Network change notification protocol(NCNP) for Multi-Layer Networks

Prof H.B.Waliker
Dept.of Computer Science,Karnataka University, Dharwad

Ramesh K.
Dept. of Computer Science,Karnataka State Women's University Bijapur

Abstract— Network changes are normal in Multi-Layer Networks. But, too many of them can have an impact on network performances and also topology change counters increment in the statistics log. Here We tried to determine what triggers a network change mechanism and described the issues related to the network change mechanism. To resolve the above said issues we propose Network change notification protocol (NCNP) for Multi-Layer Networks which advertises the network changes. We also show that our earlier developed dynamic optimal routing algorithm[4] makes use of this protocol to find an alternate shortest path for multilayer survivable networks. We have also included the NCNP in dynamic optimal routing algorithm [1].

Keywords— MAC, Survivable Network, Multi-Layer Network, aging, Change notification

I. Introduction

Routers and bridges create a table that associates to a port the Media Access Control (MAC) addresses the hosts that can be reached through this port. These tables are used to forward frames directly to their destination port. Default aging time for these tables is 300 seconds (five minutes). Only after a host has been silent for five minutes, its entry disappears from the table of the Router and bridge. Here is an example that shows why you could want this aging to be faster.

In this multilayer network, assume that bridge B1 is blocking its link to B4. A and B are two stations that have an established connection. Traffic from A to B goes to B1, B2, B3 and then B4. The scheme shows the MAC addresses table learned by the four bridges in this situation:

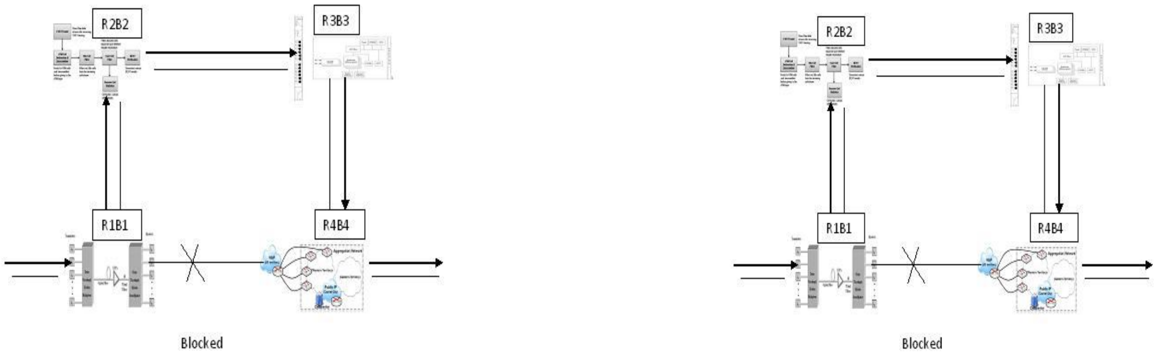

Figure 1.1Shows the need of topology change mechanism

Now, assume the link between B2 and B3 fails. Communication between A and B is interrupted at least until B1 puts its port to B4 in forwarding mode (a maximum of 50 seconds with default parameters). However, when A wants to send a frame to B, B1 still has an entry that leads to B2 and the packet is sent to a black hole. The same applies when B wants to reach the forwarding databases

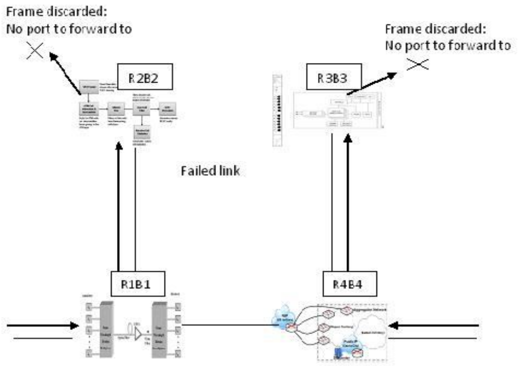

Figure 1.2This figure shows discard of frames when there are no ports to send

implemented by bridges are very efficient in a stable network. But, there are many situations where the five minute aging time is a problem after the network gets change. The network change mechanism is a work around for that kind of problem. As soon as a bridge detects a change in the network (a link that goes down or goes to forwarding), it advertises the event to the whole bridged multilayer-network.

The Principle of Operation section explains how this is practically implemented. Every bridge is then notified and reduces the aging time to forward delay (15 seconds by default) for a certain period of time (max_age +forward _delay). It is more beneficial to reduce the aging time instead of clearing the table because currently active hosts, that effectively transmit traffic, are not cleared from the table.

In this example, as soon as bridge B2 or B3 detects the link going down, it sends topology change notifications. All bridges become aware of the event and reduce their aging time to 15 seconds. As B1 does not receive any packet from B on its port leading to B2 in fifteen seconds, it goes out the entry for B on this port. The same happens to the entry for A on the port that leads to B3 on B4. Later when the link betweenB1 and B4 goes to forwarding, traffic is immediately flooded and re-learned on this link.

II. NCNP WORKING PRINCIPLE

Here we explain how NCNP advertises a network change at the Bridge Protocol Data Unit (BPDU) level. It has already been briefly explained when a bridge considers it detected a topology change. The process involves the following steps:

1. When a port that was forwarding is going down(blocking for instance).
2. When a port transition to forwarding and the bridge has a designated port. (This means that the bridge is not stand alone.)

The process to send a notification to all bridges in the network involves two steps:

3. The bridge notifies the root bridge of the spanning tree.
4. The root bridge “broadcasts” the information into the whole network.

Notifying the Root Bridge

A special BPDU called the network change notification(NCN)BPDU has been introduced. Therefore, when a bridge needs to signal a network change, it starts to send NCNs on its root port. The designated bridge receives the NCN, acknowledges it, and generates another one for its own root port. The process continues until the NCN hits the root bridge.

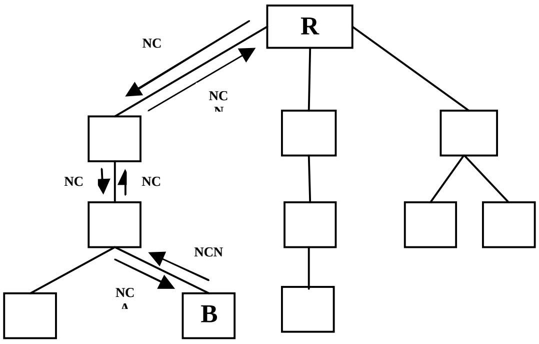

Figure 2.1 : Introduction of NCN to signal a network change

The NCN is a very simple BPDU that contains absolutely no information that a bridge sends out every hello_time seconds (this is locally configured hello_time specified in configuration BPDUs). The designated bridge acknowledges the NCN by immediately sending back a normal configuration BPDU with the network change acknowledgement (NCA) bit set. The bridge that notifies the network change does not stop sending its NCN until the designated bridge has acknowledged it. Therefore, the designated bridge answers the NCN even though it does not receive configuration BPDU from its root.

Broadcast the Event to the Network

Once the root is aware that there has been a Network change event in the network, it starts to send out its configuration BPDUs with the network change(NC) bit set. These BPDUs are relayed by every bridge in the network with this bit set.

As a result all bridges become aware of the network change situation and it can reduce its aging time to forward_delay. Bridges receive network change BPDUs on both forwarding and blocking ports.The NC bit is set by the root for a period of max_age+forward-delay seconds, which is 20+15=35 seconds by default.

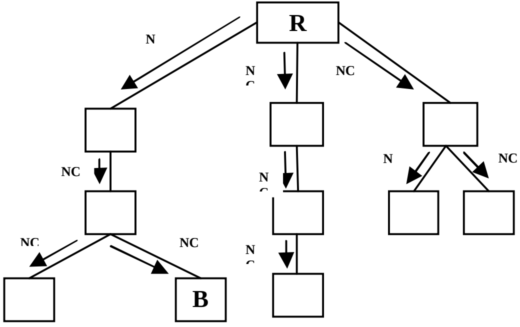

Figure 2.2 :Broadcasting of NC through the network

III. NCNP PROTOCOL DESIGN TO CONFIGURE NETWORK

hostname N1 ... interface Ethernet0 ip address 192.168.1.0 255.255.255.0 ! interface Ethernet1 ip address 192.168.1.1 255.255.255.0 ! interface Serial0 description B1 to B2 link ip address 172.16.250.0 255.255.255.0 ! interface Serial1 description B2 to B1 link bandwidth 56 ip address 172.16.251.1 255.255.255.0 ... **router R1 10** **router R2 11** **network 172.16.0.0** **network 172.16.1.0** hostname N2 --- interface Ethernet1 ip address 192.168.1.1 255.255.255.0 ! interface Ethernet2 ip address 192.168.1.2 255.255.255.0 ! interface Serial2 description B2 to B3 link ip address 172.16.251.1 255.255.255.0 ! interface Serial3 description B3 to B2 link bandwidth 57 ip address 172.16.252.2 255.255.255.0 ... **router R2 11** **router R3 12** **network 172.16.1.0** **network 172.16.2.0**	hostname N3 --- interface Ethernet2 ip address 192.168.1.2 255.255.255.0 ! interface Ethernet3 ip address 192.168.1.3 255.255.255.0 ! interface Serial2 description B3 to B4 link ip address 172.16.252.2 255.255.255.0 ! interface Serial3 description B3 to B2 link bandwidth 58 ip address 172.16.253.3 255.255.255.0 ... **router R3 12** **router R4 13** **network 172.16.2.0** **network 172.16.3.0** hostname N4 --- interface Ethernet3 ip address 192.168.1.3 255.255.255.0 ! interface Ethernet0 ip address 192.168.1.0 255.255.255.0 ! interface Serial3 description B4 to B1 link ip address 172.16.253.3 255.255.255.0 ! interface Serial0 description B3 to B2 link bandwidth 58 ip address 172.16.250.0 255.255.255.0 ... **router R4 13** **router R1 10** **network 172.16.0.0** **network 172.16.3.0**

As soon as a bridge detects a change in the topology of the network (a link that goes down or goes to forwarding), it advertises the event to the whole bridged network.NCNP can be configured to advertise the network change.

advertise ip TCNP Bridge_B1 advertise ip TCNP Bridge_B4 advertise ip TCNP IP-TCNP: 172.16.250.0/172.16.253.3 While aging > 15	advertise ip TCNP Bridge_B2 advertise ip TCNP Bridge_B3 advertise ip TCNP While aging > 15 IP-TCNP: 172.16.251.1/172.16.252.2 Embedding of TCNP in Optimal Routing and wavelength assignment algorithm

The multi-layer network may have several options to reconfigure parts of the routing in case of an equipment failure. These options are applicable or desirable depends on the technological context, long-term contracts, and other criteria. As we are dealing with transport networks, we assume that survivability against single physical link or node failures is ensured by 1+1 protection. This survivability mechanism, which is commonly used in transport networks, also provides survivability against single failures of ports or line cards. With 1+1-protection, a demand is routed on two physically node- and link-disjoint paths simultaneously, and the target node chooses the better of the two arriving signals. If any single node or link fails, at least one of these paths survives (unless one of the end-nodes fails).The assignment of backup paths for a given working path is fixed and does not depend on the failure state. One important consequence is that backup capacity cannot be shared between different demands depending on the failure state, but has to be reserved and preconfigured for each particular demand. Although this approach may require much backup capacity which is normally unused, it is attractive for network operators because nothing has to be reconfigured in case of a single node or link failure, such that all connections can be continued without disruption.

IV. WAVELENGTH ASSIGNMENT

In an IP over WDM ,SDH or ATM setting, the logical links correspond to lightpaths in the fiber network, and wavelengths must be assigned to the lightpaths such that any two lightpaths sharing a common fiber have different wavelengths. In our model assignment of wavelength is technology-specific, and second, finding a conflict-free wavelength assignment is an extremely hard problem on its own. Our models limit the number of logical links that may traverse a given physical link. The approach for choosing between different wavelengths is to simply select one of the wavelengths at random. The wavelength selection is done in a distributed manner,with only limited or outdated information, then random wavelength assignment may outperform than other assignment approaches. The reason for this behaviour is that, in other approaches, if multiple connections are attempting to set up a lightpath simultaneously, then it may be more likely that they will choose the same wavelength, leading to one or more connections being blocked.

Algorithm 1 Finding Optimal Routing

```
Call NCNP ( ) // Optimal routing algorithm calls NCNP( ) to learn current network state //
Input: commodities K, logical link cost $k_l$, logical link capacities $c_l$
Output: a routing for all commodities satisfying the demand and diversification constraints of model (3.4), and
    capacity constraints with respect to $c_l$.
5. for all $k \in K^u$ do
Hide all logical links $l \in L$ with $c_l < d^k/2$ from the graph.
Compute a shortest-path routing in the logical layer with respect to the $k_l$
end for
for all $k \in K^p$ do
Hide all logical links $l \in L$ with $c_l < d^k/2$ from the graph.
Find a shortest logical path with respect to the given logical link cost.
Hide all logical links from the graph failing together with any link from the
    first path.
Find another shortest path in the remaining logical network.
If this approach fails, call Algorithm 2
Call Algorithm 3 // to assign wavelength//
end for
```

This iterative approach succeeds in finding two failure-disjoint paths. If the algorithm fails for some protected path k ∈ K_p, we use the more sophisticated but also computationally more expensive algorithm Cover Physical Cycles, it is achieved in Algorithm 2.

Algorithm 2. It computes two disjoint paths from s_k to t_k in the physical layer and covers them with cheap logical links

```
Algorithm 2 Cover Physical Cycles
Algorithm 3 COVER PHYSICAL CYCLES
Input: a commodity $k \in K^p$, logical link cost $k_l$, logical link capacities $c_l$.
Output: a routing for commodity $k$ satisfying the routing constraints of model (3.4).
Find a shortest cycle through $s^k$ and $t^k$ in the physical graph with respect to some Physical link cost. This yields two physical paths $p_1, p_2$ $from\, s^k\, to\, t^k$.
Cover$p_1, p_2$ with few logical links using an auxiliary graph $H$ (see Figure 4.2)
for all $\boldsymbol{p} \in \{\boldsymbol{p_1}, \boldsymbol{p_2}\}$ do
    Define a node in H for every node of p.
    for all $\boldsymbol{l} \in \boldsymbol{L}$ with $c_l \leq d^k/2$do
        If $l$ is part of $p$, define an undirected edge in H with cost $k_l$.
    end for
    Search for a shortest $s^k - t^k$-path in H.
end for
```

After execution of algorithm1 & algorithm2 the algorithm3 is called to assign wavelength.
Algorithm 3 Wavelength assignment using Random selection

```
Algorithm 3 WAVELENGTH ASSIGNMENT
Input:$\boldsymbol{graph\ G(V,E)}$ $where\ V\ is\ the\ set\ of\ routers\ and\ E\ set\ of\ links$ and
Output: $finding\ of\ paths\ from\ src\ to\ dst\ \ with\ k\ wavelengths\ forming$
            $layered\ graph\ G'(V',E')$
$for\ all\ n \in V, n_i \in V'\ for\ i\ = 1,2, \dots, k. In\ addition, two\ nodes\ src'and\ dst'$
            $are\ added\ to\ the\ graph$
$For\ an\ edge(u,v) \in E, (u_i,v_i) \in E' if and\ only\ if\ the\ i_{th} wavelenth\ on\ link$
$(u,v)$$\boldsymbol{is\ available. In\ addition\ links(src',src_i)}$$for\ i\ = 1,2,..k, and\ (dst_j, dst')$
$for\ j\ = 1,2,..k\ are\ added\ to\ the\ layered\ graph.$ end for
```

V. CONCLUSIONS

In this paper we have proposed a protocol, called Network change notification protocol (NCNP) for Multi-Layer Networks which advertises the network changes. At present we have not worried about the specific compiler to compile the language but in future we plan to develop a compiler which can run on different platforms.

ACKNOWLEDGMENT

The heading of the Acknowledgment section and the References section must not be numbered. Causal Productions wishes to acknowledge Michael Shell and other contributors for developing and maintaining the IEEE LaTeX style files which have been used in the preparation of this template. To see the list of contributors, please refer to the top of file IEEETran.cls in the IEEE LaTeX distribution.

REFERENCES

1. Ramesh K;H.B.Walikar“Algorithmic Design for Multi-Layer Survivable Networks and Failure Recovery,” International Journal of Engineering and Innovative Technology (IJEIT) Volume 3, Issue 2, August 2013.
2. H.B Walikar;Ramesh K; Hanumantu K-Edge-Connectivity: a new Approach of finding Minimum Spanning Tree Ordered Minimum Spanning Tree (OMST) ,” International Journal of Engineering and Innovative Technology (IJEIT) Volume 3, Issue 2, August 2013
3. Jeroen van der Ham;Paola Grosso”Using the Network Description Language in Optcal Networks.”
4. Cigno, R.L.; M.Munafo. 1995. "RC - A Flexible Language for the Specification of ATM Networks Simulation Experiments." In Proceedings of the Third Workshop on Performance Modelling and Evaluation of ATM Networks. (Ilkey, UK, July 2-6).
5. Marsan, M.A.; A.Bianco; T.V.Do; L.Jereb; R.L.Cigno; and M.Munafo. 1995. "ATM Simulation with CLASS." Performance Evaluation 24: 137-159.
6. MIL 3, Inc. 1996. OPNET Modeling Manual, Release 3.0, Washington, DC.
7. Moldovan, D.I. 1993. Parallel Processing, from Applications to Systems, Morgan Kaufmann, 191-204.
8. Shanmugan, K.S.; V.S.Frost; W.LaRue. 1992. "A Block-Oriented Network Simulator (BONeS)." Simulation. (Feb.).
9. Varga, A.; Gy.Pongor. 1997. "Flexible Topology Description Language for Simulation Programs". In Proceedings of the 9th European Simulation Symposium. (Passau, Germany, October 19-22).