ELSEVIER

Computer Communications 25 (2002) 1640–1652

computer communications

www.elsevier.com/locate/comcom

Non-greedy minimum interference routing algorithm for bandwidth-guaranteed flows

Su-Wei Tan*, Sze-Wei Lee, Benoit Vaillaint

Center for High Speed Broadband Network, Multimedia University, Cyberjaya, Malaysia

Received 17 July 2001; revised 7 January 2002; accepted 29 January 2002

Abstract

This paper presents a new non-greedy routing and admission control algorithm for the dynamic routing of bandwidth-guaranteed tunnels. The algorithm operates online that is handling requests which arrive one at a time without prior knowledge of the traffic pattern. It combines the key concepts in the minimum interference routing algorithm and the algorithms developed based on theoretic competitive analysis to provide efficient routing. Using extensive simulations, we show that our algorithm out-performs several previously proposed algorithms on several metrics such as the acceptance rate of the tunnel set-up requests, fairness and the capability of providing priorities to tunnel requests of certain source–destination pairs in the network. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Quality of service routing; Bandwidth-guaranteed path selection; Multi-protocol label switching

1. Introduction

One of the key aspects of future Internet is the ability to establish connections capable of meeting quality of service (QoS) requirements. The demand for such capability is caused by the rapid development in multimedia applications such as video-on-demand and video conferencing that require QoS guarantees, and the establishment of virtual private network (VPN) to satisfy customer service level agreements (SLAs). To deliver performance guarantees, reservation of resources is required. In particular, the capability to allocate resources in terms of bandwidth-guaranteed tunnels is essential in future Internet.

In reservation based networks, the resources (link bandwidth) which have been allotted to a particular connection are solely occupied and thus not available for other connections during the connection period. As a result, future connection set-up requests may be denied due to insufficient network resources. Therefore, there is a need for a routing and admission strategy that efficiently addresses the problems of selecting a path to route a request and deciding on whether the request should be routed or rejected. The obvious way to achieve this is by adopting offline algorithms, which require prior knowledge of point-to-point demands. However, due to the advent of services that permit dynamic and frequent requests for capacity change, online algorithms capable of handling requests that arrive one at a time without knowledge of future traffic arrival must be adopted. In addition, the algorithms must route the traffic without splitting it into multiple paths and the rerouting of existing connections should be minimized or avoided altogether.

A natural admission control scheme is a greedy strategy in which a request is accepted as long as an admissible path exists. However, the greedy strategy can lead to inefficient use of resources. For example, it will accept a connection request even if it spans excessively over a long path. Longer path ties up more resources and thus decreases network throughput. Consequently, much research has been focused on non-greedy routing and admission control strategies. The problem has been studied extensively in the context of circuit-switched networks. However, most of the analyses and design efforts were based on the assumption that the request arrival pattern is described by simple probabilistic model with known parameters. An example of such algorithms is the real time network routing algorithm (RTNR) [10] used in the AT&T long distance network. It is only recently that a new framework for routing strategies which do not rely on any assumption about the probabilistic behavior of the traffic was developed for general topology network [2,3,11]. Based on theoretic competitive analysis, these algorithms have been shown to exhibit good performance. In particular, a non-greedy algorithm called EXP [19] has been proposed based on the competitive algorithm

---

* Corresponding author.
E-mail address: swtan@mmu.edu.my (S.-W. Tan).

0140-3664/02/$ - see front matter © 2002 Elsevier Science B.V. All rights reserved.
PII: S0140-3664(02)00055-5

developed in Ref. [2]. EXP has been shown to out-perform the greedy min-hop routing algorithm (MHA).

In light of the fact that the quasi-static knowledge on the ingress–egress points in the network should be exploited by routing algorithm to avoid potential connection rejection, Kodialam and Lakshman proposed the minimum interference routing algorithm (MIRA) [15] which utilizes such information in the context of multi-protocol label switching (MPLS) networks. MIRA was developed based on the 'minimum interference' concept that is a newly routed tunnel must follow a route that does not impart too much interference to paths of which availability may be critical in satisfying future demands. The simulation study in Ref. [15] shows that MIRA performs better than the min-hop routing and the widest shortest path algorithms [21].

In this paper, we present a new online non-greedy routing and admission control algorithm for the dynamic routing of bandwidth-guaranteed tunnels. As in Ref. [15], we will describe the algorithm in the context of MPLS networks for ease of terminology and conciseness. In MPLS networks, packets are assigned fixed-length labels at ingress routers which are then used to forward the packets along label switched paths (LSPs). The LSPs can be associated with certain constraints such as the bandwidth needed to satisfy customer SLAs. An important application of bandwidth-guaranteed LSPs is to become components of an IP VPN. Before mapping packets onto an LSP, the LSP is established using a signaling protocol such as the extension of resource reservation protocol (RSVP) [6] or label distribution protocol (LDP) [14]. A key feature of LSP is that it can be set up explicitly along a pre-specified path. This enables service providers to traffic-engineer their networks and to provide the bandwidth-guaranteed paths dynamically.

Our algorithm integrates the key concepts in the competitive strategies and minimal interference routing algorithm. Specifically, we introduce a new weight assignment scheme, which uses an exponential function to translate link's utilization and criticality into cost. The admission control scheme will accept the request if there is a sufficiently cheap path that is a path with the sum of the link costs less than a pre-defined threshold value.

We evaluate the performance of the new algorithm by carrying out an extensive set of simulations on a number of network topologies. The simulations show that our algorithm out-performs the original MIRA, min-hop routing and widest shortest path algorithms, and a variant of non-greedy scheme over a broad range of network operating environments. The simulations also show that our algorithm is capable of providing priorities to certain ingress–egress pairs in an environment which requires lower blocking probability for the tunnel set-up requests of the pairs.

The rest of this paper is structured as follows. In Section 2, a review of related work is presented. In Section 3, we will describe the system model and the problem definition. Reviews and analyses of the main concepts of minimal interference routing are presented in Section 4. In Section 5, our proposed algorithm is presented and analyzed. Discussion of simulation results and conclusion are presented in Sections 6 and 7, respectively.

2. Related work

In recent years, much research work has been carried out on various aspects of unicast QoS routing. For example, there has been research focused on path selection algorithms [2,15,19,21,23,27–30], cost and feasibility analysis [1,8] and processing cost reduction strategies [9,13,16]. In this section, we describe some early works on unicast path selection algorithm based on different metrics. A complete overview of QoS routing problem can be found in Ref. [24].

The most commonly used algorithm for the routing of bandwidth-guaranteed flows is the MHA. This algorithm chooses the least hop path from a set of feasible links that is links with sufficient capacity for the connection bandwidth request. Since the algorithm does not consider the available capacity of all feasible links in the path selection, it may potentially cause some links being blocked earlier. As a result, it can lead to poor network utilization.

In Ref. [30], two versions of shortest-widest path (SWP) algorithms for the routing of traffic with bandwidth and delay constraints were proposed. One is based on the distance-vector scheme while the other adopts the link state scheme. The SWP algorithm selects the path with the maximum bandwidth among all feasible paths. If there are several such paths, the one with the minimum hop-count is selected.

In Ref. [21], a variant of MHA, called widest-shortest path (WSP) was proposed. The algorithm selects a path with the minimum hop-count from all feasible paths. If there are several such paths, the one with the maximum bottleneck capacity is selected. By selecting the widest of all minimum hop paths, WSP attempts to load-balance the network resource usage and thus performs better than plain min-hop routing.

In Ref. [20], the authors studied the bandwidth-constrained and delay-constrained routing problems with imprecise network states. The model of imprecision is based on the probability distribution functions. The algorithms developed operate with imprecise information and are suitable to be adopted in hierarchical routing. A further study of QoS routing with imprecise state based on the probability model was carried out in Ref. [5].

More sophisticated algorithms have been proposed and analyzed based on the theoretic competitive analysis [25]. The main idea of these competitive strategies is to use an exponential function to convert the link congestion (utilization) into cost and the shortest path is chosen based on these dynamically changing weights. In Ref. [19], a non-greedy routing and admission control algorithm called EXP was developed based on the admission control scheme proposed in Ref. [2]. EXP considers the practical operating environments and its performance has been evaluated through extensive simulations. The algorithm selects among the minimum hop paths, the path that fulfills the admission control scheme. The admission control condition is given by

$$\sum_{e \in \text{path}} \mu^{u(e)} \le \rho \tag{1}$$

where $u(e)$ denotes the link utilization, μ and ρ are the configurable parameters. In EXP algorithm, the admission threshold value, ρ is set to μ in order to protect single link path.

In Ref. [15], MIRA which utilizes the knowledge of ingress–egress points in the network was proposed. One main observation in Ref. [15] is that the routing of a connection along a path can reduce the maximum permissible flow for some other ingress–egress pairs. This phenomenon is called 'interference'. The authors proved that the problem is NP-hard and developed path selection heuristics which are based on the idea of deferring the loading of certain 'critical' links. These are links which, if heavily loaded, would make it impossible to satisfy future demands of certain ingress–egress pairs.

While the above mentioned algorithms focus on finding path subject to constraints based on the combination of concave metric (for example bandwidth) and additive metric (for example delay), a more difficult routing problem is to find path, that is constrained by multiple additive metrics (for example delay, delay-jitter). This problem is known to be NP-complete [17]. Several heuristic algorithms have been developed to address this problem. Examples of them are as in Refs. [23,27–29].

In this paper, we propose to combine the concepts of minimal interference routing and the non-greedy framework. There are two main reasons which motivate the integration of these two different frameworks. Firstly, although MIRA considers ingress–egress points in the network, it does not perform admission control. This may result in excessively long path being selected, which in turn causes inefficient resource usage. Secondly, non-greedy algorithms (for example EXP) perform admission control but are oblivious to the ingress–egress pair locations in the network. As a result, some connections may be incorrectly accepted or rejected.

In this paper, we consider only the algorithms associated with traffic that requires bandwidth guarantees. This is due to the fact that the initial deployment of QoS routing will most likely focus on this kind of application. In Section 5.5, we describe some possible extensions to our algorithm to handle other QoS requirements.

3. Problem definition and system model

The network is represented as a capacitated graph $G(V, E, u)$ with n nodes and m edges, where $u(e)$ represents the capacity of the edge $e \in E$. We consider the request for an LSP i as a triple (s_i, d_i, b_i). Nodes s_i and d_i are the source and destination of the request and b_i is the bandwidth required for the LSP. We assume that the requests for LSPs arrive one at a time and there is no knowledge of future demand characteristics.

We describe, for simplicity, only a centralized routing model in this paper. In a centralized model, a route server is responsible for determining QoS routes on behalf of all the routers in a network. We further assume that the knowledge on ingress–egress pairs between LSP set-up requests does not change very frequently and is made available to the route server by a provisioning or administrative mechanism.

Each request for an LSP set-up may arrive at the route server directly (if the LSP is being set up manually) or queried through an ingress node. Based on the knowledge of current network topology and its available capacities, the route server computes an explicit path. The information on the network topology can be known administratively or via the link state database of a link state routing protocol, for example open shortest path first (OSPF) [12]. The algorithm keeps track of available capacities and we assume that the initial link capacities are known (the residual capacities information can be obtained also via routing protocol extensions as in Ref. [4]). The computed explicit route is then communicated back to the ingress router at which the LSP is to be established. The establishment and withdrawal of an LSP can be achieved via MPLS signaling protocols such as the extension of RSVP or LDP.

4. Review of minimum interference routing algorithm (MIRA)

In this section, we review the key concepts of MIRA [15]. The main concept of MIRA is to pick paths that do not interfere much with potential future LSP set-up requests of other source–destination pairs. In a capacitated network, the upper bound on the total amount of bandwidth that can be routed between a given ingress–egress pair is represented by the maximum flow (maxflow) value, v of the pair. The maxflow between the source node s and destination node d is the maximum flow value that can be sent from s to d without exceeding the capacity of any arc. The value of v decreases by D units whenever a bandwidth demand of D units is routed between s and d, or for another ingress–egress pair. Thus, the amount of interference on a particular ingress–egress pair is defined as the reduction in the maxflow value caused by the routing of an LSP for another ingress–egress pair. In the context of bandwidth-guaranteed tunnels routing, only unreserved (available) capacity can be used by new connections. Therefore, the term capacity is referred to the available capacity in the rest of this paper.

With interference defined as above, path selection heuristics based on the idea of deferred loading of certain critical links was developed. These are links over which the routing of an LSP can cause the maxflow values of one or more ingress–egress pairs to decrease. To aid the description of formal definition of critical links in the following, we adopt the notion of flow residual graph which is given as follows.

Given a flow f on a capacitated graph G, the residual capacity of an edge (i, j), $u_f(i,j)$ is the additional amount of flow that can be sent from i to j before exceeding the capacity $u(i,j)$, that is

$$u_f(i,j) = c(i,j) - f(i,j) \tag{2}$$

Given a flow f on a capacitated graph, $G(V, E, u)$, the flow residual graph, G_f of G induced by f is a flow graph on G with flow capacities $u_f(i,j)$.

Theorem 1. [15]: *Let $G(V, E, u)$ be the given directed graph. Let s represent the source node and d the destination node. Assume that a maximum flow between s and d has been computed. Let S be the set of nodes reachable from s in the flow residual graph. Let T represents the nodes that can reach the sink d in the flow residual graph. C_{sd} denotes the set of links are critical to s and d. An arc $(i,j) \in C_{sd}$ if*

- *Arc (i,j) is filled to capacity*
- *$j \notin S$*
- *$i \notin T$*
- *There is no path between i and j in the flow residual graph.*

Based on the above theorem, critical links for a node pair can be determined from the flow residual graph after the execution of max-flow algorithm. The maxflow value between two nodes in a network can be computed in time $O(n^2 m^{1/2})$ by the Goldberg-Tarjan highest label pre-flow push algorithm or in time $O(nm + m^2 \log U)$ using an excess scaling algorithm [22]. The procedure for determining all critical links for a node pair takes $O(m^2)$ time in addition to the maxflow computation.

Once the critical links are identified, MIRA generates a weight graph in which the critical links have weights which are increasing functions of their 'criticalities' in order to defer loading of critical links whenever possible. The weight function is given by:

$$w(e) = \sum_{(s,d):e \in C_{sd}} \alpha_{s,d} \tag{3}$$

where C_{sd} is the set of critical links for the ingress–egress pair (s, d) and $\alpha_{s,d}$ denotes the weight of the ingress–egress pair (s, d).

The value of $\alpha_{s,d}$ can be chosen to reflect the importance of the ingress–egress pair (s, d). The following are several possible configurations of $\alpha_{s,d}$ [15].

- If the value of $\alpha_{s,d} = 1$ for all (s, d) then the $w(e)$ represents the number of ingress–egress pairs for which link e is critical.
- The weights can be made inversely proportional to the maxflow values, that is $\alpha_{s,d} = 1/\theta_{s,d}$ where $\theta_{s,d}$ is the maximum flow value for the ingress–egress pair (s, d). This implies that the critical arcs for the ingress–egress pairs with lower maximum flow values will be weighted higher than the ones with higher maxflow value.
- If the network carries best effort traffic and if the delays are proportional to the flow on the links, the residual capacity of the link can be used to influence the weight of the link. For example, the weight of link e can be set to $w'(e) = w(e)/R(e)$ where $w(e)$ is defined as in Eq. (3) and $R(e)$ is the residual capacity of link e.

In our performance evaluation, the value of $\alpha_{s,d}$ is set according to the first approach. For links which are not critical to any ingress–egress pair, the path selection objective is to choose the one with the least number of hops. These links are assigned some small positive numbers. After the weight graph has been generated, all links with residual capacity less than the demand size are eliminated. An explicit path is selected from the reduced graph by using the Dijkstra or Bellman-Ford shortest path algorithm.

5. Non-greedy minimum interference routing algorithm

The idea of incorporating minimal interference in routing of MIRA has been shown to be effective by resulting in an algorithm which performs better than algorithms that do not consider the knowledge of ingress–egress points in the network. However, it has several limitations. Firstly, MIRA is a greedy algorithm. As the network resources are scarce, approach that attempts to admit all connection can result in inefficient resource utilization. Secondly, path selection in MIRA does not explicitly consider the available resource on each individual link, specifically min-hop approach has been adopted to select path from the non-critical links. The advantage of considering link available resource information in the path selection algorithm can be observed from the superior performance of WSP over the MHA, WSP computes a least hop path as in MHA except that it considers the residual capacities of the links for load balancing. Finally, MIRA is computationally expensive. While most algorithms such as the min-hop and WSP algorithms perform a single shortest path computation to route a request, MIRA requires a number of max-flow computations (proportional to the number of ingress–egress pairs in the network), each of which is of magnitude several times more expensive than the shortest path calculation. It is inevitable to experience increased computation for more efficient network usage, however, it is desirable to keep the computation overhead minimized.

Based on the observations, we propose a routing scheme that integrates the minimum interference concept with non-greedy admission control strategy developed based on theoretic competitive analysis. Specifically, our algorithm

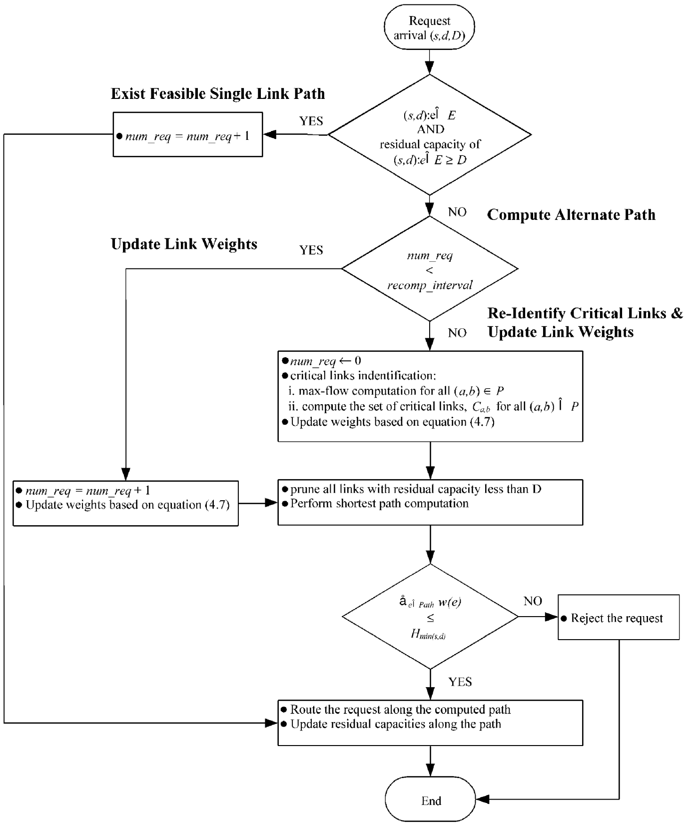

Fig. 1. Flow chart depicts the high level view of E-MIRA.

classifies links into two groups: critical and non-critical. Links are assigned costs that grow as an exponential function in utilization and 'criticality'. We name the new scheme as exponential-MIRA (E-MIRA) reflecting the significance of its exponential weight function. In the remainder of this section, we present the weight function, admission control scheme and algorithm description of the E-MIRA. After that we present detail analysis of each component.

5.1. Weight function

The new weight function adopted in E-MIRA is given by:

$$w(e) = \begin{cases} \mu_{\mathrm{nc}}^{u(e)-1} & e \notin C_{s,d} \\ \mu_{\mathrm{c}}^{u(e)+\theta_{\mathrm{dec}}-1} & e \in C_{s,d} \end{cases} \qquad \forall e \in E \tag{4}$$

where $C_{s,d}$ denotes the critical link set for ingress–egress pair (s,d), μ_{c} and μ_{nc} denote the exponent bases for critical and non-critical links, respectively. The link utilization, $u(e)$ is defined as the fraction of link capacity currently being occupied. It is given by:

$$u(e) = \frac{e_{\mathrm{cap}} - e_{\mathrm{res\_cap}}}{e_{\mathrm{cap}}} \tag{5}$$

where e_{cap} and $e_{\mathrm{res\_cap}}$ are the capacity and residual capacity of link e. The term θ_{dec} represents the reduction in total maxflow value of all source–destination pairs by routing the current demand via link e. It is given by:

$$\theta_{\mathrm{dec}} = \frac{De_{\mathrm{cc}}}{\sum_{(s,d)\in P} \theta_{s,d}} \tag{6}$$

where P denotes the set of ingress–egress pairs in the network, D is the current demand size, $\theta_{s,d}$ is the maxflow value for node pair (s,d), and e_{cc} is the criticality count of link e that is the number of ingress–egress pairs to which link e is critical.

5.2. Admission control criterion

The admission control criterion is based on the weight of the computed path, that is sum of the link costs on the path. A path is accepted if its cost is less than a pre-defined threshold value, otherwise it is rejected. E-MIRA uses the minimum hop length of each requesting node pair as the threshold value. The admission control condition is given by:

$$\sum_{e\in \mathrm{path}} w(e) \le H_{\min(s,d)} \tag{7}$$

where $H_{\min(s,d)}$ denotes the minimum hop path length between s and d.

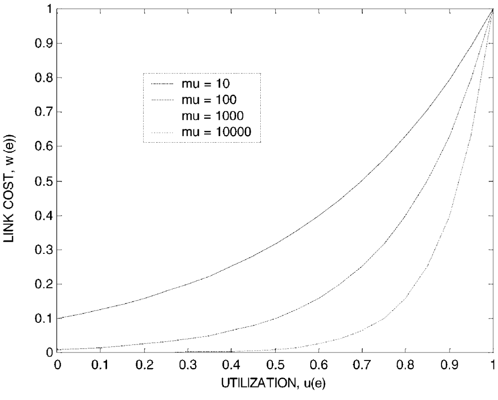

Fig. 2. Link cost vs utilization.

5.3. Algorithm description

Upon receiving an LSP set-up request, E-MIRA first attempts to route the request via any feasible direct link between the requesting source and destination. Otherwise, the algorithm invokes the critical link computation procedures as described in Section 5.2. After all critical links are identified, weights are assigned to the links based on Eq. (4). All links with residual capacity less than the demand size are pruned from the weight graph. Shortest path is then calculated from the reduced graph. Instead of routing the request directly via the selected path as in greedy algorithm, E-MIRA compares the path cost with the minimum hop path length of the requesting node pair. The path is accepted if the cost is less than the threshold value, otherwise it is rejected. If the computed path is accepted, the residual capacities along the shortest path are updated.

The proposed scheme utilizes the similar information as in the original MIRA and does not exhibits additional complexity as compared to the original algorithm. The admission control threshold value ($H_{\min(s,d)}$) needs to be computed only during system start up or rare topology changes. This information can be computed efficiently using the Folyd-Warshall all pairs shortest path algorithm [22] in time $O(n^3)$.

Although E-MIRA does not create extra algorithmic overhead compared to the MIRA, the computation of critical links is considerably expensive. To reduce the computation overhead for each run of the algorithm, critical links identification needs to be performed in a periodical manner. In this case, critical links are computed after every N number of LSP set-up requests. Between two successive critical link calculation, links remain in their status (critical or non-critical) determined during the most recent critical link identification. During each computation, link weights are updated based on Eq. (4).

In a network, a subset of ingress–egress pairs may need to be prioritized, that is their tunnel set-up requests need to be protected (having low probability of rejection). Since the E-MIRA is based on the concept of minimizing interference between ingress–egress pairs, it can be adapted to provide priority for certain node pairs in a network. For simplicity, we consider a network with two types of node pairs: normal and prioritized. In such network, E-MIRA functions as described earlier except that the admission control scheme now accepts all tunnel requests from prioritized pairs.

Fig. 1 depicts the flow chart for the E-MIRA. In the chart, *num_req* is initially set to zero and is used to keep track of the number of request for periodical computation of critical links. The computation interval is determined by the configurable parameter, *recomp_interval* in the chart.

5.4. Algorithm analysis

In this section, we discuss and analyze the main building blocks of the E-MIRA in detail.

5.4.1. Weight function

The formulation of the weight function is based on three key criteria:

(i) To minimize interference under low link utilization.
(ii) To minimize resource usage under high link utilization.
(iii) To reject a path if it consists of critical links which cause significantly large reduction in total maxflow value.

From Eq. (4), if we ignore the term θ_{dec} in the weight function for critical links, the weight is dependent on the exponent base, μ. Fig. 2 shows the plot of weight function ($w(e) = \mu^{u(e)-1}$) for different values of μ (10, 100, 1000 and 10,000). From the curves, it is clear that smaller value of μ results in higher cost. If we assign smaller value of μ to critical links' weight function (compared to non-critical links'), that is $\mu_{\text{c}} < \mu_{\text{nc}}$, the shortest path algorithm will most likely select non-critical links with low link utilization, that is an attempt to defer loading on critical links (criterion (i)).

When link utilization is high, the desirable routing behavior is the selection of path with the least number of hop-counts such that the network resource usage is minimized (criterion (ii)). From Fig. 2, under low link utilization, the difference between the costs is initially large, when the utilization increases, the difference becomes smaller and all costs approach one. This indicates that the weight function is adapting to the changes in network loading status. When the network utilization is high (most links have high utilization), as the difference in cost between critical and non-critical links become insignificant, the shortest path computed will most likely have relatively small hop-count.

By using μ_{c} of smaller value compared to μ_{nc}, the admission control scheme is dominated by the cost of those critical links (criterion (iii)). In addition, the term θ_{dec} contains information of the link criticality. From the well-known Max-Flow Min-Cut theorem [22], the maxflow value is equal to the sum of the capacities of minimum cut links. In Eq. (6), the value of θ_{dec} is upper bounded when there is

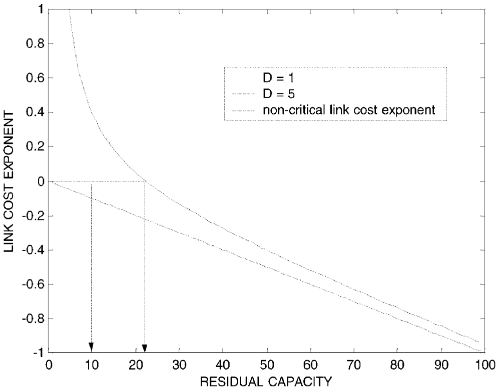

Fig. 3. Exponent term, α against residual capacity for two demand size.

only a critical link for all ingress–egress pairs. Since all critical links are minimum cut links (Theorem 1), Eq. (6) reduces to,

$$\theta_{\mathrm{dec}} = \frac{De_{\mathrm{cc}}}{e_{\mathrm{cc}}e_{\mathrm{res\_cap}}} \tag{8}$$

Since only links with requisite capacity are considered, the value of θ_{dec} will always be less than one. Based on Eqs. (5) and (8), we can rewrite the exponent term α in the critical links' weight function for the upper bound value of θ_{dec} as,

$$\alpha = \frac{D}{e_{\mathrm{res\_cap}}} - \frac{e_{\mathrm{res\_cap}}}{e_{\mathrm{cap}}} \tag{9}$$

In Fig. 3, we plot Eq. (9) for two values of demand size: 1 and 5% of link capacity. From the graph, we observe that the value of α becomes positive for small values of residual capacity (high network loading). When the value of α becomes positive, the cost of the link increases rapidly to large positive values (depends on the μ_{c}) and thus indicates that the link should be protected from being used for routing the demand. We refer this as the link protection stage. Thus, θ_{dec} acts as an indicator for the admission control scheme to reject path with high criticality under high network loading.

Fig. 3 also shows that different values of demand size cause the link cost to enter into the protection stage at different utilization levels. Thus, one may assume that the weight function will cause the path selection to favor request with small demand size and thus create unfair in an environment with a mixed demand size. However, our simulation shows that the weight function does not cause such unfairness compared to other algorithms.

5.4.2. Admission control scheme

E-MIRA attempts to accept all single link paths in which sources and destinations are directly connected. This is because single link path uses the least amount of resources. When an alternative path has to be computed, Eq. (7) is used as the admission criterion. The formulation of this admission criterion is aimed to address the problem inherent in strategies using constant threshold value. As the admission of a given path is based on the comparison of the path cost and the admission threshold, it is clear that node pair with longer hop distance will have higher path cost. If the threshold value is fixed, the network will most likely accept only requests from node pairs with short hop distance when network loading is high. Although this will improve the routing performance, that is the call acceptance rate, it results in unfairness to node pairs with longer hop distances.

From Eq. (4), the maximum link cost for non-critical link is one. Therefore, a feasible minimum hop path that does not cause any interference will always be used to route the request. From the discussion in Section 5.4.1, the cost for critical links exceeds one only under high network loading and high link criticality. Therefore, path which causes large interference to future requests will have higher probability of rejection. We will show experimentally that our admission control scheme indeed provides improved fairness compared to the approach with constant threshold value.

5.5. Extensions to handle other constraints

In this section, we describe several extensions to E-MIRA for handling other constraints such as hop-count and end-to-end delay. It is easy to incorporate hop-count requirement as the final explicit route is generated based on a shortest path algorithm. The Bellman-Ford shortest path algorithm can be used instead of Dijkstra's algorithm in order to incorporate hop-count requirements. If other constraint such as delay constraint need to be satisfied, this can be viewed as a restricted shortest path problem (RSP) [22]. In the RSP problem, the goal is to find the shortest path from a source node s to a destination node d with respect to an edge cost, among all paths that satisfy a certain criterion (for example a bounded delay). In the case of E-MIRA, after the weighted graph has been generated, a delay-constrained path can be computed using algorithms developed for the RSP problem [18] with E-MIRA generated weights as the cost to minimize.

6. Performance evaluation

In this section, we evaluate the performance of the E-MIRA and several other algorithms. The evaluation is carried out using an extensive set of simulation study. Although analytical models enable provable results be obtained, with the complexity of today's network topology, traffic distributions and service classes, it is extremely difficult to perform analytical modeling without sacrificing realism. In our simulation study, we choose a diverse range of network topologies and traffic patterns.

We compare the performance of E-MIRA with the original MIRA, MHA, WSP, a modified non-greedy algorithm based on EXP algorithm [19] and a greedy version of E-MIRA called GE-MIRA. GE-MIRA does not attempt to reject any tunnel set-up request as long as a feasible path exists. It is used in the evaluation to study the advantage of

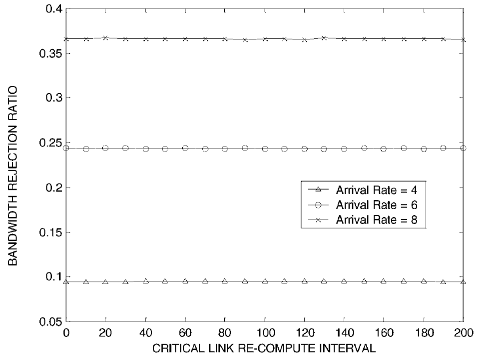

Fig. 4. LSP request rejection ratio of E-MIRA under periodical implementation for IspNet.

the new weight function over the original MIRA and the effect of admission control scheme to the routing performance. In the following sections, we denote MIRAs as the set of algorithms developed based on minimum interference concepts, that is MIRA, E-MIRA and GE-MIRA.

6.1. Simulation model and set-up

In the simulation, the arrival of LSP set-up requests was modeled as a Poisson process and the holding time was exponentially distributed. In most experiments, we fixed the mean holding time to one and varied the arrival rates to achieve different levels of network loading. As in Ref. [15], the bandwidth demands are uniformly distributed between 1 and 3 units unless specified otherwise. The details of the traffic distribution are specified in each experiment.

We used a wide range of network topologies, consisting of a widely used ISP network (19 nodes and 64 links) [1], the topology (15 nodes and 56 links) used in Ref. [15] and some randomly generated topologies based on Waxman's model [7]. In this paper, we only report the results obtained for the ISP network (hereafter referred to as IspNet). We will discuss on the similarities and differences between the results obtained from the topologies considered. The topologies may have homogeneous or heterogeneous link capacities. If not stated otherwise, the heterogeneous topologies are configured with links of 120 and 480 units of capacity. This is to model the ratio between OC-12 and OC-48. In homogeneous topologies, links may have capacity of 120 or 1000 units.

For most simulations, we configure E-MIRA and GE-MIRA with the values of μ_c and μ_{nc} as 2000 and 10,000, respectively, unless stated otherwise. This corresponds to critical links having weight that is five times heavier than non-critical links at low network load. This configuration had been found to perform well in most configurations. The length of each simulation run is 1000 time units. We ignore the first 100 time units to account for transient effects. Results are obtained by averaging five independent runs that is each run starts with a different random number seed.

To evaluate the performance of the algorithms, we consider two widely used performance metrics

$$\text{request rejection ratio} = \frac{\text{total number of request rejected}}{\text{total number of request}} \tag{10}$$

and

$$\text{bandwidth rejection ratio} = \frac{\text{total bandwidth request rejected}}{\text{total bandwidth requested}} \tag{11}$$

The request rejection ratio is used in the experiment involving equal size tunnel demands while bandwidth rejection ratio is used in the environment with different tunnel sizes.

6.2. Simulation results and discussions

The simulation results and discussions are presented in the following sub-sections.

6.2.1. *Effect of critical link computation frequency on performance*

MIRA based routing schemes are more expensive in computation than algorithms which involve only shortest path calculation, that is min-hop routing and WSP algorithms. It is desirable to reduce the computation overhead by performing the critical links identification in a periodical

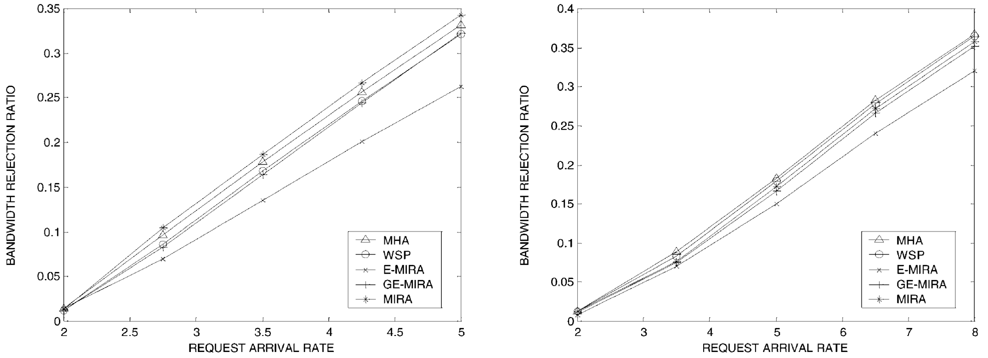

Fig. 5. Bandwidth rejection ratio as a function of traffic load under uniform workload. (a) Homogeneous capacity network. (b) Heterogeneous capacity network.

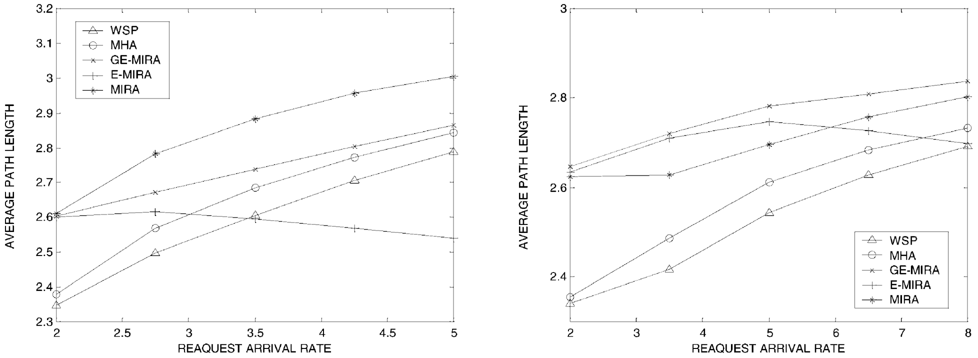

Fig. 6. Average path length computed. (a) Homogeneous capacity network. (b) Heterogeneous capacity network.

manner. This section evaluates the impact of the critical link computation frequency on the system performance.

Fig. 4 shows the result for IspNet. Similar trend is observed in experiments with other topologies. In the figure, the *y*-axis represents the bandwidth rejection ratio and the *x*-axis represents the interval between two successive critical link computations in terms of LSP request arrivals. The plots show that for different offered loads (arrival rates) there is no significant change in rejection ratio even for a large interval, such as 200. This means that most LSPs can be routed based on a shortest path calculation. In the following experiments, the critical link computation period is set to 200 for E-MIRA and GE-MIRA.

6.2.2. LSPs blocking rate

We examined the blocking rate (rejection ratio) of the five algorithms for different loads, topologies and traffic patterns. The details of the simulation configurations are described in each sub-section.

6.2.2.1. Uniform workload. Fig. 5 shows the bandwidth rejection ratio as a function of the offered load (arrival rate) for homogeneous and heterogeneous capacity networks. Traffic is evenly distributed whereby all node pairs have equal opportunity to request for LSPs set-up. In both topologies, E-MIRA significantly out-performs all other algorithms. In addition, GE-MIRA also performs better than the other three algorithms. This indicates that the new weight assignment scheme performs better than the original MIRA and the non-greedy admission control in E-MIRA can further improve the performance.

It is interesting to see that the original MIRA shows different results in both topologies; in topology with homogeneous capacity it performs the worst and in heterogeneous topology it performs better than min-hop routing and WSP algorithms. To investigate the differences observed, we examined the average path length computed for each offered load. From Fig. 6, it is clear that the average path length increases with network load except for E-MIRA in both topologies. The decrease in path length observed in E-MIRA is due to the rejection of long paths to minimize excessive resource usage. On the other hand, MIRA computed the longest path in homogeneous capacity network. Since long paths consume more resources, LSPs acceptance probability decreases and thus accounting for MIRA's worst performance among the algorithms in the homogeneous capacity topology.

Comparing the trend in LSPs rejection rate and path length distribution under both topologies, we observe that routing with long paths performs better in heterogeneous capacity network. In such topology, links with larger capacity can accommodate more LSPs than links with smaller capacity. Therefore, routing with long paths exploit the advantage of avoiding the overloading of certain links can

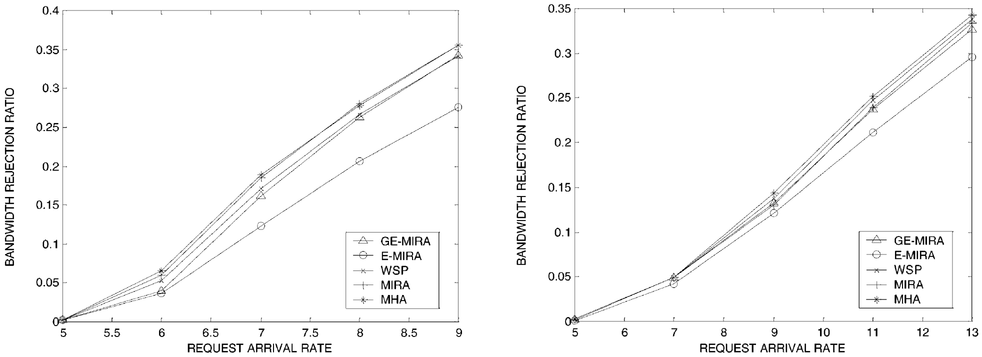

Fig. 7. Bandwidth rejection ratio as a function of traffic load under skewed workload. (a) Homogeneous capacity network. (b) Heterogeneous capacity network.

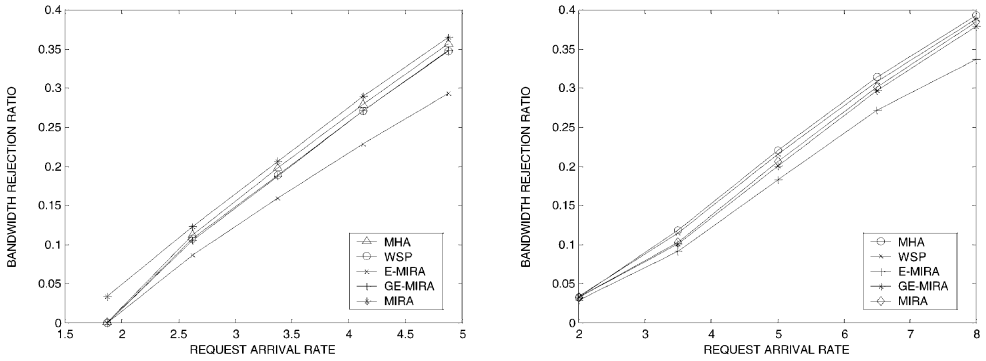

Fig. 8. Bandwidth rejection ratio as a function of traffic load under dynamic traffic pattern. (a) Homogeneous capacity network. (b) Heterogeneous capacity network.

achieve improved performance. Note that the MHA performs the worst.

6.2.2.2. Skewed workload. In practice, workload is naturally skewed and each traffic source may have different traffic arrival rates. To model a skewed workload, we assume that all node pairs have different arrival rates where a small proportion of the pairs may have a much higher traffic load compared to others. In the simulation, a large percentage (more than 90%) of node pair is assigned arrival rates that are randomly chosen between 0 and 1. The rest are assigned higher rates between 4 and 5. For both topologies, our earlier observation holds: E-MIRA performs better than the other algorithms and the original MIRA performs better than min-hop routing and WSP algorithms in heterogeneous capacity network. The results are shown in Fig. 7.

6.2.2.3. Dynamic traffic pattern. This section investigates the robustness of E-MIRA in environments with highly dynamic traffic patterns. The simulation set-up is adopted from Ref. [19]. In the simulations, each traffic source initially requests for LSP set-up with equal opportunity and arrival rate. This initial condition is referred as the base case. The traffic matrix is then randomly changed at time intervals of one mean hold duration. Each change of the traffic matrix alters the arrival rate of any node pair to a value that is uniformly distributed between 0 and twice its value in the base case. The results in Fig. 8 show that E-MIRA maintains its performance advantage over other algorithms. The relative performances of other algorithms remain as in previous simulation results.

Table 1
LSP rejection rate per bandwidth class

Bandwidth class (% of link capacity)	Per bandwidth class rejection ratio (%)				
	MHA	WSP	MIRA	GE-MIRA	E-MIRA
0.1	0.067	0.062	0.066	0.067	0.434
0.5	1.261	1.146	1.273	1.206	2.493
1.0	4.238	3.897	4.261	3.840	5.578
2.5	17.992	17.254	18.845	16.771	16.362
5.0	40.223	38.793	42.071	38.796	34.192

6.2.2.4. Mix of bandwidth classes. Previous simulations consider small bandwidth classes. From Section 5.4.1, we observe that large bandwidth demands cause the link cost to enter into the protection stage under smaller utilization level. Consequently, one may expect that E-MIRA to favor requests with smaller demand sizes. To investigate the fairness in environment with a mixed demand size, we consider the demand sizes that are uniformly distributed in the range: 0.1, 0.5, 1, 2.5 and 5% of link capacity. All topologies used are homogeneous in link capacity.

Table 1 presents the LSP rejection ratio in percentage for each bandwidth class. In general, we observe that small demand sizes correspond to relatively small rejection rates. The reason is that small bandwidth tunnels require much less resources and thus can be routed even if high-bandwidth demands are blocked.

It is interesting to see that per bandwidth class rejection rate of E-MIRA is more evenly distributed than other greedy algorithms. With greedy algorithms, tunnels of larger bandwidth classes are admitted even if they spanned over some long paths. This increases the rate of resource consumption which in turn causes only small bandwidth tunnels be routed. As for E-MIRA, long path may be rejected even though there are sufficient resources along the route, thus leaving room for future requests (small or large demand size with min-hop path).

6.3. E-MIRA dependency and sensitivity

We investigate in this section the characteristics of E-MIRA and its sensitivity to and dependency on the exponent bases (μ_c and μ_{nc}) and the admission control threshold value.

In order to investigate the effect of associating minimal interference routing to the non-greedy framework, we compare E-MIRA with a variant of EXP algorithm [19]. In the modified EXP algorithm, all links are assigned cost based on the non-critical links weight function in Eq. (4) and

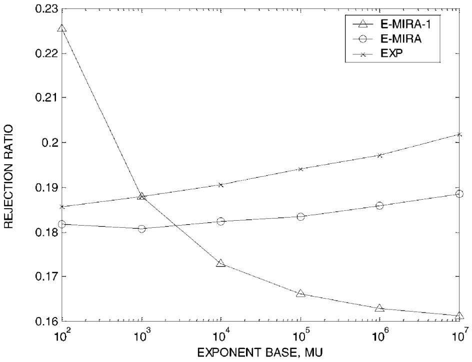

Fig. 9. Rejection ratio as a function of exponent base.

the min-hop path length of the requesting node pair is used as the admission control threshold. The modified EXP algorithm is similar to the routing component of the original EXP algorithm (called EXP-MC in Ref. [19]) except that the difference in the admission control scheme; EXP-MC is designed to protect single link path while our modified EXP attempts to be fair to all node pairs. In addition, we compare E-MIRA under two different admission criteria: threshold value of minimum hop path length and threshold value of one. These two configurations are used to examine the impact of admission threshold value on the rejection rate and fairness. In Ref. [19], it was shown that the performances of EXP-MC and EXP do not differ greatly.

In the simulations, uniform workload and homogeneous capacity topologies were used. All LSPs require bandwidth of 1% of link capacity. We conducted experiments with μ_c ranged from 2 to 1,000,000 for E-MIRA and each μ_{nc} was set five times larger of the corresponding μ_c. The exponent base of the modified EXP was assigned the value of μ_c. A large set of experiments under different offered loads was carried out.

Fig. 9 shows the simulation results for one of the test cases. In the figure, EXP denotes the modified EXP algorithm and E-MIRA-1 denotes the version of E-MIRA with admission threshold of one. Rejection ratios for exponent base with values less than 100 are not shown in the figure. From the plots, we see that E-MIRA out-performs the modified EXP algorithm for a wide range of exponent base values. This indicates that by considering the knowledge of ingress–egress points in the network, E-MIRA correctly identifies paths that are critical and rejects them to increase the future LSPs acceptance rate. Besides, we can also observe that E-MIRA is less sensitive to the choice of exponent base than EXP. E-MIRA performs well under moderate values of exponent base (for example 1000–100,000) for different offered loads over the simulated topologies.

Table 2
Rejection ratio for each hop-count

Hop length	Rejection ratio (%)				
	E-MIRA	E-MIRA-1	MHA	WSP	MIRA
1	9.23	8.75×10^{-3}	13.82	13.08	14.98
2	20.08	4.53	24.44	23.14	25.68
3	21.06	27.91	31.00	29.78	31.97
4	17.84	54.16	36.79	35.20	37.45

The results in Fig. 9 also show that with admission threshold set to one, E-MIRA can achieve lower LSP rejection rate. With such configuration, the admission control scheme becomes aggressive in preserving the network resources. As a result, paths with large number of hop-counts are more likely to be rejected to leave room for future LSP set-up requests. However, this configuration is highly unfair to node pairs with long hop distances. A good routing algorithm should not only perform well in terms of connection acceptance but also needs to provide equal opportunity to all classes of connection request to be accepted.

To investigate fairness, we classify all node pairs into some classes based on their minimum hop path lengths and record the rejection rate of each class. The blocking rate of each class is measured in terms of rejection ratio per hop-count. By using this metric, we can examine how the algorithms treat node pairs with different hop distances. Table 2 lists the rejection ratio per hop-count for several algorithms under the simulation scenario shown in Fig. 9.

From the results, in general, the rejection rate increases with the hop distance, except for the version of E-MIRA with min-hop length set as its threshold value. With admission threshold set to one, E-MIRA is highly biased towards node pairs with small hop distances (for example one and two) and thus exhibits large differences in rejection rate. For greedy algorithms, although the differences in rejection rates are smaller compared to E-MIRA-1, the algorithms still exhibit significant unfairness. In other word, our approach (E-MIRA with admission threshold set at the min-hop path length) not only out-performs the greedy algorithms in terms of connection acceptance but also in terms of fairness.

We have also carried out experiments with admission thresholds which are set between one and the min-hop path length. In particular, the threshold value is assigned as follows

$$\text{Threshold} = \text{MAX}\{1, kH_{\min(s,d)}\} \qquad k < 1 \tag{12}$$

where k is a configurable parameter.

The results (not shown here) suggest that the above threshold value yields better LSPs acceptance rate compared to E-MIRA while maintaining the fairness advantage over E-MIRA-1. The value of quantity k can be varied to provide priorities to different classes of LSPs.

6.4. Priority based routing

In a network, it is likely that some subsets of ingress–egress pairs are more important that is their LSP set-up requests require lower blocking probability. In other words, it is desirable that different levels of priority can

Table 3
Performance of each class of node pair in a two level priorities network

Algorithm	Rejection ratio (%)								
	Prioritized pairs: 2%			Prioritized pairs: 5%			Prioritized pairs: 10%		
	Overall	Prioritized	Non-prioritized	Overall	Prioritized	Non-prioritized	Overall	Prioritized	Non-prioritized
MHA	12.98	9.25	13.06	12.98	15.20	12.86	12.98	11.37	13.16
WSP	11.72	8.14	11.80	11.72	13.80	11.62	11.72	10.16	11.90
MIRA	13.27	9.51	13.35	13.35	15.02	13.26	13.17	11.68	13.34
GE-MIRA	10.93	7.33	11.01	11.18	13.03	11.09	10.94	9.50	11.10
E-MIRA	8.42	1.25	8.57	8.35	2.18	8.67	8.37	2.32	9.04

be associated with different ingress–egress pairs. The routing objective is to protect the LSPs for node pairs with higher priority while maintaining the overall performance. This section investigates the performance of the algorithms in such environment.

For simplicity, we consider only two classes of node pairs: (i) prioritized and (ii) non-prioritized. The LSP set-up requests for prioritized pairs should receive lower blocking probability. In this network environment, MIRA based algorithms compute the critical links only for prioritized pairs to defer loading on links that are critical to these pairs. E-MIRA, on the other hand, performs admission control only for LSP set-up requests from non-prioritized pairs.

In the simulations, uniform workload and homogeneous link capacity topologies were used. All LSP require bandwidth of 1% of link capacity. We assume that the proportion of prioritized pair in the network to be small, for example 2–10% of all possible node pairs. The distributions between prioritized and non-prioritized pairs were randomly selected. We carried out a large number of experiments with different distribution and proportion of prioritized pairs.

Table 3 lists the results for one of the test cases. From the results, we observe that E-MIRA exhibits the smallest rejection ratio for both prioritized and non-prioritized pairs. In the environment considered, E-MIRA attempts to defer loading on links which are critical to the prioritized pairs and performs admission control only on LSP set-up requests of non-prioritized pairs. Therefore, the lower rejection for prioritized pairs is expected. It is interesting to observe that when only a small proportion of node pairs are considered in critical link identification, E-MIRA still performs better than other algorithms. Similar results were obtained for other cases.

7. Conclusions

The primary contribution of the paper is the development of a non-greedy routing and admission control algorithm for the dynamic routing of bandwidth-guaranteed tunnels. The algorithm combines the advantages of algorithms which were developed based on the notion of minimum interference and competitive analysis. It is based on the objectives to minimize interference under low network loading and resource usage under high network loading. In addition, paths which consist of links that cause large interference should not be admitted. We have also introduced a new weight assignment scheme that adapts to the changes in the network loading and takes into account the knowledge of ingress–egress points in the networks. The admission control scheme proposed addresses the fairness in routing of tunnels for source–destination pairs with different hop distances. We showed through simulations over a wide range of network configurations that our algorithm performs better in terms of tunnel acceptance compared to the original MIRA, min-hop and WSP algorithms and a variant of non-greedy scheme. The simulations also show that the computation of the algorithm can be implemented in a periodical manner without affecting the performance. In addition, the algorithm can achieve improved fairness in environment with mix bandwidth classes and ingress–egress pairs having different hop distances as compared to the greedy algorithms. The algorithm is also capable of protecting the LSP set-up requests from certain ingress–egress pairs. Although we have described and evaluated the algorithm in the centralized routing model, the underlying concepts can be readily adapted to the distributed routing environment.

References

[1] A. Shaikh, J. Rexford, K. Shin, Evaluating the overheads of source-directed quality-of-service routing, Proc. IEEE Int. Conf. Network Protocols (ICNP'98), Austin TX, October (1998) 42–51.

[2] B. Awerbuch, Y. Azar, S. Plotkin, Throughput competitive on-line routing, Proc. 34th Annual Symp. Found. Comput. Sci., Polo Alto, California, November (1993).

[3] B. Awerbuch, Y. Azar, S. Plotkin, O. Waarts, Competitive routing of virtual circuits with unknown duration, Proc. Fifth ACM-SIAM Symp. Discrete Algorithms, January (1994) 321–327.

[4] D. Katz, D. Yeung, Traffic engineering extensions to OSPF, Internet Draft, August 2001.

[5] D. Lorenz, A. Orda, QoS routing in networks with uncertain parameters, IEEE/ACM Trans. Network. (1998) 768–778.

[6] D.O. Awduche, L. Berger, D. Gan, T. Li, G. Swallow, V. Srinivasan,

RSVP-TE: extensions to RSVP for LSP tunnels, Internet Draft, February 2001.

[7] E.W. Zegura, K.L. Calvert, S. Bhattacharjee, How to model an internetwork, Proc. IEEE INFOCOM, March (1996) 594–602.

[8] G. Apostolopoulos, R. Guerin, S. Kamat, A. Orda, S.K. Tripathi, Intra-domain QoS routing in IP networks: a feasibility and cost/benefit analysis, IEEE Network (Spec. Issue Integ. Different. Serv. Internet) 13 (5) (1999) 42–54.

[9] G. Apostolopoulos, R. Guerin, S. Kamat, S.K. Tripathi, Server based QoS routing, Proc. IEEE GLOBECOM'99, Rio de Janeiro, Brazil 1b (1999) 762–768.

[10] G.R. Ash, J. Chen, A.E. Frey, B.D. Huang, C. Lee, G.L. McDonald, Real-time network routing in the AT&T network-improved service quality at lower cost, Proc. IEEE GLOBECOM, Orlando, Florida, December (1992) 115–136.

[11] J. Aspnes, Y. Azar, A. Fiat, S. Plotkin, O. Waarts, On-line load balancing with applications to machine scheduling and virtual circuit routing, Proc. 23rd Annual ACM Symp. Theory Comput., San Diego, California, May (1993).

[12] J. Moy, OSPF version 2, RFC 2328, April 1998.

[13] J.Y. Le Boudec, T. Przygienda, A route pre-computation algorithm for integrated services networks, J. Network Syst. Mgmt. 3 (4) (1995) 427–429.

[14] L. Andersson, P. Doolan, N. Feldman, A. Fredette, B. Thomas, Label distribution protocol specification, RFC 3036, January 2001.

[15] M. Kodialam, T.V. Lakshman, Minimum interference routing with applications to MPLS networks, Proc. IEEE INFOCOM, March (2000) 884–893.

[16] M. Peyravian, A.D. Kshemkalyani, Network path caching: issues, algorithms and a simulation study, J. Comput. Commun. 20 (1997) 605–614.

[17] M.R. Garey, D.S. Johnson, Computers and Intractability, Freeman, San Francisco, 1979.

[18] R. Hassin, Approximation schemes for the restricted shortest path problem, Math. Ops. Res. 17 (1) (1992) 36–42.

[19] R. Gawlick, A. Kamath, S. Plotkin, K.G. Ramakrishnan, Routing and admission control in general topology networks, Technical Report STAN-CS-TR-95-1548, Stanford University, CA, 1995.

[20] R. Guerin, A. Orda, QoS based routing in networks with inaccurate information: theory and algorithms, IEEE/ACM Trans. Network. June (1999) 350–364.

[21] R. Guerin, D. Williams, A. Orda, QoS routing mechanisms and OSPF extensions, Proc. IEEE GLOBECOM, November (1997) 1903–1908.

[22] R.K. Ahuja, T.L. Magnanti, J.B. Orlin, Network Flows: Theory, Algorithms, and Applications, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[23] S. Chen, K. Nahrstedt, On finding multi-constrained paths, IEEE Int. Conf. Commun., June (1998).

[24] S. Chen, K. Nahrstedt, An overview of quality of service routing for next-generation high-speed networks: problems and solutions, IEEE Network November/December (1998) 64–79.

[25] S. Plotkin, Competitive routing of virtual circuits in ATM networks, IEEE JSAC (1995) 1128–1136.

[27] T. Korkmaz, M. Krunz, A randomized algorithm for finding a path subject to multiple QoS constraints, J. Comput. Networks 36 (2–3) (2001) 251–268.

[28] T. Korkmaz, M. Krunz, S. Tragoudas, An efficient algorithm for finding a path subject to two additive constraints, submitted for publication.

[29] X. Yuan, X. Liu, Heuristic algorithms for multi-constrained quality of service routing, Proc. IEEE INFOCOM 2 (2001) 844–853.

[30] Z. Wang, J. Crowcroft, Quality-of-service routing for supporting multimedia applications, IEEE JSAC 14 (7) (1996) 1228–1234.