Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 676542, 10 pages
http://dx.doi.org/10.1155/2015/676542

Research Article

Modeling and Algorithm for Multiple Spanning Tree Provisioning in Resilient and Load Balanced Ethernet Networks

Steven S. W. Lee,[1] Kuang-Yi Li,[1] and Chieh-Ching Lin[2]

[1] *Department of Communications Engineering, National Chung Cheng University, Chiayi 621, Taiwan*
[2] *Pan Signal Technology, Hsinchu, Taiwan*

Correspondence should be addressed to Steven S. W. Lee; steven.sswlee@gmail.com

Received 27 September 2014; Revised 1 January 2015; Accepted 11 January 2015

Academic Editor: Chunlin Chen

We propose a multitree based fast failover scheme for Ethernet networks. In our system, only few spanning trees are used to carry working traffic in the normal state. As a failure happens, the nodes adjacent to the failure redirect traffic to the preplanned backup VLAN trees to realize fast failure recovery. In the proposed scheme, a new leaf constraint is enforced on the backup trees. It enables the network being able to provide 100% survivability against any single link and any single node failure. Besides fast failover, we also take load balancing into consideration. We model an Ethernet network as a twolayered graph and propose an Integer Linear Programming (ILP) formulation for the problem. We further propose a heuristic algorithm to provide solutions to large networks. The simulation results show that the proposed scheme can achieve high survivability while maintaining load balancing at the same time. In addition, we have implemented the proposed scheme in an FPGA system. The experimental results show that it takes only few μsec to recover a network failure. This is far beyond the 50 msec requirement used in telecommunication networks for network protection.

1. Introduction

Ethernet has become the dominant local area network technology for decades. It has also been extended to support metropolitan area networks (MAN) and even wide area networks (WAN). Nowadays Ethernet is the major data center networking technology [1, 2]. However, conventional Ethernet has difficulties on reducing failure recovery time and avoiding network congestion. How to achieve fast failover and enhance traffic engineering capability have become important research topics.

The IEEE 802.1d spanning tree protocol [3] is used in Ethernet to prevent routing loops. As a failure interrupts a working spanning tree, it takes tens of seconds for the protocol to obtain a new tree. The long convergence time is far from acceptable for supporting timing sensitive commercial services. In order to speed up the failure recovery process, the Rapid Spanning Tree Protocol (RSTP) is later presented in IEEE 802.1w [4]. The restoration time for Ethernet running RSTP is depending on the processing time of Ethernet switches. Although the failure recovery time is reduced to fall within the range of few seconds [5], it is still far behind the 50 msec failover requirement used in telecommunication networks. The restoration time requirements for network services are summarized in [6].

In recent years, several fast failure handling schemes have been proposed in the literature. Those schemes can be classified into spanning tree reconstruction based and multi-VLAN based approaches. Since a failure link breaks a working spanning tree into two parts, the tree reconstruction based approach is to select a new link to reconnect the two subtrees. In the multi-VLAN based approaches, failure recovery is achieved by switching the traffic affected by a failure to a backup VLAN tree so as to bypass the failure device. The backup VLANs are preconfigured and stored in each switch. As a failure event is detected, the switch that is adjacent to the failure performs the failure recovery process individually without exchanging protocol information among nodes. Since each switch performs failure recovery based on local decision, the failover time is greatly reduced compared to conventional Ethernet network.

Fast Spanning Tree Reconnection (FSTR) [7] is a spanning tree reconstruction based scheme. Upon a link failure, the two adjacent nodes of the failure link are responsible for identifying an alternate path for failure recovery and informing a reconnect link to rebuild the broken spanning tree. Upon receiving the recovery messages, nodes along the alternate path use a new preplanned forwarding table to redirect the rerouting flows. In [8], an enhanced version of FSTR is presented to provide Ethernet against double link failures. Unlike multi-VLAN based schemes that can perform protection switching locally, FSTR needs time to indicate the failure link and the nodes along the alternate path. It turns out that the tree reconstruction based approaches require longer time for failure recovery than the multi-VLAN based approaches.

IEEE 802.1s [9] defines Multiple Spanning Tree Protocol (MSTP) that enables Ethernet networks to use multiple spanning trees to achieve load balancing and traffic separation. A spanning tree instance can include multiple VLANs. A VLAN that binds to a spanning tree instance can only use the links belonging to the tree to deliver frames. Multi-VLAN based network protection approaches have been proposed in [5, 10–13]. Backup VLANs are provisioned in advance to protect working VLANs. The traffic on the working VLANs is rerouted to the backup VLANs as a failure occurs. The provisioning of those VLANs can be implemented in a network management system. The topology of Ethernet network should be obtained first. There are multiple schemes that can be applied for this purpose [14–16]. Based on the network topology, a multiple VLAN based spanning trees provisioning algorithm is performed to obtain the VLAN configuration. In [10, 11], a resilient architecture for fast failure handling is proposed. In the considered network architecture, the network consists of two parts. The core part is constituted by interconnecting Ethernet switches while IP routers form the edge part of the network. Multiple VLANs are provisioned inside the Ethernet to provide protection against a single link or a single node failure. Through periodically sending probe messages on each VLAN, the IP routers in the edge can understand the health conditions of each VLAN tree. The status of those VLANs is broadcasted to all of the routers in the network such that only survivable VLANs are selected for communications. Although this probing based approach can provide fast failure handling, it wastes part of network resources in exchanging and processing control messages. In addition, upon a failure, there is a time lag for the routers to take reaction to reroute their traffic to a survived VLAN.

In [12], algorithms for planning backup VLAN are presented. There are two schemes proposed in the work. One is called connection based scheme and the other is called destination based scheme. In the former, a connection is associated with a source-destination pair. At a switch, both source address and destination address are examined. Although different connections have the same destination MAC, they can use different backup VLANs. In the destination based scheme, a switch applies only a single backup VLAN to protect all flows with the same destination. It is clear that the connection based scheme requires more preplanned backup VLANs but with better capacity usage.

Viking is also a VLAN based approach [13]. A centralized management controller is responsible for performing fault monitoring and failure recovery. As a failure occurs, the switches notify the controller through Simple Network Management Protocol (SNMP). The benefit of Viking system is that it can be built using off-the-shelf devices. However, SNMP messages are carried by User Datagram Protocol (UDP), which is not a reliable protocol. UDP messages would be lost in the network. The long failure detection and processing time make it difficult to provide high-speed failure recovery.

In [5], network protection against link failures and QoS routing are considered. An Integer Linear Programming (ILP) model is proposed to determine the routing of a pair of link-disjoint working path and backup path for each traffic demand. The model also determines the VLAN trees to accommodate these working and backup paths.

Figure 1 depicts an example of VLAN based protection scheme. In this example, VLAN 1 is the working VLAN and VLAN 2 is used to protect Link $(2, 5)$. In the normal state, node 2 uses Link $(2, 5)$ to deliver frames to destination nodes 7, 8, and 9. As Link $(2, 5)$ fails, node 2 uses VLAN 2 to send frames so as to avoid using the failure link. VLAN based protection scheme like [12] is designed to provide fast protection for any single link failure. However, it cannot be used to handle a node failure. In this example, if the failure is node 5 not Link $(2, 5)$, the backup VLAN 2 can only recover the flows with destination node 7. Since in the backup VLAN 2, node 8 and node 9 are behind node 5, traffic to those nodes cannot be recovered. Therefore, the failure results in frame losses for traffic destined to node 8 and node 9.

Since a link failure and a node failure have the same syndrome, that is, loss of signal to Ethernet switches adjacent to the failure point, it is difficult to identify the exact failure location within very short time. In fact, the only way to identify the exact failure type is to cooperate among multiple nodes in the network through time consuming message exchange processes. It prohibits a node from achieving fast protection switching.

In order to resolve the difficulty of the above problem, we propose a novel fast local protection scheme. In the proposed scheme, we configured backup VLANs to protect working VLANs on each link. We require that both adjacent switches of a protected link have to be leaf nodes in this link's backup VLAN tree. This requirement is called leaf constraint in this paper. Therefore, even if the failure event is a node failure, the backup VLANs can still guarantee to provide a survivable path for each node excluding the failure node.

Figure 2 depicts an example for the proposed scheme. To simplify the presentation, we use only one working instance tree in the example. Figures 2(a) and 2(b) are the input Ethernet topology and the working tree, respectively. In the proposed scheme, each link on a working tree is provisioned with a backup VLAN and leaf constraint is applied on both adjacent nodes of the link. Figure 2(c) plots the backup VLAN tree for Link $(2, 5)$. Since Link $(2, 5)$ does not appear in the backup VLAN tree 2, VLAN 2 can be used to protect Link $(2, 5)$. In Figure 2(d), VLAN 3 is used to protect Link $(5, 8)$.

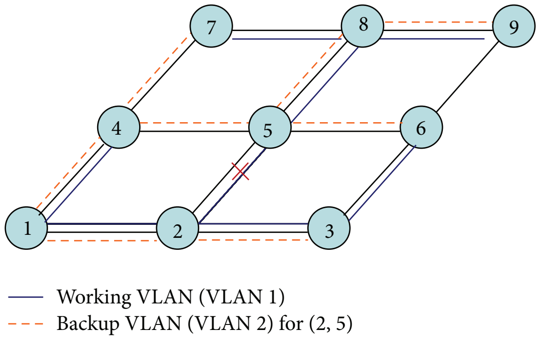


Figure 1: Example of VLAN based protection scheme.

Backup VLANs that meet the leaf constraint can protect not only any single link failure but also any single node failure. We use the same backup VLANs in Figure 2 to show how the network can use backup VLAN to protect the failure of node 5. As node 5 fails, both node 2 and node 8 will detect loss of signal from their interface to node 5. Node 2 and node 8 redirect the affected flows to VLAN 2 and VLAN 3, respectively. Since node 5 is the leaf node in both VLANs, no other nodes rely on node 5 to forward frames for them. Thus, the failure of node 5 will not affect the failure recovery for the other nodes in the network.

In this paper, we jointly take traffic engineering (TE) and network survivability into consideration. Our target is to guarantee that the provisioned VLANs are not only being able to protect any single link failure and any single node failure, but also being able to avoid traffic congestion. We denote the leaf constrained VLAN provisioning with TE consideration as LCP-TE problem. To facilitate problem formulation, we first use graph transformation technique to transform the considered Ethernet network into a two-layered graph. We propose an ILP model for this problem on the transformed graph. The objective function of the ILP problem is to minimize the utilization on the most congested link. The output of the problem includes working and backup VLANs that can guarantee 100% survivability against any single link failure and single node failure.

The LCP-TE problem is considered in our previous work [17] and a simulated annealing based algorithm is proposed as well. Although multiple backup trees are used, however, only one working tree is used to carry working traffic in that work. We have solved the single working tree based LCP-TE problem for input network with moderate size. However, for a large size network, due to the huge number of problem constraints and decision variables, it becomes difficult to solve the problem by directly applying standard integer programming techniques. In fact, the multi-VLAN provisioning problem without taking leaf constraint into consideration has been proved to be NP-complete [12]. Therefore, the LCP-TE problem is also an NP-complete problem. To reduce computation time for large size network, in [17], we decomposed the problem into several subproblems and developed a simulated annealing based heuristic algorithm to resolve the single working tree based LCP-TE problem. In this work, we extend our previous work that uses only one working tree to multiple trees for routing working traffic. We have provided an ILP for the multi-VLAN provisioning problem. A heuristic algorithm is proposed to provide solution to a large network.

The remainder of this paper is organized as follows. In Section 2, we present a formulation to the new multi-VLAN LCP-TE problem. In Section 3, we present the proposed heuristic algorithm for obtaining a solution to a large sized network. In Section 4, we demonstrate the simulation and experimental results and make performance comparisons on survivability ratio and link loads. Finally, concluding remarks are made in Section 5.

2. Problem Formulation

In this work, we formulate the muti-VLAN LCP-TE problem as an ILP problem. Our algorithm determines the working VLANs for traffic routing in the normal state and the backup VLANs for fast failover in the failure states. Leaf constraint is applied in our formulation so that both single link failure and single node failure can be handled. The objective function is to minimize the link utilization on the most congested link.

We apply graph transformation technique to facilitate problem formulation. The input graph is transformed to a two-layered directed graph. The top layer is used to determine the working VLAN trees for traffic transmission in the normal state while the down layer is used to decide backup VLANs. Those two layers are connected by some artificial bridge edges. As a failure occurs, some particular bridge edges are turned on to allow traffic moving from the top layer to the down layer. We use this idea to reduce the difficulty on formulating the multi-VLAN LCP-TE problem.

To make the notation easier for understanding, we use vertices and edges to denote switches and directional links in the transformed graph. Figure 3 depicts an example. The input graph is given in Figure 3(a) and the transformed graph is shown in Figure 3(b). In Figure 3(c), an example for failure recovery is illustrated.

The problem formulation and notations are shown below.

Given Input Constant Values

N: set of switches in the input network,

L: set of links in the input network,

S: set of network states; we denote the normal (non-failure) state by s_0 and denote state for link i failure by s_i,

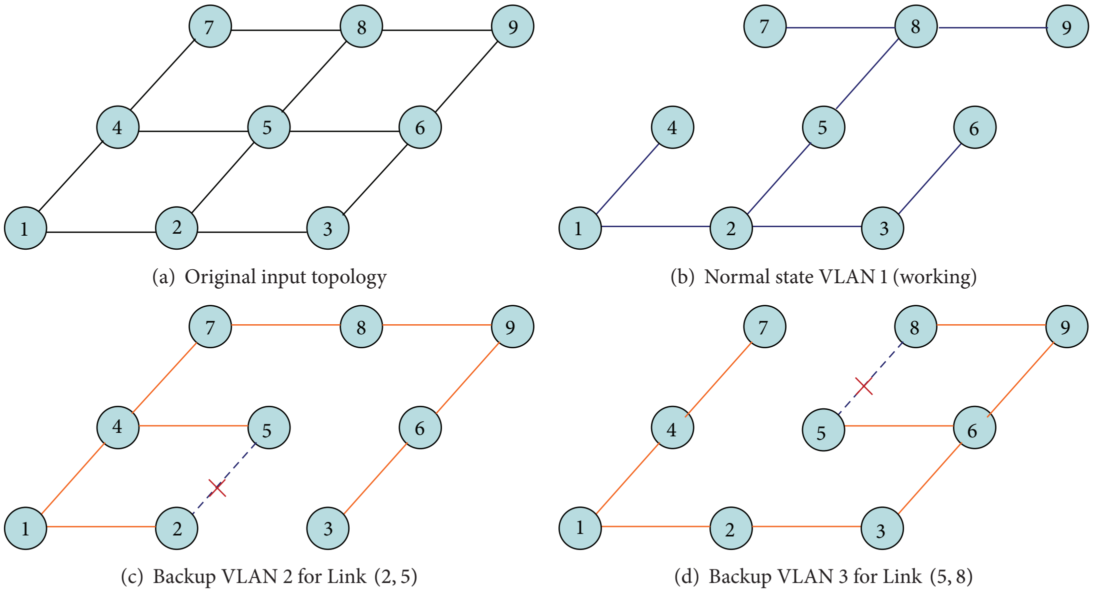
E: set of edges in the transformed graph,

E^{top}: set of edges in the top layer of the transformed graph,

E^{down}: set of edges in the down layer of the transformed graph,

E_{bridge}: set of virtual bridge edges in the network; for example, $E_{\text{bridge}} = \{(1', 1''), (2', 2''), \ldots, (9', 9'')\}$ in Figure 3(b),

E_l: edges in the transformed graph used to represent link l; for example, for $l = (1, 2)$, $E_l = \{(1', 2'), (1'', 2'')\}$ in Figure 3(b),



(a) Original input topology

(b) Normal state VLAN 1 (working)

(c) Backup VLAN 2 for Link (2, 5)

(d) Backup VLAN 3 for Link (5, 8)

Figure 2: Example of the proposed leaf constrained VLAN protection scheme.

E_v^{in}: set of edges entering vertex v,

E_v^{out}: set of edges leaving from vertex v,

E_{bridge}^s: the set containing two edges to connect the top layer and the down layer in state s; those two bridge edges are corresponding to the adjacent vertices of V_{down}^s; for example, for s = Link $(2, 5)$, $E_{\text{bridge}}^s = \{(2', 2''), (5', 5'')\}$,

E_{fail}^s: edges not allowing carrying traffic in failure state s; for example, for s = Link $(2, 5)$, $E_{\text{fail}}^s = \{(2', 5'), (2'', 5'')\}$,

$\tilde{e}$: the reverse edge of edge e; for example, if $e = (1', 2')$ then $\tilde{e} = (2', 1')$,

$v^{\text{top}}(n)$: vertex in the top layer to represent node n; for example, $v^{\text{top}}(1)$ = Node $1'$ in Figure 3(b),

$v^{\text{down}}(n)$: vertex in the down layer to represent node n; for example, $v^{\text{down}}(1)$ = Node $1''$ in Figure 3(b),

$v(n)$: the two vertices representing node n in the transformed graph; for example, $v(1) = \{1', 1''\}$ in Figure 3(b),

V_{down}^s: the two vertices in the down layer that are adjacent to the failure state s; for example, when the state is Link $(1, 2)$, $V_{\text{down}}^s = \{1'', 2''\}$,

M: a big enough number; for example, M can be set to any constant larger than the sum of all traffic demands in the network,

H_d: total traffic demand to destination node d,

h_{nd}: traffic demand volume between source node n and destination node d,

C_l: physical capacity of link l,

T: set of all spanning trees of the input graph,

K: set of working spanning tree instances; $|K|$ is the total number of working trees used in this problem.

Decision Variables

u: utilization of the most congested link,

x_{es}^k: =1, if edge e is used in state s; =0, otherwise; given k, the edge set $\{e \in E^{\text{top}} \mid x_{es_0}^k = 1\}$ forms the working tree k in the normal state and given k and state s $(s \neq s_0)$, the edge set $\{e \in E^{\text{down}} \mid x_{es}^k = 1\}$ forms the backup VLAN tree used to protect working spanning tree k in failure state s,

f_{es}^{dk}: volume of total flow following spanning tree instance k to destination node d carried on edge e in state s,

H_d^k: total traffic demand carried by spanning tree instance k to destination node d,

h_{nd}^k: traffic that is carried on spanning k to realize demand between source node n and destination node d.

Multi-VLAN LCP-TE problem (**IP**):

$$\min u \tag{1}$$

subject to

$$\sum_{k \in K} H_d^k = H_d, \quad \forall d \in N, \tag{2}$$

$$\sum_{k \in K} h_{nd}^k = h_{nd}, \quad \forall n \in N,\ d \in N \setminus n, \tag{3}$$

$$\sum_{e \in E_{v^{\text{top}}(d)}^{\text{in}}} f_{es}^{dk} + \sum_{e \in E_{v^{\text{down}}(d)}^{\text{in}}} f_{es}^{dk} = H_d^k, \quad \forall s \in S,\ d \in N,\ k \in K, \tag{4}$$

$$\sum_{e\in E^{\text{out}}_{v_{\text{top}}(n)}} f^{dk}_{es} - \sum_{e\in E^{\text{in}}_{v_{\text{top}}(n)}} f^{dk}_{es} = h^{k}_{nd}, \tag{5}$$

$$\forall n\in N,\quad d\in N\setminus n,\quad s\in S,\quad k\in K,$$

$$\sum_{e\in E^{\text{out}}_{v_{\text{down}}(n)}} f^{dk}_{es} - \sum_{e\in E^{\text{in}}_{v_{\text{down}}(n)}} f^{dk}_{es} = 0, \tag{6}$$

$$\forall n\in N,\quad d\in N\setminus n,\quad s\in S,\quad k\in K,$$

$$\sum_{k\in K}\sum_{e\in E_l}\sum_{d\in N} f^{dk}_{es} \le uC_l,\quad \forall l\in L,\ s\in S, \tag{7}$$

$$\sum_{d\in N} f^{dk}_{es} \le M\times x^{k}_{es},\quad \forall e\in E,\ s\in S,\ k\in K, \tag{8}$$

$$\bigcup_{e\in E^{\text{top}}} x^{k}_{es_0} \in T,\quad \forall k\in K, \tag{9}$$

$$\bigcup_{e\in E^{\text{down}}} x^{k}_{es} \in T,\quad \forall s\in S\setminus s_0,\ k\in K, \tag{10}$$

$$x^{k}_{es} = x^{k}_{\tilde{e}s},\quad \forall e\in E,\ s\in S,\ k\in K, \tag{11}$$

$$\sum_{e\in E^{\text{in}}_{v}} x^{k}_{es} = 1,\quad \forall v\in V^{s}_{\text{down}},\ s\in S\setminus s_0,\ k\in K, \tag{12}$$

$$x^{k}_{es} = 0,\quad \forall e\in E^{s}_{\text{fail}},\ s\in S\setminus s_0,\ k\in K, \tag{13}$$

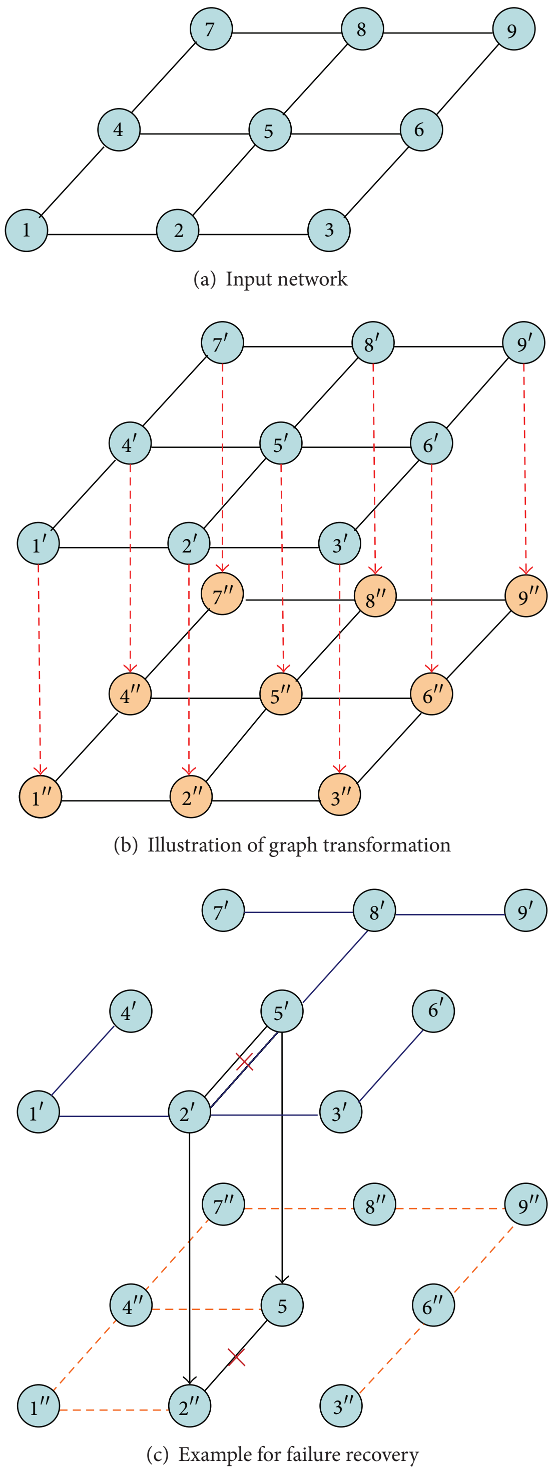
$$x^{k}_{es} = 1,\quad \forall e\in E^{s}_{\text{bridge}},\ s\in S\setminus s_0,\ k\in K, \tag{14}$$

$$x^{k}_{es} = 0,\quad \forall e\in E_{\text{bridge}}\setminus E^{s}_{\text{bridge}},\ s\in S\setminus s_0,\ k\in K, \tag{15}$$

$$x^{k}_{es} \le x^{k}_{es_0},\quad \forall e\in E^{\text{top}},\ s\in S\setminus s_0,\ k\in K, \tag{16}$$

$$x^{k}_{es} = 0 \text{ or } 1,\quad \forall e\in E,\ s\in S,\ k\in K. \tag{17}$$

The objective function is to minimize the link utilization on the most congested link. Constraints (2) and (3) require that the total demand volume destined to node d and the demand volume between any communication node pair has to be carried by the $|K|$ candidate working tree instances. Constraint (4), (5), and (6) jointly enforce the flow conservation law. Constraint (4) requires that the total demand volume to a destination node has to be carried on the input edges of the destination vertices. Because the top layer is used to determine the working VLAN, the local input flow is started from the top layer. In Constraint (5), for each node in the top layer, the difference of flow amount on the outgoing edges and the incoming edges is the local adding flow. Since there is no local adding flow to the vertices in the down layer, Constraint (6) requires that the input flow amount has to be the same as the output flow amount. Constraint (7) requires that the total flow on each link is no larger than its upper bound capacity. The upper bound capacity is determined by the most congested link in the network. Constraint (8) requires that flow can only go through the selected edges. Constraint (9) determines the working VLAN and Constraint (10) determines the backup VLANs. Constraint (9) requires that a working spanning tree be provisioned in the top layer. This tree is used for network under normal state. Constraint



(a) Input network

(b) Illustration of graph transformation

(c) Example for failure recovery

FIGURE 3: Graph transformation and failure recovery.

(10) requires a backup spanning tree in the down layer being provisioned for each failure state. Constraint (11) enforces the tree can use only two way links. Constraints (11) and (12) jointly enforce the leaf constraint. It requires that the two vertices adjacent to the failure link must be leaf vertices in the configured backup tree. Although each failure state is corresponding to a broken link in the formulation, with the help of Constraint (12), the solution obtained from the formulation can not only be used to protect any single link failure but also be used to protect any single node failure. Constraint (13) prevents the backup VLAN from using the failure link. Constraints (14) and (15) require that only two corresponding bridge edges are turned on to provide connection between top layer and down layer. For example in Figure 3(c), only $(2', 2'')$ and $(5', 5'')$ in the bridge edges are turned on. In particular, since all of the bridge edges are turned off, the working tree is determined by the top layer. In Constrain (16), an unaffected node still uses the same routes as those used in the normal state. Only the failure affected node can change VLAN, that is, to switch its flow to the down layer for failure recovery. Finally, Constraint (17) requires the decision variables for VLAN tree configurations are binary.

3. Problem Decomposition and Solution Procedure

Problem **IP** is a complicated problem due to the large number of constraints and integer variables. It is difficult to solve the problem by directly using conventional integer programming techniques. Even without taking backup traffic into account, the load balanced routing for only working traffic is an NP-complete problem. In this work, we propose a two-phase heuristic algorithm to obtain a solution to the multi-VLAN LCP-TE problem. In the first phase, only working traffic is considered. The working trees obtained from Phase I problem become the input parameters to Phase II problem.

The Phase I problem has been studied in [18–20]. Given the traffic demand matrix, a greedy based random algorithm proposed in [20] determines the multiple spanning trees and their VLAN bindings. In [18, 19], the problem for provisioning multiple spanning trees is formulated as an ILP problem. An optimization based heuristic algorithm is proposed to obtain a near optimal solution to the problem. When the input network becomes large, the algorithm proposed in [20] can obtain a near optimal solution with better performance on required computation time.

The algorithm for solving the multi-VLAN LCP-TE problem is described in Algorithm 1. In Phase I, we directly use the algorithm proposed in [20] to determine the routing for working traffic in the normal state. In Phase II, under the given working trees, the multi-VLAN LCP-TE problem can be decomposed into $|S| - 1$ subproblems; each one is corresponding to a failure state. For state s, the subproblem is named problem ($\mathbf{IP_{sub}}$ $\mathbf{s}$) in which the leaf constraint is included. Since the problem size is reduced, each subproblem can be solved by using commercial optimization tool.

Multi-VLAN LCP-TE problem subproblem ($\mathbf{IP_{sub}}$ $\mathbf{s}$):

$$\min u \tag{18}$$

subject to

$$\begin{gathered}
\sum_{e \in E^{\text{in}}_{v^{\text{top}}(d)}} f^{dk}_{es} + \sum_{e \in E^{\text{in}}_{v^{\text{down}}(d)}} f^{dk}_{es} = H^k_d, \quad \forall d \in N,\ k \in K, \\
\sum_{e \in E^{\text{out}}_{v^{\text{top}}(n)}} f^{dk}_{es} - \sum_{e \in E^{\text{in}}_{v^{\text{top}}(n)}} f^{dk}_{es} = h^k_{nd}, \\
\forall n \in N, \quad d \in N \setminus n, \quad k \in K, \\
\sum_{e \in E^{\text{out}}_{v^{\text{down}}(n)}} f^{dk}_{es} - \sum_{e \in E^{\text{in}}_{v^{\text{down}}(n)}} f^{dk}_{es} = 0, \\
\forall n \in N, \quad d \in N \setminus n, \quad k \in K, \\
\sum_{k \in K} \sum_{e \in E_l} \sum_{d \in N} f^{dk}_{es} \le u C_l, \quad \forall l \in L, \\
\sum_{d \in N} f^{dk}_{es} \le M \times x^k_{es}, \quad \forall e \in E,\ k \in K, \\
\bigcup_{e \in E^{\text{down}}} x^k_{es} \in T, \quad \forall k \in K, \\
x^k_{es} = x^k_{\tilde{e}s}, \quad \forall e \in E,\ k \in K, \\
\sum_{e \in E^{\text{in}}_v} x^k_{es} = 1, \quad \forall v \in V^s_{\text{down}},\ k \in K, \\
x^k_{es} = 0, \quad \forall e \in E^s_{\text{fail}},\ k \in K, \\
x^k_{es} = 1, \quad \forall e \in E^s_{\text{bridge}},\ k \in K, \\
x^k_{es} = 0, \quad \forall e \in E_{\text{bridge}} \setminus E^s_{\text{bridge}},\ k \in K, \\
x^k_{es} \le x^k_{es_0}, \quad \forall e \in E^{\text{top}}, k \in K, \\
x^k_{es} = 0 \text{ or } 1, \quad \forall e \in E,\ k \in K.
\end{gathered} \tag{19}$$

4. Simulation and Experimental Results

In this section, we present the numerical results obtained from computer simulations and demonstrate the protection switching time measured on an FPGA-based testbed system.

4.1. Simulation Results. We have carried out simulations on several randomly generated networks. Each network is denoted by $N(d)$, where N is the number of nodes and d is the mean degree. In those networks, capacity of each link was set to 1000 Mbps. We assume each node inside the network has traffic to communicate with each other node. The demand for each node pair is generated randomly with 5 Mbps in average.

We first make performance comparisons on the survivability ratio under single node failure scenario. The destination based algorithm [12] is implemented to obtain the results without taking leaf constraint into consideration. We make performance comparisons between our algorithm and the destination based algorithm on twelve randomly

```
    /* Phase I */
(1) Use algorithm [20] to determine load balanced routing for working traffic;
(2) The output of Step 1 is used to derive $x^{k}_{es_0}$, $H^{k}_{d}$ and $h^{k}_{nd}$ that become given constant values for Phase II;
    /* Phase II */
(3) $u_{\max} := 0$; /* initializing the load on the most congested link */
(4) for $s \in S$
(5)     solve Sub-problem $\mathbf{IP}_{\mathbf{sub}}$ $\mathbf{s}$ to obtain $u$
(6)     if $u_{\max} < u$ then $u_{\max} := u$;
(7) end for
(8) output routing $\mathbf{x}$ for VLAN provisioning and the load on the most congested link $u_{\max}$
```

Algorithm 1: The proposed heuristic algorithm.

generated networks. The results are shown in Figure 4(a). Since survivability constraint is included in our problem, our results can guarantee each network with 100% survivability against any single node failure. Observing the results shown in Figure 4(a) we find that the average connection loss ratio is 10%–20% for the destination based algorithm [12]. This result indicates the importance of the leaf constraint for protection against single node failure.

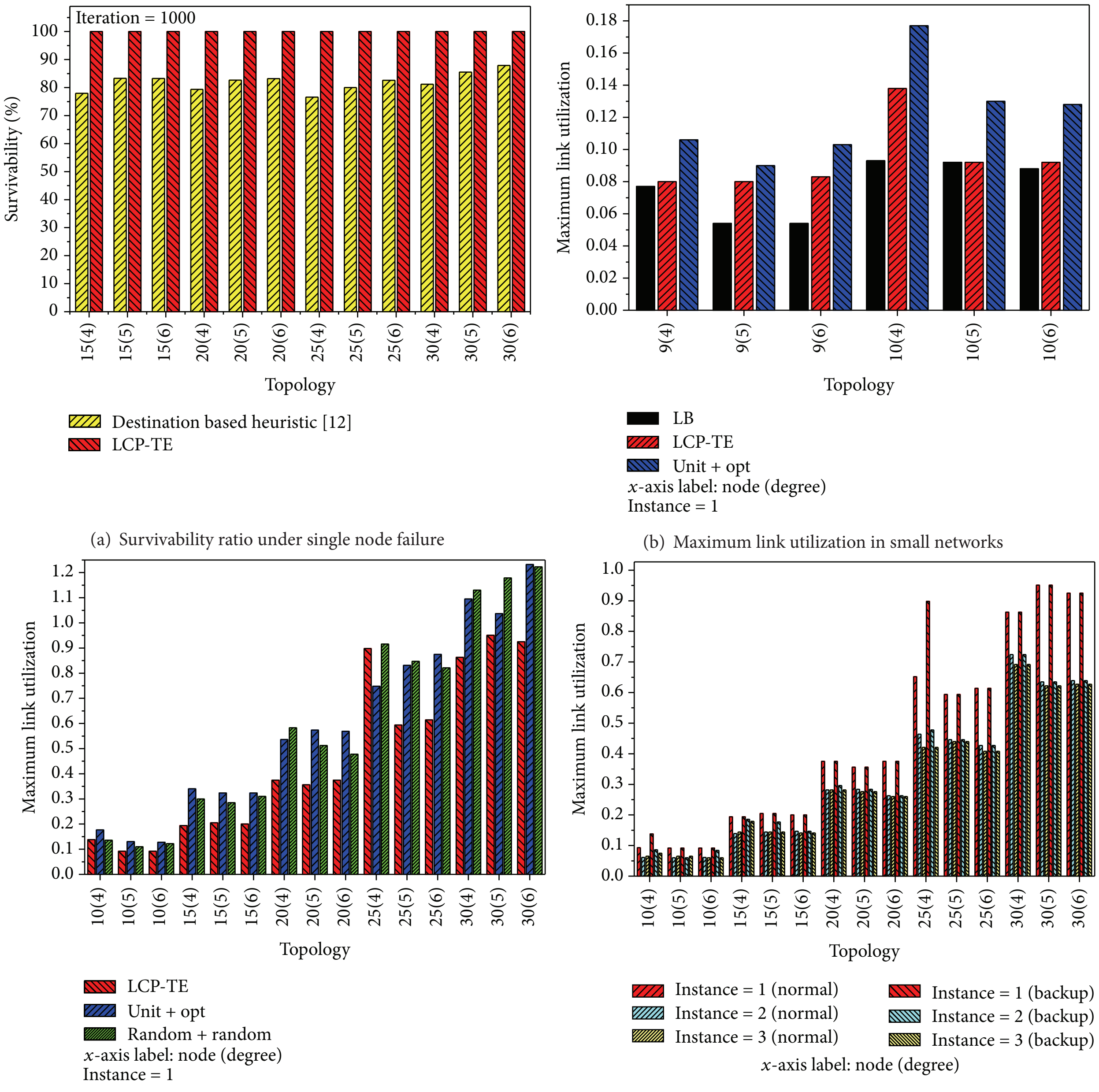
Because the multi-VLAN LCP-TE is an NP-complete problem, we cannot directly solve problem **IP** to obtain its optimal solution. Instead, we use lower bound (LB) values to justify the proposed algorithm. The lower bound values are obtained from solving a relaxed version of problem **IP**. This problem is the same as problem **IP** shown in Section 2 except that Constraint (16) is removed. We directly apply CPLEX to solve this problem. Unfortunately, the relaxed problem is still a very complicated one. CPLEX can obtain an optimal solution to the problem only if the network size is small. The results for small sized networks are shown in Figure 4(b), where the results labeled LCP-TE are obtained from the proposed algorithm shown in Algorithm 1.

Besides the relaxed problem, we also implemented the unit weight heuristic algorithm. In the algorithm, the working spanning tree is obtained using minimum spanning tree algorithm in which each link weight is set to be 1. Then we apply CPLEX to solve the model of Section 2 with the working tree fixed. We have observed that the link utilization on the most congested link obtained from applying the unit weight heuristic is much higher than the optimal solution. The results show that directly using constant weight to obtain the working VLAN is not able to provide good enough solution even if the backup VLANs are taking load balancing into consideration. On the contrary, by considering load balanced provisioning for working trees and backup trees, the proposed algorithm can provide a solution much closer to the lower bound in the test networks.

We further perform simulations to evaluate several algorithms in large networks and show the results in Figure 4(c). In this set of simulations, only one working spanning tree is used for routing working traffic. To make performance comparisons, we further implement a random based algorithm. In the algorithm, we randomly generated a working VLAN tree and its backup VLAN trees. If the leaf constraint is satisfied, this is a candidate solution. The best one among all candidate solutions obtained within 5000 trials is presented in Figure 4(c). Simulation results reveal that the routing provided by the proposed algorithm has better performance in most cases. The only exception is for the 25(4) network. The unit weight scheme and the random selection scheme have mixed behaviors. The results obtained from the random based algorithm have slightly better performance than the unit weight algorithm. Since unit weight heuristic only uses one candidate tree for working traffic, it cannot evenly distribute working traffic in the network. As a result, even if the backup VLANs take load balancing into consideration, it still consumes much bandwidth on the most congested link.

In the final set of simulations, we perform the proposed multi-VLAN LCP-TE heuristic algorithm to evaluate the performance using multiple spanning tree instances for working traffic. The results are shown in Figure 4(d). For each network, there are six data values in the figure. The left three bars indicate the bandwidth consumption by the working traffic in the normal state when the number of working spanning trees is 1, 2, and 3. It is clear that the more the working spanning tree instances, the more the routing paths provided so that the bandwidth required on the most congested link is reduced. The performance improvement between using one spanning tree and two spanning trees is significant. However, the gaps shrink as the number of trees becomes larger than 3. To make the figure easier to read, we do not include instances larger than 3 in Figure 4(d). Since the improvement becomes saturated as the number of instances is larger than 3, we suggest that 3 trees are enough for providing load balancing routing to the multi-VLAN LCP-TE problem. The right three bars depict the worst case bandwidth consumption on the most congested link after failure recovery. The difference between each pair of the right bar and the left bar is the gap of loads on the most congested link in the normal state and in the worst case failure state. Comparing the right three bars with the left three bars we discover that the increase of bandwidth consumption on the most congested link is quite small. The results show that using the proposed algorithm for VLAN provisioning the reroute traffic can successfully avoid the congested links in most network topologies.

4.2. Experimental Results. We further set up an experiment shown in Figure 5(a) to evaluate the required time for failure



(a) Survivability ratio under single node failure

(b) Maximum link utilization in small networks

(c) Performance comparisons in larger networks

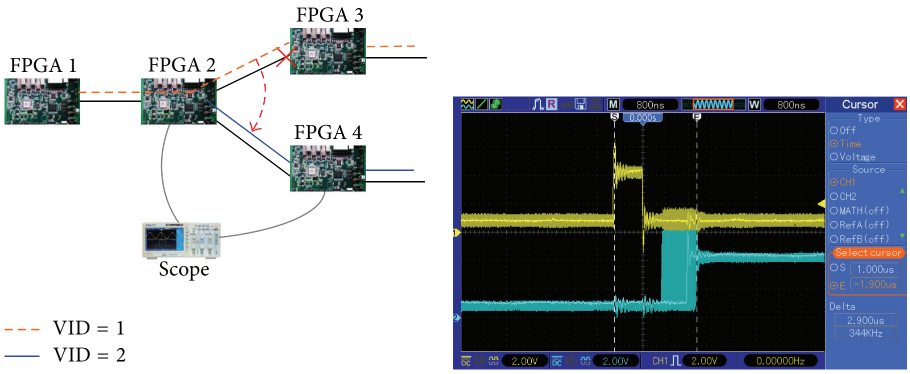
(d) Impact of multiple instances

Figure 4: Simulation results.

recovery. We implement the proposed protection scheme on FPGA boards with Gigabit Ethernet interface. In this experiment, as the line between FPGA 2 and FPGA 3 is disconnected, the traffic on VLAN 1 is switched to VLAN 2 for failure recovery. The experiment is repeated 1000 times in total. Figure 5(b) displays the accumulating results obtained from the oscilloscope. The one shot pulse in Channel 1 (upper line) is generated by FPGA 2 to indicate a failure is detected. As a reroute frame received by FPGA 4, it asserts a signal shown in Channel 2 (lower line). The worst protection switching time is 2.9 μsec in the 1000 experiments. This recovery time is much smaller than the 50 msec requirement used in telecommunication networks.

5. Conclusion

In this paper, we have proposed a novel multi-VLAN based protection scheme for fast failure recovery and congestion avoidance in Ethernet networks. By enforcing the two end nodes of each link to be the leaf nodes on its backup VLAN, our scheme is able to protect any single link failure and any single node failure. We have introduced a graph transformation technique to facilitate problem formulation for this problem. In the proposed optimization model, we take load balancing into consideration to avoid traffic congestion on the most congested link. Since this problem is an NP-complete problem, we further propose a heuristic algorithm to provide a solution to large sized networks.



(a) Experimental setup

(b) Protection switching time

Figure 5: Experimental results.

We have carried out extensive simulations on several randomly generated networks. The simulation results indicate that the proposed algorithm outperforms both the unit weight and the random heuristic algorithms. Although more working VLAN trees can provide more routing paths to improve load balancing, we have observed that the performance improvement becomes saturated as the number of working trees increases. Simulation results indicate that three working trees are enough for providing load balancing routing. Since the maximum number of VLANs in an Ethernet is 4096 VLANs, the proposed approach is free from scalability issue.

To evaluate the failure recovery time in a real system, we also implemented our approach in an FPGA system. Experimental results show that the protection switching time is within 3 μsec. It is much shorter than the 50 msec requirement used in carrier grade telecommunication networks to guarantee highest quality for timing sensitive services.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work is supported by the Ministry of Science and Technology, Taiwan, under Grant no. 103-2218-E-194-007.

References

[1] Cisco, "Data center: Load balancing data center services SRND," 2004, https://learningnetwork.cisco.com/servlet/JiveServlet/previewBody/3438-102-1-9467/cdccont_0900aecd800eb95a.pdf.

[2] J. Mudigonda, P. Yalagandula, M. Al-Fares, and J. C. Mogul, "SPAIN: COTS data-center Ethernet for multipathing over arbitrary topologies," in *Proceedings of the 7th USENIX Conference on Networked Systems Design and Implementation (NSDI '10)*, p. 18, 2010.

[3] "Media Access Control (MAC) Bridges," IEEE 802.1D, Institute of Electrical and Electronics Engineers, 1998.

[4] *Media Access Control (MAC) Bridges: Rapid Reconfiguration, IEEE 802.lw*, Institute of Electrical and Electronics Engineers, 2004.

[5] T. Cinkler, A. Kern, and I. Moldován, "Optimized QoS protection of Ethernet trees," in *Proceedings of the 5th International Workshop on Design of Reliable Communication Networks (DRCN '05)*, pp. 337–344, Naples, Italy, October 2005.

[6] H. Wessing, M. Berger, H. M. Gestsson et al., "Evaluation of restoration mechanisms for future services using carrier Ethernet," *WSEAS Transactions on Communications*, vol. 9, no. 5, pp. 322–331, 2010.

[7] J. Qiu, Y. Liu, G. Mohan, and K. C. Chua, "Fast spanning tree reconnection for resilient metro Ethernet networks," in *Proceedings of the IEEE International Conference on Communications (ICC '09)*, pp. 1–5, Dresden, Germany, June 2009.

[8] J. Qiu, G. Mohan, K. C. Chua, and Y. Liu, "Handling double-link failures in metro ethernet networks using fast spanning tree reconnection," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '09)*, Honolulu, Hawaii, USA, December 2009.

[9] *Virtual Bridged Local Area Networks-Amendment 3: Multiple Spanning Trees, IEEE 802.1s*, Institute of Electrical and Electronics Engineers, 2002.

[10] J. Farkas, C. Antal, G. Toth, and L. Westberg, "Distributed resilient architecture for Ethernet networks," in *Proceedings of the Design of Reliable Communication Networks Conference (DRCN '05)*, pp. 515–522, Naples, Italy, 2005.

[11] J. Farkas, C. Antal, L. Westberg, A. Paradisi, T. R. Tronco, and V. G. de Oliveira, "Fast failure handling in ethernet networks," in *Proceedings of the IEEE International Conference on Communications (ICC '06)*, pp. 841–846, Istanbul, Turkey, July 2006.

[12] J. Qiu, M. Gurusamy, K. C. Chua, and Y. Liu, "Local restoration with multiple spanning trees in metro ethernet networks," *IEEE/ACM Transactions on Networking*, vol. 19, no. 2, pp. 602–614, 2011.

[13] S. Sharma, K. Goplan, S. Nanda, and T. Chiueh, "Viking: a multiple-spanning tree Ethernet architecture for metropolitan area and cluster networks," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '04)*, pp. 2283–2294, Hong Kong, 2004.

[14] M. Son, L. Yongjoon, P. Cheolsig, K. Byungcheol, and J. Lee, "Physical topology discovery in large Ethernet networks," in

Proceedings of the 9th WSEAS International Conference on Communications (ICCOM '05), 2005.

[15] Y. Bejerano, "Taking the skeletons out of the closets: a simple and efficient topology discovery scheme for large ethernet LANs," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, Barcelona, Spain, April 2006.

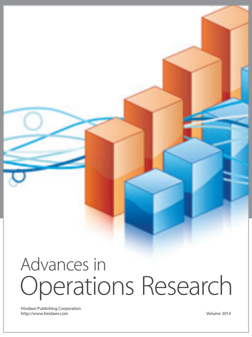
[16] J. Farkas, M. R. Salvador, V. G. de Oliveira, and G. C. dos Santos, "Automatic discovery of physical topology in heterogeneous multi-vendor Ethernet networks," in *Proceedings of the IEEE International Conference on Communications (ICC '08)*, pp. 2055–2060, Beijing, China, May 2008.

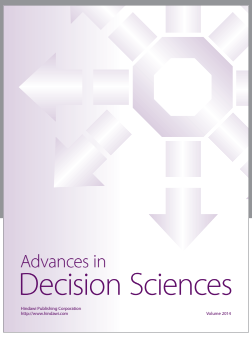
[17] S. Lee, K. Li, C. Lin, and C. Wu, "Multi-VLAN provisioning for fast protection and traffic engineering in ethernet networks," in *Proceedings of the Circuits, Systems, Communications and Computers Conference (CSCC '14)*, Santorini, Greece, July 2014.

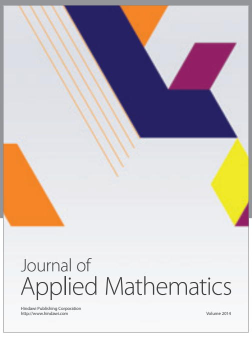
[18] D. Santos, A. De Sousa, F. Alvelos, M. Dzida, and M. Pióro, "Optimization of link load balancing in multiple spanning tree routing networks," *Telecommunication Systems*, vol. 48, no. 1-2, pp. 109–124, 2011.

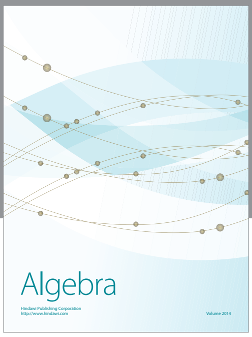
[19] D. de Santos, A. Sousa, F. Alvelos, M. Dzida, M. Píoro, and M. Zagozdzon, "Traffic engineering of multiple spanning tree routing networks: the load balancing case," in *Proceedings of the Next Generation Internet Networks (NGI '09)*, Aveiro, Portugal, July 2009.

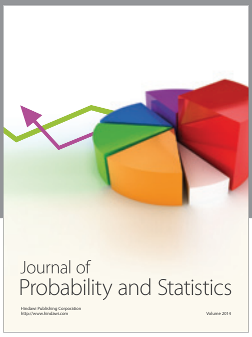
[20] A. F. de Sousa and G. Soares, "Improving load balance and minimizing service disruption on ethernet networks using IEEE 802.1S MSTP," in *Proceedings of the EuroFGI Workshop on IP QoS and Traffic Control*, pp. 25–35, 2007.


Advances in
Operations Research

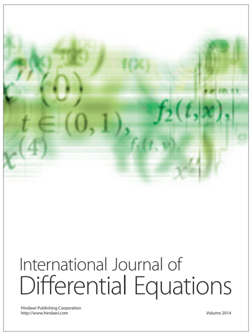

Advances in
Decision Sciences


Journal of
Applied Mathematics

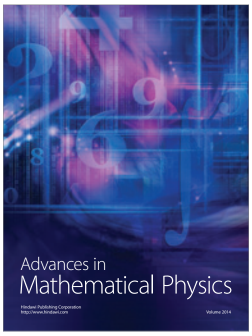

Algebra

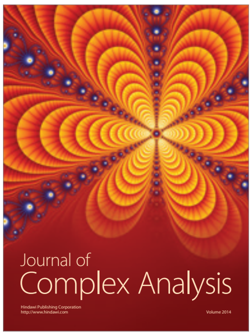

Journal of
Probability and Statistics

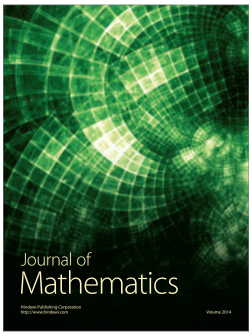
The Scientific
World Journal

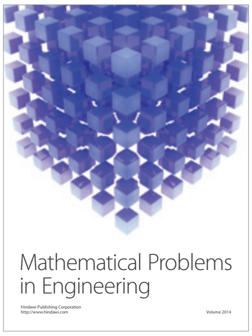

International Journal of
Differential Equations

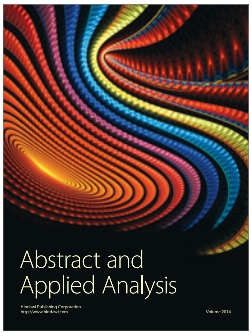
International Journal of
Combinatorics

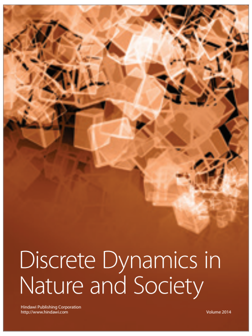

Advances in
Mathematical Physics

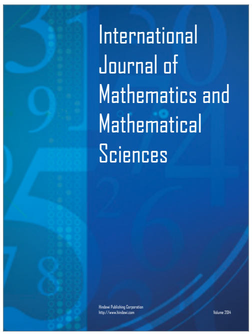

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

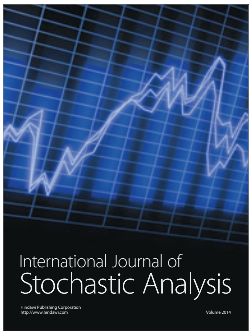

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization