Topology Aggregation for Directed Graph

Baruch Awerbuch
baruch@cs.jhu.edu
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218

*Yuval Shavitt**
shavitt@lucent.com
Bell Laboratories, Lucent Technologies
101 Crawfords Corner Rd., room 4G-627
Holmdel, NJ 07733-3030

Abstract

This paper addresses the problem of aggregating the topology of a sub-network in a compact way with minimum distortion. The problem arises from networks that have a hierarchical structure, where each sub-network must advertise the cost of routing between each pair of its border nodes. The straight-forward solution of advertising the exact cost for each pair has a quadratic cost which is not practical. We look at the realistic scenario of networks where all links are bidirectional, but their cost (or distance) in the opposite directions might differ significantly.

The paper present a solution with distortion that is bounded by the logarithm of the number of border nodes and the square-root of the asymmetry in the cost of a link. This is the first time that a theoretical bound is given to an undirected graph. We show how to apply our solution to PNNI.

Keywords: digraphs, spanners, PNNI, topology aggregation, hierarchical routing.

*Corresponding author. Tel: (732) 949 2303, Fax: (732) 949 0399

1 Introduction

As networks grow in size, the collection and maintenance of control information from the entire network, e.g., to make routing decisions, becomes difficult if not impossible. For large networks, the current solution is to partition the network into subdomains, and continue partitioning the subdomains recursively until the lower level were the subdomain is comprised of subnetworks [tc96, CCS96]. At each level, control information is aggregated before shared with peer subdomains. Specifically, for routing purposes, the internal network structure is hidden, and instead, the network presents to the outside world compressed information that enables routing entities to make intelligent decisions as to which subnetwork to use and to select the appropriate entry points to each subnetwork. The mathematical property of routing in such a scenario were recently studied by Guérin and Orda [GO97].

Methods for compact graph representation that can be used to aggregate topology information were studied in the past years. A spanner [PU88, PS89, ADD$^+$93] of a graph G is a subgraph G' that includes all the node of G but only a subset of the edges. Spanners with stretch t, t-spanners, have the property that the cost of the minimum cost path in G' between any two nodes is, at most, a factor t greater than the cost of the minimum cost path between the same nodes in G. We term this factor the max distortion of the graph compression, or simply the *distortion*. The best known algorithm [ADD$^+$93] for spanners with stretch t requires $O(b^{1+1/t})$ edges, where b is the number of border nodes, i.e., if a logarithmic distortion is acceptable the spanner size is a constant factor of the number of nodes. Bartal [Bar96] suggests a tree representation for graphs with an average distortion of $O(\log n)$, where n is the number of nodes in the graph. The tree is not built from the edges of the original graph.

All the above aggregation schemes work on undirected graphs. However, in todays application and standards [tc96], reservations for each direction of the connection take different values. As a result, the network graph becomes directional, with edges that have different weights for the two directions. For routing, the edge weight is usually a function of the residual bandwidth or the average delay on the link, and these, in case of asymmetric traffic, might have orders of magnitude differences.

For general digraphs, Peleg and Schäffer [PS89] show that for any t, a t-spanner might require $\Omega(n^2)$ links. In other words, for certain directed graphs, no compact representation with bounded distortion exists. However, in practice, the ratio between the weights of a link is bounded by some asymmetry constant, ρ, that depends on the link weight function. For example, a link delay function may be bounded from above due to finite buffer space, and from below due to processing delay.

In this paper, we show a compact representation for a directed graph that are known to have some bound on their asymmetry constant. Our compression distortion depends on the actual asymmetry constant of the graph and not on the bound for this constant. In general, we show that a subgraph of a network with b border nodes can be represented by $O(b)$ weights with $O(\sqrt{\rho}\log b)$ distortion. More precisely, the distorted broadcasted weight of an edge, whose weight is w, can be in the range $[w/\sqrt{\rho},\ w \cdot \sqrt{\rho} \cdot \log b]$. Note, that in practical cases, ρ is expected to be larger than b by several orders of magnitude.

The fact that link costs can vary over orders of magnitude might look surprising. However, both queueing theory and opportunity cost functions suggest such a wide range. If link cost is to be determined by the expected queueing delay of a packet through this link, which is the major factor in the total delay, the delay function is given by $[\mu(c-f)]^{-1}$, where c is the link capacity, f is the current flow through the link, and μ is the service rate. As f approaches c, the link cost increases to infinity. Opportunity cost functions suggest to maximize utilization by charging link usage according to a function that increases exponentially as the residual bandwidth decreases.

The rest of the paper is organized as follows. In the next section, we describe the network model, and give our notations. In section 3 we describe the aggregation algorithm and analyze its performance, and in section 4 we simulate its performance. Finally we conclude by showing the applicability of our results to the PNNI standard.

2 Model and Notation

We use the PNNI [tc96] notation. A subnetwork comprised of V, a group of connected nodes, that is called a peer group. $B \subset V$ is a set of border nodes, i.e., nodes that have a direct link to a node outside of the peer group. Each directed edge $(v,\ u) \in E$ is associated with a weight, $w(v,\ u) \in \mathbb{R}$, $\rho^{-1} \leq w(v,\ u)/w(u,\ v) \leq \rho$. w.l.o.g. let $w(v,\ u) > w(u,\ v)$, $\rho_{u,v} \equiv \rho_{v,u} \stackrel{\triangle}{=} w(v,\ u)/w(u,\ v) \leq \rho$ and let $\rho = \max_{(u,v):u,v \in V} \rho_{u,v}$ ρ is the network asymmetry constant, $\rho_{u,v}$ is the asymmetry factor for a node pair.

Since the internal structure of the graph is irrelevant to route through it, as only the the cost matrix is of relevance, the original graph, $G(V,\ E)$, is transformed to a directed clique, $\vec{K}(B)$, where B is the the group of border nodes. Each $(v,\ u)$, a directed link in $\vec{K}(B)$, is associated with a weight, $w(v,\ u) \in \mathbb{R}$, that is equal to the weight of the shortest path between v and u in $G(V,\ E)$. By $w(v,\ u)$ definition for a path, the inequality $\rho^{-1} \leq w(v,\ u)/w(u,\ v) \leq \rho$ holds for paths, as well. A full representation of the weight distances (or costs) between the border nodes requires $b(b-1) \in O(b^2)$ space, where b is the number of border nodes.

From the construction of $\vec{K}(B)$ we get the following property. For every $u,\ v,\ x \in B$:

$$w(u,\ v) \leq w(u,\ x) + w(x,\ v)$$

This directed triangle inequality suggests that although link weights can be arbitrary different in the original graph, in the resulting clique link weights are bounded by the weights of their adjacent links.

We transform the directed clique, $\vec{K}(B)$, to an undirected clique, $K(B)$, and later delete links from $K(B)$ to receive the graph $K^-(B)$. For each link $(v,\ u)$ in $K(B)$, $w_K(u,\ v)$ is the link cost. For each pair of nodes, u and v, in B, $w_{K^-}(u,\ v)$ is the cost of the shortest path between u and v in $K^-(B)$.

3 Algorithm Description and Analysis

Our suggestion for treating asymmetry is simple, yet, powerful. In the first step, the original digraph, $G(V,\ E)$, is transformed to a directed clique, $\vec{K}(B)$. A directed link between node b_i and node b_j in the clique has the weight of the shortest path between these two border nodes in G. Note that this transformation preserves all the routing information, and if the $b(b-1)$ distances between all the node pairs of the clique are broadcasted full routing information is available.

We transform the directed clique, $\vec{K}(B)$, to an undirected clique, $K(B)$, as follows. Every pair of directed links $(v,\ u)$ and $(u,\ v)$ with weights $w(v,\ u)$ and $w(u,\ v)$, respectively, is replaced with an undirected link $(v,\ u)$ with weight $w_K(v,\ u) = \sqrt{w(v,\ u) \cdot w(u,\ v)}$. At this stage we can use spanners with logarithmic distortion [ADD+93] and the result requires only $O(b)$ edges. Specifically, a $(2\log_2 b + 1)$-spanner with less than $2b$ edges can be found by a simple polynomial algorithm. We note that the distortion for the border-node pair u and v depends only on the number of border nodes in the graph and the graph asymmetry constant, ρ. The overall weight distortion for an edge $(u,\ v)$ can be in the range $[w(u\ ,v)/\sqrt{\rho},\ w(u\ ,v) \cdot \sqrt{\rho_{u,v}} \cdot \log b]$. The distortion calculation will be explained later after the discussion of the distortion of the tree construction. Remember, that in practical cases, ρ is expected to be larger than b by several orders of magnitude.

Using Bartal's tree construction [Bar96] might be more appealing due to the fact that the resulted aggregated structure has a known structure (a tree), while the distortion remains logarithmic in b. However, Bartal's algorithm cannot be applied directly to the undirected clique since it requires

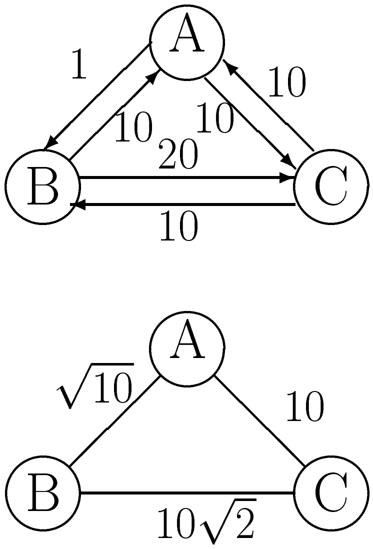

Figure 1: An example where the transformation from a directed to an undirect clique results in a triangle that does not obey the triangle inequality

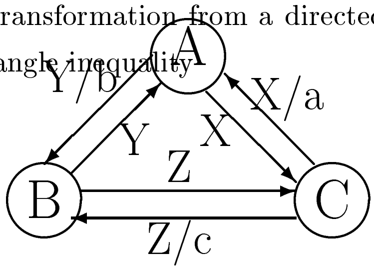

Figure 2: A general triangle

the graph to represent a metric space while the resulted clique may contain triangles that do not obey the triangle inequality as can be shown by the simple three-node example-network in figure 1. In this example, a three-node directed clique with $\rho = 10$ (Fig. 1 above) is transformed to an undirected clique (Fig. 1 below). The resulted triangle does not obey the triangle inequality as $\sqrt{10} + 10 = 13.16 < 10\sqrt{2} = 14.14$

However, the clique can be transformed to represent a metric space in the following way. list all the triangles that do not obey the triangle inequality and delete the longest edge in all of them. The resulted graph, $K^{-}(B)$, represents a metric space.

In the following, we investigate the influence the edge deletion has on the distortion. We show first that the graph remains connected, and then we give a bound on the additional distortion due to edge deletion.

Theorem 1 *The link deletion from the undirected clique results in a connected graph.*

Proof:

Assume that the graph $K^{-}(B)$ is not connected. Thus, two border nodes, u and v, exist in $K^{-}(B)$ such that no path connects them in $K^{-}(B)$. Consider a link $(w,\ x)$ on the shortest path between u and v in $K(B)$. To be deleted, $(w,\ x)$ should be longer than the sum of the other two links in a triangle, which contradicts the fact that $(w,\ x)$ belongs to the shortest path. $\square$

Lemma 1 *For a link with asymmetry ratio ρ_e, the maximum distortion due to link deletion from the undirected clique is $\sqrt{\rho/\rho_e}$.*

Proof:

Consider the general triangle in figure 2. Let

$$\begin{aligned} X' &= X/a && (1)\\ Y' &= Y/b && (2)\\ Z' &= Z/c && (3)\\ Z &\leq X+Y \quad \text{(weak triangle inequality)} && (4)\\ Z' &\leq X'+Y' \quad \text{(weak triangle inequality)} && (5) \end{aligned}$$

To maximize the distortion let $Z = X + Y$ and thus

$$\begin{aligned} X &= pZ && (6)\\ Y &= (1-p)Z && (7) \end{aligned}$$

W.l.o.g. assume $X \geq Y$, i.e., $0.5 \leq p < 1$.

The distortion is given by

$$dist = \frac{\sqrt{ZZ'}}{\sqrt{XX'}+\sqrt{YY'}} = \frac{\sqrt{ab/c}}{\sqrt{a}-p(\sqrt{a}-\sqrt{b})} \qquad (8)$$

where the second equation in (8) is due to substituting (1) – (7) and performing some simple algebraic manipulations.

If $a > b$ the maximum is achieved when p approaches 1, and the maximum distortion is then

$$dist^* = \sqrt{\frac{a}{c}} = \sqrt{\frac{\max\{a,\ b\}}{c}} \qquad (9)$$

If $a < b$ the maximum is achieved when $p = 0.5$ and its value is $\sqrt{\frac{\min\{a,\ b\}}{c}} < dist^*$.

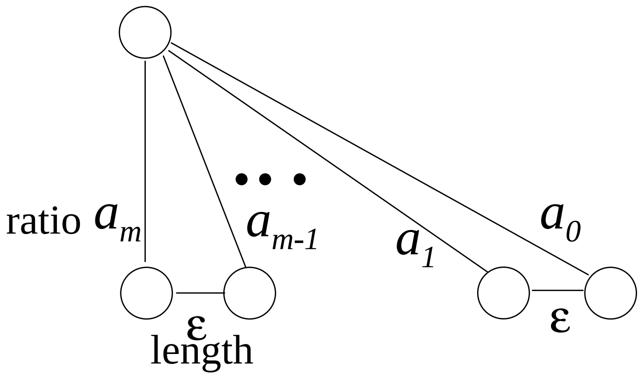

Figure 3: Maximal cascading effect

$\sqrt{\frac{a}{c}}$ is the maximum distortion for one triangle. When several triangles are cascaded, let the asymmetry ratio of the longest link be a_0, let the asymmetry ratio of the link sharing a triangle with the longest link be a_1, and define a_i, $i = 2,\ 3,\ ,\ldots, m$ in the same manner (see figure 3). The total distortion due to the deletion of the m links that violates the triangle inequality is

$$\sqrt{\frac{a_1}{a_0}} \cdot \sqrt{\frac{a_2}{a_1}} \cdots \sqrt{\frac{a_m}{a_{m-1}}} = \sqrt{\frac{a_m}{a_0}} \leq \frac{\sqrt{\rho}}{\sqrt{a_0}} \tag{10}$$

□

Theorem 2 *The maximum distortion of applying the square-root transformation on a directed clique is* $\sqrt{\rho}$.

Proof:
For $(u,\ v)$, the larger of two opposite links, we reduce the $(u,\ v)$-path cost by a factor of $\sqrt{\rho_{u,v}}$ when we multiply and take the square-root, and by lemma 1 we, at most, reduce the path cost by an additional factor of $\frac{\sqrt{\rho}}{\sqrt{\rho_{u,v}}}$ when we delete links. As a result the distortion for the long link is at most

$$\frac{\sqrt{\rho}}{\sqrt{a_{u,v}}} \cdot \sqrt{a_{u,v}} = \sqrt{\rho}. \tag{11}$$

For $(v,\ u)$, the short link, the square-root transformation increases the path cost by a factor of $\sqrt{\rho_{u,v}}$. The link deletion process can only decrease the path cost to become closer to (but always higher than) $w(v,\ u)$. □

Bartal's tree increases the distance between two nodes by $\log b$ on the average. This means that the advertised network crossing cost between two border nodes can be at least a factor of $\sqrt{\rho}$ below the actual distance, due to the averaging of the two opposite links and the distortion of the link deletion, or at most a factor of $\sqrt{\rho_e}\log b$ above the actual cost due to the averaging of the two opposite links and the tree construction.

The $(\log b)$-spanner algorithm adds a link to the spanner only if the distance between the two link end points improves by $\log b$ as a result of the addition. Obviously, only links that belong to some shortest path are used, and specifically, links that are deleted to apply Bartal's tree construction do not effect the final result. The maximum $\log b$ distortion is measured on the shortest path between two nodes in $K^-(B)$. This was shown in theorem 2 to be at most $\sqrt{\rho}$. Thus, the distortion for the spanner is the same as the one for tree, i.e., in the range $(w(u,\ v)/\sqrt{\rho},\ w(u,\ v)\cdot\sqrt{\rho_{u,v}}\cdot\log b)$.

4 Simulation Results

To check the affect of the link deletion procedure, we produce random graphs according to Waxman's method [Wax88]. In the graph creation process, links are added until all nodes reach a minimum node degree of two. Links that increase node degree over 6 are rejected. This way the resulted graphs have both the clustering effect of close nodes and a relative high diameter. In a second phase, each undirected link is replaced by two directed link and the link weights are randomly assigned in the range $(1..10^{-6})$ uniformly on the log scale.

In tables 1 and 2 we present results for graphs with 100 nodes, of which 22 and 33 are border nodes, respectively. Figures 4 and 5 depict these graphs. Link weights were checked to cover the entire range and the average was around 11000, with variance around 3.0E+9. Link asymmetry ratios were ranging from just above 1 to over 500,000.

The percentage of deleted links varied from 38% to 70%, up to 31% of the triangles were bad, i.e., did not obey the triangle inequality. The additional distortion due to the link deletion process is almost always below $\log b < 5$ (only 2 node pairs in the 8 experiments were distorted by more than 5), a negligible number when compared to $\sqrt{\rho} = 1000$.

In table 3 we present results for graphs with 100 nodes, 7 of which are border nodes. Between 2 and 10 of the 21 clique links where deleted, and up to 22 out of the 35 triangles where bad. Although the variance in several categories was very large, the additional distortion due to the link deletion process is below 4 as in table 1.

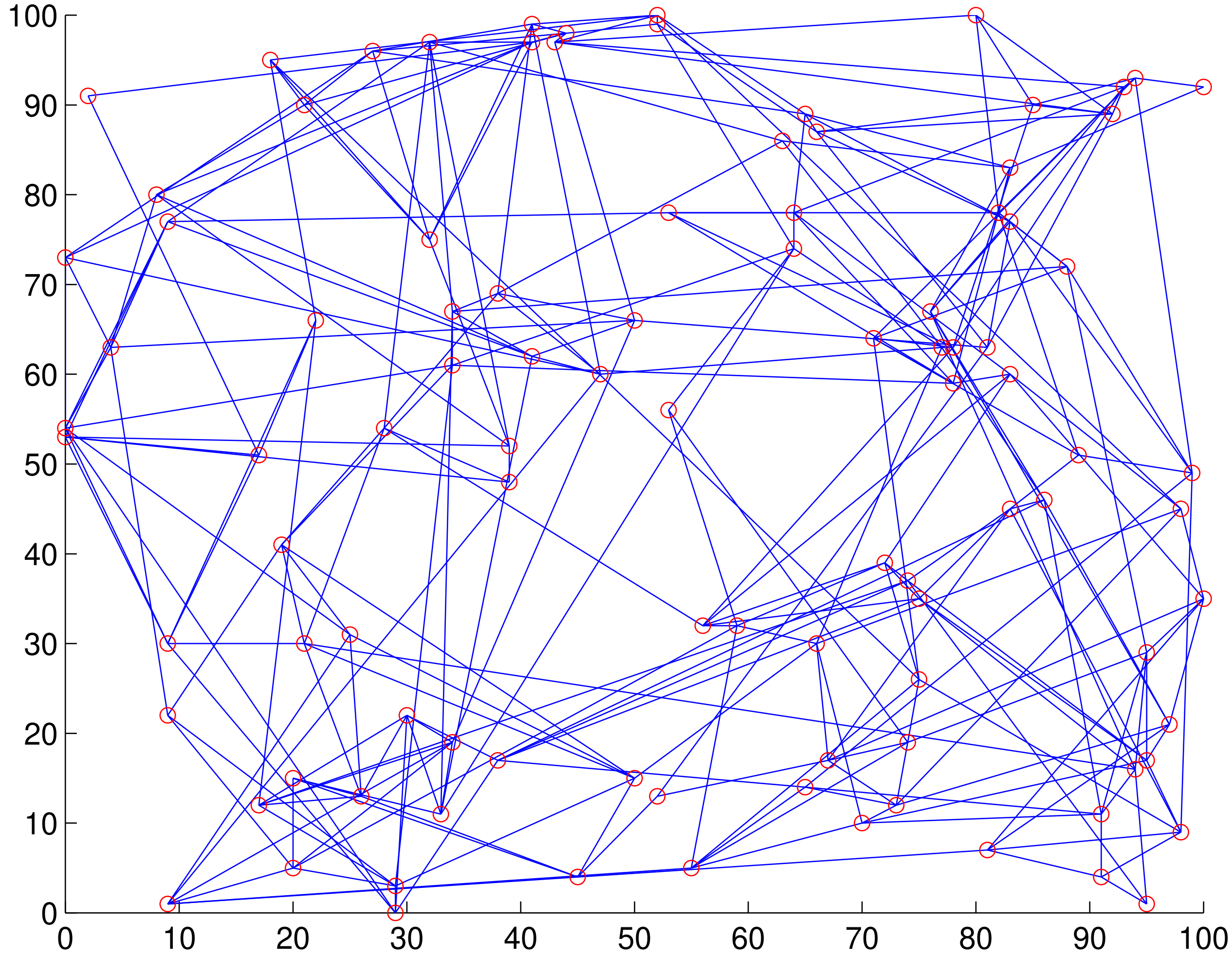

Figure 4: A random graph with 100 nodes, 22 of which are border nodes.

border nodes asymmetry ratio				No. of	links	max.
max	min	ave	var	bad △s	deleted	distortion
133.2	1.019	6.4	242	303	107	3.15
54.0	1.006	6.2	83	471	129	3.05
144.0	1.002	3.4	107	175	88	3.74
1284.1	1.002	21.5	8144	477	115	3.11

Table 1: Results for a graph with 100 nodes, 22 of them are border nodes

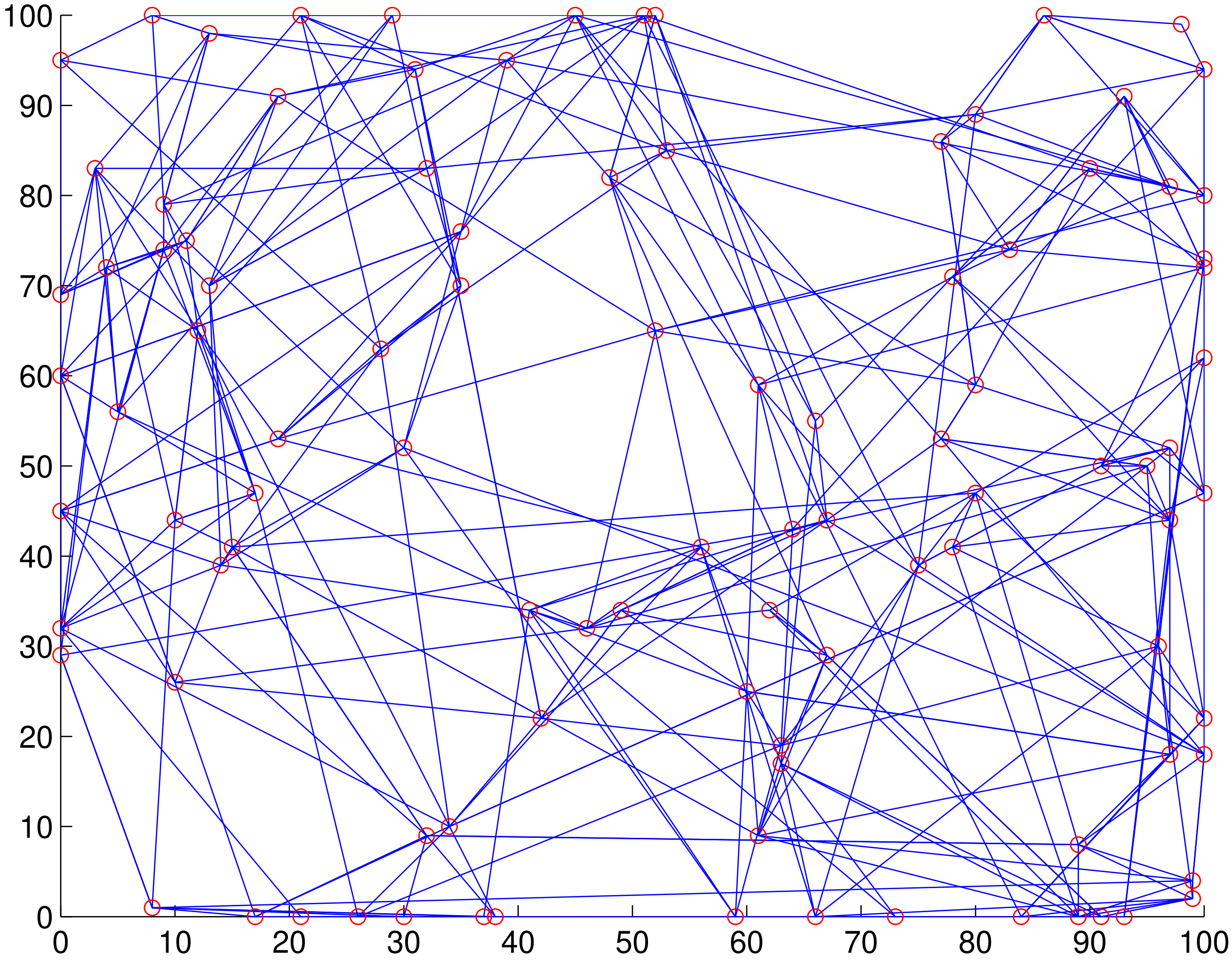

Figure 5: A random graph with 100 nodes, 33 of which are border nodes.

border nodes asymmetry ratio				No. of	links	max.
max	min	ave	var	bad △s	deleted	distortion
276.9	1.004	11.2	611	1546	341	4.95
413.1	1.002	5.2	418	823	305	4.88
864.3	1.001	8.9	1929	1101	372	5.70
100.2	1.011	4.0	45	767	338	2.72

Table 2: Results for a graph with 100 nodes, 33 of them are border nodes

border nodes asymmetry ratio				No. of	links	max.
max	min	ave	var	bad $\triangle$s	deleted	distortion
2.98	1.1003	1.90	0.3	3	2	1.04
89.31	1.0370	9.49	336	10	9	1.99
1511.74	1.0386	173.85	131223	22	10	3.83
190.59	1.0214	24.75	2022	12	8	1.47

Table 3: Results for a graph with 100 nodes, 7 of them are border nodes

5 Applicability to PNNI

In the PNNI standard [tc96, section 3.3.8], a network is aggregated ("complex node") by a star. A virtual node ("nucleus") is the interior reference point. For each state parameter, a single default value is given to represent the value of this parameter for the connection between a border node and the nucleus ("spoke") in each direction.

To cope with asymmetry, the standard allows "exceptions" of two types: a different value for a spoke of a certain border node, or a value for a "bypass" between two border nodes. In guidelines that are supplied with the standard [tc96, Appendix C] it is recommended that the number of exceptions used to configure a complex node will be kept smaller than $3b$, where b is the number of border nodes. There is no suggestion of how to select exceptions.

This paper, is the first (to the best of our knowledge) that gives a framework for aggregating "complex node" topology. The aggregations described in the sequel gives a bound on the worst case distortion in the advertised parameters which is $\log b \cdot \sqrt{\rho}$. The worst case distortion in the best star aggregation is $\sqrt{D/d}$ where D and d are the longest and shortest parameter values between pairs of border nodes, respectively. This means that for the case where the graph is undirected, our scheme distortion is only $\log b$ while the star distortion remains $\sqrt{D/d}$ that might be very large. Note that, adding $3b$ exceptions in a naive way can not fix the star distortion (e.g., assume the parameter values between border node are $1, 2, \ldots, 2^{b(b-1)-1}$).

Our suggested aggregation is based on random Bartal trees as described in section 3. However, a tree that is built by applying Bartal's algorithm directly is comprised of not only the original network nodes (the border nodes in our case) but also of up to $b-1$ additional virtual nodes. The PNNI standard assumes one virtual node which is the star nucleus, and does not provide means to describe additional virtual nodes. To report the tree aggregation one must embed the tree in the network links without loosing the logarithmic distortion. To describe our embedding, we must first provide a description of Bartal's algorithm.

Bartal's algorithm is comprised of two phases. In the first phase, the graph is recursively partitioned as follows. A node is arbitrarily selected, and all the nodes that are within a random radius from this node comprise the partition. The maximum value of the radius is a factor of k smaller than the diameter of the partition one level higher. This process continues for each partition, until all the nodes of the partition are in some sub-partition. For each sub-partition the process recurses.

In the second phase, a virtual node is assigned to each of the partitions in each level. Each virtual node, becomes the father of the virtual nodes that are assigned to the sub-partitions of its partition. The length of the links from a node to its children is half the partition diameter.

Our aim is to embed the virtual nodes in the network nodes in a way that does not increase the order of the distortion. After the partition phase, we pick a *center* for each partition P, i.e., we pick a node v whose maximal distance to any other node in the partition is minimal:

$$\max_{u \in P} w_{k^-}(v,\ u) = \min_{w \in P} \max_{u \in P} w_{k^-}(w,\ u)$$

The partition center radius is at most the partition diameter, which theoretically introduces a factor of two additional distortion. For the tree root we use the nucleus and the default spoke value for the embedding. In the worst case, when at every level a partition is divided to exactly two sub-partitions, $b-2$ nodes are embedded (the root is not embedded), and $2b-4$ bypass exceptions are used to represent the tree links. However, the selection of the node from which to start building a sub-partition is arbitrary. We choose to select the partition center to be the first node in the sub-partitioning. The resulted virtual tree does not contain cycles and thus can be represented with less than $b-1$ bypass exceptions.

To further decrease the distortion, the weight of the link between two nodes in the tree is set to the weight of the shortest path between them, rather then the partition radius. Theoretically, this change does not decrease the average distortion, but, in practice, the distortion for some of the links decreases. Note that for the farthest nodes in the partition, the factor two additional distortion may still apply due to the embedding of the virtual nodes in real nodes. However, it is not expected to be large in the high levels of the tree, where the distances are large, because at these levels, the number of nodes in a partition is fairly large and the center radius is expected to be close to half the partition diameter.

We suggest two structures based on the above trees. The first, a *t-tree*, is the union of t random trees. For this purpose one might want to build more than t trees and pick the t combination that gives the best performance. A potentially better construction called quality partition forest (QPF) is a union of some highest level partitions. We note that all the random trees share the same nucleus, and the same high-level diameter, that is advertised as the default diameter. This gives rise to the following QPF construction. Build several trees, and select one of them in random to the aggregated graph. Add additional first-level sub-partitions according to some quality measure until the threshold of $3b$ exceptions is reached. The measure for the quality of a first-level sub-partition can be the maximal (or average) improvement of the distances over all pairs of node due to the addition of the sub-partition.

We tested the t-tree construction on the graphs of figures 4 and 5. Trees were iteratively created until the cumulative number of links (exceptions) exceeded $2b$. Tables 4 and 5 summarize the link distortions for the same random weight assignments that were used in tables 1 and 2, respectively. The average distortion is shown to be about 1.5 when the number of border nodes is 22, and about 2 when 33 border nodes are aggregated. However, more important are the histograms showing for a column i the number of node-pairs whose distance distortion is above $i-1$ but not above i. The histograms show that most of the node-pair distances are distorted by less than $\log b$, and a negligible fraction of the node pairs are distorted by more than $2\log b$.

The maximal distortions when a single tree is used for the random assignments used in table 4 are 198, 13, 57.4, and 15.5. These distortions are two orders of magnitude higher than the maximum distortion of 6 obtained with the t-tree. In addition, the tail of highly distorted node-pair distances is fat, meaning that many pairs are highly distorted. The maximal distortions for a 1-tree that correspond to the results in table 5 are 81.2, 26.6, 34.4, and 68.8, up to 10 times more than the results obtained by the t-tree. Thus, we can conclude that by joining several random trees, we successfully avoid the highly distorted node-pairs that are part of any single tree, and suggest a construction that is both implementable and with reasonable distortion.

Note, that the actual number of exceptions used in our t-tree is closer to $2b$ than to $3b$ (the last column in tables 4 and 5). Since the number of node-pairs whose distance cost is badly distorted is smaller than b, we can add a phase to the algorithm in which bypass exceptions between the badly distorted node-pairs are added, eliminating the thin tail in the histograms.

max	ave	var	Histogram 1	2	3	4	5	6	7	8	9	10	links used
2.68	1.31	0.10	31	192	8	0	0	0	0	0	0	0	54
2.90	1.32	0.13	33	180	18	0	0	0	0	0	0	0	52
4.63	1.46	0.47	45	150	22	11	3	0	0	0	0	0	45
6.00	1.64	0.44	21	161	38	9	1	1	0	0	0	0	52

Table 4: Summary of the distortion for the graph of figure 4 (22 border nodes) when the t-tree structure is used for aggregation

max	ave	var	Histogram 1	2	3	4	5	6	7	8	9	10	>10	links used
8.83	1.78	1.21	38	388	62	15	9	6	2	6	2	0	0	72
11.18	2.01	1.25	27	282	173	25	4	10	3	1	1	0	2	67
4.39	1.38	0.17	82	407	37	1	1	0	0	0	0	0	0	67
23.72	2.75	4.64	33	243	68	77	53	16	17	8	4	3	5	67

Table 5: Summary of the distortion for the graph of figure 5 (33 border nodes) when the t-tree structure is used for aggregation

Acknowledgement

We would like to thank Amitabh Chaudhary and Amitabha Bagchi for their help in coding Bartal's algorithm.

References

[ADD+93] I. Althofer, G. Das, D. Dopkin, D. Joseph, and J. Soares. On sparse spanners of weighted graphs. *Discrete and Computational Geometry*, 9:81 – 100, 1993.

[Bar96] Yair Bartal. Probabilistic approximation of metric space and its algorithmic applications. In *37th Annual IEEE Symposium on Foundations of Computer Science*, October 1996.

[CCS96] I. Castineyra, J. N. Chiappa, and M. Steenstrup. The nimrod routing architecture, February 1996. Internet Draft, Nimrod Working Group.

[GO97] Roch Guérin and Ariel Orda. QoS-based routing in networks with inaccurate information: Theory and algorithms. In *IEEE INFOCOM'97*, pages 75 – 83, April 1997.

[PS89] David Peleg and Alejandro A. Schäffer. Graph spanners. *Journal of Graph Theory*, 13(1):99 – 116, 1989.

[PU88] David Peleg and Eli Upfal. A tradeoff between space and efficiency for routing tables. In *20th ACM Symposium on the Theory of Computing*, pages 43 – 52, May 1988.

[tc96] The ATM Forum technical committee. Private network-network interface specification version 1.0 (PNNI), March 1996. af-pnni-0055.000.

[Wax88] Bernard M. Waxman. Routing of multipoint connections. *Journal on Selected Areas in Communications*, 6:1617 – 1622, 1988.