Technical Report

Perspectives of QoS routing in the Internet:

Preliminary Study

Peng Zhang

Laboratory of Telecommunication Technology,

Helsinki University of Technology

Email: Pgzhang@tct.hut.fi

Tel: +358 9 4515454

1999-09-09

Abstract

The emerging Gigabit-per-second high-speed switching technologies and the rapid growth of the Internet enable IP_based networks to support a variety of services in particular for requests with specific QoS constraints, e.g., distributed multimedia communication applications. These requirements for services might contain a number of QoS constraints possibly including delay, delay jitter, loss ratio, bandwidth and so on, which raise new challenges for the next-generation high-speed networks. Among them, one of the key issues is QoS routing, whose basic function is to find feasible paths under a single or multiple QoS constraints. It is becoming one of the key components to build up a QoS_based network, and one of the most active areas in the Internet Community. Many recent work focus on this area by presenting a number of unicast or multicast QoS routing algorithms. However, there still lack the overall and thorough considerations, even judgements on this area. In this report, we attempt to present a thorough consideration on QoS routing as well as QoS control in the Internet. We first give the background information on QoS routing in the Internet. Then, we investigate issues involved in QoS routing in the Internet, identify basic requirements for QoS routing and examine the feasibility of QoS routing in respect of performance/cost ratio. We then present each class of QoS routing algorithms. Finally, we outline our future work on QoS routing in the Internet. As an important assumption, we believe the benefit of QoS routing overwhelms its complexity and cost under the suitable configuration and design for the next-generation high-speed networks.

1 Introduction

The current global Internet based on IP protocol supports only the best effort service, i.e., network resources are contended and fairly shared by all traffic injected into the networks. Data packets of a session may follow different paths to the destination. However, with the success of Internet in recent years, IP networks are also expected to support various services, not only the traditional services (e.g., email, ftp), but also the upcoming high-speed and real-time services (e.g., audio-video real-time transmission, virtual private networks). The latter ones exhibit much different traffic characteristics from the former ones in terms of bit rate and burst, and they require fixed QoS assurances in the duration of transmission. Considering this problem, how to support the QoS requirements is becoming a hot topic in the Internet community[J1~J4].

QoS_based routing, however, has been recognized as a missing part in the evolution of QoS_based service offerings in the Internet [I1~I4]. The basic function of QoS routing is to find such a feasible path (tree) that has the sufficient residual resources to satisfy the QoS constraints of a connection. The objectives of QoS_based routing scheme is to aid the efficient utilization of network resources by improving the total network throughput. Moreover, QoS_based routing provides flexibility in support for various service requirements by customers.

More and more attentions are attracted to the field of QoS routing: the QoS Routing Workgroup is established in IETF and has fulfilled a RFC on the general framework for QoS routing in the Internet[I1]; many other work on this area have presented a number of unicast or multicast QoS routing algorithms [J7~J15].

Nevertheless, there still lack the detailed and complete considerations in this area. This report aims to investigate the overall issues involved in QoS routing problem. In particular, we attempt to examine the necessity and feasibility of QoS routing in the existing IP networks. The rest of this report is organized as followed. In next section, the background information on QoS routing in the Internet is introduced. We firstly present the QoS requirements for Internet and candidates of QoS architectures. Then we show the issues involved in QoS routing. We give the QoS routing model and its one example implementation. In sections 3, we present the QoS routing problems and its classification.

We summarize the state of art of QoS routing strategies in section 4, the possible extension of this topic is also presented in this section. In the final section, we outline our succeeding steps on this field.

2 QoS Routing in the Internet

2.1 QoS in the Internet

2.1.1 What compels QoS in the Internet?

In general, two major strengths drive the need for QoS in the Internet [J1-J3].

One strength comes from the development of technologies in the fields of high-speed networks, image processing, and audio/video compression. These developments achieved make it possible to support multimedia traffic in the communication networks.

Another strength derives from the need in market for various high-speed, delay sensitive services offered by network providers. The distributed multimedia applications with in-expensive price and quality assurance are in particular required by the customers. The need in market increases so fast that strongly drives the need for QoS in the Internet.

2.1.2 QoS Provisioning Steps and components [J2 – J5]

In order to provide QoS in the Internet, by using resource reservation and scheduling, the following steps must be performed in turn at each system and component participating in the end-to-end application:

- *QoS Requirements*: the expected QoS must be specified to enable the system to determine whether and which QoS can be provided.

- *QoS calculation*: When an application issues its QoS requirements, the admission control of the system must check whether these demands can be satisfied taking existing reservations into account. If so, the best-possible QoS which can be provided is calculated and the application is given a certain QoS guarantee accordingly.

- *Resource Reservation*: According to the QoS guarantees given, appropriate resource capacities, e.g., transmission or processing bandwidth, must be reserved.

- *Enforcement of QoS guarantees*: The guarantees must be enforced by the appropriate scheduling of resource access. For instance, an application with a short guaranteed delay must be served prior to an application with a less strict delay bound.

This functionality can be divided into two distinct phases. The set-up phase (also called `*QoS negotiation*') consists of the first three steps. The specified QoS requirements are used for capacity test and QoS computation which finally results either in resource reservation or in rejection of the reservation attempt if the QoS cannot be met (due to a lack of resources). After the negotiation phase has been successfully completed, in the data transmission phase, the resources used to process the user data are scheduled with respect to the reserved resources (also called `*QoS enforcement*').

If a connection-oriented approach for the provisioning of QoS during data transmission is used, the QoS negotiation steps are typically part of the connection setup. If no connections but (soft-state based) flows are used, these steps are performed as part of the flow setup, they mark nevertheless the beginning of QoS support for data transmission because no QoS can be provided without these reservation.

Overall, several resource management components interact to provide QoS assurance: Applications, QoS translators, admission control, resource scheduler. Additionally, further components are needed, for example, a resource reservation protocol to communicate QoS specifications among participating systems and a resource monitor which measures the availability of resources and whether indeed the promised QoS is provided.

2.1.3 QoS Requirements

An application of multimedia communication service may have several requirements, which can be divided into traffic and functional requirements [J4]. The functional requirements are multicast transmission and the ability to define coordinated set of unidirectional streams. The traffic requirements include transmission bandwidth, delay, delay jitter and reliability. They depend on the used kind, number, and quality of data streams. In general, the traffic requirements are mostly concerned and they can be satisfied by the use of resource management mechanisms. There are a number of parameters of traffic requirements:

- *Bandwidth*, as the most prominent QoS parameter, specifies how much data (maximum or average) is to be transferred within the network system. In general, it is not sufficient to specify the rate only in terms of bits, as the QoS scheme shall be applicable to various networks as well as to general-purpose end-systems. For instance, in the context of protocol processing, issues like buffer management, timer management, and the retrieval of control information play an important role. The costs of these operations are all related to the number of packets processed (and are mostly independent of the packet size), emphasizing the importance of a packet-oriented specification of the data rate. Information about the packetization can be given by specifying the maximum and the average packet size and the packet rate.

- *Delay* as the second parameter specifies the maximum delay observed by a data unit on an end-to-end transmission.

- *Delay jitter* measures the delay variance. It is the result from varying delays during processing and transmitting the data. It can be smoothed by buffering at the receiver side which, however, increases the end-to-end delay.

- *Reliability* pertains to the loss and corruption of data. To some extent, it can be represented by the loss probability.

The flow QoS requirements can be represented by a set of constraints, i.e., link constraints, path constraints or tree constraints [J7]. A link constraint specify a restriction on the use of links, e.g., bandwidth and delay. A path/tree constraint specifies the end-to-end QoS requirement on a single path or a entire tree.

2.1.4 QoS Architectures [J2-J3]

In order to provide support for QoS in an overall framework, a general QoS architecture is needed. The objectives of QoS architecture is to define a set of quality of service configurable interfaces that formalize quality of service in the end-system as well as network, providing a framework for the integration of quality of service control and management mechanisms. Since meeting QoS guarantees is fundamentally an end-to-end issue, that is, from application-to-application, the QoS architecture essentially comprises of QoS control and management in network as well as in end-system. Hence, some QoS architectures are generated to consider both under the same ceiling. Meanwhile, numerous important works within the network region have been carried on, for instance, the Integrated Service (IntServ) and Differentiated Services (DiffServ) in IETF, and TINA in ITU-T. These architectures are briefly introduced as follows.

- Integrated Services [I5-I6]

The Integrated Services, in relation with the RSVP protocol, provide a general solution for QoS guarantees in the future Internet by defining the int-serv architecture with the QoS framework. The RSVP protocol is used to transport flow specifications (FlowSpecs) that adhere to Intserv rules. There are two types of descriptions used for the QoS specification: the traffic specification (TSpec) describes the behavior and characteristics of a flow, and the service request specification (RSpec) describes the service requested under the condition that the flow adheres to the restrictions of TSpec.

On the basis of TSpec and RSpec, the following int-serv services are offered in addition to best effort: controlled load service, which attempts to provide several delay which the application can choose from; guaranteed service, which provides an absolute guaranteed delay bound.

Quality of service is implemented for a particular data flow by mechanisms collectively called "traffic control". The IntServ architecture is restricted to the network but can be applicable in the end-system. These mechanisms comprises of four components [I6] :

- a packet classifier, which determines the QoS class (and perhaps the route) for each packet.
- a packet scheduler, which forwards packets streams using a set of queues and timers;
- an admission controller, which determines whether a new flow can be admitted or denied;
- a reservation setup protocol, i.e., RSVP, which is used to create and maintain flow-specific state in the routers along the path of the flow.

Especially, reservation protocols are needed to exchange and negotiate QoS requirements between the participating end systems and routers. Reservation protocols are only the vehicles to transfer information about resource requirements and to negotiate QoS values between the end-systems and the intermediate network routers -- they leave the reservation itself to local resource management modules. However, RSVP is not a routing protocol, but depends on a separate routing protocol. Nevertheless, the network nodes need to know always the paths of data flows for making reservations, e.g., for physical transmission lines with asymmetric capacity.

- Differentiated Services [J17, I7-I9]

The differentiated services architecture is based on a simple model where traffic entering a network is classified and possibly conditioned at the boundaries of the network, and assigned to different behavior aggregates. It is composed of a number of functional elements implemented in network nodes, including a small set of per-hop forwarding behaviors, packet classification functions, and traffic conditioning functions including metering, marking, shaping, and policing. This architecture achieves scalability by implementing complex classification and conditioning functions only at network boundary nodes, and by applying per-hop behaviors to aggregates of traffic which have been appropriately marked using the DS field in the IPv4 or IPv6 headers. Per-hop behaviors are defined to permit a reasonably granular means of allocating buffer and bandwidth resources at each node among competing traffic streams. Per-application flow or per-customer forwarding state need not be maintained within the core of the network. Differentiated services mechanisms can be utilized to aggregate Integrated Services/RSVP state in the core of the network.

- TINA [J18]

The TINA QoS Framework describes a framework for specifying QoS aspects of distributed telecommunications within the context of the Computing Architecture. The QoS Framework address the context of the computing and engineering viewpoints of distributed telecommunication applications. Multimedia services utilize the Distributed Processing Environment (DPE) and underlying communication capability. Therefore, by stating QoS requirements through DPE, QoS of applications can be supported with the relief of complex resource management mechanism. The TINA QoS framework is partly based on work in the literature of ANSA QoS Framework and CNET Framework.

The TINA QoS Framework provide a means for supporting QoS guarantee in the realm of telecommunication management networks (TMN), which enable an unified interface on applications for customers. It is also possible to comprise of various QoS architectures into a single architecture, e.g., Quality of Management Network [J19].

- QoS-A [J20]

The Quality of Service Architecture (QoS-A) is a layered architecture of services and mechanisms for quality of service management and control of continuous media flows in multiservice networks. The architecture incorporates the following key notions: flows, which characterize the production, transmission and eventual consumption of single media streams (both unicast and multicast) with associated QoS; service contracts, which are binding agreements of QoS levels between users and providers; and flow management, which provides for the monitoring and maintenance of the contracted QoS levels. The realization of the flow concept demands active QoS management and tight integration between device management, end-system thread scheduling, communications protocols and networks.

Discussions:

Although the above architectures hold the similar objectives, they are different in many aspects. First, each architecture offers a different set of services capability to applications. For example, IntServ supports Controlled-load services and Guaranteeded Services, while DiffServ provides a wide range of services with respect to requirements. Second, the scope of these architectures concerned is different , for example, the work in IETF (Intserv and DiffServ) does not specify the part of QoS control (e.g., flow scheduling, flow shaping, flow synchronization etc) in the end-system, however QoS-A addresses them in detail. Third, there exist differences in the methods to set up the flow connection and reserve the resource. For example, IntServ and DiffServ utilizes the Soft-State, while QoS-A adopts the hard-state solution to network level QoS provision. Moreover, IntServ is much different from DiffServ that IntServ performs traffic classification on the basis of hop-by-hop, while DiffServ performs it at the edge of the network.

Overall, all above architectures anticipate for the QoS routing capability in their frameworks. For instance, in IntServ architecture, TSpec and RSpec of a flow will impose the network to find a path which can satisfy these specifications. Otherwise, the flow may experience performance degradation or rejection to set up a QoS_guaranteeded flow path even with sufficient resources in other paths. It is obvious that resource reservation is closely related to QoS routing.

2.2 Background on QoS Routing

2.2.1 QoS routing vs. best-effort routing

In traditional data networks, routing is primarily concerned with connectivity.

Current routing protocols (e.g., OSPF and RIP), usually characterize the network with a single metric such as hop-count or delay and use shortest-path algorithms for path computation. These protocols do not use alternate paths with acceptable but non-optimal cost to route traffic.

In comparison with best-effort routing, QoS Routing support traffic using various services with requirements for additional routing metrics, e.g., delay, and available bandwidth. QoS routing also provides support for alternate routing. If the best existing path cannot admit a new flow, the associated traffic can be forwarded in an adequate alternate path. QoS routing algorithms can prevent traffic shifting from one path to another "better" path only if the current path meet the service requirements of the existing traffic.

2.2.2 Benefit and Cost of QoS routing [J8]

QoS routing determines the paths for flows under the knowledge of network resource availability, as well as the requirements of flows. It aims to dynamically choose a feasible path from multiple choices, according to and policy constraints, such as path cost, provider selection, etc. As a result, the performance of applications is guaranteeded or improved in comparison with that without QoS routing. Meanwhile, QoS routing optimize the resource usage in the network by improving the total network throughput.

However, these benefits of QoS routing also incur the cost of developing new routing protocols or extending the existing ones. Moreover, it potentially increases higher communication, processing and storage overheads. QoS routing raises some following issues [I1]:

- How do routers determine QoS capability and reserve resource?
- How do routers determine QoS_based paths computed for unicast/multicast flows?
- How is scalability achieved?
- ...

2.2.3 QoS routing and RSVP [J10]

RSVP is a reservation protocol in the Internet suite, which can be used in conjunction with QoS routing. Specially, RSVP PATH messages can serve as the trigger to query QoS routing. During the processing of RSVP PATH messages, RSVP queries

QoS routing to obtain the next hop for forwarding the PATH message. The PATH message is then forwarded on the interface returned by QoS routing.

On the other hand, because of the variations in the availability of resources in the network, routes between the same source and destination and for the same QoS, may often differ depending on when the request is made. Thus, it is important to "*pine*" or "*unpine*" the path. Path pinning or unpinning considered as RSVP domain operations particularly impacts the frequency of processing QoS routing, which also affects the performance.

2.2.4 QoS routing and OSPF [J9]

OSPF is a widely used link-state routing protocol, which is designed to be run internal to a single autonomous system. Each OSPF router maintains an identical database describing the autonomous system topology. From this database, a routing table is calculated by constructing a shortest-path tree. OSPF has some distinguished features, such as fast, loopless convergence, support of precise metrics, support of multiple paths to a destination, separate representation of external routes and so on. OSPF can be extended to provide QoS routing, called QOSPF [I4]. These extensions include: *1)* link advertisement with additional QoS metrics, e.g., bandwidth; *2)* mechanisms to trigger the link state update; *3)* algorithms on computing the QoS_based paths.

There are some alternatives of link state update mechanisms such as on the basis of the variation of available resource or just periodically. The path can also be calculated according to various algorithms in replace of that in QOSPF.

2.2.5 QoS routing and MPLS [J21-J22, I10-I11]

In MPLS, explicit routing may be needed in order to allow each stream to be individually routed, and to eliminate the need for each switch along the path of a stream to compute the route for each stream. Given that MPLS allows efficient explicit routing, it follows that MPLS also facilitates QoS routing. MPLS allows the explicit route to be carried only at the time that the label switched path is set up, and not with each packet. This implies that MPLS makes explicit routing practical. This in turn implies that MPLS can make possible a number of advanced routing features which depend upon explicit routing.

2.2.6 QoS routing and ATM [I12]

In the field of research on ATM networks, the work related to QoS routing is done in the ATM Forum PNNI protocol, which is comprised of signaling part and routing part. The PNNI routing protocol is a dynamic, hierarchical link state protocol that propagates topology information by flooding it through the network. The topology information is the set of resources (e.g., nodes, links and addresses) which define the network. Resources are qualified by defined sets of metrics and attributes (delay, available bandwidth, jitter, etc.) which are grouped by supported traffic class. The PNNI routing protocol supports source routing, crankback and alternate routing. PNNI source routing allows loop free paths. Also, it allows each implementation to use its own path computation algorithm. Furthermore, source routing is expected to support incremental deployment of future enhancements such as policy routing. Overall, the PNNI protocol is designed to scale to very large networks and support QoS.

Moreover, Integrated PNNI (I-PNNI) has been designed from the start to take advantage of the QoS Routing capabilities that are available in PNNI and integrate them with routing for layer 3. This would provide an integrated layer 2 and layer 3 routing protocol for networks that include PNNI in the ATM core. The I-PNNI specification has been under development in the ATM Forum and, at this time, has not yet incorporated QoS routing mechanisms for layer 3.

2.3. QoS routing model and an example implementation [I1, J11-J13]

A typical QoS routing protocol consists of three core functional components, that is, 1) distribution of resource availability information; 2) topology database with resource information; 3) QoS route computation. We illustrate them by building the QoS routing protocol based on OSPF.

2.3.1 Distribution of resource availability information

One extension needed for QoS routing is that of timely updates for resource availability. Link State Advertisements in OSPF carry administrative cost metrics for each link, and there is a provision for advertising multiple cost metrics using TOS fields. These fields can be effectively used to encode information such as available bandwidth and delay associated with a particular link. Moreover, additional update mechanisms are still needed to determine when updates are to be sent and what their content should be. It is particularly important in both the protocol overhead and the performance of QoS routing.

The design of an update triggering function involves several alternatives. Advertising every change in resource level provides the most accurate information for computing paths, but its communication overhead is not acceptable. A simple alternative is to rely on a timer to periodically issue such advertisements. This provides a direct control over the communication overhead, but does not ensure timely propagation of significant changes. Another approach is to base the triggering decision solely on the significance of the change in available resources. For example, a threshold based method triggers a new advertisement if the percentage change in resource level since the previous advertisement exceeds a specific threshold. Alternatively, resource levels such as available bandwidth may be divided into preset ranges or classes, and new advertisements may be issued each time a class boundary is crossed. Such triggering mechanisms provide direct control over the accuracy of information used by QoS routing, but may incur significant communication overhead, and in particular may cause transient overloads during periods of rapid changes. In practice, a reasonable trade-off can be achieved by combining several of the above methods, e.g., by augmenting a threshold policy with a timer to enforce a minimum spacing between consecutive updates.

2.3.2 Storing Resource Information in the Topology Database

The standard OSPF protocol already provides each router with a complete network map, which is stored in a topology database. This database is easily extended to also include the resource availability information needed by QoS routing. This extension is facilitated by the fact that resource updates are themselves communicated using existing OSPF mechanisms, i.e., flooding of extended LSAs as described above. As a result, both processing of resource updates and their inclusion in the topology database, can be added with minimal modifications. It should also be noted, that because best effort metrics are still kept unchanged in the topology database, the computation of best effort routes is unaffected, so that best effort and QoS routing can readily coexist within the same router.

2.3.3 QoS Route Computation

Route computation is the component whose implementation differs most from its best effort counter part. QoS routes are computed based on requests characteristics, e.g., how much bandwidth is required, and the resource information provided in the topology database. Differences with the best effort model are along two axes: The algorithms used to compute routes, and when those algorithms are actually executed. The latter is a major factor in the computational overhead associated with QoS routing, as well as the quality of the routes being computed. One approach is to compute paths on demand, i.e., run the algorithm for each new request. However, if requests arrive too frequently, this approach may prove costly even if the algorithm is of relatively low complexity, e.g., for bandwidth constraints, a shortest feasible path can be computed through a standard Dijkstra's shortest path algorithm on a pruned topology containing only those links that meet the bandwidth constraint. As a result, it is desirable to explore alternatives of lower computational complexity.

3 QoS routing Problems [J7-J8]

3.1 link metrics

There are some considerations in defining suitable link and node metrics [I1]. First, the metrics must represent the basic network properties of interest, which include residual bandwidth, delay etc. Moreover, the flow QoS requirements have to be mapped on path metrics. Second, the path computation based on a certain metrics or a combination of metrics must not be too complex as to render them impractical. A common strategy to allow flexible combinations of metrics is to utilize "*sequential filtering*", that is, paths based on a primary metric and so forth until a single path is found.

Once suitable link and node metrics are defined, a uniform representation of them is required. Particularly, encoding of the maximum, minimum, range, and granularity of the metrics are needed.

3.2 Graph Model

A network can be modeled as a graph (V, E). Nodes (V) of the graph represents switches, routers, and hosts. Edges (E) represent communication links. The edges are undirected only if the links are always symmetric. For most real networks the communication links are asymmetric, hence each link is represented by two directed edges in the opposite directions.

Thus, the network can be modeled as a weighted graph measured by the QoS metrics concern. For example, the link state is a triple consisting of residual bandwidth, delay, and cost. The cost of a link is defined in dollars or as a function of the buffer or bandwidth utilization. Moreover, each node also has a state on node resource, e.g., CPU bandwidth. The state of a node can be consider in conjunction with the link state.

3.3 QoS routing problems [J7]

The QoS routing problems can be divided into two major classes: unicast routing and multicast routing. The unicast routing problem is defined as follows: given a source code *s*, a destination *t*, a set of QoS constraints *C*, and possibly an optimization goal, find the best feasible path from *s* to *t* which satisfies *C*. The multicast routing problem is defined as follows: given a source code *s*, a set *R* of destination nodes, a set of QoS constraints *C*, and possibly an optimization goal, find the best feasible tree covering *s* to all nodes in *R* which satisfies *C*.

In each class, routing problems can be partitioned into subclasses according to the QoS based metrics. In unicast routing domain, for metrics of bandwidth and residual buffer space, the routing problems can be divided into two subclasses*: link-optimization routing* and *link-constrained routing*. An example of link (bandwidth)-optimization routing is to find a path that has the largest bandwidth on the bottleneck link. An example of link (bandwidth)-constrained routing is to find a path whose bottleneck bandwidth is above a required value. For other metrics such as delay, delay jitter and cost, the routing problems can be divided into two subclasses: *path-optimization routing* and *path-constrained routing*. Many composite routing problems can be derived from the above four basic problems. Unfortunately, there are two NP-complete problem classes, that is, *path-constrained path-optimization routing* (PCPO) and *multi-path-constrained routing* (MPC). An example of PCPO is delay-constrained least-cost routing, which is to find the least-cost path with bounded delay. An example of MPC is delay-delay-jitter-constrained routing, which is to find a path with both bounded delay and bounded delay jitter.

Multicast routing is different from unicast routing that an optimization or a constraint must be applied to the entire tree instead of a single path. There are several well-known multicast routing problems. The *Steiner tree* problem is to find the least-cost tree, the tree covering a group of destinations with the minimum total cost over all links. The *constrained Steiner tree* problem is to find the least-cost tree with bounded delay. The delay-delay-jitter-constrained multicast routing problem belongs to the multi-tree-constrained routing problem class. The above multicast routing problems are all NP-complete.

4 Introduction to QoS Routing Algorithms

4.1 General Requirements

There are some requirements for a routing algorithm:

Generality - Multimedia applications tend to have diverse QoS requirements on bandwidth, delay, delay jitter, cost, and so on. From a network designer's point of view, it would be beneficial to develop a generic routing algorithm instead of implementing different routing algorithms for different types of QoS requirements independently. The generic algorithm captures the common messaging and computational structure.

Extensibility - As the network infrastructure evolves and capacity increases, new applications are made possible. It requires the routing algorithms to adapt in order to accommodate new service types. It is important to design extensible algorithms and make them adapt to new applications, because the networks become increasingly complex and the deployment of new routing algorithms is very costly.

Simplicity - The simplicity of a routing algorithm in terms of time/logical complexity often allows efficient implementation, debugging and evaluation. It also makes the algorithm easier to understand, maintain, and upgrade.

Scalability - QoS_based routing should be scalable.

4.2 Routing Strategies [J7]

There are three routing strategies: source routing, distributed routing and hierarchical routing. They are classified according to how the state information is maintained and how the search of feasible paths is carried on.

In source routing, each node maintains the complete global state, including the network topology and state information of every link. Based on the global state, a feasible path is locally computed at the source node. A control message is then sent out along the selected path to inform the intermediate nodes of their precedent and successive nodes. A link-state protocol is used to update the global state at every node.

In distributed routing, the path is computed by a distributed computation. Control messages are exchanged among the nodes, and the state information kept at each node is collectively used for the path search. Most distributed routing algorithms need a distance-vector protocol (or a link-state protocol) to maintain a global state in the form of distance vectors at each node. Based on the distance vectors, the routing is done on a hop-by-hop basis.

In hierarchical routing, nodes are clustered into groups, which are further clustered into higher-level groups recursively, creating a multilevel hierarchy. Each physical node maintains an aggregated global state. This state contains detailed state information about the nodes in the same group and aggregate state information about the other groups. Source routing is used to find a feasible path on which some nodes are logical nodes representing groups. A control message is then sent along this path to establish the connection. When the border node of a group represented by a logical node receives the message, it uses the source routing to expend the path through the group.

4.3 Descriptions of routing algorithms

In this subsection, we just list the various routing algorithms and present the summary. The more detailed content of this subsection can be found in [J7] and its references.

4.3.1 Unicast Routing Algorithms

Source routing algorithms

- Wang-Crowcraft algorithm
- Guerin-Orda algorithm
- Chen-Nahrstedt algorithm
- Awerbuch et al. algorithm

The above algorithms required global state to be maintained at every node. Most algorithms transform the routing problem to a shortest-path problem and then solve it by Dijkstra's or the Bellman-Ford algorithm.

Distributed routing algorithm

- Wang-Crowcroft algorithm
- Salama et al. algorithm
- Sun-Landgendorfer algorithm
- Cidon et al. algorithm
- Shin-Chou algorithm
- Chen-Nahrstedt algorithm
- Ticket-Based Probing

Hierarchical routing algorithm

- PNNI

4.3.2 Multicast routing algorithms

Source routing algorithms

- MOSPF
- Kou et al. algorithm
- Takahashi-Matsuyama algorithm
- Kompella et al. algorithm
- Sun-Langendoerfer algorithm
- Widyono algorithm
- Zhu et al. algorithm
- Rouskas-Baldine algorithm

All the above algorithms require global state to be maintained at every node. Most heuristic algorithm for the NP-complete multicast routing problems construct a constrained tree incrementally by adding one destination into the tree each time based on certain selection criteria.

Distributed routing algorithm

- Kompella et al. algorithm
- Chen-Nahrstedt algorithm
- Carlberg-Crowcoft algorithm

4.4 Possible extensions to QoS Routing

There are a number of possible extensions in the field of QoS routing:

- efficient unicast/multicast routing algorithms:

Most source heuristic algorithms for the NP-complete routing problems are not scalable due to prohibitively high time to complexity, especially in the case of multicast routing. New efficient algorithms are required to achieve a good balance between the computation time and the connection-success ratio so that the time complexity can be reduced to the shortest-path computation range while the success ratio is still acceptable. The more dynamic and adaptive routing algorithms are possibly based on the fuzzy or intelligent theory [J23].

- routing with imprecise state information [J24-J25]

Most existing routing algorithms assume the availability of precise state information. However, state information is inherently imprecise in a distributed network environment. The imprecision directly affects routing performance. Therefore, the design of routing algorithms for large networks should take the information imprecision into consideration.

- routing in network advance reservation [J12]

Advance reservation are likely to become increasingly important as networks and distributed application become functionally richer, and there has been a number of previous works and investigations that have explored various related aspects. The impact of advance reservations on path selection is a topic that has been left largely untouched. There are several service models for advance reservations, which range from the traditional basic model of reserving a given amount of bandwidth for sometime in the future, to more sophisticated models aimed at increasing the flexibility of services available through advance reservations.

- QoS routing in wireless networks [J26-J27]

The emergence of nomadic applications have generated a lot of interest in wireless network infrastructures which support multimedia services. The QoS routing can inform the source of the bandwidth and quality of service available to any destinations in the wireless network. It also enables the establishment of QoS connection within the wireless networks and the efficient support of real time, multimedia traffic.

- integration with or extensions to IntServ or DiffServ model

The QoS architectures of IntServ and DiffServ provide the need for QoS routing, however, the considerations on the implementation of QoS under these architectures are incomplete. The interfaces to other components in these architectures need to be specified.

5 Future Works

We have some major interests in this field, which include:

- feasibility and benefit/cost analysis of QoS routing:
- Link state update algorithms
- Route calculation algorithms

Specially, we are interested in examining the determination of the timer value in periodic algorithm and the threshold value in the threshold_based algorithm.

An integrated software platform should be developed to examine our interests. This platform must contain a suit of components such as state maintenance and distribution, path calculation, resource reservation and so on. Moreover, the platform should provide an easy interface to measure the cost and facilitate our study in other aspects of QoS routing area in future study.

GATED software program can be used to build our environment platform. GATED provides a platform for implementing routing protocols on machines running the Unix operating system. It includes implementation of some routing protocols, such as OSPF. It offer a number of services which can facilitate implementations of QoS routing algorithms. Source code of GATED unicast routing version is free on http://www.gated.org. RSVP code can be found on http://www.isi.edu/div7/rsvp/. (More information can be found in these two sites.)

Based on this platform, we intend to investigate the cost of QoS routing deployed in the intra-domain IP networks. Some detailed results involved in various routing algorithms and link state update algorithms are also expected to present. Importantly, we expect to achieve substantial experience on a complete implementation of QoS routing.

References

Journals and Conferences Papers:

[J1] Quality of Service - Delivering QoS on the Internet and in Corporate Networks. P. Ferguson, G. Huston. John Wiley & Sons, 1998.

[J2] Quality of Service - An overview. J. Schmitt and L. Wolf. http://www.kom.e-technik.tu-darmstadt.de/~lars/lars.html

[J3] Quality of Service: Where are We? R. Steinmetz, L. C. Wolf. http://www.kom.e-technik.tu-darmstadt.de/~lars/lars.html

[J4] Multimedia Communication. L. C. Wolf, C. Griwodz, R. Steinmetz. Proceedings of IEEE, Vol. 85, No.12, December 1997, pp. 1915-1933.

[J5] A Survey of QoS Architectures. C. Aurrecoechea, A. T. Campbell, L. Hauw. http://www.ctr.columbia.edu/comet/members.html

[J6] A Quality of Service Architecture. A. Campbell, G. Coulson, D. Hutchison. http://www.ctr.columbia.edu/comet/members.html

[J7] An overview of Quality of Service Routing for Next-Generation High-Speed Networks: Problems and Solutions. S. Chen and K. Nahrstedt. IEEE Network. Novermber/December 1998.

[J8] Quality of Service Routing for Supporting Multimedia Applications. Z. Wang and J. Crowcroft. IEEE JSAC, Vol.14, No.7, September 1996.

[J9] QoS Routing Mechanisms and OSPF Extensions. R. A. Guerin, A. Orda and D. Williams. Global Telecommunications Conference, 1997. GLOBECOM'97., IEEE. Volume: 3 , 1997 , Page(s): 1903 -1908 vol.3

[J10] QoS Path Management with RSVP. Guerin, R.A.; Kamat, S.; Herzog, S.Global Telecommunications Conference, 1997. GLOBECOM'97., IEEE. Volume: 3 , 1997 , Page(s): 1914 -1918 vol.3

[J11] Intra-Domain QoS Routing in IP Networks: A Feasibility and Cost/Benefit Analysis. G. Apostolopoulos, R. Guerin, S. Kamat, S. K. Tripathi. http://www.seas.upenn.edu/~guerin/

[J12] Networks with Advance Reservations: The Routing Perspective. R. Guerian and A. Orda. http://www.seas.upenn.edu/~guerin/

[J13] Implementation and Performance Measurements of QoS Routing Extensions to OSPF. G. Apoatolopouloa, R. Guerin, S. Kamat. Proceedings of INFOCOM'99, New York, NY, March 21-25, 1999. Also available at: http://www.seas.upenn.edu/~guerin/

[J14] Market Pricing of Differentiated Internet Services. N. Semret, R. R.-F. Liao, A. T. Campbell and A. A. Lazar. http://www.ctr.columbia.edu/comet/members.html

[J15] Peering and Provisioning of Differentiated Internet Services. N. Semret, R. R.-F. Liao, A. T. Campbell and A. A. Lazar. http://www.ctr.columbia.edu/comet/members.html

[J16] Routing in the Internet. Christian Huitema. Prentice Hall PTR, 1995.
[J17] Differentiated Services for the Internet. Kalevi Kilkki. Macmillan Technical Publishing. 1999.
[J18] The QoS Framework. TINA-C, Internal technical report, 1995.
[J19] QoS Management Network Mechanism. D. Xiao. Unpublished paper, 1999.
[J20] A Quality of Service Architecture. A. Campbell, A. Coulson, D. Hutchison. ACM Computer Communication Review, April 1994.
[J21] Evolution of Multiprotocol Label Switching. A. Viswanathan, N. Feldman, Z. Wang, R. Callon. IEEE Communication Magazine, May 1998.
[J22] MPLS Routing Dynamics. S. Ayandeh and Y. Fan. Draft-ayandeh-mpls-dynamics-00.txt. March 1998.
[J23] Fuzzy Interface system for QoS Routing in B-ISDN. E. Aboelela, C. Douligeris. IEEE Canadian Conference on Electrical and Computer Engineering, 1998.
[J24] QoS_based Routing in Networks with Inaccurate Information: Theory and algorithms. IEEE Infocom'97, Japan, April 1997.
[J25] Distributed QoS Routing with Imprecise State Information. S. Chen, K. Nahrstedt. Computer Communications and Networks, 1998. Proceedings of 7th International Conference on, Pages: 614-621.
[J26] QoS Routing Performance in Multihop, Multimedia, Wireless Networks. T. Chen, J. Tsai, M. Gerla. IEEE 6th International Conference on Uniiversal Personal Communications Record, 1997.
[J27] QoS Routing in Multihop Packet Radio Environment. Y. Hsu, T. Tsai, Y. Lin. Third Symposium on Computers and Communications (ISCC), 1998.
[J28] Virtual Path Control for ATM Networks with Call Level Quality of Service Guarantees. N. Anerousis, A. A. Lazar. IEEE/ACM Transactions on Networking, Vol. 6, No. 2, April 1998, pp222-236.

IETF Draft and RFCs:

[I1] A Framework for QoS-based Routing in the Internet. RFC 2386. August 1998.
[I2] QoS Routing Mechanisms and OSPF Extensions. draft-guerin-qos-routing-ospf-05.txt. April 1998.
[I3] Additional OSPF Extensions for Traffic Engineering and QoS Routing. draft-wimer-ospf-traffic-00.txt. February 1999.
[I4] Quality of Service Extensions to OSPF. draft-zhang-qos-ospf-01.txt. September 1997.
[I5] Integrated Services. RFC2210 – 2216.
[I6] RSVP. RFC2205 - 2209. RFC2328.
[I7] A Framework for Differentiated Services. draft-ietf-diffserv-framework-02.txt. February 1999.

[I8] Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers. RFC2474. December 1998.

[I9] An Architecture for Differentiated Services. RFC2475. December 1998

[I10] A Framework for Multiprotocol Label Switching. draft-ietf-mpls-framework-04.txt. July 1999.

[I11] Multiprotocol Label Switching Architecture. draft-ietf-mpls-arch-05.txt. April 1999.

[I12] Private Network-Network Interface Specification Version 1.0 (PNNI 1.0). af-pnni-0055.000. March 1996.

Related Links:

[1] Prof. Rochin Guerin's Homepage

http://www.seas.upenn.edu/~guerin/

[2] Columbia Comet Group

http://www.ctr.columbia.edu/comet/

[3] Prof. Lars Wolf's Homepage

http://www.kom.e-technik.tu-darmstadt.de/~lars

[4] Prof. Lixia Zhang's Homepage

http://www.cs.ucla.edu/csd/people/faculty_pages/lixia.html

[5] IETF QoS Routing WG

http://www.ietf.org/html.charters/qosr-charter.html

[6] IETF Integrated Services WG

http://www.ietf.org/html.charters/intserv-charter.html

[7] IETF Differentiated Services WG

http://www.ietf.org/html.charters/diffserv-charter.html

[8] Prof. Sally Floyd's Homepage

http://www-nrg.ee.lbl.gov/floyd/

[9] Gated Web Site

http://www.gated.org/

[10] RSVP Project Homepage

http://isi.edu/div7/RSVP/