A semantic model for complex computer networks : the network description language

van der Ham, J.J.

Publication date
2010

Chapter 5

Introduction to Network Topology Aggregation

5.1 Introduction

In the previous chapters we have introduced the Network Description Language and shown its applications. However, applying NDL to real-life inter-domain networks is not straightforward. Network operators do not always wish to share their full topology, either for security, business, or for scalability reasons. On the other hand it is necessary to share some degree of topology data to enable inter-domain lightpath planning. It is possible to create a less detailed view of a network topology using *aggregation*. This means that details of the internal topology are aggregated into a virtual topology, which can then be published to other domains.

Topology aggregation is not a new research topic. It has been applied in ATM (Asynchronous Transfer Mode) networks, where it is also necessary to exchange topologies. Asynchronous Transfer Mode (ATM) is a network framing protocol developed in the mid 1980s. It encapsulates traffic into small fixed-length cells, over which virtual circuits can be defined, creating a circuit switched network. The topologies for ATM networks have been represented as graphs, with attributes for QoS such as capacity, available bandwidth, link delay, et cetera.

The rest of this chapter is organised as follows, in section 5.1.1 we discuss how pathfinding is performed in aggregated topologies. Then in section 5.2 we describe different aggregation strategies. In section 5.3 we summarize three performance studies of common aggregation methods, the first two study ATM, while the latter focuses on purely optical networks. We finish the chapter with a summary and comparison of the different studies in section 5.4.

5.1.1 Hierarchical Routing

One of the first standards on the area of topology aggregation is in the Private Network-to-Network Interface (PNNI) Specification[83] of the ATM forum. As the name of the standard implies, it specifies the interface between neighbouring networks. This standard provides some pointers on how aggregation can be performed, and it also defines how nodes can find routes through the aggregated network.

Routing through aggregated topologies is performed using hierarchical routing. Each node in the network has perfect knowledge of its local domain, and an aggregated view of the global topology. Using this knowledge, the node selects a path with aggregated hops. These hops get expanded at the border nodes into an underlying physical path. Together these expanded paths form a complete path between the node and its chosen end-point.

An example is shown in figure 5.1, we see a path request from node A1 to node C2 at the top, as seen from node A1. Notice that node A1 has complete visibility of its own domain, but only aggregated visibility of the other domains. Node A1 only knows that node C2 is in domain C, it does not know how it is connected to the rest of Cs network.

As soon as the request reaches domain B, the first node in B, B1 maps the path through the aggregated topology to the physical topology in B. B1 then forwards the request to B3, which forwards the request to domain C. In domain C the node C3 maps the path to the physical topology, and the path to C2 is completed. At the bottom of figure 5.1 we show the path mapped to the physical topology.

Note that in the aggregated view nothing is said about the availability inside the domains. If the links in domain B had no available capacity, then the path request would have failed.

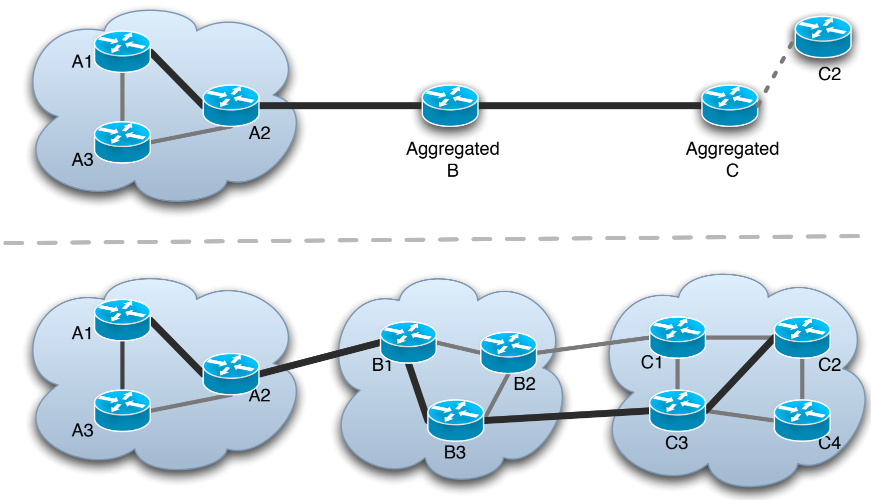

Figure 5.1: *An example of hierarchical routing through the aggregated topology (top) and the physical topology (bottom)*

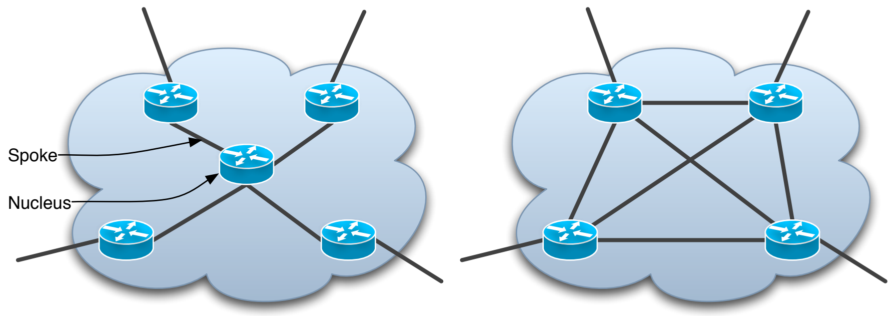

Figure 5.2: *Examples of Symmetric Star (left) and Full Mesh (right) aggregated topologies*

5.2 Topology Aggregation

There are different ways of creating aggregated topologies. In the example in figure 5.1 the domains are aggregated into single nodes and single links. Lee[84] provides an overview of this and two other aggregation methods. Below we describe the three most common aggregation strategies.

The *Simple Node* approach is the aggregation method that we have shown in figure 5.1. This is also called *Symmetric* or *Single Node* approach. The topology of a complete domain is replaced by a single node, which is directly connected to other domains. A single metric is advertised for the connectivity through this node. This single parameter implies that all connectivity through the node is considered to be symmetrical.

In the *Symmetric Star* approach the topology of a domain is aggregated using a central node. The border nodes of the original topology and their inter-domain connections are preserved. The intra-domain connections are represented by virtual links, *spokes*, to a virtual central node, the *nucleus*. All connectivity in this topology runs through the nucleus. This is often referred to as *Star* aggregation. An example is shown in figure 5.2 on the left, where we see an aggregated topology of a domain that is connected to four other domains. This is called a symmetric approach because a default definition for the properties of the spokes is used. All the spokes then have the same properties, unless explicitly specified otherwise.

On the right side of figure 5.2 is an example of the *Full Mesh* aggregation.

This aggregation preserves the most information of the original topology. Like in the Star aggregation, the border nodes are kept in the aggregated topology. Instead of a central node in the aggregated topology, there is a full mesh of connections between the border nodes. These aggregated connections between the border nodes can accurately describe the properties of the path through the domain, thus preserving the most important information for connections crossing the domain.

5.3 Performance Evaluation of Topology Aggregation

An aggregated topology hides details about the intra-domain connectivity. This means that inter-domain pathfinding with aggregated topologies is not always optimal. There are two types of performance impacts that can occur due to this lack of information.

First, in the aggregated topology there is no discernible difference between a domain that has many internal hops and a domain that has only one hop. This means that the paths found in the aggregated topology may not always be the shortest paths in the physical topology.

Second, a path found in the aggregated topology may not always have resources available to map it to the physical topology. This means that the path found in the aggregated topology is a false positive.

The impact on performance of the above effects can be measured in different ways. The rest of this section describes three different studies. Section 5.3.1 describes the study performed by Guo and Matta on a single emulated ATM network[85]. Section 5.3.2 presents the findings of Awerbuch et al. on several emulated topologies with ATM networking[86]. The study on aggregated topologies in an optical network by Liu et al.[87] is described in section 5.3.3. Finally in section 5.4 we compare the results of the three studies.

5.3.1 Performance Evaluation Study by Guo and Matta

Guo and Matta[85] have examined the performance of aggregation methods using the PNNI specification in ATM networks. They have examined the Simple Node, Star, and Full Mesh aggregation methods and analysed the performance on a single topology, randomly divided into domains. In their simulations they use two different traffic workloads:

Uniform : source and destination pairs are uniformly distributed over the network,

Skewed : some nodes are selected as destination for the majority of the connections.

In either case, traffic between two end-points is defined using different 'services'. A service is defined using time-dependent arrival rate, average life-time of the connection, maximum delay, and transmission rates (both average, and peak, including the length of the busy period). An example of a service is a video service, with a life-time of 20 minutes, an average rate of 0.7Mbps, a peak rate of 2.1Mbps, with a busy period of 0.3 seconds, and a statistical delay requirement $P(\text{end-to-end packet delay} > 50\text{msec}) < 10^{-4}$.

They also examine the performance of different route selection methodologies in the Full Mesh aggregation, based on different sets of metrics: utilization, utilization and hop-count, utilization and feasibility, or all three. Since pathfinding based on multiple metrics is NP-complete[88], they apply a heuristic. The authors first find a set of shortest paths, and then pick one of the set using the metric selection. This set is formed by the shortest paths in the aggregated graph, with length *minhop*, and paths of length $minhop + 1$. The decision of picking the right path is performed by the source node for the whole path as well as each of the border nodes for their segment of the path.

Their results show that under the skewed load, the Simple Node performs better or as well as Full Mesh and Star aggregations. Under uniform workload, both Full Mesh and Star outperform Simple Node significantly. In their simulations, the Star approach performs slightly worse than the Full Mesh as the former provides a less detailed view of the available bandwidth.

5.3.2 Performance Evaluation Study by Awerbuch et al.

Another performance evaluation of routing with aggregated topologies is given by Awerbuch et al. in [86]. The authors compare the following aggregation schemes:

Full Mesh The full cost matrix between the border nodes is advertised,

Diameter Star aggregation with a metric of half the diameter of the original network for each spoke,

Average Star aggregation with a metric of half the average cost between all pairs of border nodes,

MST Minimum Spanning Tree of the border nodes,

RST Random Spanning Tree of the border nodes,

Spanner A t-spanner, which is a sub-graph that guarantees a worst-pair distortion of at most a factor t. The authors have used $t = 2$ for their experiments.

These aggregation schemes are examined in combination with two methods for metric values, a constant and an exponential cost function. In the constant case, the link metric is fixed, regardless of the available bandwidth on the link, this is equivalent to (weighted) min-hop routing. The exponential cost function on the other hand uses a metric that increases exponentially as the available bandwidth decreases.

In their simulations they examine both specifically designed topologies, as well as randomly generated networks. The former topologies have been designed to maximize the penalty for routing errors, so that difference in routing schemes is emphasized. They have used ring-like topologies, as well as a self-similar hierarchical topology, which is a three-level hierarchical topology, where each level is a similar multi-stage graph.

The random networks were generated using an adapted Waxman method[89]. In this method nodes are randomly placed on a grid, and the probability of a link between two nodes decays exponentially with their Euclidean distance. They adapted the method to prevent nodes exceeding a degree of four, but the degree must always be at least two.

In the simulation the requests are modelled by a Poisson process, and the holding time is exponentially distributed. The inter-arrival time of the requests are exponentially distributed with a mean of 1 when the traffic load is 100%. The load on the network is defined as (avg. request rate $\times$ avg. hold time)/ capacity of the avg. min. cut (over all source-destination pairs). The results are analysed using two parameters:

Throughput defined as the fraction of attempted connections that are realized,

Control Load the average number of crank-backs required per realized connection.

A *crank-back* is performed when a path that has been found in the aggregated topology is not available in the physical topology. The information regarding this false-positive is then used to locally update the view of the network, and is then used in subsequent attempts to find a correct path. The amount of crank-backs also gives an indication for the set-up delay, because route recalculation when crank-backs occur is a time consuming task.

The results of the simulations confirm the authors' earlier theoretical work[90] that an exponential metric performs much better than a constant metric. In fact, even the worst aggregation strategy in the exponential metric simulation performs better than the best aggregation method in the constant metric simulation.

In their simulations the authors also varied the link-delay, so that topology updates between domains, and reservation requests take longer to travel through the network. The effect of the higher delay is that the performance difference between all the aggregation methods becomes more pronounced.

On ring-like topologies the Random Spanning Tree and the two Star aggregation methods perform worse than the other methods. The Random Spanning Tree method shows significantly the worst performance, with crank-back rates of up to four times higher than the other aggregation methods. The performance difference of the two Star aggregation methods is less pronounced, but still significant. The difference becomes more pronounced in a self-similar hierarchical topology, were the Star methods perform significantly worse than the other methods, including the RST.

Also in the random topologies the two Star aggregation methods and the RST perform the worst. The Star aggregation methods require more crank-backs, with a slightly worse throughput, and the RST method requires significantly more crank-backs while providing a lower throughput.

In the above simulations each aggregated domain topology is updated once the bandwidth availability of any of its constituent links changes. The authors propose a more practical re-aggregation policy called the logarithmic update re-aggregation policy. In this case the bandwidth b of a link is divided into $(log\ b)+1$ blocks. Re-aggregation of the domain topology is only triggered when the residual bandwidth crosses these division boundaries. With this update scheme in the simulations, there is no significant difference in throughput, and only a slight increase in the number of crank-backs for all aggregation schemes. The only exception is MST, where there is a 5-20% increase in the number of crank-backs with the new re-aggregation policy. This re-aggregation policy requires significantly less amount of updates, and even under high-load settings it requires over 30% less updates than the regular policy.

5.3.3 Aggregated Topologies in Optical Networks

While aggregated topologies have been studied extensively for ATM networks, they have not been studied extensively for pure optical networks. Liu et al.[87] have applied the Simple Node and Full Mesh aggregations to inter-domain WDM networks. In both cases a wavelength availability vector is used to represent the available wavelengths.

The aggregation strategies are tested using two different lightpath provisioning strategies. They use the term *transparent* for lightpaths that have the same end-to-end wavelength, and *translucent* for lightpaths that use different wavelengths using optical-electrical-optical (OEO) conversions.

In the transparent case paths are selected using a K-shortest path algorithm and a widest-shortest approach[91]. A wavelength for this path is then selected using either the most-used or least-used wavelength. The most-used or least-used wavelength selection method mean that the available wavelength on that path are ordered by their use in the rest of the network[92].

In the translucent case paths are also first selected using K-shortest path, but then the candidate path is selected using either minimum hop count, or minimum converter count (based on the aggregated view only). The same wavelength selection strategies are used, but this time only for segments between converters. This makes the translucent routing inherently more complex.

The authors show results of their simulation in OPNET[93], a network simulator tool, on a topology with 9 domains and 19 inter-domain links, using 8 or 16 wavelengths. The results show that selecting the most-used wavelength outperforms selecting the least-used wavelength. For transparent lightpaths Full Mesh performs better than Single Node aggregation, with a difference of up to 45% at 120 Erlang[1] with 16 wavelengths. The authors note that the difference in blocking reduction becomes smaller when the intra-domain connectivity decreases.

In the translucent case, it makes little difference in terms of blocking probability whether paths are selected based on the least number of hops or least number of converters. There is still a difference between the two aggregation schemes, but the difference is smaller than in the transparent case.

The same authors show more results on the same topology in [94]. They show that using conversion capabilities on all nodes provides little improvement over just converting at the border nodes. However, note that in the test topology they are using, there are relatively few non-border nodes.

[1] Erlang is a unit to describe statistical measure of the traffic volume. Traffic of one Erlang refers to a single resource being in continuous use, or two channels being at fifty percent use each, and so on.

5.4 Summary

In this chapter we have discussed different methods for topology aggregation, and described three studies evaluating the performance of pathfinding in aggregated topologies. Unfortunately, all three studies have used a different measure of performance.

The results that Guo and Matta show in their paper use revenue as an indication of the pathfinding performance. They define revenue as the total amount of bandwidth in use by connections at an instant t.

In the study by Awerbuch et al. the performance has been measured by throughput and control load. The throughput is defined by the fraction of attempted connections that were realized. The control load is represented by the average number of crank-backs per realized connection.

Finally in the study by Liu et al. the performance is determined by the blocking probability compared to the load on the network.

The different performance measurements make it hard to directly compare the results of the three studies. However, it is possible to detect a general trend in the results. As can be expected, the Full Mesh aggregation performs best in all studies, and the Single-Node aggregation performs worst in most cases compared to the other strategies.

The Star aggregation strategy has only been examined in the first two studies. Unfortunately these studies provide conflicting results. In the study by Guo and Matta the Star aggregation performs as well as or better than the Full Mesh, while in the study by Awerbuch et al. several Star aggregation strategies are examined, and they all perform significantly worse than Full Mesh.

The first two studies above have also been performed with ATM networks in mind, and this does not readily map to inter-domain optical and hybrid networks. While both ATM and optical networks are circuit based, the availability of circuits is different. A request in ATM takes up a portion of an edge, while in optical networks a request often takes up the whole edge. It is therefore not completely clear what impact topology aggregation has on inter-domain pathfinding in optical and hybrid networks. In the next chapter I describe an emulation experiment and its results to accurately determine this impact.