On the Use of Duality and Geometry in Layouts for ATM Networks

Shmuel Zaks

Department of Computer Science, Technion, Haifa, Israel.
zaks@cs.technion.ac.il, http://www.cs.technion.ac.il/∼zaks

Abstract. We show how duality properties and geometric considerations are used in studies related to virtual path layouts of ATM networks. We concentrate on the one-to-many problem for a chain network, in which one constructs a set of paths, that enable connecting one vertex with all others in the network. We consider the parameters of load (the maximum number of paths that go through any single edge) and hop count (the maximum number of paths traversed by any single message). Optimal results are known for the cases where the routes are shortest paths and for the general case of unrestricted paths. These solutions are symmetric with respect to the two parameters of load and hop count, and thus suggest duality between these two. We discuss these dualities from various points of view. The trivial ones follow from corresponding recurrence relations and lattice paths. We then study the duality properties using trees; in the case of shortest paths layouts we use binary trees, and in the general case we use ternary trees. In this latter case we also use embedding into high dimensional spheres.
The duality nature of the solutions, together with the geometric approach, prove to be extremely useful tools in understanding and analyzing layout designs. They simplify proofs of known results (like the best average case designs for the shortest paths case), enable derivation of new results (like the best average case designs for the general paths case), and improve existing results (like for the all-to-all problem).

1 Introduction

In path layouts for ATM networks pairs of nodes exchange messages along predefined paths in the network, termed *virtual paths*. Given a physical network, the problem is to design these paths optimally. Each such design forms a layout of paths in the network, and each connection between two nodes must consist of a concatenation of such virtual paths. The smallest number of these paths between two nodes is termed the *hop count* for these nodes, and the *load* of a layout is the maximum number of virtual paths that go through any (physical) communication line. The two principal parameters that determine the optimality of the layout are the maximum load of any communication line and the maximum hop count between any two nodes. The hop count corresponds to the time to set up a connection between the two nodes, and the load measures the size of the routing tables at the nodes.

Following the model presented in [15,5], this problem has been studied from various points of view (see also Section 8).

M. Nielsen and B. Rovan (Eds.): MFCS 2000, LNCS 1893, pp. 114–131, 2000.

$h \backslash l$	1	2	3	4 ...
1	P	P	P	P ...
2	P	NPC	NPC	NPC ...
3	NPC	NPC	NPC	NPC ...
4	NPC	NPC	NPC	NPC ...
...	...	...	...	...

Fig. 1. Tractability of the one-to-all problem

The existence of a design, with given bounds on the load $\mathcal{L}$ and the hop count $\mathcal{H}$ between a given node and all the other nodes was shown to be NP-complete except for few cases. This was studied in [7], and the results - whether the problem is polynomially solvable or whether it is NP-complete - are summarized in thetable depicted in Fig. 1 (a related NP-complete result was presented in [15]).

Two basic problems that have been studied are the *one-to-all* (or *broadcast*) problem (e.g., [5,13,11,6]), and the *all-to-all* problem (see, e.g., [5,11,16,17,1,6]), in which one wishes to measure the hop count from one specified node (or all nodes) in the network to all other nodes.

In this paper we focus on chain networks, with an emphasis on duality properties and the use of geometry in various analytic results. Considering a chain network, where the leftmost vertex has to be the root (the one broadcasting to all others using the virtual paths), and where each path traversed by a message must be a shortest path, a family of ordered trees $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ was presented in [13], within which an optimal solution can be found, for a chain of length $\mathcal{N}$, with $\mathcal{N}$ bounded by $\binom{\mathcal{L}+\mathcal{H}}{\mathcal{L}}$. This number, which is symmetric in $\mathcal{H}$ and $\mathcal{L}$, is also equal to the number of lattice paths from $(0,0)$ to $(\mathcal{L},\mathcal{H})$, that use horizontal and vertical steps. Optimal bounds for this shortest path case were also derived for the average case , which also turned out to be symmetric in $\mathcal{H}$ and $\mathcal{L}$.

Considering the same problem but without the shortest path restriction, termed the *general path* case, a family of tree layouts $\mathcal{T}(\mathcal{L},\mathcal{H})$ was introduced in [6], for a chain of length $\mathcal{N}$, not assuming that the root is located at its leftmost vertex, and with $\mathcal{N}$ bounded by $\sum_{i=0}^{min\{\mathcal{L},\mathcal{H}\}} 2^i\binom{\mathcal{L}}{i}\binom{\mathcal{H}}{i}$ [12]. This number, which is also symmetric in $\mathcal{H}$ and $\mathcal{L}$, is equal to the number of lattice points within an $\mathcal{L}$-dimensional l_1-Sphere or radius $\mathcal{H}$, and is also equal to the number of lattice paths from $(0,0)$ to $(\mathcal{L},\mathcal{H})$, that use horizontal, vertical or (up-)diagonal steps. The main tool in this discussion was the possibility to map layouts with load bounded by $\mathcal{L}$ and hop count bounded by $\mathcal{H}$ into this sphere.

As a consequence, the trees $\mathcal{T}(\mathcal{L},\mathcal{H})$ and $\mathcal{T}(\mathcal{H},\mathcal{L})$ have the same number of nodes, and so do the trees $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ and $\mathcal{T}_{short}(\mathcal{H},\mathcal{L})$. It turns out that these dualities bear a lot of information regarding the structure of these trees, and exploring this duality, together with the use of the high dimensional spheres, proved to be extremely useful in understanding and analyzing layout designs: they simplify proofs of known results, enable derivation of new results, and improve existing results.

We use one-to-one correspondences, using binary and ternary trees, in order to combinatorially explain the duality between these two measures of hop count and load, as reflected by these above symmetries. These correspondences shed more light into the structure of these two families of trees, allowing to find for any optimal layout with $\mathcal{N}$ nodes, load $\mathcal{L}$ and minimal (or minimal average) hop count $\mathcal{H}$, its *dual* layout, having $\mathcal{N}$ nodes, maximal hop count $\mathcal{L}$ and minimal (or minimal average) load $\mathcal{H}$, and vice-versa. Moreover, they give one proof for both measures, whereas in the above-mentioned papers these symmetries were only derived as a consequence of the final result; we note that the average-case results were derived by a seemingly-different formulas, whereas the worst-case results were derived by symmetric arguments. In addition, these correspondences also provide a simple proof to a new result concerning the duality of these two parameters in the worst case and the average case analysis for the general path case layouts. Finally, it is shown that an optimal worst case solution for the shortest path and general cases, is also an optimal average case solution in both cases, allowing a simpler characterization of these optimal layouts. We then introduce the relation between high dimensional spheres and layouts for the general case. This is then used in simplifying proofs of known results, in derivation of new results (like the best average case designs for the general paths case), and in improving existing results (like for the all-to-all problem).

This survey paper is based on results presented in previous studies, as detailed in the following description of its structure. In Section 2 the ATM model is presented, following [5]. In Section 3 we discuss the optimal solutions; the optimal design for the shortest path case follows the discussion in [13], and the optimal design for the general case follows the discussion in [6,8]. We encounter the duality of the parameters of load and hop count, which follows via recurrence relations. In Section 4 we discuss relations with lattice paths. In Section 5 we describe the use of binary and ternary trees to shed more direct light on these duality results, and the use of high dimensional spheres is discussed in Section 6, both following ([6,8]). The applications of the tools of duality and geometry are presented in Section 7, following [8]). We close with a discussion in Section 8.

2 The Model

We model the underlying communication network as an undirected graph $G = (V, E)$, where the set V of vertices corresponds to the set of switches, and the set E of edges corresponds to the physical links between them.

Definition 1. *A* rooted virtual path layout (*layout* for short) Ψ *is a collection of simple paths in* G*, termed* virtual paths *(VPs for short), and a vertex* $r \in V$ *termed the* root *of the layout (denoted* $root(\Psi)$*).*

Definition 2. *The* load $\mathcal{L}(e)$ *of an edge* $e \in E$ *in a layout* Ψ *is the number of VPs* $\psi \in \Psi$ *that include* e*.*

Definition 3. *The* load $\mathcal{L}_{max}(\Psi)$ *of a layout* Ψ *is* $\max_{e \in E} \mathcal{L}(e)$*.*

Definition 4. *The* hop count $\mathcal{H}(v)$ *of a vertex* $v \in V$ *in a layout* Ψ *is the minimum number of VPs whose concatenation forms a path in* G *from* v *to* $root(\Psi)$*. If no such VPs exist, define* $\mathcal{H}(v) = \infty$*.*

Definition 5. *The* maximal hop count *of* Ψ *is* $\mathcal{H}_{max}(\Psi) = \max_{v \in V}\{\mathcal{H}(v)\}$*.*

In the rest of this paper we assume that the underlying network is a *chain*. We consider two cases: the one in which only shortest paths are allowed, and the second one in which general paths are considered.

To minimize the load, one can use a layout Ψ which has a VP on each physical link, i.e., $\mathcal{L}_{max}(\Psi) = 1$, however such a layout has a hop count of $\mathcal{N} - 1$. The other extreme is connecting a direct VP from the root to each other vertex, yielding $\mathcal{H}_{max} = 1$, but then $\mathcal{L}_{max} = \mathcal{N} - 1$. For the intermediate cases we need the following definitions.

Definition 6. $\mathcal{H}_{opt}(\mathcal{N}, \mathcal{L})$ *denotes the* optimal hop count *of any layout* Ψ *on a chain of* $\mathcal{N}$ *vertices such that* $\mathcal{L}_{max}(\Psi) \leq \mathcal{L}$*, i.e.,*

$$\mathcal{H}_{opt}(\mathcal{N}, \mathcal{L}) \equiv \min_{\Psi}\{\mathcal{H}_{max}(\Psi) : \mathcal{L}_{max}(\Psi) \leq \mathcal{L}\}.$$

Definition 7. $\mathcal{L}_{opt}(\mathcal{N}, \mathcal{H})$ *denotes the* optimal load *of any layout* Ψ *on a chain of* $\mathcal{N}$ *vertices such that* $\mathcal{H}_{max}(\Psi) \leq \mathcal{H}$*, i.e.,*

$$\mathcal{L}_{opt}(\mathcal{N}, \mathcal{H}) \equiv \min_{\Psi}\{\mathcal{L}_{max}(\Psi) : \mathcal{H}_{max}(\Psi) \leq \mathcal{H}\}.$$

Definition 8. *Two VPs constitute a* crossing *if their endpoints* l_1, l_2 *and* r_1, r_2 *satisfy* $l_1 < l_2 < r_1 < r_2$*. A layout is called* crossing-free *if no pair of VPs constitute a crossing.*

It is known ([13,1]) that for each performance measure ($\mathcal{L}_{max}$, $\mathcal{H}_{max}$, $\mathcal{L}_{avg}$, $\mathcal{H}_{avg}$) there exists an optimal layout which is crossing-free. In the rest of the paper we restrict ourselves to layouts viewed as a planar (that is, crossing-free) embedding of a tree on the chain, also termed *tree layouts*. Therefore, when no confusion occurs, we refer to each VP in a given layout Ψ an *edge* of Ψ.

$\mathcal{N}_{\text{short}}(\mathcal{L}, \mathcal{H})$ denotes the length of a longest chain in which one node can broadcast to all others, with at most $\mathcal{H}$ hops and a load bounded by $\mathcal{L}$, for the case of shortest paths. The similar measure for the general case is denoted by $\mathcal{N}(\mathcal{L}, \mathcal{H})$.

3 Optimal Solutions and Their Duality

In this section we present the optimal solutions for layouts, when messages have to travel either along shortest paths or general paths. We'll show the symmetric role played by the load and hop count, and explain it via the corresponding recurrence relations.

3.1 Optimal Virtual Path for the Shortest Path Case

Assuming that the leftmost node in the chain has to broadcast to each node to its right, it is clear that, for given $\mathcal{H}$ and $\mathcal{L}$, the largest possible chain for which such a design exists is like the one shows in Fig. 2.

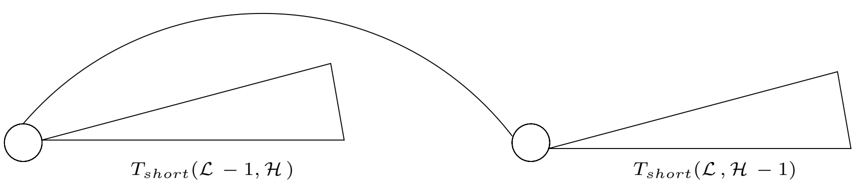

Fig. 2. The tree layout $\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})$

The design depicted in Fig. 2 uses the trees $\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})$ defined as follows.

Definition 9. *The tree layout $\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})$ is defined recursively as follows. $\mathcal{T}_{short}(\mathcal{L}, 0)$ and $\mathcal{T}_{short}(0, \mathcal{H})$ are tree layouts with a unique node. Otherwise, the root of a tree layout $\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})$ is the leftmost node of a $\mathcal{T}_{short}(\mathcal{L}-1, \mathcal{H})$ tree layout, and it is also the leftmost node of a tree layout $\mathcal{T}_{short}(\mathcal{L}, \mathcal{H}-1)$*

Recall that $\mathcal{M}_{\text{short}}(\mathcal{L}, \mathcal{H})$ is the length of the longest chain in which a design exists, for a broadcast from the leftmost node to all others, for given parameters $\mathcal{H}$ and $\mathcal{L}$. $\mathcal{M}_{\text{short}}(\mathcal{L}, \mathcal{H})$ clearly satisfies the following recurrence relation:

$$\mathcal{M}_{\text{short}}(0, \mathcal{H}) = \mathcal{M}_{\text{short}}(\mathcal{L}, 0) = 1 \quad \forall\, \mathcal{H}, \mathcal{L} \geq 0 \tag{1}$$
$$\mathcal{M}_{\text{short}}(\mathcal{L}, \mathcal{H}) = \mathcal{M}_{\text{short}}(\mathcal{L}, \mathcal{H}-1) + \mathcal{M}_{\text{short}}(\mathcal{L}-1, \mathcal{H}) \quad \forall\, \mathcal{H}, \mathcal{L} > 0 \ .$$

It easily follows that

$$\mathcal{M}_{\text{short}}(\mathcal{L}, \mathcal{H}) = \binom{\mathcal{L} + \mathcal{H}}{\mathcal{H}}. \tag{2}$$

The expression in 2 is clearly symmetric in $\mathcal{H}$ and $\mathcal{L}$, which establishes the first result in which the load and hop count play symmetric roles.

Note that it is clear that the maximal number $\mathcal{N}_{\text{short}}(\mathcal{L}, \mathcal{H})$ of nodes in a chain to which one node can broadcast using shortest paths, satisfies

$$\mathcal{N}_{\text{short}}(\mathcal{L}, \mathcal{H}) = 2\binom{\mathcal{L} + \mathcal{H}}{\mathcal{H}} - 1 .$$

Using these trees, it is easy to show that $\mathcal{L}_{max}(\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})) = \mathcal{L}$ and $\mathcal{H}_{max}(\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})) = \mathcal{H}$. The following two theorems follow:

Theorem 1. *Consider a chain of $\mathcal{N}$ vertices and a maximal load requirement $\mathcal{L}$. Let $\mathcal{H}$ be such that*

$$\binom{\mathcal{L} + \mathcal{H} - 1}{\mathcal{L}} < \mathcal{N} \leq \binom{\mathcal{L} + \mathcal{H}}{\mathcal{L}}.$$

Then $\mathcal{H}_{opt}(\mathcal{N}, \mathcal{L}) = \mathcal{H}$.

Theorem 2. *Consider a chain of $\mathcal{N}$ vertices and a maximal hop requirement $\mathcal{H}$. Let $\mathcal{L}$ be such that*

$$\binom{\mathcal{L} + \mathcal{H} - 1}{\mathcal{H}} < \mathcal{N} \leq \binom{\mathcal{L} + \mathcal{H}}{\mathcal{H}}.$$

Then $\mathcal{L}_{opt}(\mathcal{N}, \mathcal{H}) = \mathcal{L}$.

Optimal bounds were also derived in [13,14] for the average case, using dynamic programming; the results use different recursive constructions, but end up in structures that are symmetric in $\mathcal{H}$ and $\mathcal{L}$. These results are stated as follows:

Theorem 3. *Let n and $\mathcal{H}$ be given. Let $\mathcal{L}$ be the largest integer such that $N \geq \binom{\mathcal{L}+\mathcal{H}}{\mathcal{L}}$, and let $r = N - \binom{\mathcal{L}+\mathcal{H}}{\mathcal{L}}$. Then*

$$\mathcal{L}_{tot}(N, \mathcal{H}) = \mathcal{H}\binom{\mathcal{L} + \mathcal{H}}{\mathcal{L} - 1} + r(\mathcal{L} + 1).$$

Theorem 4. *Let N and $\mathcal{L}$ be given. Let $\mathcal{H}$ be the maximal such that $N \geq \binom{\mathcal{L}+\mathcal{H}}{\mathcal{H}}$, and let $r = N - \binom{\mathcal{L}+\mathcal{H}}{\mathcal{H}}$. Then*

$$\mathcal{H}_{tot}(N, \mathcal{L}) = \mathcal{L}\binom{\mathcal{L} + \mathcal{H}}{\mathcal{H} - 1} + r(\mathcal{H} + 1).$$

Note that these last two theorems imply that in $\mathcal{T}_{short}(\mathcal{L}, \mathcal{H})$ the average hop is $\frac{\mathcal{L}}{\mathcal{L}+1}\mathcal{H}$, and the average load is slightly smaller than $\frac{\mathcal{H}}{\mathcal{H}+1}\mathcal{L}$.

3.2 Optimal Virtual Path for the General Case

In the case where not only shortest paths are traversed, a new family of optimal tree layouts $\mathcal{T}(\mathcal{L},\mathcal{H})$ is now presented.

Definition 10. *The tree layout $\mathcal{T}(\mathcal{L},\mathcal{H})$ is defined recursively as follows. $\mathcal{T}_{right}(\mathcal{L},0)$, $\mathcal{T}_{right}(0,\mathcal{H})$, $\mathcal{T}_{left}(\mathcal{L},0)$ and $\mathcal{T}_{left}(0,\mathcal{H})$ are tree layouts with a unique node. Otherwise, the root r is also the rightmost node of a tree layout $\mathcal{T}_{right}(\mathcal{L},\mathcal{H})$ and the leftmost node of a tree layout $\mathcal{T}_{left}(\mathcal{L},\mathcal{H})$, when the tree layouts $\mathcal{T}_{left}(\mathcal{L},\mathcal{H})$ and $\mathcal{T}_{right}(\mathcal{L},\mathcal{H})$ are also defined recursively as follows. The root of a tree layout $\mathcal{T}_{left}(\mathcal{L},\mathcal{H})$ is the leftmost node of a $\mathcal{T}_{left}(\mathcal{L}-1,\mathcal{H})$ tree layout, and it is also connected to a node which is the root of a tree layout $\mathcal{T}_{right}(\mathcal{L}-1,\mathcal{H}-1)$ and a tree layout $\mathcal{T}_{left}(\mathcal{L},\mathcal{H}-1)$ (see Fig. 3). Note that the root of $\mathcal{T}_{left}(\mathcal{L},\mathcal{H})$ is its leftmost node. The tree layout $\mathcal{T}_{right}(\mathcal{L},\mathcal{H})$ is defined as the mirror image of $\mathcal{T}_{left}(\mathcal{L},\mathcal{H})$.*

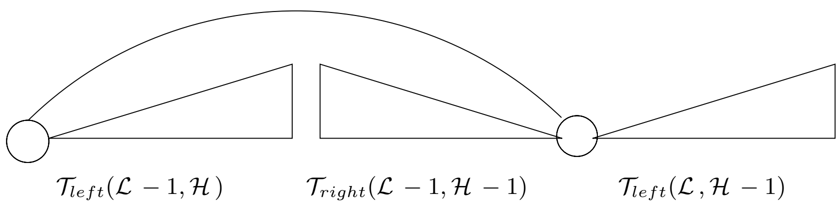

Fig. 3. $\mathcal{T}_{left}(\mathcal{L},\mathcal{H})$ recursive definition

Denote by $\mathcal{N}(\mathcal{L},\mathcal{H})$ the longest chain in which it is possible to connect one node to all others, with at most $\mathcal{H}$ hops and the load bounded by $\mathcal{L}$. From the above, it is clear that this chain is constructed from two chains as above, glued at their root. $\mathcal{N}(\mathcal{L},\mathcal{H})$ clearly satisfies the following recurrence relation:

$$\mathcal{N}(0,\mathcal{H}) = \mathcal{N}(\mathcal{L},0) = 1 \quad \forall\, \mathcal{H},\mathcal{L} \ \geq\ 0 \tag{3}$$
$$\mathcal{N}(\mathcal{L},\mathcal{H}) = \mathcal{N}(\mathcal{L},\mathcal{H}-1) + \mathcal{N}(\mathcal{L}-1,\mathcal{H}) + \mathcal{N}(\mathcal{L}-1,\mathcal{H}-1) \quad \forall\, \mathcal{H},\mathcal{L} > 0\ .$$

Again, the symmetric role of the hop count and the load are clear both from the definition of the corresponding trees and from the recurrence relations that compute their sizes.

It is known ([12]) that the solution to the recurrence relation (3) is given by

$$\mathcal{N} = \sum_{i=0}^{min\{\mathcal{L},\mathcal{H}\}} 2^i \binom{\mathcal{L}}{i}\binom{\mathcal{H}}{i}. \tag{4}$$

4 Correspondences with Lattice Paths

The recurrence relation (1) clearly corresponds to the number of lattice paths from the point (0,0) to the point $(\mathcal{L},\mathcal{H})$, that use only horizontal (right) and vertical (up) steps.

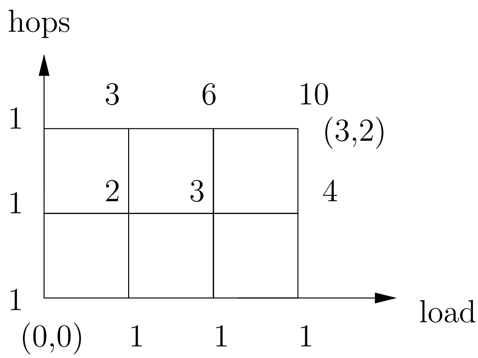

Fig. 4. Lattice paths with regular steps

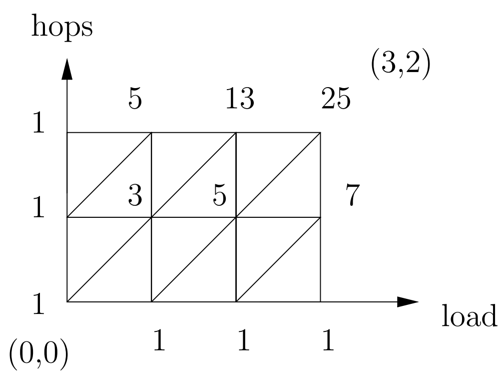

Fig. 5. Lattice paths with regular and diagonal steps

In Fig. 4 each lattice point is labeled with the number of lattice paths from (0,0) to it; the calculation is done by the recurrence relation 1. For the case $\mathcal{L}=3$ and $\mathcal{H}=2$ one gets $\binom{3+2}{2}=10$; this corresponds to the number of nodes in the tree $\mathcal{T}_{short}(3,2)$ (see Fig. 6), and to the number of paths that go from (0,0) to (3,2).

The recurrence relation (3) clearly corresponds to the number of lattice paths from the point (0,0) to the point $(\mathcal{L},\mathcal{H})$, that use horizontal (right), vertical (up), and diagonal (up-right) steps. In Fig. 5 each lattice point is labeled with the number of lattice paths from (0,0) to it. For the case $\mathcal{L}=3$ and $\mathcal{H}=2$ one gets 25 such paths. This corresponds to the number of nodes in the tree $\mathcal{T}(3,2)$ that is constructed of two trees, glued at their roots, the one $(\mathcal{T}_{left}(3,2))$ depicted in Fig. 6 (and containing 13 vertices), and its corresponding reverse tree.

We also refer to these lattice paths in Section 6.

5 Duality: Binary Trees and Ternary Trees

We saw in Section 3 that the layouts $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ and $\mathcal{T}_{short}(\mathcal{H},\mathcal{L})$ and also $\mathcal{T}(\mathcal{L},\mathcal{H})$ and $\mathcal{T}(\mathcal{H},\mathcal{L})$ have the same number of vertices. We now turn to show that each pair within these virtual path layouts are, actually, quite strongly related. In Section 5.1 we deal with layouts that use shortest-length paths, and show their close relations to a certain class of binary trees, and in Section 5.2 we deal with the general layouts and show their close relations to a certain class of ternary trees.

5.1 $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ and Binary Trees

In this section we show how to transform any layout Ψ with hop count bounded by $\mathcal{H}$ and load bounded by $\mathcal{L}$ for layouts using only shortest paths, into a layout $\overline{\Psi}$ (its *dual*) with hop count bounded by $\mathcal{L}$ and load bounded by $\mathcal{H}$. In particular, this mapping will transform $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ into $\mathcal{T}_{short}(\mathcal{H},\mathcal{L})$.

To show this, we use transformation between any layout with x virtual paths (depicted as edges) and binary trees with x nodes (in a binary tree, each internal node has a left child and/or a right child). We'll derive our main correspondence between $\mathcal{T}_{short}(\mathcal{H},\mathcal{L})$ and $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ for $x = \mathcal{N} - 1$, where $\mathcal{N} = \binom{\mathcal{L}+\mathcal{H}}{\mathcal{L}}$. Our correspondence is done in three steps, as follows.

Step 1: Given a planar layout Ψ we transform it into a binary tree $T = b(\Psi)$, under which each edge e is mapped to a node $b(e)$, as follows. Let $e = (r, v)$ be the edge outgoing the root r to the rightmost vertex (to which there is a VP; we call this a *1-level* edge). This edge e is mapped to the root $b(r)$ of T. Remove e from Ψ. As a consequence, two layouts remain: Ψ_1 with root r and Ψ_2 with root v, when their roots are located at the leftmost vertices of both layouts. Recursively, the left child of node $b(e)$ will be $b(\Psi_1)$ and its right child will be $b(\Psi_2)$. If any of the layouts Ψ is empty, so is its image $b(\Psi)$ (in other words, we can stop when a Ψ that consists of a single edge is mapped to a binary tree that consists of a single vertex).

Step 2: Build a binary tree $\overline{T}$, which is a reflection of T (that is, we exchange the left child and the right child of each vertex).

Step 3: We transform back the binary tree $\overline{T}$ into the (unique) layout $\overline{\Psi}$ such that $b(\overline{\Psi}) = \overline{T}$

In Fig. 6 the layouts for $\mathcal{L} = 2, \mathcal{H} = 3$ and $\mathcal{L} = 3, \mathcal{H} = 2$ are shown, together with the corresponding trees $\mathcal{T}_{short}(2, 3)$ and $\mathcal{T}_{short}(3, 2)$, and the corresponding binary trees constructed as explained above. The edge e in the layout $\mathcal{T}_{short}(3, 2)$ is assigned the vertex $b(e)$ in the corresponding tree $b(\mathcal{T}_{short}(3, 2))$.

Given a non-crossing layout Ψ, we define the *level* of an edge e in Ψ, denoted $level_{\Psi}(e)$ (or $level(e)$ for short), to be one plus the number of edges above e in Ψ. In addition, to each edge e of the layout Ψ we assign its farthest end-point from the root, $v(e)$.

In Fig. 6 the edge e in the layout $\mathcal{T}_{short}(3, 2)$ is assigned the vertex $v(e)$ in this layout, and its level $level(e)$ is 2.

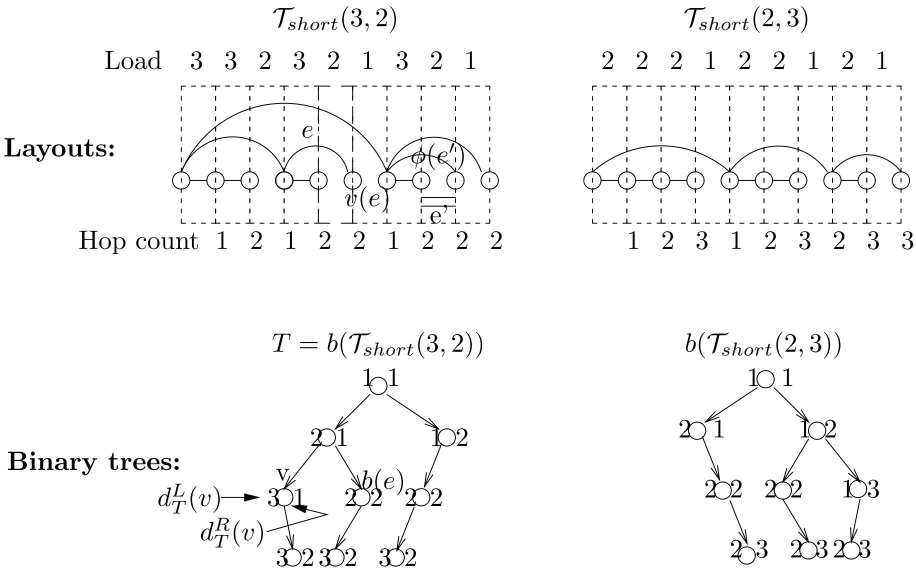

Fig. 6. An example of the transformation using binary trees

One of our key observations is the following theorem:

Theorem 5. *For every $\mathcal{H}$ and $\mathcal{L}$, the trees $b(\mathcal{T}_{short}(\mathcal{L}, \mathcal{H}))$ and $b(\mathcal{T}_{short}(\mathcal{H}, \mathcal{L}))$ are reflections of each other.*

This clearly establishes a one-to-one mapping between these trees, and thus establishes the required duality.

To further investigate the structure of these trees, we now turn to explore the properties of the binary trees that we have defined above. We prove the following theorem:

Theorem 6. *Given a layout Ψ, let $T = b(\Psi)$ be the binary tree assigned to it by the transformation above. Let $d_T^L(v)$ ($d_T^R(v)$) be equal to one plus the number of left (right) steps in the path from the root r to v, for every node v in T. Then, for every edge e in the layout Ψ:*

1. $\mathcal{H}_\Psi(v(e)) = d_T^R(b(e))$, *and*
2. $level(e) = d_T^L(b(e))$.

Given a non-crossing layout Ψ, for each physical link e' we assign an edge $\phi(e')$ in Ψ that includes it and is of highest level (such a path exists due to the connectivity and planarity of the layout; see edge e' and physical edge $\phi(e')$ in Fig. 6). It can be easily proved that:

Lemma 1. *Given a non-crossing tree layout Ψ, the mapping of a physical link e' to an edge $\phi(e')$ described above is one-to-one.*

Proposition 1. *Given a non-crossing tree layout Ψ over a physical network, let $T = b(\Psi)$ be the binary tree assigned to it. Then $\mathcal{L}(e') = level(\phi(e'))$ for every edge e' in the physical network.*

Given a layout Ψ over a chain network, if we consider the multiset $\{d_T^R(v)|v \in b(\Psi)\}$ we get exactly the multiset of hop counts of the vertices of this network (by Theorem 6), and if we consider the multiset $\{d_T^L(v)|v \in b(\Psi)\}$ we get exactly the multiset of loads of the physical links of this network (by Theorem 6 and Proposition 1). By using this and finding the dual layout $\overline{\Psi}$ with the multisets $\{d_T^R(v)|v \in b(\overline{\Psi})\}$ of hop counts of its vertices and $\{d_T^L(v)|v \in b(\overline{\Psi})\}$ of loads of its physical edges of $\overline{\Psi}$, we observe that the multiset of hop counts of $\overline{\Psi}$ is exactly the multiset of load of Ψ, and the multiset of loads of $\overline{\Psi}$ is also the multiset of hop counts of Ψ, thus deriving a complete combinatorial explanation for the symmetric results of Section 3.1 for either the worst case trees or average case trees:

Theorem 7. *Given an optimal layout Ψ with $\mathcal{N}$ nodes, load bounded by $\mathcal{L}$ and optimal hop count $\mathcal{H}_{opt}(\mathcal{N}, \mathcal{L})$, its dual layout $\overline{\Psi}$ has $\mathcal{N}$ nodes, hop count bounded by $\mathcal{L}$ and optimal load $\mathcal{H}_{opt}(\mathcal{N}, \mathcal{L})$.*

Theorem 8. *Given an optimal layout Ψ with $\mathcal{N}$ nodes, hop count bounded by $\mathcal{H}$ and optimal load $\mathcal{L}_{opt}(\mathcal{N}, \mathcal{H})$, its dual layout $\overline{\Psi}$ has $\mathcal{N}$ nodes, load bounded by $\mathcal{H}$ and optimal hop count $\mathcal{L}_{opt}(\mathcal{N}, \mathcal{H})$.*

Theorem 9. *Given an optimal layout Ψ with $\mathcal{N}$ nodes, load bounded by $\mathcal{L}$ and optimal average hop count, its dual layout $\overline{\Psi}$ has $\mathcal{N}$ nodes, hop count bounded by $\mathcal{L}$ and optimal average load.*

Theorem 10. *Given an optimal layout Ψ with $\mathcal{N}$ nodes, hop count bounded by $\mathcal{H}$ and optimal average load, its dual layout $\overline{\Psi}$ has $\mathcal{N}$ nodes, load bounded by $\mathcal{H}$ and optimal average hop count.*

5.2 $\mathcal{T}(\mathcal{L}, \mathcal{H})$ and Ternary Trees

We now extend the technique developed in Section 5.1 to general path case layouts; we show how to transform any layout Ψ with hop count bounded by $\mathcal{H}$ and load bounded by $\mathcal{L}$ into a layout $\overline{\Psi}$ (its *dual*) with hop count bounded by $\mathcal{L}$ and load bounded by $\mathcal{H}$. In particular, this mapping will transform $\mathcal{T}(\mathcal{L}, \mathcal{H})$ into $\mathcal{T}(\mathcal{H}, \mathcal{L})$.

To show this, we use transformation between any layout with x edges (VP s) and ternary trees with x nodes (in a ternary tree, each internal node has a left child and/or a middle child and/or a right child). Our correspondence is done in three steps, as follows.

Step 1: Given a planar layout Ψ we transform it into a ternary tree $T = t(\Psi)$, under which each edge e is mapped to a node $t(e)$, as follows. Let $e = (r, v)$ be

the edge outgoing the root r to the rightmost vertex (to which there is a VP; we call this a *1-level* edge). This edge e is mapped to the root $t(r)$ of T. Remove e from Ψ. As a consequence,three layouts remain: Ψ_1 with root r and and Ψ_3 with root v (when their roots are located at the leftmost vertices of both layouts) and Ψ_2 with root v (when v is its rightmost vertex). Recursively, the left child of node $t(e)$ will be $t(\Psi_1)$, its middle child will be $t(\Psi_2)$ and its right child will be $t(\Psi_3)$. If any of the layouts Ψ is empty, so is its image $t(\Psi)$ (in other words, we can stop when a Ψ that consists of a single edge is mapped to a ternary tree that consists of a single vertex).
Step 2: Build a ternary tree $\overline{T}$, which is a reflection of T (that is, we exchange the left child and the right child of each vertex; the middle child does not change).
Step 3: We transform back the ternary tree $\overline{T}$ into the (unique) layout $\overline{\Psi}$ such that $t(\overline{\Psi}) = \overline{T}$

See Fig. 7 for an example of this transformation.

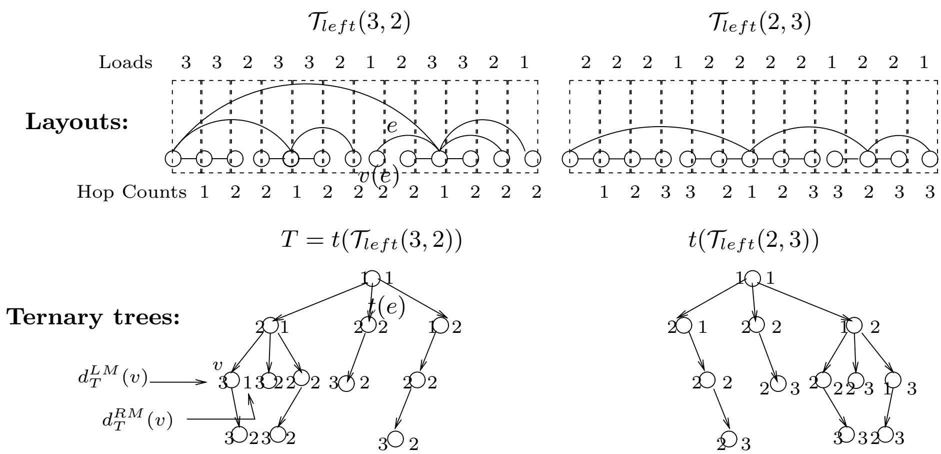

Fig. 7. An example of the transformation using ternary trees

One of our key observations is the following theorem:

Theorem 11. *For every $\mathcal{H}$ and $\mathcal{L}$, the trees $t(\mathcal{T}(\mathcal{L},\mathcal{H}))$ and $t(\mathcal{T}(\mathcal{H},\mathcal{L}))$ are reflections of each other.*

This clearly establishes a one-to-one mapping between these trees, and thus establishes the required duality.

To further investigate the structure of these trees, we now turn to explore the properties of the ternary trees that we have defined above. We prove the following theorem. Note that the definition of *level* (of an edge) and ϕ (of a physical link) remain exactly the same as in Section 5.1.

Theorem 12. *Given a layout Ψ, let $T = t(\Psi)$ be the ternary tree assigned to it by the transformation above. Let $d_T^{LM}(v)$ ($d_T^{RM}(v)$) be equal to one plus the* *number of left and middle (right and middle) steps in the path from the root r to v, for every node v in T. Then, for every edge e in the layout Ψ:*

1. $\mathcal{H}_{\Psi}(v(e)) = d_T^{RM}(t(e))$*, and*
2. $level(e) = d_T^{LM}(t(e))$*.*

Proposition 2. *Given a non-crossing tree layout Ψ over a physical network, let $T = t(\Psi)$ be the ternary tree assigned to it. Then $\mathcal{L}(e') = level(\phi(e'))$ for every edge e' in the physical network.*

Given a layout Ψ over a chain network, if we consider the multiset $\{d_T^{RM}(v)|v \in t(\Psi)\}$ we get exactly the multiset of hop counts of the vertices of this network (by Theorem 12), and if we consider the multiset $\{d_T^{LM}(v)|v \in t(\Psi)\}$ we get exactly the multiset of loads of the physical links of this network (by Theorem 12 and Proposition 2). By using this and finding the dual layout $\overline{\Psi}$ with the multisets $\{d_T^{RM}(v)|v \in t(\overline{\Psi})\}$ of hop counts of its vertices and $\{d_T^{LM}(v)|v \in t(\overline{\Psi})\}$ of loads of its physical edges of $\overline{\Psi}$, we observe that the multiset of hop counts of $\overline{\Psi}$ is exactly the multiset of load of Ψ, and the multiset of loads of $\overline{\Psi}$ is also the multiset of hop counts of Ψ, thus deriving a complete combinatorial explanation for the symmetric results of either the worst-case trees or average-case trees in the general path case.

Following the above discussion, we obtain the exact four theorems (Theorems 7, 8, 9 and 10) extended to the general path case layouts.

6 Use of Geometry

Consider the set $Sp(\mathcal{L}, \mathcal{H})$ of lattice points (that is, points with integral coordinates) of an $\mathcal{L}$-dimensional l_1-Sphere of radius $\mathcal{H}$. The points in this sphere are $\mathcal{L}$-dimensional vectors $\boldsymbol{v} = (v_1, v_2, \ldots, v_{\mathcal{L}})$, where $|v_1| + |v_2| + \ldots + |v_{\mathcal{L}}| \leq \mathcal{H}$. Let $Rad(N, \mathcal{L})$ be the radius of the smallest $\mathcal{L}$-dimensional l_1-Sphere containing at least $\mathcal{N}$ internal lattice points. For example, $Sp(1, 2)$ contains 5 lattice points, and $Rad(6, 2) = 3$.

We show that

Theorem 13. *The tree $\mathcal{T}(\mathcal{L}, \mathcal{H})$ contains $|Sp(\mathcal{L}, \mathcal{H})|$ vertices.*

The exact number of points in this sphere is given by equation (4). (This was studied, in connection with codewords, in [12].)

Moreover, we can show that

Theorem 14. *Consider a chain of $\mathcal{N}$ vertices and a maximal load requirement $\mathcal{L}$. Then $\mathcal{H}_{opt}(\mathcal{N}, \mathcal{L}) = Rad(N, \mathcal{L})$.*

These theorems are proved by showing a one-to-one mapping between the nodes of any layout with hop count bounded by $\mathcal{H}$ and load bounded by $\mathcal{L}$ into $Sp(\mathcal{L}, \mathcal{H})$. This mapping turns out to be a very useful tool in derivations of analytical results (see Section 8). The details of this embedding can be found in [6,8]. A short description of this embedding is now described, followed by an example.

In this embedding, a node for which the hop count is h will be mapped onto a point $(x_1, x_2, \cdots, x_{\mathcal{L}})$, such that $|x_1| + |x_2| + \cdots + |x_{\mathcal{L}}| = h$. This embedding starts by mapping the root of the given layout onto the origin $(0, 0, ..., 0)$. The algorithm continues in $\mathcal{L}$ phases. In the first phase we consider the paths, emanating from the root in both directions. The nodes on one path are mapped to the points $(1, 0, ..., 0), (2, 0, ..., 0)$, and so on, and the nodes on the other path to the points $(-1, 0, ..., 0), (-2, 0, ..., 0)$, and so on. In each subsequent phase, we continue in the same manner from each node that already got mapped, and for each such node, the new nodes are mapped to points that differ from it in the second component.

We present now an example of this embedding . We illustrate our algorithm on the tree layout T shown in Figure 8(A). $first(T) = a$, $last(T) = d$, and $root = c$. The path P_1 is $first(T) = a - b - c = root$ and the path P_2 is $root = c - d = last(T)$. We thus map in the first stage ($\xi = 1$) the nodes a, b, c and d to the points (2,0,0), (1,0,0), (0,0,0) and (-1,0,0), respectively (see Figure 8(B)). We then delete these edges from T, and the remaining graph (forest) is shown in Figure 8(C). At this 2nd stage ($\xi = 2$), the nodes b, c and d are roots of non-trivial layouts, and the algorithm maps the nodes e, f, and g to the points (1,-1,0), (0,-1,0) and (-1,1,0), respectively. Note that $LABEL(e)[1] = LABEL(b)[1] = 1$. The corresponding edges are then deleted from the layout, and we result in the graph depicted in Figure 8(D), which results in a similar mapping for nodes h and i.

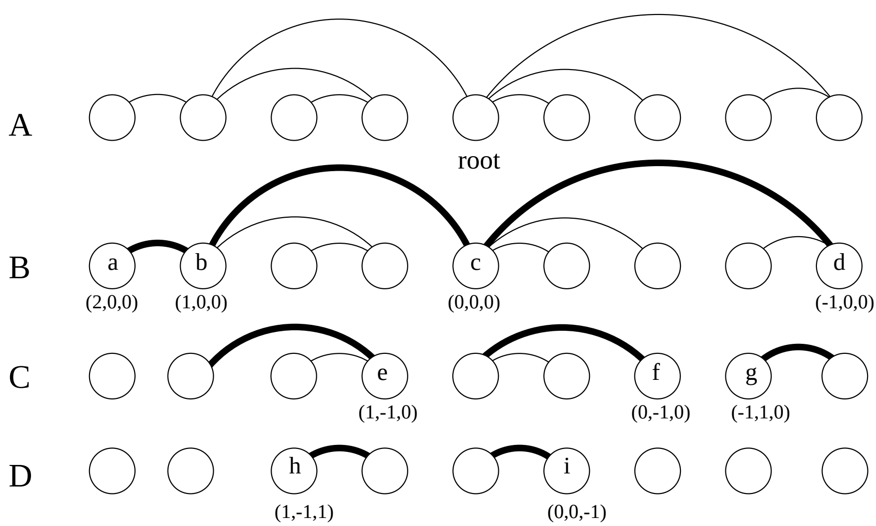

Fig. 8. Embedding of a tree layout of load 3

We now sketch a one-to-one mapping between the set of lattice points of the $\mathcal{L}$-dimensional sphere of radius $\mathcal{H}$ and the set of lattice paths from $(0,0)$ to $(\mathcal{L},\mathcal{H})$ that use horizontal, vertical or (up-)diagonal steps. We first describe a function which maps every vector $\boldsymbol{v} = (v_1,\ldots,v_{\mathcal{L}})$ in $Sp(\mathcal{L},\mathcal{H})$ into such a lattice path. Starting from $(0,0)$ make $|v_1|$ vertical steps and one horizontal step, make $|v_2|$ vertical steps and one horizontal step,..., make $|v_{\mathcal{L}}|$ vertical steps and one more horizontal step, ending with $\mathcal{H} - \sum_{i=1}^{i=l} v_i$ horizontal steps. After that, for every negative v_i component of $\boldsymbol{v}$, we replace the $|v_i|$th vertical step and the subsequent horizontal step done during the translation of this component by an (up-)diagonal step. A close look at the properties of these paths enables us to further explore the properties of these trees. Returning to the discussion of the layouts $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ that use only shortest paths, it is possible to find a similar correspondence between the vertices of these trees and lattice paths from $(0,0)$ to $(\mathcal{L},\mathcal{H})$ that use only vertical and horizontal steps, and to view some properties of these trees using these lattice paths.

7 Applications

The insight gained by the duality properties of the solutions for both the shortest path case and the general path case, and the one-to-one correspondence between layouts with a hop count bounded by $\mathcal{H}$ and load bounded by $\mathcal{L}$ with the lattice points in $Sp(\mathcal{L},\mathcal{H})$, have proved to be quite powerful in deriving analytical results.

1. Using the insight we got for the trees $\mathcal{T}_{short}(\mathcal{L},\mathcal{H})$ due to the duality between the hop count and the load, we managed to supply very short proofs for the optimal average hop count and load in the shortest path case, as detailed in Theorems 3 and 4 (for details, see [8]). Moreover, the duality properties imply that a solution for a certain setting of the parameters implies a solution for the same setting, where the roles of the hop count and the load are interchanged. See Theorems 7, 8, 9 and 10, and the last sentence of Section 5.
2. Using the duality properties, and especially using the high dimensional spheres, the following theorem can be proved (see [8]), regarding the optimal average hop count and load in the general path case.

   Theorem 15. *Let N and $\mathcal{H}$ be given. Let $\mathcal{L}$ be the maximal l such that $|Sp(l,\mathcal{H})| \le N$, and let $r = N - |T(\mathcal{L},\mathcal{H})|$. Then*

   $$\mathcal{L}_{tot}(N,\mathcal{H}) = (\mathcal{L} + \frac{1}{2})|Sp(\mathcal{L},\mathcal{H})| - \frac{1}{2}|Sp(\mathcal{L},\mathcal{H}+1)| + r(\mathcal{L}+1).$$

   Theorem 16. *Let N and $\mathcal{L}$ be given. Let $\mathcal{H}$ be the maximal h such that $|Sp(\mathcal{L},h)| \le N$, and let $r = N - |T(\mathcal{L},\mathcal{H})|$. Then*

   $$\mathcal{H}_{tot}(N,\mathcal{H}) = (\mathcal{H} + \frac{1}{2})|Sp(\mathcal{L},\mathcal{H})| - \frac{1}{2}|Sp(\mathcal{L}+1,\mathcal{H})| + r(\mathcal{H}+1).$$

3. Using the above correspondences and discussion, it can be shown that the layout that we managed to design - for a given N and $\mathcal{L}$ - with an optimal worst case hop count is also optimal with respect to the average case hop count, and the layout that we managed to design - for a given N and $\mathcal{H}$ - with an optimal worst case load is also optimal with respect to the average case load. This holds for either the shortest path designs or the general path designs (for details, see [8]).
4. Using volumes of high-dimensional polyhedra, we show a trade-off between the hop count and load, as follows (for a detailed discussion, see [6,8]):

   Theorem 17. *For all $\mathcal{L}$ and N:*

$$max\{1/2 \cdot (\mathcal{L}\,!N)^{\frac{1}{\mathcal{L}}} - \mathcal{L}\,/2, 1/2 \cdot N^{\frac{1}{\mathcal{L}}} - 1/2, \frac{\log N}{\log\,(2 \cdot \mathcal{L}\, + 1)}\} \leq$$

$$\leq Rad(N, \mathcal{L}\,) < 1/2 \cdot (\mathcal{L}\,!N)^{\frac{1}{\mathcal{L}}} + 1/2$$

5. While the one-to-one problem is naturally related to the radius of a network, the all-to-all problem is related to its diameter. By using the fact that the diameter lies between the radius and twice the radius, and using the approximation to $Rad(N, \mathcal{L}\,)$ as discussed above, we manage to significantly improve results regarding the all-to-all problem, presented in [16,17,1]; for a detailed discussion, see [6,8]).

8 Discussion

We showed how duality properties and geometric considerations are used in studies related to virtual path layouts of chain ATM networks. These dualities follow immediately from the recurrence relations, but a clearer insight was gained with the aid of binary trees (in the shortest path case) and ternary trees (in the general path case). For the general path case we also presented the relation with high dimensional spheres. The duality nature of the solutions, together with the geometric approach, proved to be extremely useful tools in understanding and analyzing the optimal designs. We managed to simplify proofs of known results, derive new results, and improve existing ones.

It might be of interest to further explore such duality relations in various directions. This can be done either for related parameters (such as load measured at vertices, as discussed in [11,9]), or for other topologies (such as trees ([5,11]), meshes ([3,2,11]), or planar graphs ([11])). An interesting direction for extension is suggested for directed networks, following [4]. One might also consult the survey in [18] for a general discussion of these and other extensions.

Of special interest was the use of the high dimensional spheres. The discussion of the use of these spheres and of the applications of this embedding technique suggest this as a promising direction for a further investigation.

References

1. W. Aiello, S. Bhatt, F. Chung, A. Rosenberg, and R. Sitaraman, *Augmented rings networks*, *11th Intl. Conf. on Math. and Computer Modelling and Scientific Computing (ICMCM & SC)* (1997); also: Proceedings of the *6th International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, Lacanau-Océan, France, 1999, pp. 1-16.
2. L. Becchetti, P. Bertolazzi, C. Gaibisso and G. Gambosi, *On the design of efficient ATM routing schemes*, submitted, 1997.
3. L. Beccheti and C. Gaibisso, *Lower bounds for the virtual path layout problem in ATM networks*, *Proceedings of the 24th Seminar on Theory and Practice of Informatics (SOFSEM)*, Milovny, The Czech Republic, November 1997, pp. 375-382.
4. J-C. Bermond, N. Marlin, D. Peleg and S. Pérennes, *Directed virtual path layout in ATM networks*, *Proceedings of the 12th International Symposium on Distributed Computing (DISC)*, Andros, Greece, September 1998, pp. 75-88.
5. I. Cidon, O. Gerstel and S. Zaks, *A scalable approach to routing in ATM networks. 8th International Workshop on Distributed Algorithms (WDAG)*, Lecture Notes in Computer Science 857, Springer Verlag, Berlin, 1994, pp.209-222.
6. Y. Dinitz, M. Feighelstein and S. Zaks, *On optimal graph embedded into path and rings, with analysis using l_1-spheres*, *23th International Workshop on Graph-Theoretic Concepts in Computer Sciences (WG)*, Berlin, Germany, June 1997.
7. T. Eilam, M. Flammini and S. Zaks, *A Complete Characterization of the Path Layout Construction Problem for ATM Networks with Given Hop Count and Load*, Proceedings of the *24th International Colloquium on Automata, Languages and Programming (ICALP)*, Bologna, Italy, pp. 527-537, July 1997.
8. M. Feighelstein, *Virtual path layouts for ATM networks with unbounded stretch factor*, , M.Sc. Dissertation, Department of Computer Science, Technion, Haifa, Israel, May 1998.
9. M. Flammini, E. Nardelli and G. Proietti, *ATM layouts with bounded hop count and congestion*, *Proceedings of the 11th International Workshop on Distributed Algorithms (WDAG)*, Saarbrüecken, Germany ,September 1997, pp. 24-26.
10. M. Feighelstein and S. Zaks, *Duality in chain ATM virtual path layouts*, Proceedings of the *4th International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, Monte Verita, Ascona, Switzerland, July 24-26, 1997, pp. 228-239.
11. O. Gerstel, *Virtual Path Design in ATM Networks*, Ph.D. thesis, Department of Computer Science, Technion, Haifa, Israel, December 1995.
12. S. W. Golomb and L. R. Welch, *Perfect Codes in the Lee Metric and the Packing of Polyominoes. SIAM Journal on Applied Math.*,vol.18,no.2, January, 1970, pp. 302-317.
13. O. Gerstel, A. Wool and S. Zaks, *Optimal layouts on a chain ATM network*, *Discrete Applied Mathematics* , special issue on Network Communications, 83, 1998, pp. 157-178.
14. O. Gerstel, A. Wool and S. Zaks, *Optimal Average-Case Layouts on Chain Networks*, Proceedings of the *Workshop on Algorithmic Aspects of Communication*, Bologna, Italy, July 11-12, 1997.
15. O. Gerstel and S. Zaks, *The Virtual Path Layout Problem in Fast Networks*, Proceedings of the *13th ACM Symposium on Principles of Distributed Computing (PODC)*, Los Angeles, CA, U.S.A., August 1994, pp. 235-243.

16. E. Kranakis, D. Krizanc and A. Pelc, *Hop-congestion tradeoffs for ATM networks, 7th IEEE Symp. on Parallel and Distributed Processing*, pp. 662-668.
17. L. Stacho and I. Vrt'o, *Virtual Path Layouts for Some Bounded Degree Networks. 3rd International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, Siena, Italy, June 1996.
18. S. Zaks, *Path Layout in ATM Networks*, Proceedings of the *24th Annual Conference on Current Trends in Theory and Practice of Informatics (SOFSEM)*, Lecture Notes in Computer Science 1338, Springer Verlag, Milovy, The Czech Republic, November 22-29, 1997, pp. 144-160.