

Topology optimization of IP over ATM

L. Freléchoux, M. Osborne, R. Haas
{frl, osb, rha}@zurich.ibm.com

IBM Research, Zurich Research Laboratory,
Säumerstrasse 4,
CH-8803 Rüschlikon,
Switzerland

Abstract:

This paper discusses an extension to the PNNI augmented routing (PAR) protocol where PAR servers taking part in the Private Network to Network Interface (PNNI) protocol, and therefore aware of the PNNI topology, tag the PAR information delivered to the routers with topological metrics. These topological metrics reflect the location of the node that originated the PAR information. Routers can use this information to decide intelligently with which other routers to peer and establish adjacencies, thus optimizing the IP topology overlaying the ATM topology. Two optimization scenarios are described, one in the context of an enterprise network running OSPF, the other in the context of mobile networks running unicast RIP v2.0 to communicate among themselves.

Keywords: PNNI, OSPF, PNNI augmented routing, topology abstraction, mobile networks

1. Introduction

The complexity of running IP overlaying ATM in networks is commonly regarded as a drawback to the deployment of ATM networks. ATM, however, with its extended support for quality of service, is often deployed by Internet Service Providers as a physical layer for their IP network. This results in the well-known picture of an ATM cloud with IP routers attached at its edge.

Several protocols have been developed and standardized to run IP overlay ATM. The most common ones are LAN Emulation (LANE) and Classical IP (CLIP). Whereas LANE emulates an Ethernet-type network, CLIP provides only the ARP resolution of IP addresses to ATM addresses and broadcast capability. Both of these protocols were designed at a time when ATM was still foreseen to extend to the desktop and therefore was expected to provide scalability and flexibility to handle thousands of "dumb" end systems.

In many cases, however, these protocols are used today to establish connectivity between IP routers, so that IP routing protocols can be run between the routers. Both the

use of Classical IP and LANE require consequent configuration on the routers and is often perceived as a major cause of the complexity of ATM technology.

Furthermore, routers running IP routing protocols such as Open Shortest Path First (OSPF) and Routing Internet Protocol (RIP) can only discover peering routers by multicasting information (e.g. OSPF hello). To circumvent this problem, OSPF has been extended to support Non-Broadcast Multi-Access (NBMA) networks. OSPF routers running in the NBMA mode no longer require broadcast capability from the media, and can therefore use Classical IP to perform the ATM ARP resolutions. Running a router in NBMA mode, however, requires the list of remote routers it must peer with to be configured.

PNNI Augmented Routing (PAR) and proxy-PAR have been standardized at the ATM Forum and were developed to provide an integrated mechanism that would reduce the configuration necessary in IP routers attached to an ATM network. Proxy-PAR allows IP routers to register their protocol interfaces (e.g. OSPF interfaces) within an ATM network and query about the protocol interfaces that other routers attached to the ATM network have registered. The ATM network is in charge of reliably flooding the PAR information through the Private-to-Private Network Interface (PNNI) routing protocol. PAR and proxy-PAR not only make the configuration of neighbors in OSPF superfluous, but also provides the IP-to-ATM address resolution and therefore preclude the need for Classical IP.

The proxy-PAR protocol has been designed as a simple client/server protocol with most of the intelligence on the server side (i.e. the ATM switch). As mentioned above, the protocol allows a router to query its serving ATM switch about the protocol interfaces registered by other routers attached to the ATM network. Whereas the PAR information about a protocol interface delivered by an ATM switch to a router contains all the necessary information to establish communication between two routers, no indication is given as to the location where the information was generated.

This paper presents a new extension to the proxy-PAR protocol where the PAR information delivered by an ATM switch to a router is tagged with abstracted topological information about the location where the information was generated. Described and illustrated by examples, the topological information can be used by the routers to optimize the IP topology overlaying the ATM topology and, for example, to avoid a full mesh of adjacencies between the OSPF routers.

This paper is organized as follows. Section 2 gives a brief introduction to the concepts in the PNNI protocol, introduces the PNNI augmented routing protocol, and describes the common solutions to run OSPF overlay of an ATM network. Section 3 introduces new filtering methods to remove redundant PAR information delivered by an ATM switch to a router based on topological considerations. Section 4 builds on the new filtering and describes in detail the proposed tagging of PAR data with abstracted topology information. Section 5 illustrates the application of the abstracted topological metrics with the help of two scenarios: an enterprise network running OSPF overlay ATM, and a group of ATM mobile networks running RIP v2.0 on a physical ATM network. A summary of the tagging of PAR information with abstracted topological information as well as the conclusion constitute the last section.

2. Background

2.1 Private Network to Network Interface (PNNI)

An ATM network that uses the PNNI routing protocol [1] is arranged in hierarchical peer groups (groups of switches). At the bottom level, ATM switches are connected in arbitrary topologies. In each peer group one ATM switch is elected as a peer group leader (PGL). The PGL represents its peer group at the next level of the hierarchy, which is labeled the logical group node (LGN). An LGN and its corresponding PGL are always found on the same physical machine. The process of clustering nodes into peer groups is recursively applied at each level of the hierarchy. Peer groups are formed by configuring each node with a peer group id (PGID). Neighboring switches exchanged hello messages with each other to learn each other's configuration and to determine whether they belong to the same peer group. The clustering of LGNs into peer groups is based on the same process, each LGN also being configured with a PGID. A PNNI hierarchy is illustrated in Figure 1.

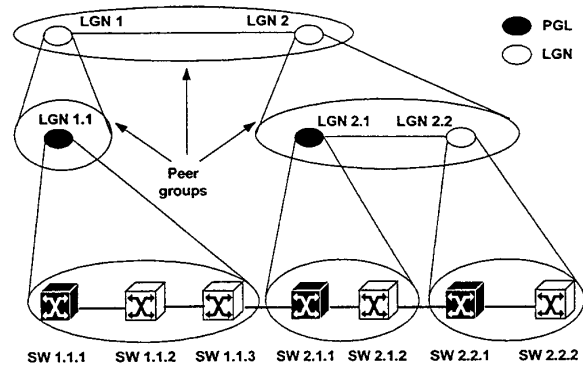


Figure 1: PNNI hierarchy of an ATM network.

Routing information is encapsulated in so-called PNNI Topology State Packets (PTSP). PTSPs contain one or more PNNI Topology State Elements (PTSE). Each PTSE is uniquely identified by a PTSE ID, and further qualified with, among other things, a sequence number and a time to live. The unit of encapsulation is the information group (IG). PTSEs comprises one or more IGs. Each IG has a standard header, which contains, among other things, the IG type and its length in bytes.

PTSPs are exchanged between the nodes in a peer group. PTSPs contain information about the topology (node and links), reachability information (which end systems are reachable through this node), and the local conditions of each node and link (Metrics). PTSPs are flooded within a peer group, and each node maintains a local copy of the routing information it receives. The flooding mechanism ensures that each node will end up with the same routing information, something essential for link-state routing.

In each peer group, the PGL is responsible for summarizing the routing information (topology, reachability, metrics) and regenerating it at the next level of the hierarchy under its own name (Node ID). Routing information is also flooded downwards in the hierarchy by the LGNs that send to their child nodes the routing information collected from within their own peer group. The routing elements that are flooded downwards are not regenerated, they are merely propagated and as such retain the encapsulation information from the node that generated or regenerated them. Thus an ATM switch receives the complete routing information from its bottom-level peer group in addition to the summarized routing information from all of its ancestor peer groups.

The addressing scheme used in ATM is similar to that of the IP addressing scheme, in that it is based on hierarchical addresses. A logical group node can only summarize the reachability information of its child nodes if its child nodes share a common address prefix. If an address cannot be summarized, it is regenerated as a foreign address by the LGN.

The scalability of PNNI resides in having peer groups with a reasonable number of nodes, and in its capability of summarizing routing information available in one peer group to advertise more abstract and less detailed information at the next level of the hierarchy.

2.2 PNNI Augmented Routing (PAR)

PAR [2] is an enhancement to PNNI that simplifies the integration of IP and ATM networks. Today, when IP networks are interconnected through an ATM backbone network, the configuration necessary is complex and cumbersome. Either PVCs have to be configured to connect each router, or an emulation protocol has to be introduced, such as LANE [3] or Classical IP over ATM [4].

PAR makes use of the PNNI protocol to reliably distribute IP-related information. The IP information is encapsulated within PNNI PTSEs and transparently propagated throughout an ATM network. PAR integrates IP and ATM in terms of external services: PNNI is extended to carry IP-related information and flood it transparently throughout the PNNI hierarchy. The IP information remains opaque to PNNI to a certain extent, as we will describe next.

Introduction to proxy-PAR

Proxy-PAR is a client-server protocol in which a proxy-PAR Client runs inside the router, and a proxy-PAR Server inside the switch to which the router is directly attached. The reserved VPI:VCI (0:18) is used for the client-server communication. The proxy-PAR protocol is actually composed of three different protocols, namely the proxy-PAR Hello Protocol, the proxy-PAR Registration Protocol, and the proxy-PAR Query Protocol. The Hello protocol is similar to the PNNI Hello protocol, and monitors the state of the connection between the client and the server. The proxy-PAR Registration Protocol allows the client to send the information it wants the server to register on its behalf in the PNNI database. Finally, the proxy-PAR Query Protocol allows the client to obtain information extracted by the server from the PNNI database. The Registration and the Query protocols guarantee reliable transfers between the client and the server. The client registers the services it supports to its server by sending Service Description packets. A Service Description packet contains the ATM address of the client, the UNI scope at which the information has to be flooded (it is then translated by the server to the corresponding PNNI scope), and one or more PAR IPv4 Service Description IGs or PAR VPN ID IGs.

Figure 2 shows an example of an ATM network, composed of four switches. Two routers are attached to the ATM network, and interconnect with Ethernet networks. The router on the left-hand side runs a full PAR implementation, and can therefore inject PAR PTSEs directly into the PNNI network. The router on the right-hand side has the lightweight proxy-PAR client, which runs the proxy-PAR protocol with the server.

Note that the proxy-PAR protocol is completely client-driven. It is the responsibility of the client to reregister its services periodically before they expire in the PNNI database of the server. The client also has to query regularly for the services in which it is interested.

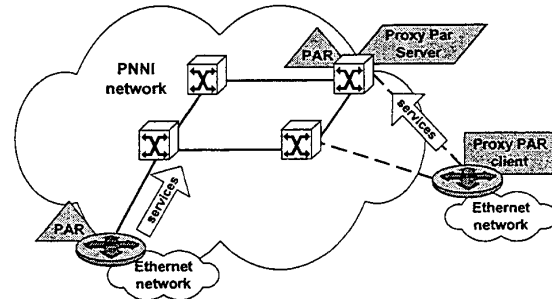


Figure 2: Service integration with PAR.

PAR Scoping

By setting appropriate scopes to the services registered through proxy-PAR, it is possible to control how the IP routing hierarchy is formed, as will be shown in the following example. The scope defines the highest level in the PNNI hierarchy that the registered services should be flooded.

Figure 3 shows a network consisting of an ATM backbone that interconnects a number of IP routers. The switches in the ATM network are arranged in a two-layer hierarchy with two peer groups at level 88 forming the lower layer. These peer groups are interconnected with a common peer group at level 64. The Switches all run proxy-PAR and PAR. The IP hierarchy consists of two Autonomous Systems (AS1 and AS2) running OSPF and interconnected with Border Gateway Protocol (BGP) [5]. Routers R2 and R3 run both OSPF and BGP. The IP topology matches that of the ATM topology: where the IP routers of within a single AS all connect to ATM switches in the corresponding peer group. All routers use proxy-PAR to register their OSPF service capabilities with the corresponding proxy-PAR server. The OSPF services are registered with local scope, i.e., with PNNI level 88.

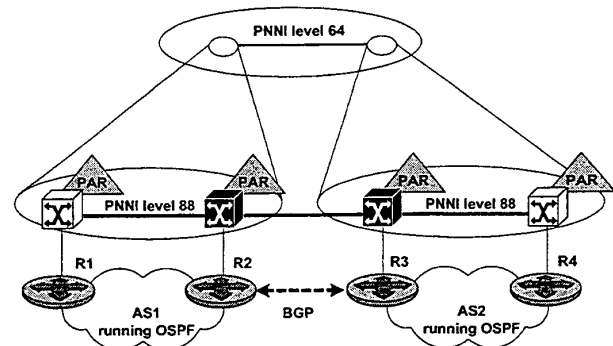


Figure 3: Hierarchical IP network with PAR.

Consequently, the service announcements are distributed only within the local peer group. Because of the alignment of the ATM and the IP hierarchies, only routers within the same AS receive each other's OSPF announcements. Routers R2 and R3 in addition register their BGP service capabilities with scope "one above local", which the proxy-PAR server translates into PNNI level 64. Hence, router R2's BGP service announcements are flooded throughout the entire network and eventually reach R3 (and vice versa), allowing the two routers to form a BGP adjacency.

2.3 Open Shortest Path First (OSPF)

OSPF [6] is an Interior Gateway Protocol (IGP) used inside IP networks. In a similar way to PNNI, OSPF routers share the information about their local connectivity in a globally replicated, distributed database. This database is used by a link-state routing algorithm to calculate routes. While OSPF supports only two levels of hierarchy (PNNI has a limit of 104), it can work over a large a variety of network types, such as:

- point-to-point networks: a network that joins a single pair of routers;
- broadcast networks: networks supporting more than two attached routers, together with the capability of addressing a single physical message to all of the attached routers (broadcast). Neighboring routers are discovered dynamically on these networks using the OSPF Hello Protocol. The Hello Protocol itself takes advantage of the broadcast capability. The protocol makes further use of multicast capabilities, if they exist. An Ethernet is an example of a broadcast network;
- non-broadcast networks: networks supporting more than two attached routers, but having no broadcast capability. Neighboring routers are maintained on these nets using OSPF's Hello Protocol. However, due to the lack of a broadcast capability, some configuration information is necessary for the correct operation of the Hello Protocol. On these networks, OSPF protocol packets that are normally multicast need to be sent to each neighboring router in turn. An X.25 Public Data Network (PDN), or an ATM network is an example of a non-broadcast network.

OSPF runs in one of two modes over non-broadcast networks. The first mode, called non-broadcast multi-access (NBMA), simulates the operation of OSPF on a broadcast network. The second mode, called Point-to-MultiPoint, treats the non-broadcast network as a collection of point-to-point links. Non-broadcast networks are referred to as NBMA networks or Point-to-MultiPoint networks, depending on OSPF's mode of operation over the network.

Neighbor discovery in OSPF over ATM

When using a broadcast-simulation-based over an ATM network, such as LANE or ARP over ATM together with Multicast Address Resolution Service (MARS) [7], OSPF treats the ATM network as a broadcast network, so that OSPF neighbors can be discovered automatically through the broadcasting of Hello messages. In the latter alternative, in addition to configuring the address of the servers, each router has to register itself as part of the AllOSPF IP multicast group to the MARS Server.

The opposite is true of an ATM network running Classical IP without MARS. Here the OSPF neighbors have to be manually configured in each router.

With the help of MARS, point-to-multipoint SVCs can be used for the IP multicast traffic. Such multicast solutions complement broadcast-simulation-based solutions, which are used for address resolution. The major drawback of these solutions, is that they suffer from the single point of failure of their respective servers (ATM ARP server, LANE server, MARS server).

Whereas client-side configuration of these server-based solutions can be completely automated, either through proxy-PAR [8] or other mechanisms [9, 10], these solutions still require substantial configuration on the server side, especially if the network needs to be structured with multiple different servers.

In the case of OSPF (as well as many other protocols, such as RIP, IGRP, IS-IS, BGP, etc., see [2]), it is possible to circumvent these servers completely and still benefit from the automatic neighbor discovery process by using proxy-PAR in each router to discover the potential neighbor routers. This allows OSPF to run in NBMA or point-to-multipoint modes, with no prior configuration of the neighbors' ATM addresses [11]: neighbors are discovered dynamically through proxy-PAR queries from the router to the ATM switch where they are attached, together with their corresponding ATM address.

In the specific case of a PVC environment, Inverse ATMARP [4] offers a simpler discovery mechanism that does not require a central server: Inverse ATMARP allows the IP address discovery of the device located at the remote end of a given PVC. But one should note that this mechanism, unlike proxy-PAR, does not return whether the remote device is an OSPF router. It also means that most of the configuration work has been completed when initially setting up the PVCs.

Automatic OSPF overlay set-up over ATM

Unlike server-based solutions or PVC-based solutions, which require substantial up-front configuration, proxy-PAR allows the seamless installation of OSPF routers in an ATM network, and lets them discover each other automatically. Clearly, the resulting OSPF overlay topology depends on the PNNI hierarchy that has been defined prior to this. The goal of the mechanisms

introduced in this paper is to allow the automatic setup of an optimal OSPF overlay over a hierarchical ATM-PNNI network. Using proxy-PAR with the appropriate PNNI scoping can only partly help define an optimal overlay. We introduce additional mechanisms into proxy-PAR that result in better overlays.

3. Filtering of redundant PAR data

Before PAR information is subject to retrieval by client proxy-PAR queries, the proxy-PAR server can use several methods to filter redundant information. Note that, whereas this information can be removed from the client query responses, it cannot be removed from the PNNI PTSE database, where it is necessary to conform with the PNNI protocol.

PAR information is considered redundant in the following two cases:

- **non-connected originator**, where the PAR PTSE was generated by a node that has no connectivity to the proxy-PAR server. This may happen if a peer group is partitioned due to link failure, if an ATM switch becomes defect, or in the case of mobile PNNI after a mobile node moves from one peer group to another.

- **LGN duplicates**, where the PAR PTSE contents were duplicated by successive LGN processing. The problem occurs with the LGN regeneration of PAR PTSEs in a PNNI hierarchy. As the PAR information contained within PTSEs is essentially opaque to PNNI, the only method to distribute the contents throughout a portion of the hierarchy is to regenerate (duplicate and repackage) the PAR PTSEs at each LGN. Contrary to PNNI PTSEs there is no summarization process carried out on the IGs contained within the PTSEs. As a result and especially in PNNI hierarchies with a significant number of layers, PAR IGs will be duplicated several times, each time encapsulated in a unique PTSE that has been generated by an LGN one level higher in the hierarchy.

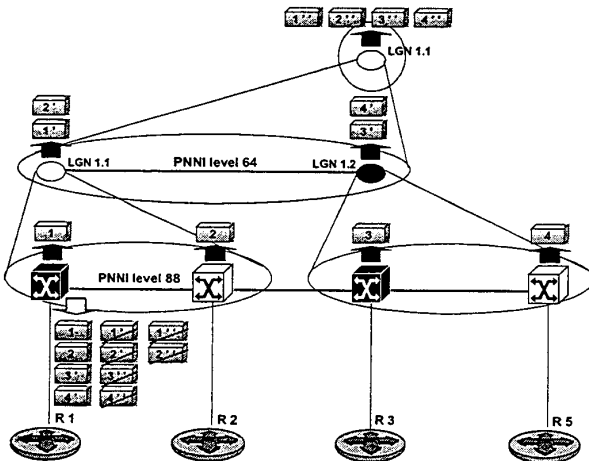


Figure 4: Filtering of redundant PAR data.

Take for example the case in Figure 4. If we look at the case of PAR PTSE 4 that has been registered by router R6 with a global flooding scope. It is flooded by the serving switch throughout the peer group at level 88. The PGL at level 88 receives the PTSE and the corresponding LGN 1.2 regenerates the information in PAR PTSE 4', which is then flooded within the peer group at level 64 and subsequently all child peer groups. Recursively, the PGL at level 64 received the PTSE and the LGN one layer higher regenerates it as PAR PTSE 4'' and it is flooded within its own peer group and all child peer groups. The end result is that the proxy-PAR server attached to router R1 will receive PAR PTSE 4' and 4'', both containing the same PAR IG. If fact Figure 4 shows that, of the 10 PAR PTSEs in its database, only four of them are not duplicates.

4. Tagging of PAR data with abstracted topology information

Currently, IP routers clustered at the edge of an ATM cloud have no means, aside from complex manual configuration, to generate optimal IP overlay networks that topologically match the underlying ATM network.

Whilst proxy-PAR allows these IP routers to discover automatically their peer routers on the ATM network, there is no indication of the suitability of a peer router as a direct IP adjacency. The resulting full mesh of IP adjacencies, which run oblivious to the underlying topology, is more often than not highly undesirable.

In order for such an IP router to make informed decisions about the IP adjacencies that it builds, additional information has to be generated by the ATM switch (proxy-PAR server) and passed back to the router.

The existing proxy-PAR standard has been designed such that the majority of the intelligence is integrated inside the ATM switch while keeping the IP router extensions as simple as possible. The extensions presented in this section follow that same philosophy.

One method would be to provide the router with access to the PNNI routing database, or some subset thereof. However, this solution would entail substantially more intelligence being integrated into the router. This runs against the proxy-PAR philosophy of keeping the client as simple as possible. The solution presented in this paper is to reduce the PNNI routing topology into a few simple abstract attributes. These basic attributes, while simple for a router to manage, provide a powerful extension to the proxy-PAR protocol.

The abstracted attributes are passed to the proxy-PAR client in the form of tags attached to the standard proxy-PAR information groups returned in a client (IP Router) query.

This paper deals with three such examples of abstracted topology information that can be used as tags:

- (1) Topological Distance, which represents the distance between an ATM switch returning PAR information to a router and the ATM switch that originated the PAR information.
- (2) Topological Level, which represents the number of hierarchy layers above that of the ATM switch returning PAR information to a router that PAR information was originated.
- (3) Physical Neighbor Tag, which is a flag indicating whether the PAR IG contained within the PTSE was generated by an ATM switch that is a physical neighbor.

Topological Distance

This is a measure of the distance between any two nodes in a PNNI network. This measure can be modeled in various ways, for simplicity we will use a static measure, the 'hop' count. The hop count between two nodes in a PNNI network is given by the minimum number of links that have to be traversed to reach one node from the other node.

We will use as an example the three-level hierarchy shown in Figure 5.

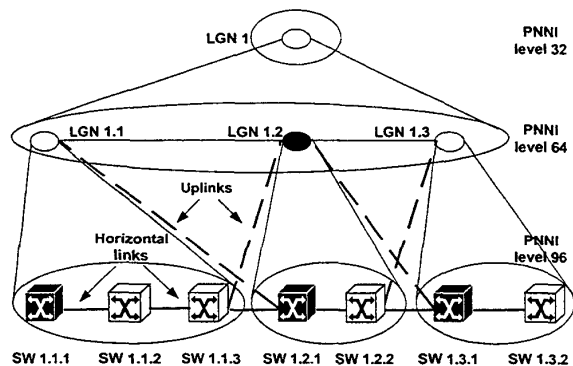


Figure 5: PNNI hierarchy.

Reducing this to the portion of the hierarchy as seen by nodes 1.1.1, 1.1.2 and 1.1.3 we arrive at the hierarchy shown in Figure 6.

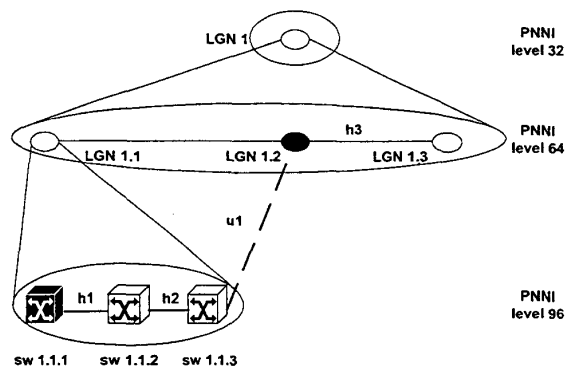


Figure 6: PNNI hierarchy viewed from node 1.1.1.

Each physical link internal to a bottom-level peer group (Figure 6, h1 and h2) counts as a single hop, as does each uplink connecting an uncommon lower-level peer group to a common peer group at a higher level of the hierarchy (Figure 6, u1), and the LGN horizontal links between common logical group nodes at higher levels of the hierarchy (Figure 6, h3).

This method of calculating topological distance is related to the shortest hop-routing algorithm used in call path selection. Here, calls are routed in such a way as to minimize the resource usage in a network.

Looking at the number of hops between the two extreme nodes sw1.1.1 and LGN 1.3 results in a count of 4. ($h1 + h2 + u1 + h3$)

Hop counts have been generated to all destinations for each of the three nodes 1.1.1, 1.1.2 and 1.1.3 and are shown in Table 1.

	1.1.1	1.1.2	1.1.3
1.1.1		1	2
1.1.2	1		1
1.1.3	2	1	
1.2	3	2	1
1.3	4	3	2

Table 1: Hop counts pre-generated for Nodes 1.1.x.

For more complex arrangements of nodes and links where there are multiple paths between any two nodes, the shortest path is required. In most PNNI implementations this information is already available in the shortest path-spanning trees used to calculate PNNI routes.

An extended model for static topology distances would include the use of the PNNI administrative weights over the shortest path.

More advanced models may adapt the tagging behavior based on the dynamic metrics of a PNNI network. An alternative to the shortest path algorithm is the widest-path algorithm. The priority in this case is to load balance calls across the entire network. In terms of topological distance, this would correspond to using an abstraction of the QoS metrics, for example available bandwidth or delay, as a tag.

Taking dynamic metrics into account would allow a router to decide with whom to peer, based on the minimum available bandwidth between its proxy-PAR server and the node that originated the PAR PTSE (e.g. in a mobile environment avoiding satellite links whenever possible).

Topological Level

This is a measure of the hierarchy layer at which PAR information was introduced into the routing domain. It is

defined in terms of the number of layers of difference between a proxy-PAR server in a bottom-level peer group and the node that introduced a PAR PTSE into the hierarchy.

This value is not to be confused with the Scope field already contained in the PAR Service Description Packet.

Taking as an example the topology shown in Figure 5, let us consider a PAR Service Description Packet that has been registered via proxy-PAR with server 1.2.2. The flooding scope of this packet is global, and as such it will be flooded throughout the entire PNNI hierarchy. After a successful registration, server 1.2.2 will encapsulate the PAR Service description packet in a PAR PTSE, with the originating node set to 1.2.2, and flood this PTSE throughout the peer group at level 96. On receipt of this PAR PTSE, the PGL of this peer group, node 1.2.1 will see from the flooding scope that the contents need to be propagated at the next higher level by the LGN 1.2. The PAR Service Description Packet will be re-encapsulated in a PAR PTSE with the originating node set to that of the LGN, 1.2 and flooded throughout the peer group at level 64. This PTSE is eventually received by LGN 1.1, which in turn propagates it down to its child peer group at level 96. A proxy-PAR server in switch 1.1.1 we see this PTSE and look at the level of the node the generated it, in this case LGN 1.2. As LGN 1.2 resides one hierarchy level higher than the bottom level node in switch 1.1.1 the resulting topological level will be set to 1. The same server will also receive PTSEs generated from switch 1.1.2 and switch 1.1.3. As these are generated at the same level as switch 1.1.1 they will be tagged with a topological level set to 0.

Physical Neighbor Tag

There are several cases where it is useful to know whether a PAR IG was originated by an immediate physical neighbor. Within a bottom-level peer group this is already represented by a hop-count value of 1. However, in the case where physical neighbors are in different peer groups at the bottom level, this is not obvious. This is due to the LGN abstraction of child nodes and the regeneration of PAR PTSEs. The mechanism that we propose in order to generate this tag is to match the 13-byte prefix of the remote ATM address returned in the hello protocol with the ATM prefix registered within a PAR Service IG. This will enable a border node acting as a proxy-PAR server to tag PAR PTSEs containing IGs registered with this same ATM address prefix.

4.1 OSPF in a campus network

In the following example (see Figure 7), we consider a simple network consisting of a single OSPF area and a single IP subnet. More sophisticated scenarios with routers belonging to different areas and subnets can be derived from this scenario. Note that the configuration of a more realistic network requires the definition and partitioning into OSPF areas, as well as subnet and address assignment.

As the goal here is not to stress the OSPF configuration task but instead to show how the adjacencies between OSPF routers can be formed to best reflect the underlying topology, we consider this example representative enough.

In the network under consideration, OSPF runs in the point-to-multipoint mode. This mode of operation is more robust than the NBMA mode, especially where adjacencies between neighbor routers tend to fail. One disadvantage is that because the topology is computed in each node instead of using a Designated Router, the load incurred on each router is higher. Unlike NBMA a full mesh of VCs between all routers is neither necessary nor recommended. Such meshes are not only resource-consuming, they also hide the real cost of an underlying network from the OSPF routers (each router thinks that he is only one hop away from every other router). A further problem is scalability; when a network expands it becomes difficult to maintain adjacencies with all other routers without a significant performance penalty in the protocol processing.

It is interesting to note that MPLS [12] adopts a similar solution in a slightly different context: each MPLS router peers only with its directly attached MPLS routers (if no PVCs are established). This way, the cost of traversing the network is shown to the IP layer: as each switch is also an MPLS router, there is a one-to-one mapping between the ATM topology and the IP overlay. In our case, as we do not assume that each switch is also a router, the problem of building an efficient IP overlay network is not trivial.

Each router initially has the same configuration, besides the IP address, which has to be different. We show the principle elements of this configuration in Table 2.

IP address	9.0.0.x (x goes from 1 to 6)
IP subnet mask	255.0.0.0
OSPF area	1.1.1.1
OSPF interface type	point-to-multipoint
PAR scope	64 (highest possible scope)
Neighbor selection policy	Full-mesh in same layer AND one hop-count away, physical neighbor AND from layer above.

Table 2: Basic configuration for all routers.

We use a simple policy to select with which other routers a given router should peer, based on the underlying PNNI hierarchy: a full-mesh of adjacencies is established between routers of the same peer-group, and a single adjacency is established between the two physically closest routers from different peer groups that are only a hop count of 1 apart.

The resulting topology is depicted with the links connecting the six routers together.

Thanks to the topological distance, the level indicators, and the physical neighbor tag, the T1 link interconnecting the two buildings together does not remain hidden in the IP overlay layer. Each rectangle in the figure represents a PAR IPv4 Service Definition IG, wherein an PAR IPv4 OSPF Service Definition IG is nested. The values contained in these IGs are shown in Table 2.

Once all routers have registered with proxy-PAR, they start querying for services, more particularly OSPF routers in the same subnet. Out of the replies received from the proxy-PAR servers, only OSPF routers in the same area and with the same interface type are considered.

Router 3, for instance, receives five advertisements from potential neighbor routers. With the given selection

process, it first picks routers 1 and 2 as they are at a level-count of 0, i.e. in the same peer group. Second, looking at routers 4, 5 and 6, which have equal hop-counts and a level-counts of 1, it uses the value of the physical neighbor tag to break the tie. The result is that Router 4 is selected.

Router 1, on the other hand, does not peer with any of the routers 4, 5 or 6. The reason is that none of the other routers tagged with level 1 have a hop-count value of 1.

This simple example shows how, with identical initial configurations (except IP addresses), OSPF routers can automatically form an optimal overlay that reflects the cost of traversing the underlying network. A successful result relies of course on the coherency of the PNNI hierarchy, from where the OSPF overlay is eventually derived.

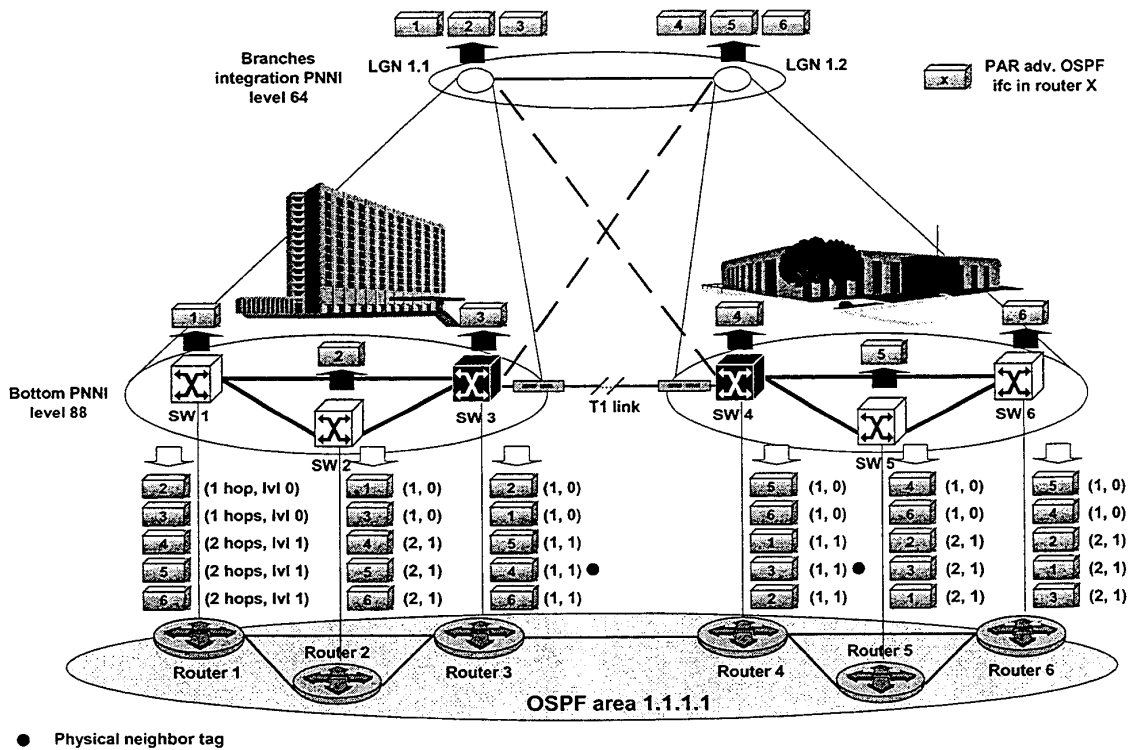


Figure 7. OSPF in a campus network.

4.2 RIP in an ad-hoc network

Ad-hoc networks [13] are formed when a group of mobile ATM networks (e.g. ships with onboard networks) establish communication between each other and dynamically create a PNNI hierarchy, similarly to [14]. Figure 8 illustrates an ATM ad-hoc network with four ships communicating via line-of-sight laser links. In this example, each of the ships has a network composed of one ATM switch and an attached router.

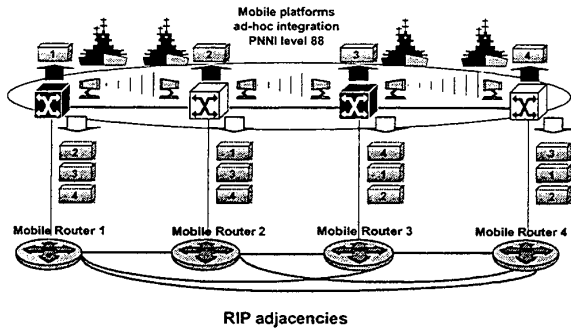


Figure 8: Use of RIP unicast in an ad-hoc network.

Ad-hoc networks are characterized by the presence of low bandwidth links (e.g. satellite links) and the absence of central servers because no device in an ad-hoc network is expected to remain indefinitely.

The common solutions for running IP overlay ATM such as Classical IP, MARS, LANE are therefore not suitable in an ad-hoc network environment. All of these solutions requires a centralized server (ARP server, MARS server, or LAN Emulation server) to operate and an ad-hoc network must be operational without reach-back to a fixed networking infrastructure. An alternative to these solutions would be to configure the adjacencies between the mobile routers, with the drawback that whenever a mobile network leaves or joins an ad-hoc network, manual reconfiguration of the routers has to be performed to maintain IP connectivity.

PAR on the other hand with its fully distributed nature is ideally suited for such an environment. Mobile routers register their protocol interfaces to their proxy-PAR server and query the protocol interfaces of other mobile routers in the ad-hoc network. This allows a mobile router to bring up adjacencies with the other mobile routers present in the ad-hoc network.

Similarly to running OSPF over an NBMA network, RIP version 2.0 [15] can be run as a unicast routing protocol. Each router maintains a list of neighboring RIP interface and when a multicast RIP advertisement must be sent, the router sends a unicast IP packet to each of its RIP neighbors. Coupling RIP unicast with PAR, a router can automatically discover its RIP neighbors in a network. Furthermore, when deployed in an ad-hoc network, a mobile router is automatically notified of a change in the

connectivity between the mobile networks. If a mobile network with a router leaves or joins an ad-hoc network, the mobile routers present in the ad-hoc network are notified of the departure or arrival, respectively, of a mobile router the next time they query their PAR server.

As shown in Figure 8, each mobile router registers one RIP interface with its serving proxy-PAR server. The information is then flooded under the form of PAR PTSEs in the PNNI hierarchy of the ad-hoc network and eventually reach the other mobile routers. This solution suffers from the need to create a full mesh of adjacencies between all the mobile routers present in the ad-hoc network. This results in the establishment of a full mesh of SVCs between the mobile routers. Whereas this is affordable between the routers inside each mobile network, ideally one would minimize the number of SVCs established through a wireless link, such as the laser link between the two mobile networks in figure 8.

The tagging of PAR data with abstracted topological information provides an elegant solution to this problem and allows mobile routers to choose with which mobile routers to establish RIP adjacencies. For example, as shown in Figure 9, a mobile router can decide to build a RIP adjacency only with another mobile router if the latter is no more than one hop away. This results in a one-to-one mapping of the IP topology with the ATM topology.

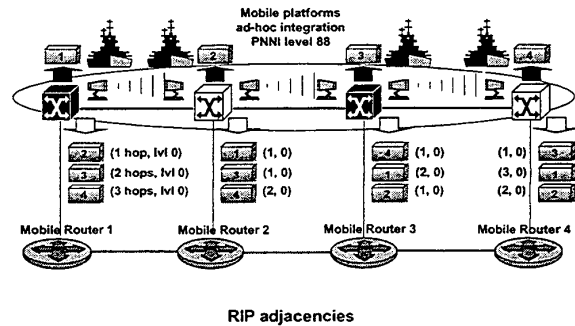


Figure 9: Use of abstracted topology to optimize the RIP adjacencies in an ad-hoc network.

5. Summary and conclusion

In this paper we focused on an extension to PAR and proxy-PAR that allows an ATM switch running PNNI to associate topological information to the PAR information delivered to a router. The proposed information describes in an abstract fashion the topology that connects the proxy PAR server delivering the information to the PNNI node that originated the PAR data. The abstracted topological information is composed of a hop count and a level count. The hop count indicates the number of hops, the sum of horizontal links and uplinks, between the PAR server and the PNNI node that originated the PAR PTSE. The level

count indicates at which level of the hierarchy the PAR information was generated.

With the help of two metrics, a router now gains knowledge about the location of the originator of a piece of PAR information, and this without having to implement the PNNI protocol to gather the PNNI topology of the current hierarchy.

A router can use the abstracted topological information to decide with which routers in the ATM network to peer, allowing the IP topology to be optimized over an ATM network. The hop-count metric can be used, for instance, to map the overlay IP topology to the physical ATM topology, avoiding a full mesh of SVCs between OSPF routers. As discussed in this paper, such optimization is critical in a mobile environment, where low bandwidth links are involved. The level count on the other hand reflects the clustering of the ATM switches in peer groups and can be used to adopt different peering behavior between the routers as a function of the PNNI hierarchy of the underlying ATM network.

Whilst the proposed hop count and the level count metric can be considered in a wired network as "static" values, more dynamic metrics for the abstracted topology could be of interest. The use of the minimum available bandwidth between its proxy-PAR server and the node that originated the PAR PTSE could be of interest and is open for further research. It would allow the establishment of adjacencies between routers as a function of the load of the networks, and could be used as a self-controlled mechanism to avoid overloading states.

6. References

- [1] ATM Forum, "Private Network to Network Interface Specification Version 1.0", doc. af-pnni-0055.000, March 1996.
- [2] ATM Forum, "PNNI Augmented Routing (PAR) Version 1.0", doc. af-ra-0104.000, January 1999.
- [3] ATM Forum, "LAN Emulation over ATM 1.0", doc. af-lane-0021.000, January 1995.
- [4] M. Laubach and J. Halpern, "Classical IP and ARP over ATM", Internet RFC 2225, April 1998.
- [5] Y. Rekther and T. Li, "A Border Gateway Protocol 4 (BGP-4)", Internet RFC1771, March 1995.
- [6] J. Moy, "OSPF Version 2", Internet RFC 2328, April 1998.
- [7] G. Armitage, "Support for Multicast over UNI 3.0/3.1 based ATM Networks", Internet RFC 2022, November 1996.
- [8] P. Droz and T. Przygienda, "Proxy-PAR", Internet RFC 2843, May 2000.
- [9] M. Davison, "ILMI-Based Server Discovery for ATMARP", Internet RFC 2601, June 1999.
- [10] M. Davison, "ILMI-Based Server Discovery for MARS", Internet RFC 2602, June 1999.
- [11] T. Przygienda, P. Droz and R. Haas, "OSPF over ATM and Proxy-PAR", Internet RFC 2844, May 2000.
- [12] E. Rosen, A. Viswanathan and R. Callon, "Multiprotocol Label Switching", Internet Draft, work in progress, draft-ietf-mpls-arch-06.txt, August 1999.
- [13] S. Corson and J. Macker, "Mobile Ad Hoc Networks (MANET): Routing Protocol Performance Issues and Evaluation Consideration", Internet RFC 2501, January 1999.
- [14] L. Freléchoux, D. Dykeman, I. Iliadis, P. Scotton, "Resource Location in Mobile ATM Networks", ICATM'98, Colmar, France, June 1998.
- [15] G. Malkin, "RIP Version 2", Internet RFC 2453, November 1998.