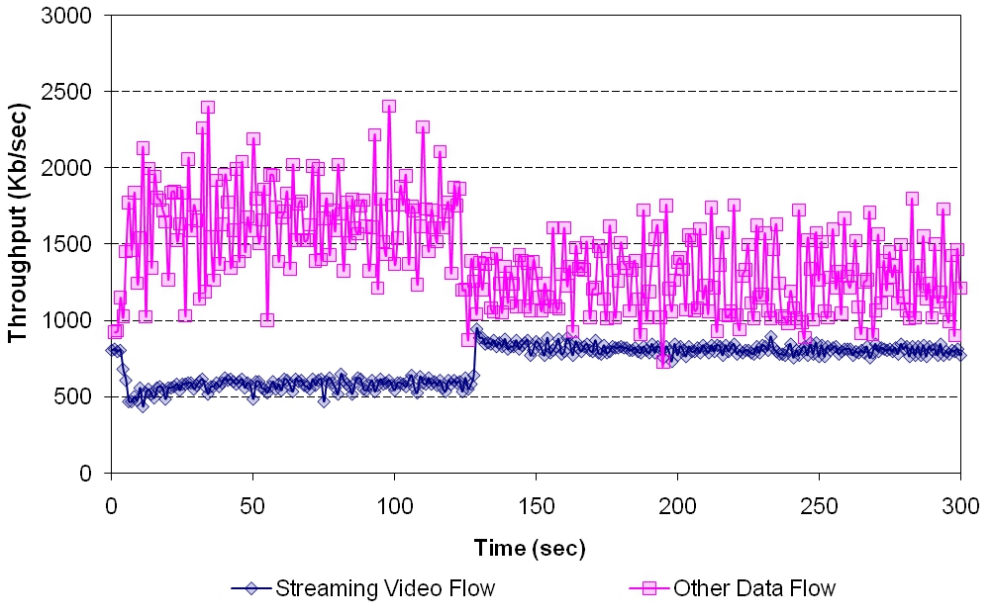
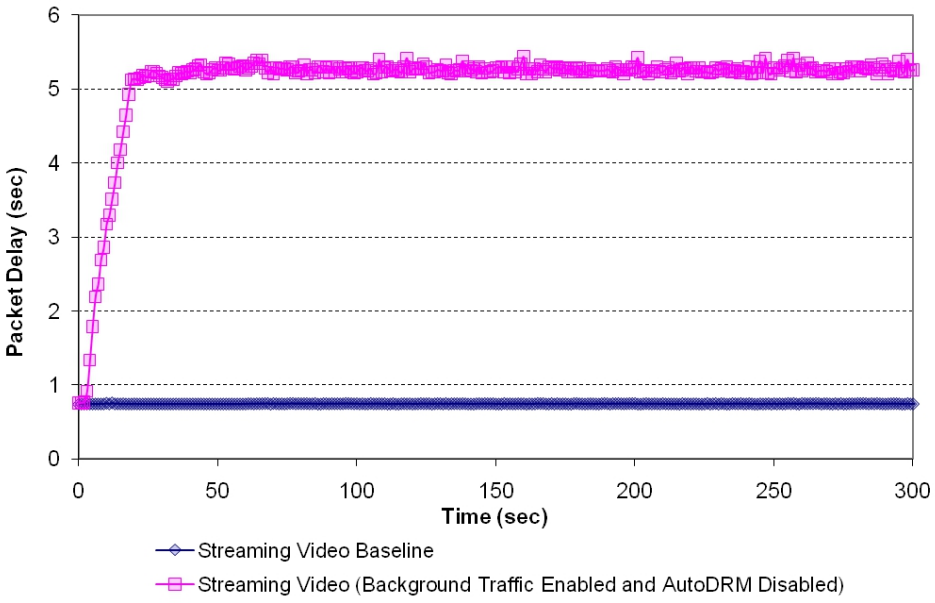
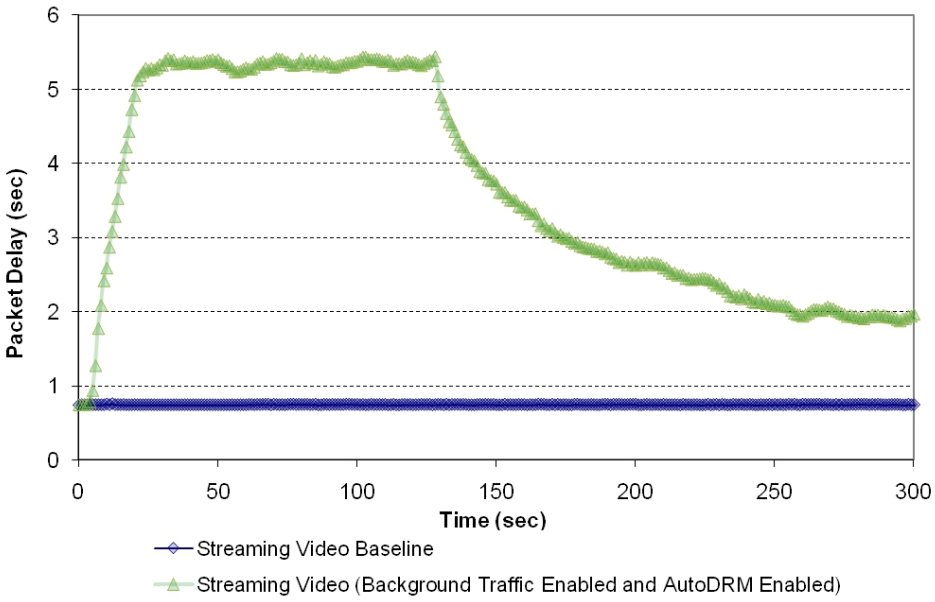


Figure 4.7: AutoDRM Throughput Results

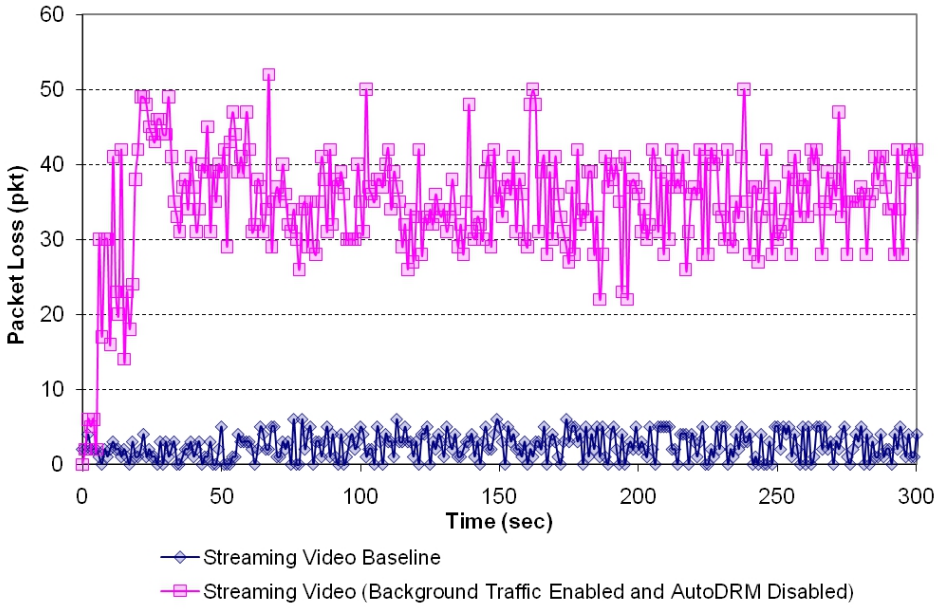


(a) Packet Delay (sec) – AutoDRM Disabled

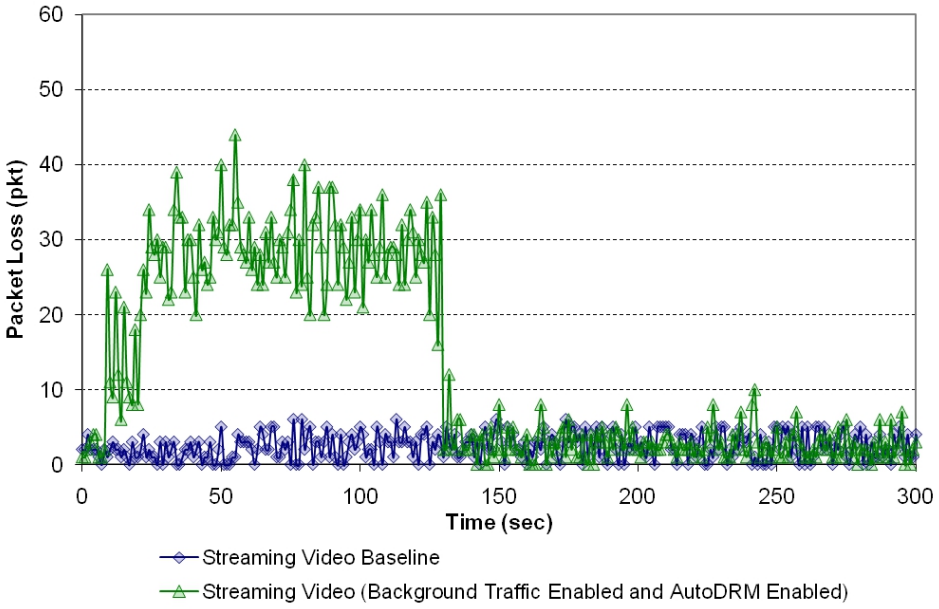


(b) Packet Delay (sec) – AutoDRM Enabled

Figure 4.8: AutoDRM Packet Delay Results

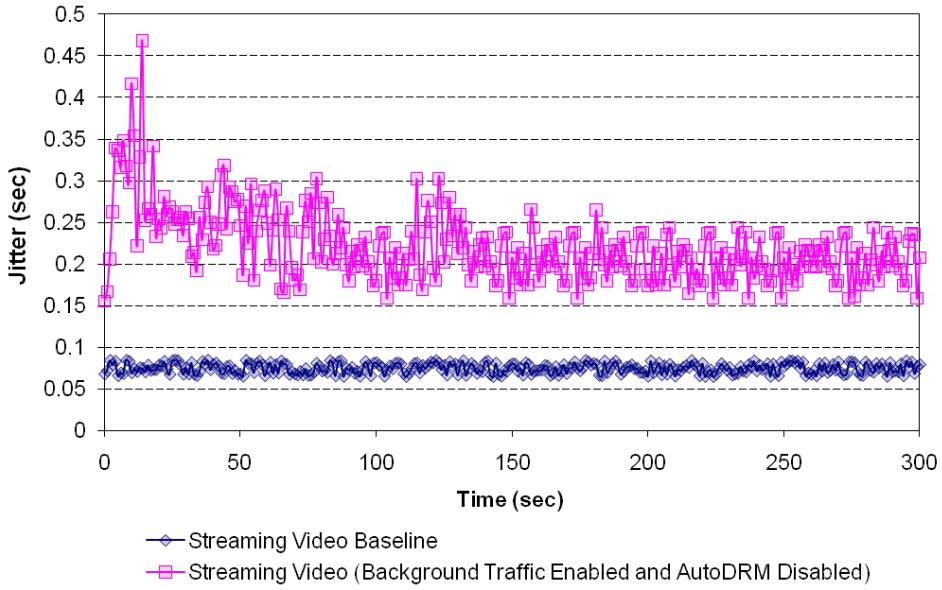


(a) Packet Loss (pkt) – AutoDRM Disabled

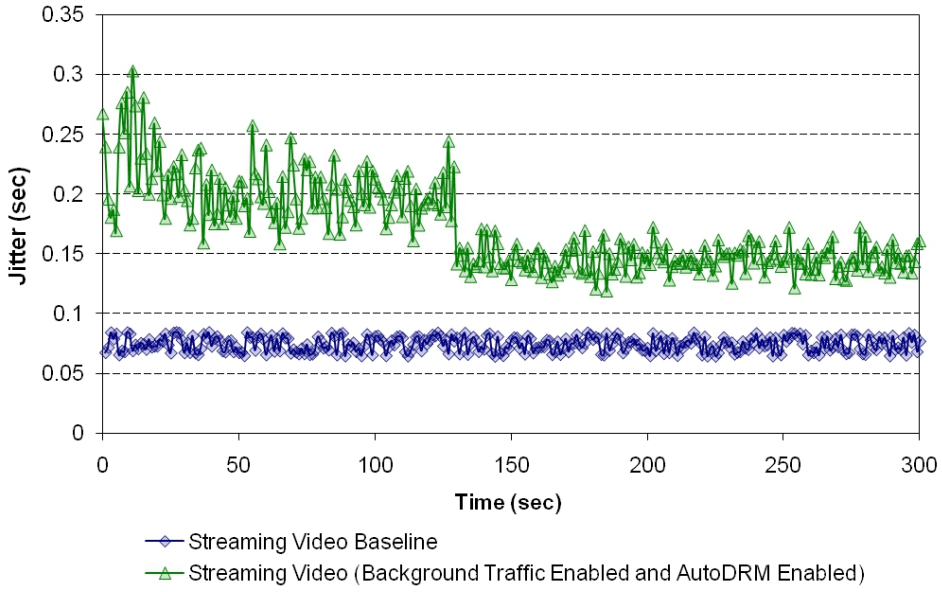


(b) Packet Loss (pkt) – AutoDRM Enabled

Figure 4.9: AutoDRM Packet Loss Results



(a) Jitter (sec) – AutoDRM Disabled



(b) Jitter (sec) – AutoDRM Enabled

Figure 4.10: AutoDRM Jitter Results

4.7 AutoDRM Conclusion

The framework of Automatic Dynamic Resource Management architecture has been developed in this experimental study. An end-to-end network prototype test bed consisting of real network devices, a simulated SATCOM link, and an OPNET SITL scenario was also developed to host the AutoDRM system. Three test scenarios representing three different network traffic conditions were executed. The test results demonstrate improved QoS performance in terms of packet delay and packet loss. The results from these scenarios indicate that AutoDRM system can be a vital function to dynamically improve the network QoS performance.

Developing a practical approach for providing ETE QoS management services through automatic and dynamic mechanisms has attracted interest from the user communities. To achieve the objective, it is important to recognize the necessity to incorporate network performance and connectivity data with service request information associated with the application and the specific services available or supported by the network. Future research efforts include using the developed end-to-end prototype test bed for exploring any dynamic QoS mechanism objective specifically recognizing the need to derive network performance from the Commander's Intent.

Chapter 5

Reliable Data Aggregation and Dissemination Framework

This chapter investigates a reliable data aggregation and dissemination framework. The framework takes a hybrid approach of combining disruption tolerant networking advantages and an adaptive sensor data aggregation method to ensure reliable data delivery. An experimental prototype system architecture is developed and implemented to demonstrate the capabilities of the proposed data aggregation and dissemination framework. A relevant demonstration scenario based on an example data aggregation map is developed for performing system evaluation. The proposed framework can be a promising solution beneficial to current and future system-level design of tactical network architectures.

5.1 Introduction to Data Aggregation and Dissemination

Data aggregation and data dissemination techniques have been investigated in the area of wireless sensor networks [63] [64]. Previous efforts focus on maintaining energy efficiency of the wireless sensor networks when applying context-based data

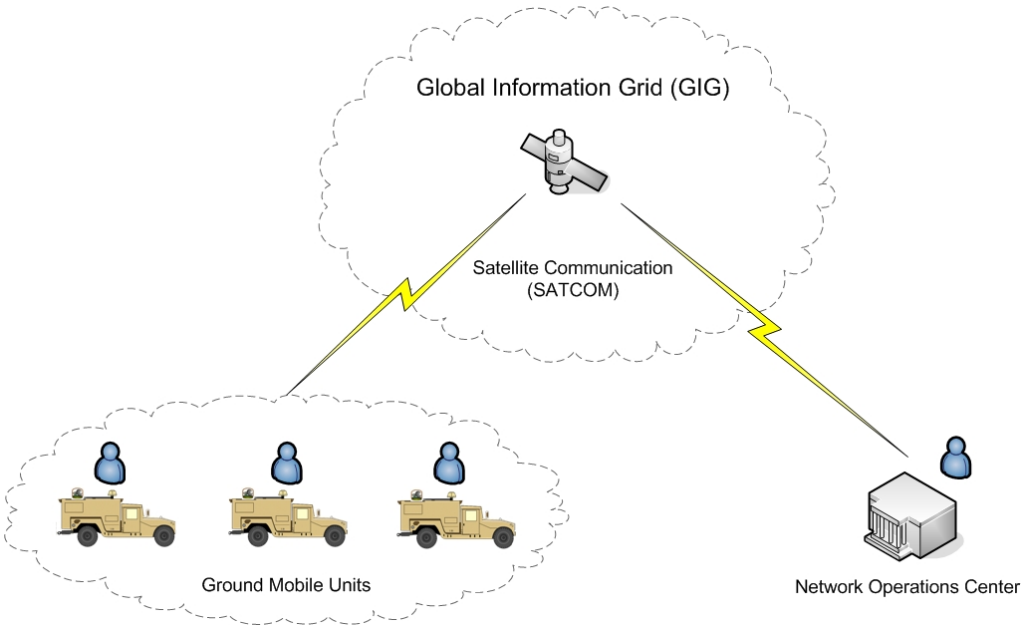


Figure 5.1: Operational View of Tactical Networks

aggregation and data dissemination techniques. While energy efficiency is an important issue in wireless sensor networks, previous efforts typically did not consider the disruptive nature of the network uplinks commonly found in tactical network architectures. The general assumption is that each sensor base station also serves as the final sink node in the network. In practice, a tactical network often includes not only local sensor components but also multiple Wide-Area Network (WAN) uplinks (*e.g.* SATCOM, Wi-Fi, WiMAX..*etc.*) for delivering meaningful messages to a remote network operations center running Command and Control (*C2*) applications. The proposed framework aims to address this shortfall in previous efforts by combining disruption tolerant networking capabilities [65] with an adaptive sensor aggregation method in order to achieve reliable end-to-end data delivery in tactical networks.

Wireless sensor networks have been designed for various applications including military surveillance and environmental monitoring [63]. This study assumes

that a wireless sensor network collects real-time health status of a ground mobile unit. By applying the adaptive sensor data aggregation method, the large amount of raw sensor data can be reduced and transformed into a smaller number of alert messages that reliably identify the health status of each mobile unit. Each ground mobile unit is equipped with one or more WAN uplinks for disseminating alert messages locally or to a *C2* application at a remote network operations center. A disruption tolerant networking approach is then utilized for delivering the aggregated alert messages to the network operations center. This reliable data aggregation and dissemination framework can be integrated with many tactical network architectures.

5.2 System Requirements

In the context of tactical networks, the design of a reliable data aggregation and dissemination framework must include several key requirements:

- *Fault Tolerant and Reliable* - The framework must be able to reliably infer the current status of each individual system under monitoring based on sensor data aggregation. Fault tolerant capabilities must be built-in to the framework in order to support data dissemination across the intermittent network connectivity. An example of a fault tolerant capability would be data retransmission upon recovery from network link disruption.

- *Interoperable and Flexible* - The framework must be interoperable with the GIG and other tactical communications systems to ensure that information can be sent to and from anywhere at any time. In order to minimize the system integration cost, the framework must be flexible, highly configurable, and easily integrated with existing *C2* applications.

- *Maintainable* - The system must provide a management function with a user friendly interface for maintenance personnel to enable continuous and rapid diagnostics for detecting problems and performing repairs [66].

- *Secure* - Ensuring integrity, confidentiality, and authenticity of the data is necessary in tactical networks. In tactical operations, the authentication of the sensitive data must always be verified first and the delivery of sensitive data must be protected at all times. Modified or falsified data result in miscommunication which can potentially jeopardize the personnel involved in tactical operations.

This study focuses on *Fault Tolerant and Reliable*, and *Interoperable and Flexible* aspects of the system requirements. The *Maintainable* aspect of the study involves the development of a system management solution which is outside the scope of this research effort. Nevertheless, a graphical user interface (GUI) application was implemented in the experimental prototype system for the purpose of system evaluation. This study also assumes that the system operates in a physically *Secure* environment, the security features of the framework are left for future investigation.

5.3 Disruption Tolerant Network

Tactical networks frequently encounter various forms of network disruptions due to energy resources (*e.g.* power outages), interferences (*e.g.* noise or radio frequency denied environments), mobility (*e.g.* out-of-range radio signals), or environmental hostility (*e.g.* attacks). These networked systems lack continuous connectivity which causes some technical challenges in reliable data delivery. Disruption Tolerant Networking (DTN), or a similar effort known as Delay Tolerant Networking, is an approach to support end-to-end reliability by addressing disruptions in networks. The DTN architecture takes a store-and-forward approach reminiscent of how an electronic mail system works [65]. Since an end-to-end path between the source and destination is not necessarily guaranteed for the duration of a communication session, the DTN architecture exploits opportunistic routing where data is essentially stored by the intermediate nodes collaborating to forward packets reliably [67].

In the DTN architecture, the bundle protocol [68] is utilized to form an overlay network that employs persistent storage in all intermediate nodes to alleviate network interruption problems and for its store-and-forward function. This overlay network provides transfer of reliable delivery responsibility (*i.e.* custody transfer), optional end-to-end acknowledgment, and several diagnostic and management capabilities [65]. The bundle protocol packages a unit of application data along with any required control information into a "bundle" which is essentially a virtual message analogous to an e-mail message. A DTN node running an instance of the bundle protocol can then apply rules or policies on how to handle the forwarding of each received bundle.

Since the DTN architecture creates an overlay network that focuses on delivering virtual messages rather than packet switching, one key advantage of the DTN architecture is that it can be implemented at any Open System Interconnection (OSI) layer to meet any application-specific requirement. It can be implemented at the application layer, as a middleware service to an application on top of the traditional TCP/IP protocols, or as a replacement for TCP/IP at the transport and the network layer respectively. This advantage offers tremendous flexibility and customization opportunities depending on the specific applications and the operating environment. The proposed data aggregation and dissemination framework exploits DTN as a middleware service between application and transport layers.

5.4 Data Aggregation and Dissemination Framework

Both data aggregation and data dissemination functions are two common communications patterns supporting tactical operations [69]. In data aggregation, multiple source nodes must aggregate data streams at a single sink node known

as the aggregation node[1] . Data aggregation is a common communications pattern found in wireless sensor networks as multiple sensor nodes stream data to a base station in the network cluster. In data dissemination, a single source node known as the dissemination node streams data to one or more sink nodes. This is analogous to a base station in a wireless sensor network sharing information with other base stations in neighboring networks or a remote higher-level management entity. In the tactical scenarios from [70] and [71], the base stations are mobile nodes which essentially form a mobile ad hoc network that often disseminate data through various network uplinks to neighboring nodes or a remote network operations center. Communications patterns for data aggregation and data dissemination are similar in that they both have a single central node and several other nodes, but differ in the direction of data transfer. In order to control the data transfer, rules or policies must be applied to both functions to satisfy application-specific requirements. In the reliable data aggregation and dissemination framework, an adaptive sensor aggregation method is developed by applying policies to infer alert messages from raw sensor data. The aggregated alert messages are then disseminated upstream using the reliable data delivery features of the DTN architecture described in Section 5.3.

5.5 Prototype System Architecture

An experimental prototype system architecture was developed and implemented in order to evaluate the proposed reliable data aggregation and dissemination framework. Development of the prototype system architecture follows the Department of Defense (DoD) Open Architecture strategy which utilizes commercial off-the-shelf (COTS) hardware and software products. As illustrated in Figure 5.2, the prototype system architecture represents an end-to-end tactical network that includes two key network entities: a mobile node and a network operations

[1] In the context of this research discussion, data aggregation *node* refers to a physical node/device in a network, whereas a data aggregation *point* refers to a data collection point in the software process.

Algorithm 1 Adaptive Sensor Data Aggregation Method

Input: A_i, L_i, T_i, *timestamp*
Output: M_j

repeat
  for $i = 1$ to n **do**
    {Sensor Level:}
    Get values of $\{A_i, L_i, T_i\}_{timestamp}$;
    {Threshold Level:}
    Compare values A_i against acceleration threshold;
    Compare values L_i against light intensity threshold;
    Compare values T_i against temperature threshold;
    {Duration Level:}
    Initialize timers for tracking each duration;
    if A_i, L_i, or T_i = thresholds **then**
      Track timers for "at threshold" duration;
    else if A_i, L_i, or $T_i >$ thresholds **then**
      Track timers for "exceeded threshold" duration;
    else
      Track timers for "cleared" duration;
    end if
    {Alert Level:}
    if Duration timers = Triggering conditions **then**
      Trigger alerts;
    else
      Retire alerts;
    end if
    {Combination Level:}
    if Similar types of alerts received **and** $M_j \in \mathbb{M}$ **then**
      Combine alerts into a single M_j;
      Return M_j;
    else
      Return M_j;
    end if
  end for
until *Program Termination*

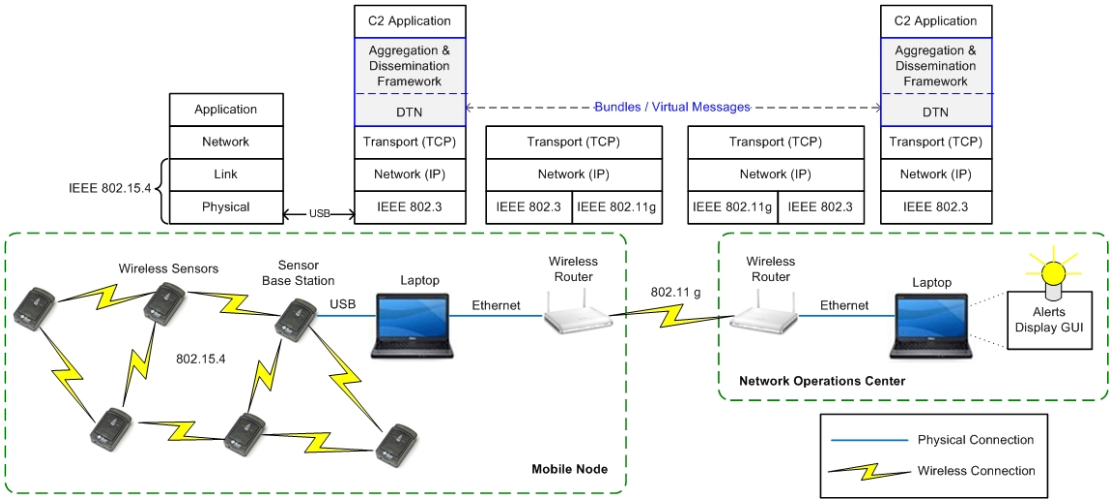


Figure 5.2: Experimental Prototype System Architecture

center. Generally, each network entity has its own system infrastructure. In tactical scenarios, a mobile node could be a logistics vehicle, a combat vehicle, an afloat unit, or other type of mobile unit. The network operations center is where the *C2* applications such as situational awareness system management solutions are deployed for monitoring real-time information from the battlefield. From a networking perspective, all source nodes must directly or indirectly share information with the network operations center which is generally considered as a major sink node in a tactical network.

In the experimental prototype system architecture, a wireless router and a laptop computer are used to represent a simplified version of the network operations center. The mobile node comprises a wireless router, a laptop computer, and several SunSPOT [72] sensor nodes that form a wireless sensor network. For the ease of lab evaluation, this experimental study assumes that IEEE 802.11g is the WAN connection between the mobile node and the network operations center. Each wireless router has an IEEE 802.11g access point and several IEEE 802.3 Ethernet ports for wired connections.

In the mobile node, a wireless sensor network is implemented with several IEEE 802.15.4 compliant SunSPOT sensor nodes. One of the SunSPOT sensor

nodes acts as a sensor base station which connects to the laptop computer through its USB 2.0 interface. Raw sensor data collected at each SunSPOT sensor node are routed through intermediate sensor nodes and forwarded to the sensor base station connected with the laptop computer. The adaptive sensor data aggregation method is running at the laptop computer to infer alert messages which are then disseminated across the IEEE 802.11g WAN connection to the network operations center.

5.6 Experimental Setup

5.6.1 System Configurations

Each component in the experimental prototype system architecture is configured with software as shown in Table 5.1. Custom applications are developed in these software suites. The laptop computers are running Ubuntu Linux distribution with DTN software [73] installed. The wireless routers are flashed with DD-WRT firmware [74] for its flexibility in configuring IEEE 802.11g wireless access points. SunSPOT base station adapter application written in Java programming language was implemented using SunSPOT Software Development Kit (SDK). This Java application is running in the laptop computer at the mobile node for translating raw sensor data from SunSPOT sensor nodes to the adaptive sensor aggregation method. A Java GUI application is also implemented at the laptop computer in the network operations center for displaying incoming alert messages.

Table 5.1: Summary of System Configurations

Name	**Software Versions**
DD-WRT firmware	v24 pre-sp2
DTN Software	2.6
SunSPOT SDK	Blue 4.0
Ubuntu Linux OS	8.10

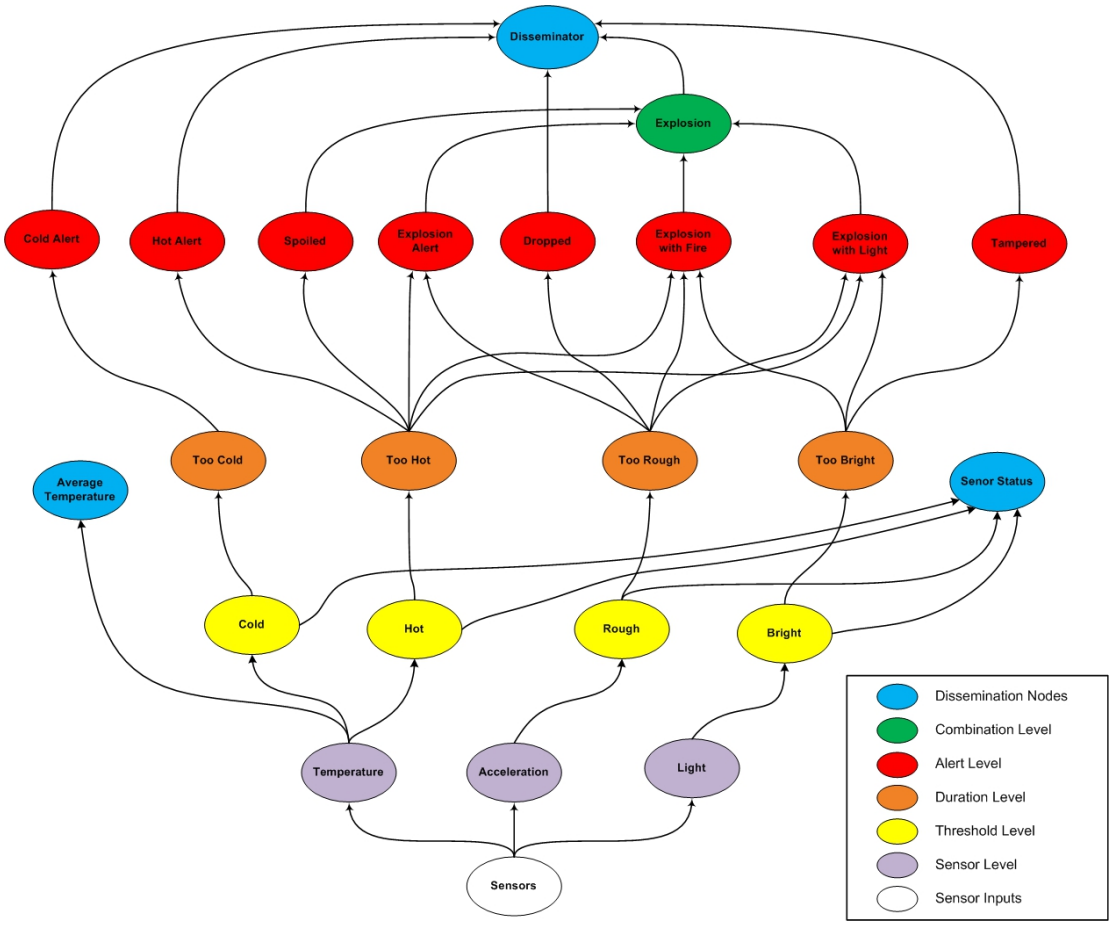


Figure 5.3: An Example of the Adaptive Sensor Data Aggregation Map

5.7 Framework Demonstration Scenario

A demonstration scenario was developed for evaluating the proposed framework in the experimental prototype system architecture. The scenario is based on the example of the data aggregation map shown in Figure 5.3, which demonstrates the concept of the adaptive sensor data aggregation method described in Section 5.4. This data aggregation map illustrates an example of inferring alert messages based on raw sensor data collected from the wireless sensor network at the mobile node. Each oval in the map represents an data aggregation point in the software process. The data aggregation map demonstrates the layered approach in the adaptive sensor data aggregation method and should be read from bottom up. The raw sensor data is first analyzed at the sensor level, then at the threshold level, then at the duration level, and finally at the alert level where the aggregated alert message is generated. An additional step is taken at the combination level to further reduce any redundancy in the alert messages. The final alert messages will accurately represent the health status of the mobile node and will be ready for dissemination to any external entity in the tactical network.

The test scenario ran for approximately 30 minutes with a five-minute network interruption at the WAN uplink during the test run. The network interruption was introduced by physical disconnection of the Ethernet ports between the laptop and the wireless router. Several wireless sensors were randomly placed at various locations in the lab environment. Two sensors were manually moved around in the lab environment for stimulating sensor data while the remaining sensors acting as intermediate router nodes. The Ad hoc On Demand Distance Vector (AODV) routing protocol was enabled in all deployed SunSPOT sensor nodes. Additionally, thirteen simulated sensor nodes were configured to generate more raw sensor data for validating the example data aggregation map. The SunSPOT base station adapter application translates these raw sensor data and presents them to the proposed framework which executes the data aggregation map. At each aggregation point, an evaluator determines the type of sensor data and whether the

data has breached a threshold specified by the system policy. Each threshold represents a system condition of *Cold* (< 50 °F), *Hot* (> 105 °F), *Rough* (> 3g), or *Bright* (> 500 lm). A system condition of *Too Cold*, *Too Hot*, *Too Rough*, or *Too Bright* is represented upon breaching of a threshold, a timer then starts to track the duration of each threshold breached. If the duration is longer than what the system policy specified (> 300 ms), an alert message is then generated. The alert messages in the data aggregation map include *Cold Alert*, *Hot Alert*, *Spoiled*, *Explosion Alert*, *Dropped*, *Explosion with Fire*, *Explosion with Light*, and *Tampered*. At the combination level, the alert message is verified again to see if it can be further combined with other similar type of alert messages. In this test scenario, *Explosion Alert*, *Explosion with Fire*, *Explosion with Light* conditions were further aggregated into *Explosion* condition. The final alert message was then disseminated to a configured recipient known as a disseminator. The implementation also provides options for applying system policies to the disseminator for choosing different forwarding path if multiple network uplinks exist. Since the experimental prototype system architecture only implements one WAN uplink at this time, the disseminator always forwards the alert messages through the default WAN uplink.

Alert Message Types

Sensor Status Table

	Sensor ID	Avg Temp	Cold Alert	Hot Alert	Spoiled	Dropped	Tampered	Explosion
Real Sensors	0014.4F01.0000.49A3	85.916	OKAY	OKAY	OKAY	ALERT	ALERT	OKAY
	0014.4F01.0000.50D6	112.650	OKAY	ALERT	ALERT	ALERT	ALERT	explosion-with...
Simulated Sensors	81da.1f33.0000.1005	108.443	OKAY	ALERT	OKAY	ALERT	ALERT	OKAY
	81da.1f33.0000.1007	40.621	ALERT	OKAY	OKAY	ALERT	OKAY	OKAY
	81da.1f33.0000.1006	105.520	OKAY	ALERT	OKAY	OKAY	OKAY	OKAY
	81da.1f33.0000.1001	65.000	OKAY	OKAY	OKAY	OKAY	ALERT	OKAY
	81da.1f33.0000.1013	106.621	OKAY	ALERT	OKAY	ALERT	OKAY	OKAY
	81da.1f33.0000.1002	110.180	OKAY	ALERT	ALERT	OKAY	OKAY	explosion-with...
	81da.1f33.0000.1011	107.443	OKAY	ALERT	OKAY	ALERT	ALERT	OKAY
	81da.1f33.0000.1003	43.000	ALERT	OKAY	OKAY	OKAY	OKAY	OKAY
	81da.1f33.0000.1008	61.443	OKAY	OKAY	OKAY	OKAY	ALERT	OKAY
	81da.1f33.0000.1009	60.520	OKAY	OKAY	OKAY	OKAY	OKAY	OKAY
	81da.1f33.0000.1010	60.621	OKAY	OKAY	OKAY	ALERT	OKAY	OKAY
	81da.1f33.0000.1012	73.030	OKAY	OKAY	OKAY	OKAY	OKAY	OKAY
	81da.1f33.0000.1004	31.003	ALERT	OKAY	OKAY	OKAY	OKAY	OKAY

Figure 5.4: Demonstration Scenario Results

5.8 Framework Test Results and Disscusion

The GUI application implemented at the network operations center captured the test results shown in Figure 5.4 from running the demonstration scenario described in Section 5.7. Since there was a five-minute network interruption during the test run, the DTN bundles were verified to ensure that all bundles were properly received after system recovery from the network interruption, and no data packet loss occurred during the 30-minute test run. For each sensor node, the sensor ID and the current average temperature reading were displayed for the purpose of lab demonstration and test results verification. The first two sensor nodes listed on the top two rows in the *Sensor Status Table* were live sensor nodes monitoring the environmental conditions in the lab. The remainder of the sensor nodes were simulated in order to generate more raw sensor data sets for stimulating the proposed framework as well as verifying the correctness in the implementation of the example data aggregation map shown in Figure 5.3.

As each aggregated alert message was received or retired at the network operations center, the *Sensor Status Table* highlighted or de-highlighted the corresponding alert message type to signal a change in the environmental conditions. Both sensor node 49A3[2] and sensor node 50D6 were manually shaken to stimulate their accelerometers and were placed close to the fluorescent lights to increase the light intensity readings. Thus, *Dropped* and *Tampered* alert messages were inferred by sensor readings from both node 49A3 and sensor node 50D6. Furthermore, sensor node 50D6 was placed near a heat source to increase the average temperature over time. A *Hot* alert message was first displayed followed by the *Spoiled* alert message after the specified time duration. Since the combination level verified that all conditions in *Explosion* alert were met, the *Explosion* alert was then highlighted for sensor node 50D6. The robustness of the proposed framework was validated through many similar scenarios with large simulated data sets.

[2] Last four Hex digits in the Sensor ID is used as the identifier in this discussion.

5.9 Data Aggregation and Dissemination Framework Conclusion

A reliable data aggregation and dissemination framework is developed and implemented in the context of tactical networks. An adaptive sensor data aggregation method is developed and combined with a DTN architecture to achieve the frameworks key requirements for fault tolerance, reliability, interoperability, and flexibility. An experimental prototype system architecture is implemented in order to demonstrate the capabilities of the proposed framework. Test results validated an example data aggregation map and demonstrated that the proposed framework is a promising solution to reliably deliver data across disruptive connections in tactical networks.

The implementation of the experimental prototype system architecture identifies more opportunities for related research. The DTN software can be embedded in the wireless router to reduce the physical footprint of the system design. By expanding the number of WAN uplinks in the prototype system, it provides an opportunity to investigate appropriate routing policies for various sensor deployment scenarios with multiple communications uplinks. Furthermore, DTN configuration parameters such as network queue sizing and data prioritization are planned to be studied to further improve the performance of the DTN bundle protocol. Other future research include design and implementation of a gateway node, better integration with existing C2 applications, development of a network management solution for the framework, and investigation of the security aspect of the framework.

Chapter 6

Conclusion and Future Work

Achieving Information Superiority in the current and future net-centric operations demand rapid growth of applications, services, systems, and number of supporting personnel. To satisfy this ever-increasing demand, there is a pressing need to address the deficiencies of current and future tactical communications system architectures. A tactical communications system serves an imperative objective of providing information to any authorized user at any geographical location and at any time. This thesis discusses the technical challenges, design principles, and recommends prudent approaches towards the development of a large scale tactical communications system. Technical challenges involving system design and development based on series of large scale communications systems are presented in the previous chapters. This chapter reviews all results found in this thesis and recommends future work in the related research area.

6.1 Thesis Conclusion

The research presented in this thesis has filled some gaps in the area of designing large scale tactical communications systems. First of all, while quality of service design in the commercial networks had been studied in the past, "How to properly

develop an adequate QoS design strategy for a tactical platform?" is not a well-understood problem until recently. Research conducted in Chapter 2 utilizes a WAN communications system as a vehicle to investigate system-level QoS design strategy on a tactical afloat platform. The system performance evaluation study is based on OPNET modeling and simulation methodology. Several test scenarios are developed to quantify and measure the system performance from an end-user's perspectives at the application-level. Simulation results confirmed that QoS design strategy must be considered at the high-latency WAN link in order to improve overall system performance. In addition, system integration cost can be further reduced as it has been shown that deployment of QoS design may not be necessarily required everywhere in the afloat platform network.

The second part of this thesis proposes a consolidated network architecture as an initial attempt to support the net-centric communications system design paradigm shift from a traditionally stovepipe system approach to a fully integrated system architecture. The overall reduction of system components in the consolidated system architecture also translates into additional system cost savings. The research presented in Chapter 3 proposes a new concept of system design to consolidate a large scale network architecture which provides LAN services. By applying appropriate security measures, multiple security network domains are joined into a common network infrastructure which provides a common set of LAN services. To investigate the system performance aspect of the proposed consolidated network architecture, OPNET modeling and simulation was also conducted in this research. A simulation test bed was implemented to characterize both a fail-over scenario and a traffic scalability scenario as well as measuring the performance of several key applications in the tactical edge LAN environment. Furthermore, simulation results validated the concept of consolidated network architecture.

The third part of this thesis addresses the resource management problem in a large scale communications system at a tactical edge network. Chapter 4 describes a software architecture known as AutoDRM system which provides a supplemental function to the commercial system management solution for alleviating the

resource management problem. The AutoDRM system can efficiently manage computing and networking resources at the tactical edge network. The development of the AutoDRM system architecture leverages the performance monitoring capability of a commercial network management system and policy-based QoS capability in the network domain. A large scale prototype system simulating a realistic tactical network environment was implemented to host the AutoDRM system for demonstrating its operational concept. The prototype system included computing resources, networking devices, SATCOM simulator, and an OPNET SITL scenario to simulate a realistic networking environment in the GIG domain. Through extensive lab experiments using the large scale prototype system, test results demonstrated improved network performance when the AutoDRM system is deployed at tactical edge network.

Finally, the last part of this thesis explores the research area in achieving reliable data delivery under intermittent networking conditions commonly found in the tactical network environment. In Chapter 5, a reliable data aggregation and dissemination framework was proposed for the tactical network architecture. The proposed framework combines disruption tolerant networking advantages and an adaptive sensor data aggregation method to ensure reliable data delivery in tactical networks. Disruption tolerant network protocol is a known method to reliably deliver the data across an intermittently connected communication link. The adaptive sensor data aggregation method deals with overwhelming amounts of raw sensor data by reducing the size of the required data across bandwidth-limited WAN interface. The proposed framework overlays on top of a heterogeneous computing and networking environment that has disruptive communications. An integrated prototype system architecture was developed to demonstrate the capabilities of the proposed reliable data aggregation and dissemination framework.

In conclusion, this thesis addresses a series of system-level design issues and provides invaluable system design principles. The research results reported herein has significantly advanced the state of the art in designing large scale tactical communications system.

6.2 Future Work

The current thesis paves the way toward providing a foundation in system-level design of large scale communications system. Nevertheless, many concerns remain and should be investigated in the future work that are partially described in this section.

In order to implement a better QoS design strategy, detailed analysis of the network traffic characteristics in the existing tactical environment is required. Network traffic characterization from various tactical communications systems is very essential to establish an accurate baseline traffic model which will enhance the results of system performance analysis using discrete event network simulation methodology. This traffic characterization should incorporate numerous system-level factors such as disk I/O, memory, storage, operating system, communication protocols, network topologies, network interface interaction, and other system related factors. Precise network traffic characterization will help understanding the system design trade-offs to aid future development of a large scale communications system. Unfortunately, performing network traffic characterization is not a trivial effort as many of these systems are not operating openly in the public network domains. Security measures need to be taken into count when performing such a network traffic characterization task. Information needs to be well protected while performing the data analysis on the collected network traffic.

While this thesis primarily focuses on utilizing a discrete event network simulation tool such as OPNET, care should be taken to investigate the radio frequency aspect of the communications system. One area of future work is to include accurate radio frequency models to represent the physical environments and physical layer behaviors. The models should be able to accurately translate physical characteristics into network characteristics. These radio frequency models can be incorporated into real-time simulation through a co-simulation methodology. Such a simulation technique combines real-time continuous simulation (*e.g.* MATLAB) which computes physical layer effects for a discrete-event network simulation (*e.g.*

OPNET) that models the behavior of the network protocol stacks. Co-simulation techniques can further enhance the overall quality of the network simulation study.

Developing a practical approach for providing resource management has attracted interest from the network user communities. Part of this thesis develops the AutoDRM system as an initial attempt to efficiently manage computing and network resources at a tactical edge network. In order to enhance the current concept of the AutoDRM system and develop it into a deployable software solution, more research is still needed in the resource management area. One example would be to understand the interaction between the commander's intents with the automated QoS plan derived from SLA and other higher-level communications plans. The problem of how to optimize and prioritize mission critical traffic when combining the commander's intents with pre-arranged automated QoS plan remains to be a much needed future work.

In the area of reliable data delivery for a tactical communications system, disruptive tolerant network configuration parameters such as network element queue size and data prioritization require further optimization to improve the overall system performance. DTN software can be embedded in a wireless gateway node to reduce the physical footprint of the large scale communications system design. Moreover, data aggregation and dissemination methods need to be refined with accurate system policies for the specific network environment to ensure its robustness in inferring accurate information from raw sensor data in a timely fashion. Finally, the security aspect of the proposed framework requires further research so that data in the delivery process is not compromised.

References

[1] (2001, Aug.) GIG Capstone Requirements Document. GIGCapstoneRequirementsDoc30AUG01.pdf. [Online]. Available: http://www.trow.tma.osd.mil/jmis/download/EA-Ref/GIGCapstoneRequirementsDoc30AUG01.pdf

[2] D. S. Alberts, J. J. Garstka, and F. P. Stein, *Network Centric Warfare: Developing and Leveraging Information Superiority.* Washington, DC: Command and Control Research Program (CCRP) Publication Series, 2000.

[3] A. K. Cebrowski and J. J. Garstka, "Network-Centric Warfare: Its Origin and Future," *Naval Institute Proceedings*, vol. 124/1/1, no. 139, pp. 1–10, Jan. 1998.

[4] (2007, Jun.) Department of Defense Global Information Grid Architectural Vision. GIGArchVision.pdf. [Online]. Available: http://cio-nii.defense.gov/docs/GIGArchVision.pdf

[5] (2010, May) Global Information Grid. [Online]. Available: http://en.wikipedia.org/wiki/Global_Information_Grid

[6] (2009, Mar.) Tactical communications system. [Online]. Available: http://en.wikipedia.org/wiki/Tactical_communications_system

[7] T. J. Strei, "Open Architecture in Naval Combat System Computing of the 21st Century," in *Proceedings of 8th Interational Command and Control Research and Technology Symposium (8th ICCRTS)*, Washington, DC, Jun. 17–19 2003, pp. 1–5.

[8] A. S. Peng and D. J. Lilja, "Performance Evaluation of Navy's Tactical Network using OPNET," in *Proceedings of IEEE Military Communication Conference (MILCOM 2006)*, Washington, DC, Oct. 23–25 2006, pp. 1–7.

[9] A. S. Peng, B. R. Eickhoff, T. He, and D. J. Lilja, "Toward Consolidated Tactical Network Architecture: A Modeling and Simulation Study," in *Proceedings of IEEE Military Communication Conference (MILCOM 2008)*, San Diego, CA, Nov. 16–19 2008, pp. 1–7.

[10] A. S. Peng, D. M. Moen, T. He, and D. J. Lilja, "Automatic Dynamic Resource Management Architecture in Tactical Network Environments," in *Proceedings of IEEE Military Communication Conference (MILCOM 2009)*, Boston, MA, Oct. 18–21 2009, pp. 1–7.

[11] A. S. Peng, D. M. Moen, T. He, and D. Lilja, "Dynamic Resource Allocation for Network Aware Applications," in *AFCEA-GMU Symposium 2010: Critical Issues in C4I*, Fairfax, VA, May18–19 2010.

[12] A. S. Peng, D. M. Moen, J. A. Spinks, L. M. Meredith, T. He, and D. J. Lilja, "Reliable Data Aggregation and Dissemination Framework in Tactical Network Architecture," in *Proceedings of IEEE Military Communication Conference (MILCOM 2010)*, San Jose, CA, Oct. 31–Nov. 3 2010, pp. 1–7.

[13] J. A. Sullivan, "Management of Autonomous Systems in the Navy's Automated Digital Network System," Master's thesis, Naval Postgraduate School, Sep. 1997.

[14] J. N. Ptasinski and Y. Congtang, "The Automated Digital Network System (ADNS) Interface to Transformational Satellite Communications System (TSAT)," in *Proceedings of IEEE Military Communication Conference (MILCOM 2007)*, Orlando, FL, Oct. 29–31 2007, pp. 1–5.

[15] J. Sun, M.-C. Wang, L. Prior, T. Gibbons, and JeffWysocarski, "Dynamic Routing with Link State Information in ADNS and Future SATCOM Network," in *Proceedings of IEEE Military Communication Conference (MILCOM 2009)*, Boston, MA, Oct. 18–21 2009, pp. 1–7.

[16] W. Youm, A. Heaberlin, M. Acevedo, and M. Acevedo, "Modeling and Simulation to Support the Development of the Navy's Extremely High Frequency TDMA Interface Processor (EHF-TIP)," in *Proceedings of IEEE Military Communication Conference (MILCOM 2005)*, vol. 2, Atlantic City, NJ, Oct. 17–20 2005, pp. 1159–1166.

[17] B. D. Rehard, "Analysis of Quality of Service Over the Automated Digital Network System," Master's thesis, Naval Postgraduate School, Sep. 1997.

[18] N. Freije, "ForceNet Enterprise Networking IT-21 Overview," Presentation, ForceNet, 2005.

[19] C. Alspaugh and A. K. Legaspi, "A Violation of Order: IP-QoS for Tactical Traffic," in *Proceedings of IEEE Military Communication Conference (MILCOM 2002)*, vol. 2, Anaheim, CA, Oct. 7–10 2002, pp. 1275–1280.

[20] D. A. Barsaleau and M. Tummala, "Testing of DiffServ Performance Over a U.S. Navy Satellite Communication Network," in *Proceedings of IEEE Military Communication Conference (MILCOM 2004)*, vol. 1, Monterey, CA, Oct. 31–Nov. 3 2004, pp. 528–534.

[21] D. Barsaleau and M. Tummala, "Testing of DiffServ Performance Over a U.S. Navy Satellite Communication Network," in *Proceedings of 22nd AIAA International Communications Satellite Systems Conference and Exhibit*, Monterey, CA, May 9–12 2004.

[22] "OPNET Model User Guide," Version 11.5.A, OPNET Technologies, Inc., 2005.

[23] J. Moy, "OSPF Version 2," RFC 2328, Apr. 1998. [Online]. Available: http://www.ietf.org/rfc/rfc2328.txt

[24] T. R. Henderson and R. H. Katz, "Transport Protocols for Internet-compatible Satellite Networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 2, pp. 326–344, Feb. 1999.

[25] S. Subramanian, S. Sivakumar, W. J. Phillips, and W. Robertson, "Investigating TCP Performance Issues in Satellite Networks," in *Proceedings of the 3rd Annual Communications Networks and Services Research Conference*, Halifax, Nova Scotia, Canada, May 16–18 2005, pp. 327–332.

[26] "NetWars Model Development Guide," Version 1.8a, Defense Information Systems Agency, Mar. 2005.

[27] J. C. Mogul, "Squeezing More Bits Out of HTTP Caches," *IEEE Network*, vol. 14, no. 3, pp. 4–14, May 2000.

[28] M. Liu, F. Wang, D. Zheng, and L. Yang, "An Overview of World Wide Web Caching," in *IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5, Tucson, AZ, Oct. 7–10 2001, pp. 3045–3050.

[29] L. Fan, P. Cao, J. Almeida, and A. Z. Broder, "Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol," *IEEE/ACM Transactions on Networking*, vol. 8, no. 3, pp. 281–293, Jun. 2000.

[30] B. M. Duska, D. Marwood, , and M. J. Feeley, "The Measured Access Characteristics of World-Wide-Web Client Proxy Caches," in *Proceedings of USENIX Symposium on Internet Technology and Systems*, Monterey, CA, Dec. 8–11 1997, pp. 23–35.

[31] "Juniper Networks WX Application Acceleration Evaluation Summary," Evaluation Summary Report, Juniper Networks, Sep. 2005.

[32] D. Washburn. (2007, Oct.) Networks, Information Assurance and Enterprise Services (PMW160). Delores Washburn presentation.pdf. [Online]. Available: http://129.7.151.14/Home/DoDconferencepresentations/

[33] C. Miller. (2007, Oct.) Navy C4I Open Architecture Strategy. [Online]. Available: https://www.softwaretechnews.com/stn_view.php?stn_id=43&article_id=89

[34] P. C4I. (2007, Jul.) Consolidated Afloat Network and Enterprise Services (CANES) Industry Day. CANESINDUSTRYDAY20-7-27-07.pdf.

[35] P. Turner. (2007, Dec.) The CANES Initiative: Bringing the Navy Warfighter onto the Global Information Grid. CANES.pdf. [Online]. Available: www.chips.navy.mil/archives/07_Dec/PDF/

[36] G. Malkin, "RIP Version 2," RFC 2453, Nov. 1998. [Online]. Available: http://www.ietf.org/rfc/rfc2453.txt

[37] C. Christou, W. Hall, and K. Sheu, "Applying Service Class Treatment Aggregates to the Global Information Grid (GIG)," in *Proceedings of IEEE Military Communication Conference (MILCOM 2006)*, Washington, D.C., Oct. 23–25 2006, pp. 1–5.

[38] "Global Information Grid Net-Centric Implementation Document: Quality of Service (T300)," Aug. 2006.

[39] *Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications*, IEEE Std. 802.3, 2008.

[40] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 2007.

[41] B. Doshi, P. Kim, B. Liebowitz, K. I. Park, and S. Wang, "Service Level Agreements and QoS Delivery in Mission Oriented Networks," in *MITRE Corporation*, May 2006.

[42] A. Kantawala, D. Voce, and D. Gokhale, "QoS Architecture for Session Oriented GIG Applications," in *Proceedings of IEEE Aerospace Conference*, Big Sky, Montana, Jul.4–11 2006, p. 9.

[43] C. Gedo, Y. Xue, J. Evans, C. Dee, and C. Christou, "GIG QoS Inter-Domain Interoperability Challenges," in *Proceedings of IEEE Military Communication Conference (MILCOM 2007)*, Orlando, FL, Oct.29–31 2007, pp. 1–7.

[44] M. Albuquerque, A. Ayyagari, M. A. Dorsett, and M. S. Foster, "Global Information Grid (GIG) Edge Network Interface Architecture," in *Proceedings of IEEE Military Communication Conference (MILCOM 2007)*, Orlando, FL, Oct.29–31 2007, pp. 1–7.

[45] C. Hamilton, "Ship Acquisition," in *NDIA 10th Annual Expeditionary Conference*, Panama City, FL, Oct.24–27 2005.

[46] P. Lardieri, J. Balasubramanian, D. Schmidt, G. Thaker, A. Gokhale, and T. Damiano, "A Multi-layered Resource Management Framework for Dynamic Resource Management in Enterprise DRE Systems," *Journal of System and Software*, vol. 80, no. 7, pp. 984–996, Jul. 2007.

[47] R. Rajkumar, C. Lee, J. P. Lehoczky, and D. P. Siewiorek, "A Resource Allocation Model for QoS Management," in *In IEEE Real-Time Systems Symposium*, San Francisco, CA, Dec.3–5 1997.

[48] R. Rajkumar, C. Lee, J. P. Lehoczky, and D. Siewiorek, "Practical Solutions for QoS-based Resource Allocation Problems," in *In IEEE Real-Time Systems Symposium*, Madrid, Spain, Dec.2–4 1998.

[49] F. Harada, T. Ushio, and Y. Nakamoto, "Adaptive Resource Allocation Control for Fair QoS Management," *IEEE Transactions on Computers*, vol. 56, no. 3, pp. 344–357, Mar. 2007.

[50] J. A. Stankovic, T. He, T. F. Abdelzaher, M. Marley, G. Tao, S. H. Son, and C. Lu, "Feedback Control Scheduling in Distributed Systems," in *22nd IEEE Real-Time Systems Symposium (RTSS 2001)*, London, UK, Dec. 3–6 2001, pp. 59–70.

[51] B. Dasarathy, S. Gadgil, R. Vaidyanathan, A. Neidhardt, K. P. B. Coan, A. McIntosh, and F. Porter, "Adaptive Network QoS in Layer-3/Layer-2 Networks as a Middleware Service for Mission-Critical Applications," *Journal of System and Software*, vol. 80, no. 7, pp. 972–983, Jul. 2007.

[52] C. Lee, J. Lehoczky, D. Siewiorek, R. Rajkumar, and J. Hansen, "A Scalable Solution to the Multi-Resource QoS Problem," Carnegie Mellon University, PA, Tech. Rep. CMU-CS-99-144, May 1999.

[53] M. M. Akbar, E. G. Manning, G. C. Shoja, and S. Khan, "Heuristic Solutions for the Multiple-Choice Multi-Dimension Knapsack Problem," in *Proc. of Int. Conf. Computational Science*, vol. 2074, May 2001, pp. 659–668.

[54] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services," RFC 2475, Dec. 1998. [Online]. Available: http://www.ietf.org/rfc/rfc2475.txt

[55] J. Babiarz, K. Chan, and F. Baker, "Configuration Guidelines for DiffServ Service Classes," RFC 4594, Aug. 2006. [Online]. Available: http://www.ietf.org/rfc/rfc4594.txt

[56] Microsoft. (2006) Policy-based QoS Architecture in Windows Server "Longhorn" and Windows Vista. [Online]. Available: http://www.microsoft.com/technet/community/columns/cableguy/cg0306.mspx

[57] L. G. Shattuck, "Communicating Intent and Imparting Presence," in *Military Review*, Mar. 2000, pp. 66–72.

[58] *Staff Organization and Operations*, Field Manual No. 101-5, Department of the Army, 1997.

[59] B. Doshi, P. Kim, B. Liebowitz, K. I. Park, and S. Wang, "Service Level Agreements and QoS Delivery in Mission Oriented Networks," *White Paper, MITRE Corporation*, May 2006.

[60] D. Harrington, R. Presuhn, and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks," RFC 3411, Dec. 2001. [Online]. Available: http://tools.ietf.org/html/rfc3411.txt

[61] OpenNMS. (2009) About the OpenNMS Project. [Online]. Available: http://www.opennms.org/index.php/FAQ-About

[62] "D-ITG V.2.6.1d Manual," Version 2.6.1d, University of Naples Federico II, 2008. [Online]. Available: http://www.grid.unina.it/software/ITG/codice/D-ITG2.6.1d-manual.pdf

[63] T. He, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "AIDA: Adaptive Application-Independent Data Aggregation in Wireless Sensor Networks," *ACM Transations on Embedded Computing Systems*, vol. 3, no. 2, pp. 426–457, May 2004.

[64] M.-G. Lee and S. Lee, "Data Dissemination for Wireless Sensor Networks," in *Proceedings of Tenth IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC 2007)*, Santorini Island, Greece, May7–9 2007, pp. 172–180.

[65] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM 2003)*, Karlsruhe, Germany, Aug.25–29 2003, pp. 27–34.

[66] L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalnagar, L. Nachman, and M. Yarvis, "Design and Deployment of Industrial Sensor Networks: Experiences from a Semiconductor Plant and

the North Sea," in *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys 2005)*, San Diego, CA, Nov.2–4 2005, pp. 1–6.

[67] H. Liu, B. Zhang, H. Mouftah, X. Shen, and J. Ma, "Opportunistic Routing for Wireless Ad Hoc and Sensor Networks: Present and Future Directions," *IEEE Communications Magazine*, vol. 47, no. 12, pp. 103–109, Dec. 2009.

[68] K. Scott and S. Burleigh, "Bundle Protocol Specification," RFC 5050, Nov. 2007.

[69] S. Okamoto and K. Sycara, "Augmenting Ad Hoc Networks for Data Aggregation and Dissemination," in *Proceedings of IEEE Military Communication Conference (MILCOM 2009)*, Boston, MA, Oct.18–21 2009, pp. 1–7.

[70] H. E. Weigner and J. E. Laudan, "MTS: A Success Story for Battlefield Logisticians," *Army Logistician*, vol. 37, no. 5, pp. 1–5, Sept–Oct 2005.

[71] S. Parikh and R. C. Durst, "Disruption Tolerant Networking for Marine Corps CONDOR," in *Proceedings of IEEE Military Communication Conference (MILCOM 2005)*, Atlantic City, NJ, Oct.17–20 2005, pp. 325–330.

[72] "SunSPOT Owner's Manual (Blue Release 4.0)," White Paper, Sun Microsystems, Aug. 2008.

[73] (2008, Jul.) DTN2 (Version 2.6.0). [Online]. Available: http://sourceforge.net/projects/dtn/files/

[74] (2009, Oct.) DD-WRT (Version v24 pre-sp2). [Online]. Available: http://www.dd-wrt.com

Appendix A

Acronyms

This appendix contains a table of acronyms and their meaning.

Table A.1: Acronyms

Acronym	Meaning
ADNS	Automated Digital Network System
AutoDRM	Automatic Dynamic Resource Management
BER	Bit Error Rate
C2	Command and Control
CANES	Consolidated Afloat Networks and Enterprise Services
CCE	Common Computing Environment
CENTRIXS	Combined Enterprise Regional Information Exchange System
COTS	Commercial Off-The-Shelf
DiffServ	Differentiated Services
DITG	Distributed Internet Traffic Generator
DoD	Departement of Defense
DSCP	Differentiated Service Code Point
DTN	Disruption Tolerant Network
ETE	End-To-End

Continued on next page

Table A.1 – continued from previous page

Acronym	Meaning
FTP	File Transfer Protocol
GEO	Geosynchronous Earth Orbit
GIG	Global Information Grid
GPO	Group Policy Object
GUI	Graphical User Interface
HAIPE	High Assurance Internet Protocol Encryptor
IP	Internet Protocol
IT	Information Technology
LAN	Local Area Network
LOS	Line of Sight
NMS	Network Management System
NOC	Network Operations Center
NSA	National Security Agency
OA	Open Architecture
OSPF	Open Shortest Path First
PHB	Per-Hop Behavior
QoS	Quality of Service
RF	Radio Frequency
RIP	Routing Information Protocol
SATCOM	Satellite Communications
SNMP	Simple Network Management Protocol
SLA	Service Level Agreement
SLS	Service Level Specifications
SOA	Service Oriented Architecture
STIL	System-In-The-Loop
TCP	Transmission Control Protocol
TTM	Time to Market

Continued on next page

Table A.1 – continued from previous page

Acronym	Meaning
UDP	User Datagram Protocol
VLAN	Virtual Local Area Network
WAN	Wide Area Network

Appendix B

Glossary

This appendix defines jargon terms in a glossary.

- **CipherText Core** – refers to a network transporting encrypted IP data.
- **Commander's Intent** – is a concise expression for the purpose of the operation and the desired end state that serves as the initial impetus for the mission planning process.
- **PlainText Core** – refers to a network transporting unencrypted IP data.
- **System of Systems** – is a collection of independently operated systems integrated together to form a new and more complex large scale system which enables more functionality and performance.
- **Service Oriented Architecture** – a system architecture based on a loosely-integrated suite of services that can be used within multiple separate systems from several business domains.
- **Tactical Communications System** – is a secure communications system designed to meet the challenging system requirements in various tactical situations including harsh physical conditions. The system is used within or in support of tactical forces.