Quality of Service guarantees and fault-tolerant TCP services in mobile wireless optical networks

Ionut Cardei*

Florida Atlantic University,
Boca Raton, 33431 FL, USA
E-mail: icardei@cse.fau.edu
*Corresponding author

Allalaghatta Pavan

Honeywell Labs,
Minneapolis, 55417 MN, USA
E-mail: allalaghatta.pavan@honeywell.com

Riccardo Bettati

Texas A&M University,
College Station, 77843 TX, USA
E-mail: bettati@cs.tamu.edu

Abstract: In this paper we address the problem for communications Quality of Service (QoS) in wireless networks of Mobile Optical Free-space networks. We propose an architecture for end-to-end statistical delay guarantees. A delay model using the concept of virtual traffic accommodates variations in link capacity variations as well as transient outages. The QoS architecture uses an admission control technique that limits link utilisation for real-time traffic flows in order to reduce queuing delay. Admission control uses a 2-phase commit protocol for handling QoS negotiation and adaptation. We also present a mechanism for deploying QoS-enabled dependable TCP services in this network. The primary-backup TCP replication mechanism is supported by the routing infrastructure and improves server deployment transparency compared to previous work. We illustrate application performance improvement with simulation results.

Keywords: Quality of Service; QoS; wireless networks; optical networks; channel model; statistical delay guarantees.

Reference to this paper should be made as follows: Cardei, I., Pavan, A. and Bettati, R. (2008) 'Quality of Service guarantees and fault-tolerant TCP services in mobile wireless optical networks', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Ionut Cardei is an Assistant Professor at the Department of Computer Science and Engineering at the Florida Atlantic University, in Boca Raton, Florida. He received his PhD in Computer Science from the University of Minnesota. His main research interests are in communication protocols for wireless networks, Quality of Service and resource management.

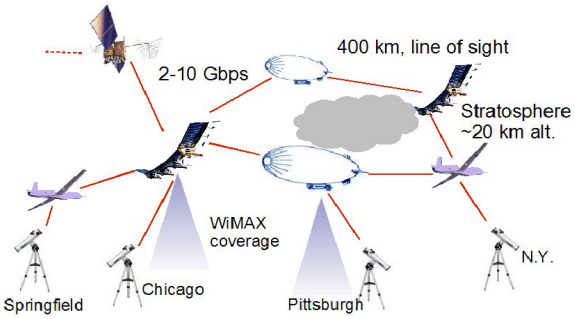
Allalaghatta Pavan is a Principal Research Scientist at Honeywell Labs in Minneapolis. He has served as a Principal Investigator on several projects sponsored by DARPA, ONR and AFRL. He has over 30 publications in the areas of QoS, resource management, WDM and multimedia systems. He has a PhD (CS) from the University of Minnesota.

Riccardo Bettati received his Diploma in Informatics from the Swiss Federal Institute of Technology (ETH), Zuerich, Switzerland, in 1988 and his PhD from the University of Illinois at Urbana-Champaign in 1994. He is currently an Associate Professor in the Department of Computer Science at Texas A&M University, where he leads the Real-Time Systems Research Group. His research interests are in traffic analysis and privacy, real-time distributed systems, real-time communication and network support for resilient distributed applications.

1 Introduction

Recent developments in wireless optical technologies in the areas of beam pointing, acquisition and tracking (Stadler and Duchak, 2004) have opened the possibility of building mobile optical wireless (or free-space) networks (MOFS networks) consisting of ground and airborne terminals carried on airplanes or airships. Long-endurance flight platforms such as AeroVironment's Helios UAV http://www.nasa.gov/centers/dryden/history/pastprojects/Erast/helios.html) or SansWire's Stratellite (http://www.stratellite.net/), could provide continuous networking coverage via direct wireless laser links and Radio-Frequency (RF) links (e.g., WiMAX) to entire cities or remote communities. Figure 1 illustrates such a network. A multi-hop ad-hoc topology running the IP protocol suite has increased ability to route traffic around areas with adverse weather (Stadler and Duchak, 2004). At stratospheric altitude wireless optical links can achieve 2–10 Gbps over hundreds of kilometers.

Figure 1 MOFS network concept



MOFS networks raise significant problems for the deployment of applications demanding QoS, such as end-to-end delay bounds. Link quality is time-variable and hard to model theoretically. Signal fading is caused by scintillation from atmospheric turbulence, especially on long links. Cloud obscuration makes links unusable due to beam absorption. With multi-aperture terminals, more alternate links can be set up to form routes avoiding space with adverse transmission conditions.

Beyond the technical problems related to the hardware platform, a major issue still remains communication QoS. Providing the same level of delay and bandwidth guarantees as a wired network is not possible. In the same time, a MOFS network is very different from a typical mobile ad-hoc RF wireless network (MANET), requiring a new approach for QoS. There is no broadcast medium, link duration is expected to be much longer than in a MANET and the network topology is much sparser and with a mostly deterministic and predictable mobility pattern.

The main contribution of our research is an architecture, OptiExpress, that provides

- statistical guarantees for end-to-end communication QoS
- QoS-enabled fault tolerant TCP services in a MOFS network.

OptiExpress includes a QoS mechanism capable of accommodating variations in network quality at different levels, and still provide acceptable communication services. It is able to handle link quality degradation causing link capacity variation, short term link outages (<10 ms), topology changes, and long term link outages. Our approach leverages standard IP protocols and is implementable on commercial IP routing equipment with little changes. We base the OptiExpress architecture on a statistic model for link delay that uses descriptions of the optical link capacity and application traffic. The variation of link capacity is modelled as virtual traffic schedulable with the highest priority. The model described in Section 2.3 determines the probability that a random packet exceeds the delivery deadline on a link (Delay Violation Probability (DVP)). The model is extended to end-to-end probabilistic guarantees on the route from a flow source to the destination. The QoS architecture can handle changes to routes through a process of QoS reconfiguration that involves flow adaptation. OptiExpress admission control assigns a differentiated services code point (Blake et al., 1998) for bandwidth, delay and DVP levels that fall within the required QoS interval. The admission algorithm limits the total bandwidth allocated per class for each link to a safe utilisation level. This guarantees that all real-time flows will experience delay limit violations within configured per-class delay limits. This approach for flow admission is called Utilisation Based Admission Control and was first introduced in NetEx (Wang et al., 2001).

The second contribution of this work consists of an architecture for fault-tolerant TCP services provisioned with communication QoS. We extended the HydraNet-FT (Shenoy et al., 2000) primary-backup approach for service replication. Server coordination and client communication are accomplished with an atomic ordered multicast protocol that uses primarily the TCP flow control messages. In HydraNet-FT an 'ACK chain' is built from the backup servers and the primary server to control message delivery and retransmission. In our approach, named HydraNet-DS (from DS), the ACK chain includes only server-side routers belonging to the MOFS network (instead of the servers themselves), allowing services and server machines to be deployed fully transparent. We provisioned QoS for all client-servers flows and the ACK chain and we measured improved performance.

We implemented the OptiExpress architecture in the OPNET discrete event network simulator (http://www.opnet.com) and we evaluated its performance for various operational scenarios and applications.

1.1 Related work

Earlier results related to this project have been published in Wang et al. (2004) and Cardei et al. (2005). MOFS networks are a new technology. We believe this effort is the first to address the aspects of QoS in this type of network. However, QoS frameworks for wireless ad-hoc networks with RF links have been studied extensively. The biggest obstacle for QoS guarantees in these networks is that the link

capacity is variable and connectivity is intermittent. Frameworks for delay guarantees on wired lines (Knightly, 1998; Liebeherr et al., 1996) can not be applied directly in this context.

The work in Chaporkar and Sarkar (2002) proposes an approach for statistical delay and drop guarantees in single-hop wireless networks using admission control and earliest deadline first scheduling. The delay modes requires instantaneous channel state. In contrast, we use for QoS models channel disturbance as virtual traffic, which is used to reduce the utilisation limit on a link. Other efforts for QoS in MANETs look into bandwidth reservation and QoS routing. Insignia (Lee et al., 2001) is an in-band signalling system for bandwidth reservation in IP MANETs compatible with multiple routing protocols. QoS routing is proposed in Shigang and Nahrstedt (1999) and Chenxi and Corson (2002).

Mechanisms for building fault-tolerant TCP services have been proposed before. FT-TCP (Zagorodnov et al., 2003) uses wrappers around TCP server code to forward TCP traffic to a logger that stores the server state. In case the primary fails a new server will be initialised using the state stored from the previous primary. This system may experience long failover delays due to server initialisation and incurs the overhead of storing each packet. With M-TCP (Sultan et al., 2002) clients open TCP connections to any server from a pool of replicas. Servers log the state of their client connections. Policies control how clients can migrate their connections to another server depending on connection performance. This solution does not allow full client and server transparent failover. Finally, ST-TCP (Marwah et al., 2003) uses a primary-backup scheme, with the backup server intercepting the client-server TCP stream at MAC level. The backup detects primary failures with a heartbeat protocol, allowing rapid failover. This solution is also not transparent to the server and requires backup servers to be located on the same LAN as the primary server.

The next section describes the delay analysis for a link server, the end-to-end guarantees mechanisms and the QoS reconfiguration. Section 2 presents the QoS architecture in detail, including the network model, the QoS model and the protocol architecture. The architecture for fault-tolerant TCP services is described in Section 3. Section 4 continues with a review of performance results. The paper concludes in Section 5 and comments on future work.

2 Quality of Service (QoS) architecture

We start by describing the network model, followed by the delay model on which the QoS architecture is based on. We build on the statistical QoS guarantee model published in Wang et al. (2004). In this paper we adapt its mechanisms to the MOFS network running the TCP/IP communication protocols. The second part of this section presents the QoS protocol architecture and the admission control.

2.1 The network model

A notional MOFS topology is depicted in Figure 1. The network's objective is to connect at high-speed remote and large geographical areas, such as cities. Airborne terminals can also provide blanket connectivity with an network, such as WiMAX (http://www.wimaxforum.org/home). Air vehicles, such as airplanes, UAVs, airships carry on board a router and multiple wireless optical I/O interfaces (apertures), implementing an ad-hoc mobile wireless network. Payload sensors aboard aircraft may also generate traffic.

Point-to-point links are established when two nodes point a transmitter to each other's optical aperture. Wireless laser links will soon achieve data rates between 45 Mbps for passive terminals and 2–10 Gbps for active terminals, on distances for up to hundreds of kilometers, at high altitudes (20 km), where adverse atmospheric effects are minimal. A thorough review of emerging technologies relevant to MOFS networks is given in Stadler and Duchak (2004).

When a link is obscured by clouds or terrain, an airborne relay node reroutes traffic on another link already established. If no alternate links are available, the relay node points the laser beam to another node within Line-Of-Sight (LOS) to establish a new connection. The network topology configuration process and the beam pointing and acquisition procedure are managed through a secondary RF network. The work in Zhuang et al. (2004) presents algorithms for topology control in MOFS networks.

The admission control algorithm need access to routes and link state. In our implementation the EIGRP routing protocol (Cisco, 2002) exports routes and reports node reachability.

2.2 Quality of service (QoS) representation

We define some terms to describe the relationship between QoS and applications. A *managed* flow is a packet flow that went through admission control and for which the network provides statistical delay guarantees. A *legacy* application is ported from another network without changes to code or binaries. A *QoS-aware* application requires real-time communication services from the network and is dependent on bandwidth and bounds to end-to-end packet delay.

Before a packet flow can be admitted in the network, it must undergo a admission control. A flow admission request is submitted to the admission control component, the Bandwidth Broker (BB). An admission request has a series of parameters describing flow characteristics:

- protocol type (TCP, UDP)
- source, destination IP addresses
- flow priority (0 – for best effort, up to 255)
- a descriptor for end-to-end network QoS:

- maximum burst size M
- delay interval $[D_{min}, D_{max}]$
- average transmission rate interval $[R_{min}, R_{max}]$
- Delay Violation Probability interval $[E_{min}, E_{max}]$.

The DVP is the probability that a random packet will exceed its end-to-end delay D_{max}, $P\{D \geq D_{max}\} \in [E_{min}, E_{max}]$. Using intervals for QoS allows the admission control algorithm to be flexible in assigning resources, as most real-time applications tolerate a range of network performance. These applications are capable to *adapt* to a variation in bandwidth and delay and to continue to operate satisfactory. The *Current Operational Point* (COP) is defined by $< M, D, R, E >$ and describes the instantaneous values of the flow QoS measures perceived by the application.

The network manager defines the set of traffic classes and assigns for each class an operational point in the (M, D, R, E) QoS parameter space. An IP queuing policy is set that maps the traffic class (Differentiated Services Code Point (DSCP)) to a scheduling priority. For the simulation study, we assigned higher priority for traffic classes that require lower delay. As a condition for providing probabilistic delay guarantees, connection admission control assisted by traffic policing at the network edge routers keep the total managed traffic data rate from class i below $v\alpha_i C_k$ on each link k, where

$$\sum_{\text{class } i} \alpha_i \leq 1,$$

Ck is the data rate capacity on link k, including effects from coding and ARQ, and n is the safe bandwidth utilisation limit. The difference to 100% is available for best effort traffic. In Section 2.5 we present how the safe utilisation limit is determined during QoS reconfiguration. A sample traffic class configuration is shown in Table 1. IPv6 packets larger than 64 KB (jumbograms) are discussed in http://www.faqs.org/rfcs/rfc2675.html. The delay specification D, includes only the transmission and queuing delays along the end-to-end path. Propagation and processing delays can be estimated and factored out.

Table 1 Sample class QoS specification

Class	*Priority*	*M* (kB)	*R* (Mbps)	*D* (ms)	α_I (%)	*E* (%)	*Description*
0	8	128	10	80	5	0.5	10 Mbps sensors
1	7	128	20	100	5	0.8	MPEG2 20 Mbps
2	6	521	45	100	15	1.0	CDL 45 Mbps
3	5	521	100	100	25	1.0	VPN
4	4	1024	274	100	45	1.0	CDL 274 Mbps

2.3 Link delay analysis

This analysis applies to networks that use static priority packet schedulers. Packets from real-time flows are assigned to a traffic class at network ingress and each class is associated with a DSCP (Blake et al., 1998) that defines how routers service the packet, inclusive the scheduling priority. For a packet of class i, the *probabilistic delay guarantee* on a link is a bound on the probability that a random packet will exceed a deadline ε_i, and can be expressed as:

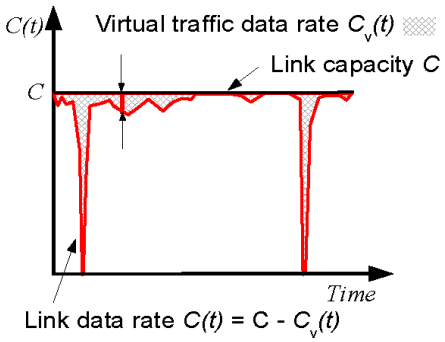
$$P(D_i > d_i) \leq \varepsilon_i.$$

The packet delay D_i is a random variable, and d_i is the maximum acceptable deadline for the link. The delays discussed in this paper include only the transmission and queuing delays, as the propagation and processing delays can be estimated and factored out.

The data links in the MOFS have variable data rate and suffer from short term interruptions. Define $C(t)$ as the capacity available for traffic on a link as a function of time. The maximum capacity on a link is C. C is perceived above the data link layer and reduced by the effects of Forward Error Correction (FEC) and Automatic-Repeat Request (ARQ) overhead. Then $C(t) = C - C_v(t)$, where $C_v(t)$ is part of link capacity that is not available due to link quality variation and outages.

A delay model where the effective capacity for transmission on a link is a time-variable function $C(t)$, has the same delay characteristics as a model where the link capacity is constant C, and where a *virtual* traffic with data rate $C_v(t) = C - C(t)$ arrives at the link and is scheduled with the absolute highest priority. Figure 2 shows the concept of virtual traffic.

Figure 2 Link capacity and virtual traffic



Packet delays in the link model with the variable capacity $C(t)$ are equal to delays in the model with the virtual traffic. Define $S(t)$ the stochastic service curve of the wireless optical link. $S(t)$ describes the traffic amount serviced during a time interval $[0, t]$ by a channel of time-variable capacity $C(\tau)$:

$$S(t) = \int_0^t C(\tau)\, d\tau.$$

Then, the virtual traffic has a service curve $B'(t) = C \cdot t - S(t)$. Now, assume G_i is the group of flows from class i arriving at a link of capacity C, and $b_{ij}(t)$ and $B_{i,j}(t)$ are the deterministic and statistic traffic envelopes for the traffic arrival of flow j from G_i. The DVP on a link with variable quality becomes:

$$P(D_i > d_i) \le \max P(B'(t+d_i) + B^*(t+d_i) \ge C \cdot (t+d_i)), \quad (1)$$

where $B^*(t)$ is the service curve of the aggregated traffic of class $\le i$. By convention class 1 has the highest scheduling priority.

$$B^*(t+d_i) = \sum_{q=1}^{i-1} \sum_{j \in G_q} B_{q,j}(t+d_i) + \sum_{j \in G_i} B_{i,j}(t). \quad (2)$$

$B^*(t + d_i)$ and $B'(t + d_i)$ are independent and the c.d.f. of their sum can be computed by convolution. It is important to notice that equation (1) is valid independent of the wireless link model.

If the number of flows is large enough and if their arrival processes are independent, the distribution of $B^*(t + d_i)$ can be approximated with the Central Limit Theorem, as in Knightly (1998). Let $n_j = |G_j|$, the number of flows in group j on a link. A deterministic leaky bucket arrival envelope for flows in G_j is $b_{i,j}(t) = \sigma_j + \rho_j t$, where ρ_j is the average flow data rate for class j and σ_j, is the maximum packet size. By using a Gaussian approximation over intervals the c.d.f. of $B^*(t + d_i)$ is bounded by a normal distribution $N(\phi_i(t), RV_i(t))$:

$$P(B^*(t+d_i) < x) \le \Phi\left(\frac{x - \phi_i(t)}{\sqrt{RV_i(t)}}\right), \quad (3)$$

where $\phi_i(t)$ is the mean aggregate data rate of the flows forming B^*:

$$\phi_i(t) = (t + d_i) \sum_{q=1}^{i-1} n_q \rho_q + t\, n_i \rho_i,$$

and $RV_i(t)$ is the aggregate rate variance envelope of the B^* flows,

$$RV_i(t) = (t + d_i) \sum_{q=1}^{i-1} n_q \rho_q \sigma_q + t\, n_i \rho_i \sigma_i,$$

and

$$\Phi(a) \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{a} e^{-x^2/2} dx$$

is the c.d.f. of the normal distribution.

2.4 End-to-end delay

The end-to-end delay guarantee along a path can be determined from the per-link delay guarantees. The end-to-end deadline for a flow of priority i, d_i, is divided into delay limits for each link along the path **R**:

$$d_i = \sum_{k \in R} d_i^k.$$

The end-to-end delay delay guarantee is satisfied when

$$P(D_i^e > d_i) \le 1 - \prod_{k \in R} (1 - P(D_i^k > d_i^k)). \quad (4)$$

One way to compute an end-to-end delay breakdown is to split it equally per link: $d_i^k = d_i / |R|$, for all links $k \in R$. Some links may be considerably slower than others, so a better approach is to assign a delay per link inverse proportional to the link capacity:

$$d_i^k = \frac{d_i}{C_k \sum_{j \in R} 1/C_j},$$

for all links k belonging to route R. Before a flow can be admitted in the network, the computation equation (4) checks whether delay guarantees can be satisfied or not. This computation is time-consuming and may impose a high processing overhead in scenarios with heavy load.

To avoid this overhead, our QoS architecture uses Utilisation Based Admission Control (UBAC), first proposed for NetEx (Wang et al., 2001). UBAC uses a delay computation that is insensitive to flow population. All flows from a class i share a dedicated fraction of the link capacity, $\alpha_i C$. Hence, admission control limits the total number of flows from a class i on a particular link to

$$n_i = \lfloor \alpha_i C / \rho_i \rfloor \quad (5)$$

with $\alpha_i \in (0, 1)$ and

$$\sum_{\text{class } i} \alpha_i \le 1.$$

The isolation between classes guarantees that the assumptions for equations (2) and (3) are respected and, therefore, the delay guarantees are satisfied. Using equation (5) the mean rate and the rate variance for the aggregate traffic of the same or higher priority from equation (3) are upper bounded, eliminating the dependence on flow count:

$$\phi_i(t) = (t + d_i) \sum_{q=1}^{i-1} \alpha_i C + t \alpha_i C$$
$$RV_i(t) = (t + d_i) \sum_{q=1}^{i-1} \alpha_i \sigma_i C + t \alpha_i \sigma_i C. \quad (6)$$

These equations can be easily integrated into the computation of equation (3) and the end-to-end delay guarantee condition (4).

2.5 QoS reconfiguration and adaptation

From a practical perspective, the network manager or user defines the QoS specification for traffic classes, including σ, ρ, d_i^e and the flow priority. The user also defines the capacity partition $< \alpha_i >$, $i = 1, \ldots, m$, where $\alpha_i C$ is exclusively reserved for class i traffic. It is very probable that, with the original QoS parameters, the end-to-end delay constraints computed with equation (4) may not be satisfied for at least one flow class on at least one end-to-end path. In this case, the user could either increase the maximum

delay limits or increase the DVP for the classes that do not have the end-to-end delay constraints satisfied. But both methods are not feasible, as they interfere with the application QoS requirements.

The approach in NetEx to mitigate this issue is to uniformly reduce the link capacity allocated to all real-time traffic classes to a fraction $v \in (0, 1)$, called *safe utilisation bound.* The maximum number of flows from a class i admissible on a link of capacity C becomes

$$n_i = \left\lfloor \frac{v\alpha_i C}{\rho_i} \right\rfloor, \tag{7}$$

preferably with $v \to 1$. To select the highest possible value for the safe utilisation bound we use binary search. The selection condition is whether the delay guarantees hold for all classes and for all source-destination pairs. The search is in a continuous space and it stops when v converges. Most scenarios used in our experiments reached utilisation bounds of 70–90%.

To summarise, UBAC requires the computation of the safe utilisation bound when routes change. Then, UBAC replaces the expensive admission-time convolution computation with a simple check whether the maximum number of flows of class i has exceeded limit n_i, on each link along a route.

After the safe utilisation bound is computed, UBAC reconsiders admission for all flows for which the source-destination path has changed. If the new maximum permitted class i flow count on a link is n_i^*, n_i class i flows are already admitted on that link, and $n_i > n_i^*$, then $n_i - n_i^*$ flows must be adapted: either switched to a different traffic class or terminated. Selecting the $n_i - n_i^*$ victim flows and how to adapt the flows are operations driven by policies reflecting application and network management requirements. One approach for victim selection is to pick flows with lower priority. Admission control then attempts to assign these flows to classes that have operational points (DVP, delay and data rate) within the initial flow QoS requirement. After reassignment to another class, QoS-aware applications receive an adaptation notification from the OptiExpress middleware.

This process of recomputing the safe utilisation level followed by re-admission for some flows is called 'QoS reconfiguration'. It is performed only when routes change and not during admission.

2.6 Link estimation

In order to compute the safe utilisation level during the reconfiguration process the virtual traffic time-varying capacity must be estimated. The constant (maximum) link capacity C is determined from the data link parameters, such as nominal data rate, encoding and FEC rate. Research in mobile free-space data links is new and theoretic link models are not yet available. Measurements for link faults (Communication from ITT Industries, 2002) on wireless optical links indicated link outages of $\tau = 10$ ms that occur every $T = 1.5$ s. We approximate the variable capacity of the virtual traffic with a leaky bucket model. Data link rate history data is used to compute the parameters σ_v and ρ_v. A simple interpretation is to consider that between outages the bucket fills in with rate ρ_v. When it is full, a link outage 'drains' the bucket at link capacity, C.

Thus, $\sigma_v = C\tau$ when the bucket 'empties' during outage at full link rate, and $\sigma_v = \rho_v(T - \tau)$ when the bucket is 'filled in' between two consecutive outages. So, for this link fault scenario,

$$\sigma_v = C\tau \text{ and } \rho_v = C\tau/(T - \tau). \tag{8}$$

In general, equation (8) measuring the effective data link capacity $C(t)$ may not be practical, since it requires the link to be continuously active at full load. One approach is to have the link layer record link outages and their duration. Let $< t_i, \tau_i >_{i=1,\ldots,m}$ be the set of measured faults, where t_i represents the time when fault $i-1$ has completed and τ_i represents the fault i duration. Assuming that prior fault history is a predictor for the future behaviour of the wireless optical link, a leaky bucket model envelope can be defined as follows:

$$\sigma_v = C \max_{i=1,\ldots,m} \tau_i \text{ and } \rho_v = \max_{i=1,\ldots,m} \frac{\sigma_v}{t_{i+1} - t_i - \tau_i}. \tag{9}$$

2.7 Admission and adaptation

Applications or network management components submit admission requests to the BB. Admission control uses a 2-phase commit protocol, as described in Cardei et al. (2000). In the first phase resource (bandwidth) availability is assessed for the links along the source-destination route and, if necessary, lower priority flows are marked for adaptation. This would mean they would be switched to a different class or terminated. In the second phase, after receiving a commit message from the requester, admission control commits the reservation and notifies any adapted flows of their QoS change. If the requester cancels the flow admission, or its reply does not arrive in time, admission control will abort the admission and all reservations and adaptations will be rolled back – bringing the network resource state back to the initial state.

The following algorithm implements resource reservation, the first phase of admission control, and is run when the BB receives a request for admission, as in Section 2.2:

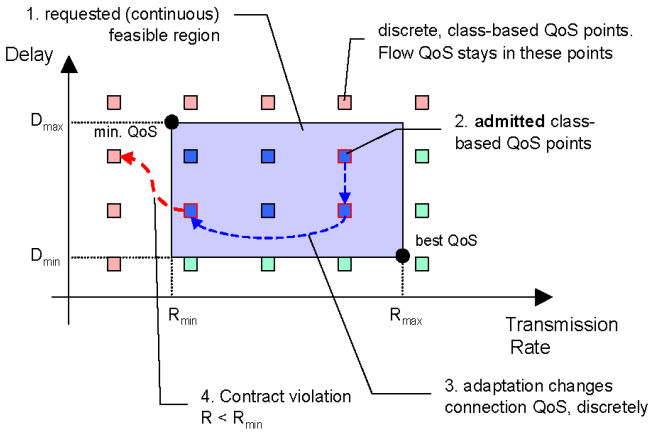
Resource Reservation:

1 select a class i from the set of classes whose corresponding $<M, D, R, E>_i$ fall within the requested QoS region.
2 if no such class exists, than selects a class i with $R_i \geq R_{min}, D_i \leq D_{max}, E_i \leq E_{max}$ and $M_i \geq M$.
3 if no such class exists, than
  3.1 admission is denied, and the negotiating party can resubmit the request with a different QoS demand. return FAILURE.
4 set $L = \varnothing$
5 for each link k along the source – destination path
  5.1 if n'_i, the number of class i flows on link k is at maximum, $n'_i = n_i$ (n_i from eq. 7)
    5.1.1 there is not enough bandwidth available on link k
    5.1.1 save link k: $L = L \cup \{k\}$
  5.2 else admit new flow on link k for class i: $n'_i = n'_i + 1$
6 if $L = \varnothing$, i.e., flow admitted on all links
  6.1 start timeout waiting for reply from requester
  6.2 return SUCCESS
7 else find the lowest priority flow f_a that spans at least all links from set L
8 if no such flow exist,
  8.1 return FAILURE.
9 **Adapt**(f_a)
10 start timeout waiting for reply from requester
11 return SUCCESS

The algorithm for adaptation (line 9) searches for an acceptable traffic class with bandwidth availability on flow f_a route or preempts it. These algorithms can be further improved such that link load is maximised and flow preemptions are limited.

An admission/adaptation scenario is shown in Figure 3. The application specifies a continuous QoS feasible region and admission control maps it into a set of discrete traffic classes. After admission the flow is assigned to a class (DSCP). During application execution, the flow can be adapted, its traffic class later changed within its acceptable QoS limits. Assuming all assumptions for correct operation are true (including traffic independence, limits on link outage duration), *contract violation* should not occur.

Figure 3 QoS mapping to traffic class and adaptation



The admission request is valid for a fixed time interval and must be renewed using a lease mechanism. A negotiated flow for which the lease expires ceases to receive QoS. Its packet classification policy will be revoked from the ingress router and it will be marked with best effort traffic.

Admission and adaptation are controlled by a set of policies designed to meet certain objectives. For instance, selection of flows to adapt/preempt in order to admit a higher priority flow can be based strictly on priority, or on a combination of priority, endpoint address, and packet content type. The range of policies depend on the network's main mission and are not elaborated in this paper.

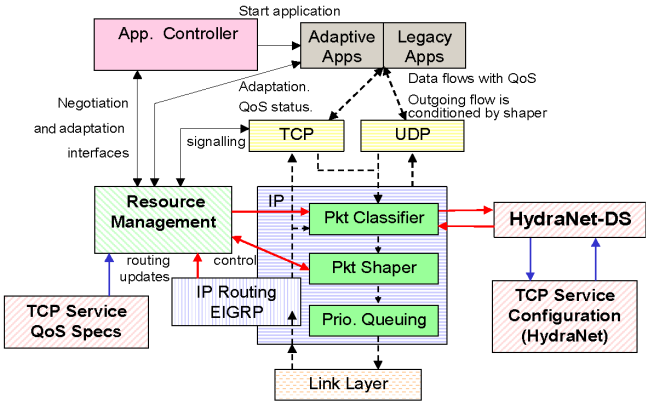
2.8 Protocol architecture

The OptiExpress architecture works with the standard IP protocol stack and routing protocols. Core routers are not directly connected to managed traffic endpoints. The only specific requirement for them is static priority packet scheduling at the IP layer to support delay guarantees. Figure 4 illustrates the protocol architecture for an edge router in the MOFS network involved in routing and also executing managed applications. Such a node could be aboard an aircraft, assisting with traffic relay and also transmitting sensor data to destinations on the ground. The *Resource Management* component:

- includes an API library for negotiation and adaptation with the BB
- controls policies and configuration for IP packet classification, shaping and queuing
- monitors link and routing state. Non-transient changes to node connectivity (e.g., new link established, or link down) are reported to the BB to speed up QoS reconfiguration.

The architecture components communicate with the BB using TCP connections.

Figure 4 Architecture of a MOFS node/edge router



NetEx Router Architecture

The *Application Controller* component interfaces OptiExpress with network management applications, or other QoS configuration platforms. Configuration tasks

include setting policies (e.g., for admission/adaptation) and configuration (class flow specs, link capacity partition per class). A network manager or an 'application manager' can submit flow admission requests on behalf of legacy applications that need delay guarantees.

The *Applications* component represent any application that transmits managed flows. QoS-aware, adaptive applications are designed to interface with OptiExpress. Legacy QoS-aware applications, such as H.323 video conferencing are provisioned managed flows through application/network management tools that translate application specific QoS metrics, such as frame rate, video frame size into the QoS parameters understood by OptiExpress. The work in Cardei et al. (2000) presents a case of QoS translation. Applications transmit managed TCP and UDP packet flows using the standard socket libraries.

The *IP* component does packet classification, policing and scheduling. The *classifier* is used to mark packets from managed flows at network ingress with the right DSCP, corresponding to the negotiated traffic class. The packet *shaper* enforces flow bandwidth according to the leaky bucket specification associated to the flow class. The packet *scheduler performs* static priority scheduling.

The *IP Routing Protocol* exports routing information to the BB. Routes are used for computing the safe utilisation level and the maximum per-class flow count n_i (equation (7)) during QoS reconfiguration. Currently, our architecture only supports single-path routing, although, multi-path routing with load balancing is possible with a tighter integration with the routing protocol.

Finally, the *HydraNet-DS* component is responsible for implementing fault-tolerant TCP services according to user-specified TCP service configuration describing server replication and the router-redirector topology.

The BB implements the Resource Management component and is executed as a TCP service, preferably on a MOFS router with high-bandwidth links and good connectivity. The BB needs current routing information to determine the network topology, and, therefore, interfaces with the routing protocol. The next section proposes some ways to improve the BB scalability and fault tolerance.

2.9 Bandwidth Broker scalability

The QoS reconfiguration process, involving delay and safe utilisation limit computation, as well as admission control need access to routing information and link capacities. In the current architecture these are executed as a TCP service on a central server, the BB. A distributed version for admission control is a challenging research subject.

To improve the scalability of centralised admission control, we propose to partition a large MOFS network into *QoS domains***,** each with a BB managing resources and providing end-to-end probabilistic delay guarantees withing its boundaries. For a flow that spans multiple QoS domains, the BB in the source QoS domain breaks down the end-to-end delay requirement into segments, one for each spanned QoS domain. The request is then passed for admission towards the BB from the destination domain. Each BB along the path processes the corresponding request. This protocol is similar to the admission/adaptation protocol from Dynamique (Cardei et al., 2004).

The MOFS network's dynamic topology may be partitioned. This would make our QoS architecture fail. To provide a degree of reliability, we consider implementing the BB service as replicated service using the HydraNet-DS framework for fault tolerant TCP services.

2.10 Provisioning support for legacy applications and subnetworks

One of the objectives of the QoS architecture was to support real-time communications for legacy applications – existing applications, ported to the MOFS network, that do not directly negotiate with OptiExpress or are not QoS aware. Since the mechanisms for delay guarantees are implemented inside the IP layer (classification/policing/scheduling), any IP packet flow can be effectively managed, provided the above IP functions are properly set up. We describe next three mechanisms designed for QoS provisioning:

- *QoS request negotiation*. Application controller negotiates a request for the duration of the flow. The request will be renewed periodically using the lease mechanism.
- *Automatic flow provisioning*. The application controller or a network manager sets up a policy for QoS with the BB that indicates a QoS admission request to be processed when a new flow meeting certain criteria is encountered. Flow descriptors may include transport protocol, source/destination address/port number, date-time, etc. When an ingress edge router classifier identifies a flow according to these rules, it signals the BB. If admission succeeds, the flow will have delay guarantees. Otherwise, the flow will be assigned to 'best effort' class or will be denied admission – dropped, depending on policy. A flow that stays idle for a configurable amount of time will experience a switch to best effort (deallocation of resources) or termination.
- *Persistent flow provisioning*. With this mechanism flow resources are assigned permanently, or preallocated. Persistent flows do not 'expire' and do not require lease renewal. This mechanism can be used for low data rate, very low latency flows, for which negotiation overhead must be avoided. The OptiExpress signaling protocol uses persistent flows.

All three mechanisms can be used for legacy applications. In addition, using automatic or persistent flow provisioning, traffic generated from a subnetwork that is not part of a BB's QoS domain can be managed starting from the ingress point in the MOFS network. Classification policies at the ingress router allow association of packets to flows from the subnetwork. Through marking with the proper class' DSCP

these packets will be provisioned the same degree of QoS as flows that undergo the regular negotiation process.

2.11 Architecture issues

There are several issues with the approach for QoS guarantees presented in this section.

- One concern is that the method for computing the safe utilisation limit ν, artificially lowers the maximum link utilisation because ν is computed to satisfy all links in the network – possibly fitting one critical link and underestimating utilisation for all others. With information on traffic patterns, safe utilisation limits can be computed for different regions of the network, considering also link capacity.
- Dependency on routing information for ν computation requires a centralised solution. This is the trade-off for having UBAC, with a cheap admission control process that simply limits the number of flows per class on each link. Route information is necessary in order to avoid storing flow state inside the network. With UBAC, flow state and reservation state is maintained only by the BB.
- Scalability. For large networks, the complexity of the QoS reconfiguration process may cause high overhead and increased latency penalties during transient periods when routes have not converged yet. We present in Section 2.9 an approach for improving scalability by partitioning the MOFS network into QoS domains.

3 Architecture for fault-tolerant TCP services

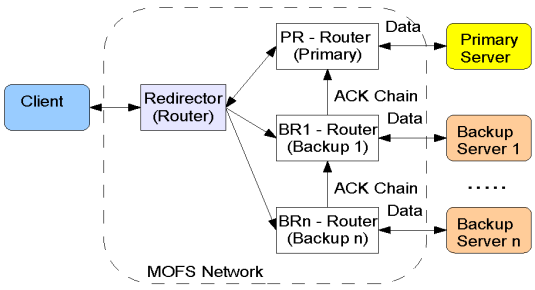
Service replication is an established method for improving reliability of critical applications in scenarios where servers and network connections can fail. In the MOFS network connections may experience failures due to weather effects that may not be mitigated quickly enough by rerouting. We present a primary-backup scheme, HydraNet- DS derived from HydraNet-FT (Shenoy et al., 2000). In HydraNet, a TCP service is replicated on multiple host servers distributed in the network. One of the replicas is designated as a primary server. Clients across the network connect to the primary server.

A TCP-based failure detection mechanism allows a backup to quickly take over the primary's role in case the client-primary connection breaks or QoS drops below a threshold. A key feature of HydraNet is that the replication and failover are entirely transparent to client and server processes. With our extensions, replication is transparent to the servers as well, requiring no changes to the machines or to the server software.

3.1 Service replication architecture

Figure 5 shows a typical HydraNet-DS architecture. This architecture implements a TCP fault-tolerant service by replicating the server application on multiple machines. All replicas bind to the same TCP port. Clients open one or more TCP connections to the designated primary server addresses. The service replication architecture implements atomic, ordered, one-to-many communication from clients to the server group and many-to-one communication from servers to the clients.

Figure 5 HydraNet – DS architecture



The supporting infrastructure consists of Redirectors (routers which connect clients to the MOFS network), the router connecting the Primary Server to the MOFS network (PR) and routers connecting backup servers to the network (BR_i).

A Redirector intercepts packets sent by a client to the primary server. The Redirector has a redirection table describing SAPs for each replicated service. When it intercepts a packet from a client, the Redirector checks the destination address/port number against the redirection table. If there is a match with a replicated service, the packet will be forwarded towards the specified primary server and tunneled to the backup server-side routers, $BR_{1,\ldots,n}$, using IP-in-IP encapsulation. If no match, the packet will be forwarded to the next hop, according to routing table entries.

A primary/backup server-side router connects the server replicas to the network. In addition to a physical network interface, which is exposed to the MOFS network, all backup server hosts configure one (possibly virtual) interface to have the same IP address as the primary server. The replica service binds to this address on all server hosts. The backup server-side routers keep this address private and use it only to communicate packets belonging to a configured replicated service. The backup server-side routers receive the tunnelled packets from the clients, extracts and forwards them to the backup servers.

3.2 Server coordination and failover

For correct and transparent operation, the communication between clients and servers must employ an atomic, ordered communication protocol that follows these rules:

- packets must be delivered at all active service replicas or at none
- all active service replicas must receive packets from a clients in the same order
- a packet from the primary server can be delivered to the client only after all backup servers have replied
- a client should receive a reply packet only from the primary server.

With these rules, all replicas operate in virtual synchrony and total client transparency is achieved. As we will see, replication is also transparent to server hosts and services.

As in HydraNet-FT, our implementation builds an Acknowledgment (ACK) chain. But instead of host servers, we put the primary and backup server-side routers in the ACK chain: $BR_n \rightarrow BR_{n-1} \rightarrow \cdots \rightarrow BR_1 \rightarrow PR$. All chain routers receive packets from clients, but only BR_n immediately extracts the packet and forwards it to replica n. When BR_n receives a TCP ACK from replica n, it forwards only the ACK number to BR_{n-1}, on the ACK chain. Now can BR_{n-1} forward the client's packet to replica $n-1$. This process $delivery_i \rightarrow ACK_i \rightarrow forward_{i-1}$ propagates down the chain. Router PR eventually delivers the packet to the primary server. This process guarantees that the primary server receives a packet only after confirmed delivery on all other replicas. If a backup server misses a packet or its ACK is lost before it reaches the backup server-side router, than the client will eventually timeout and retransmit. Server-side routers will forward the retransmitted packet as described before, but the TCP stack on the host servers will detect the retransmission and will prevent repeat delivery to the service application. The ACK messages will be, thus, retransmitted.

The handling of server response is similar in the ways it uses the TCP ACK chain for coordination. When BR_n receives a TCP packet from replica n, it strips the TCP sequence number and it sends it to BR_{n-1}. Only when BR has received the message with the sequence number for a byte k from BR_{i+1}, and also a packet with a byte k from replica i, can BR forward k's sequence number up the chain, towards PR. When PR receives this sequence number it will forward the corresponding packet to the client after the primary replies. Only the packets sent from the primary server will be transmitted to the client. Packets originating from backup servers will not be transmitted on the MOFS network. We note that server-side routers must translate the sequence number of packets from the replicas because each TCP connection has a different initial sequence number.

This protocol makes sure that a client receives a packet only after all active replicas have processed the previous client packets, and all replicas have transmitted a corresponding reply packet (atomicity). Message ordering is provided by the TCP sequence number mechanism employed by the TCP stack on the host servers.

Replica failover is based on server-side routers and redirectors detecting repeat retransmissions. If a packet is lost, a replica service or a router fails, the TCP flow control will detect a timeout and will retransmit. The routers on the ACK chain identify the failure point by analysing repeated retransmissions and comparing ACK and sequence numbers. Using a signalling protocol, server-side routers and the redirectors reconfigure the replicated service by isolating the failure point and removing it from the ACK chain. If the primary server fails current TCP connections will not change on the client side. On the ACK chain, BR1 will become the new primary server. New client connections will go directly to this server.

Backup server-side routers keep the reply packets from backup servers outside the MOFS network. The only packets from the replicas that spill into the MOFS network consist only of TCP flow control messages.

Operation of the service replication transparent on multiple levels. First, clients know about the primary server only and open connections to the primary server. Server applications (TCP services) are not aware of the replication scheme and can execute without any changes. They bind to a common server address on the service port number(s). This server address is supported by a physical interface, or a virtual interface (multi-homing). Incoming TCP connections are handled through the server-side routers that hide the fact that connections are tunnelled and flow control messages pass through the ACK chain.

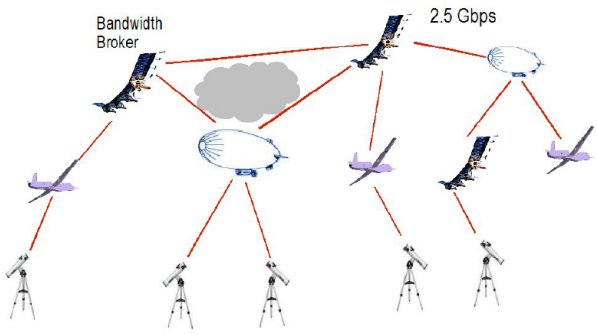
3.3 QoS-enabled TCP services

We integrated HydraNet-DS, the fault-tolerant TCP services mechanism, with the OptiExpress QoS protocols. The main motivation was to estimate the improvement in end-to-end application performance, if all packet flows receive better QoS from the network. We configured the TCP ACK chain to be pre-provisioned with a high-priority, low delay traffic class. The TCP connection between client and the primary server-side router and the IP-in-IP tunnels have been configured with automatic provisioning (Section 2.10) based on service description that includes primary server IP address, port numbers and QoS specification for client connections – (M, R, D, E). Next section shows performance results for simulations with a HTTP replicated service.

4 Performance evaluation

We present in this section a summary of the QoS architecture performance results we measured with OPNET simulations. We used MOFS topologies similar to the one in Figure 6. All links have a 2.5 Gbps capacity and experience intermittent link faults of variable duration. The network runs the EIGRP routing protocol, configured to test neighbour reachability every 250 ms. The IP forwarding code has been modified to hold off transmitting packets to the link layer when the outgoing link is interrupted.

Figure 6 Topology for simulation scenario



The ground nodes submit admission requests for managed flows connecting other ground nodes. The network is also loaded with best effort traffic (http and e-mail). In all, we focused on application performance under heavy loaded, as this is when a QoS architecture provides most value.

The safe utilisation bound for real-time flows with independent link faults of 10 ms (1.5 s interarrival time) was 80.55%. With 25 ms faults, the utilisation limit decreased to 74.98%. With faults longer than 50 ms, the utilisation limit drops gradually from 72% to 0%. Longer link faults increase the queuing delay for packets arriving during a fault and later, as the queue needs extra time to flush. Faults also affect packets that were in transit when the outage started

The average, maximum end-to-end delays and the standard deviation are show in Figure 7 for a four scenario combination: QoS enabled/disabled, with/without link faults (10 ms duration/1.5 s interarrival time). When QoS is disabled all the same traffic flows (including OptiExpress signaling) are scheduled with FIFO policy and delay cannot be guaranteed. The maximum delay dropped from 205 ms to 24 ms when OptiExpress was enabled (with link faults). Similar improvement scale can be noticed for both average delay and jitter (stdev). All scenarios were run with the network saturated with best effort traffic. Per class average delay is shown in Figure 8. The delay is considerably lower when traffic is provisioned QoS, as expected. Priority scheduling gives precedence to real-time traffic flows, reducing the queuing delay, at the expense of best effort traffic.

Figure 7 End-to-end delay for managed flows

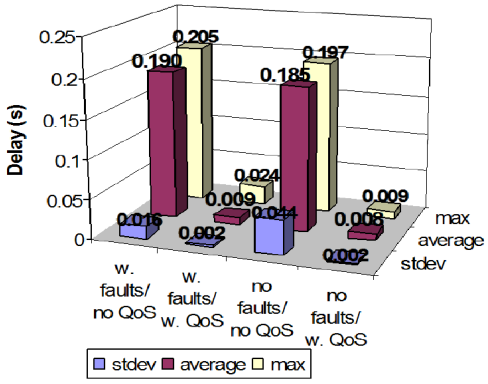
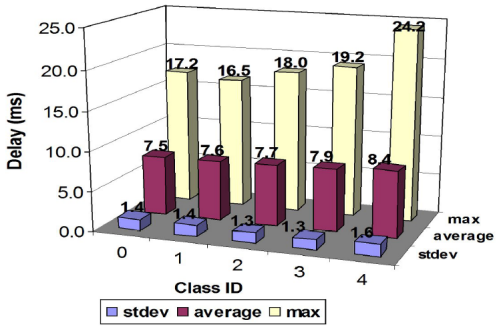


Figure 8 Per class end-to-end delay



Next we measured the performance improvement for applications that are not QoS aware ('legacy applications'). For a video conferencing application (500 kbps CBR 20 fps), configured as a legacy application the average delay dropped from 221 ms to 6.95 ms when OptiExpress automatically provisioned a flow with class 0. The jitter dropped eight times to 4.46 ms with QoS.

The BB was set to do a QoS reconfiguration whenever routes changed, as indicated by the routing tables exported by EIGRP. During reconfiguration, the BB recomputes the safe utilisation bounds for all links, recomputes admission for all flows and signals QoS adaptation to OptiExpress components on hosts and edge routers. We measured an average reconfiguration delay of 337 ms, including EIGRP route convergence time. An objective is to further reduce reconfiguration delay, as during this period some flows affected by rerouting may suffer delay violations. The reconfiguration delay consists mostly of the time needed to obtain information on topology changes.

The average admission delay (success or failed) was 115.89 ms in a scenario with 10 ms link faults. The OptiExpress signaling protocol overhead was measured in sustained load conditions with 68 admission requests per second. The measured aggregated peak overhead reached 3540 kbps (1.5% of link capacity), while the average overhead was 206 kbps (0.0082% of link capacity). Peak overhead measured transmissions related to admission requests and the admission 2-phase commit protocol. The overhead includes flow lease renewals every 60 s. We notice that the mean protocol overhead for signalling is very low when compared to the link capacity.

Another performance metric we focused on is the Request Satisfaction Ratio (RSR). This indicator measures the fraction of flows in the entire network that experienced contract violation during the simulation (packets delivered later than the negotiated delay). For all the experiments the RSR was 100% when QoS was enabled. In comparison, with QoS disabled, the RSR varied between 66% and 93%. More affected were the classes with lower delay.

4.1 Performance of QoS-enabled TCP services

In this subsection we present performance results for the fault-tolerant TCP service mechanism. A HTTP TCP service has been configured to run as a HydraNet-DS replicated services. We defined scenarios with a primary and one backup server. We first compare the HTTP client performance in terms of page load delay, TCP delay for scenarios where service replication and QoS are alternatively enabled/disabled. Each scenario is run with either no background traffic or with a heavy traffic load that uses all available capacity.

Table 2 shows performance numbers for HydraNet-DS for a HTTP service with one primary and one backup server. We measured the average delay for loading a complete 5-object HTML web page, the TCP segment delay and the delay penalty from the ACK chain. The web

page consists of 100 kB HTML code and five objects (e.g., images) of an average of 100 kB per object.

Table 2 HydraNet-DS performance (ms)

Background load	*HydraNet configuration*	*Page delay*	*TCP delay*	*TCP segment delay*	*ACK chain delay penalty*
0% load	Disabled	1326	301	3.669	0
	Enabled	1348	307	6.350	5.796
100% load	Disabled	2177	537	28.388	0
	Enabled	2614	703	66.516	25.062
	Enabled with QoS	1380	310	10.000	11.573

We first notice that the increase in page loading delay from replication is 22 ms (1.65%). Under a heavy load the page load delay penalty from replication rises to 20%. Once we enable QoS, the page load delay drops to 1380 ms, with an increase of just 4% relative to the scenario without QoS where the network is not utilised. The QoS mechanism reduces the delay penalty from the ACK chain by 2.16 times when compared to the case with no QoS.

We also measured the overhead from the ACK chain. The average throughput between the backup and primary server-side routers was limited to 26 kbps, while the average http flow throughput measured 5.7 Mbps. The measured TCP open connection delay penalty was measured to 182 μs and the failover delay is 53 ms when QoS was enabled.

We conclude that the OptiExpress QoS system improves communication performance by reducing the end-to-end delay. Improvements are consistent (8.5 reduction in maximum end-to-end delay) in networks that experience a heavy load, while the overhead from admission control and adaptation is low even at peak operation. Permitted link utilisation by real-time flows drops rapidly when link faults exceed 50 ms.

The TCP fault-tolerant service mechanism incurs relatively little delay penalty when background traffic utilisation is low, but increases to up to 20% under heavy load. QoS provisioning for all TCP flows, redirector-router tunnels and the ACK chain reduces the delay penalty measured at clients to 4% for the HTTP scenario run with a replicated service.

5 Conclusions

In the near future Mobile Optical Free Space networks have the potential to provide low-delay and high-speed connectivity over large geographical areas using IP routed multihop topologies. The dynamic quality of the wireless optical link hinders deployment of real-time distributed applications demanding end-to-end delay guarantees. We present in this paper an architecture for Quality of Service and a mechanism for TCP service replication in MOFS networks. The QoS architecture implements statistical delay guarantees for end-to-end communication. The architecture employs a model with statistical service curves representing link capacity and traffic arrival. A model for virtual traffic arrival estimates the capacity variation of the optical links due to atmospheric effects, and is integrated with delay violation computation. An UBAC technique limits the capacity that can be allocated to real-time traffic on each link in order to limit queuing delays. Application controllers interface with the BB through a 2-phase commit protocol.

The HydraNet-DS architecture for reliable TCP services uses a primary-backup scheme with transparent deployment and failover for clients and server replicas. Our solution differentiates from previous work by including in the ACK chain edge routers connecting the server replicas to the MOFS network. The TCP connections, IP-in-IP packet tunnels and the point-to-point flows that make up the ACK chain have been configured in simulation experiments for QoS. Simulations with OPNET for various topologies and application scenarios have demonstrated reduced end-to-end delay and improved application performance.

The OptiExpress protocols can be further improved. The centralised BB can be replaced with a distributed architecture that uses RSVP or MPLS for enforcing link utilisation limits. To improve architecture reliability we consider replicating the BB service with HydraNet-DS.

Acknowledgement

This material is based upon work supported by the US. Air Force and DARPA under Contract No. F33615-02-C-1247. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the USA. Air Force and DARPA.

References

Black, D., Carlson, M., Davies, E., Wang, Z. and Weiss, W. (1998) *An Architecture for Differentiated Services*, RFC 2475, http://www.ietf.org/rfc/rfc2475.txt

Cardei, I., Jha, R., Cardei, M. and Pavan, A. (2000) 'Hierarchical architecture for real-time adaptive resource management', *The IFIP/ACM International Conference on Distributed Systems Platforms and Open Distributed Processing*, April, New York, USA.

Cardei, I., Varadarajan, S., Pavan, A., Graba, L., Cardei, M. and Min, M. (2004) 'Resource management for ad-hoc wireless networks with cluster organisation', *Journal of Cluster Computing in the Internet*, Kluwer Academic Publishers, Vol. 7, No. 1, January, pp.91–103.

Cardei, I., Pavan, A. and Bettati, R. (2005) 'Architecture for delay-sensitive communication in mobile optical free-space networks', To appear in *Proc. 2nd IEEE International Conference on Mobile Ad-Hoc and Sensor Systems*, November, Washington DC.

Chaporkar, P. and Sarkar, S. (2002) 'Providing stochastic delay guarantees through channel characteristics based resource reservation in wireless network', *Proc. 5th ACM Int. Workshop on Wireless Mobile Multimedia*, Atlanta, Georgia, 28 September.

Chenxi, Z. and Corson, M.S. (2002) 'QoS routing for mobile ad hoc networks', *INFOCOM 2002, Proc. 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 2, 23–27 June, New York, NY, USA.

Cisco (2002) *Enhanced Interior Gateway Routing Protocol*, http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/en_igrp.htm

Communication from ITT Industries (2002) *Personal Communication from ITT Industries in 2002*.

Knightly, E. (1998) 'Enforceable quality of service guarantees for bursty traffic streams', *Proc. IEEE Infocom*, March, San Francisco, CA, USA.

Lee, S.B., Ahn, G.S. and Campbell, A.T. (2001) 'Improving UDP and TCP performance in mobile ad hoc networks with INSIGNIA', *IEEE Communications Magazine*, Vol. 39, No. 6, June, pp.156–165.

Liebeherr, J., Wrege, D. and Ferrari, D. (1996) 'Exact admission control in networks with bounded delay services', *IEEE/ACM Transactions on Networking*, Vol. 4, No. 6, December, pp.885–901.

Marwah, M., Mishra, S. and Fetzer, C. (2003) 'TCP server fault tolerance using connection migration to a backup server', *Proc. IEEE Int. Conf on Dependable Systems and Networks*, June, San Francisco, CA.

Shenoy, G., Satapati, S.K. and Bettati, R. (2000) 'HYDRANET-FT: network support for dependable services', *The 20th Int. Conf. on Distributed Computing Systems*, April, Taipei.

Shigang, C. and Nahrstedt, K. (1999) 'Distributed quality-ofservice routing in ad hoc networks', *IEEE JSAC*, Vol. 17, No. 8, August, pp.1488–1505.

Stadler, B. and Duchak, G. (2004) 'Terahertz operational reachback (THOR) a mobile free space optical network technology program', *Proc. IEEE Aerospace Conference*, Big Sky, Montana.

Sultan, F., Srinivasan, K., Iyer, D. and Iftode, L. (2002) 'Migratory TCP: connection migration for service continuity over the internet', *Proc. 22th IEEE Int. Conference on Distributed Computing Systems*, July, Vienna, Austria.

Wang, S., Xuan, D., Bettati, R. and Zhao, W. (2001) 'Providing absolute differentiated services for real-time applications in static-priority scheduling networks', *Proc. IEEE Infocom*, Anchorage, AK, USA.

Wang, S., Nathuji, R., Bettati, R. and Zhao, W. (2004) 'Providing statistical delay guarantees in wireless networks', *Proc. IEEE International Conference on Distributed Computing Systems*, March, Tokyo, Japan.

Zagorodnov, D., Marzullo, K., Alvisi, L. and Bressoud, T.C. (2003) 'Engineering fault-tolerant TCP/IP servers using FTTCP', *Proceedings of the 2003 International Conference on Dependable Systems and Networks (DSN 2003)*, 22nd–25th June, San Francisco, CA.

Zhuang, J., Casey, M.J., Milner, S.D., Gabriel, S.A. and Baecher, G. (2004) 'Multi-objective optimisation techniques in topology control of free space optical networks', *Proc. IEEE MILCOM*, November, Monterey, CA.

Websites

Stratellite airship, http://www.stratellite.net/

Helios solar-powered UAV, http://www.nasa.gov/centers/dryden/history/pastprojects/Era st/helios.html

OPNET, http://www.opnet.com

RFC 2675 – IPv6 Jumbograms, http://www.faqs.org/rfcs/rfc2675.html

WiMAX, http://www.wimaxforum.org/home