On the design of efficient ATM routing schemes *

Luca Becchetti[†] Paola Bertolazzi[‡] Carlo Gaibisso[§]
Giorgio Gambosi[¶]

Abstract
In this paper we deal with the problem of designing virtual path layouts in ATM networks when the hop-count is given and the load has to be minimized. We first prove a lower bound for networks with arbitrary topology and arbitrary set of connection requests. This result is then applied to derive lower bounds for the following settings: i) one-to-all (one node has to be connected to all other nodes of the network) in arbitrary networks; ii) all-to-all (each node has to be connected to all other nodes in the network) in several classes of networks, including planar and k-separable networks and networks of bounded genus. We finally study the all-to-all setting on two-dimensional meshes and we design a virtual path layout for this problem. When the hop-count and the network degree are bounded by constants, our results show that the upper bounds proposed in this paper for the one-to-all problem in arbitrary networks and for the all-to-all problem in two-dimensional mesh networks are asymptotically optimal. Moreover, the general lower bound shows that the algorithm proposed in ([10]) for the all-to-all problem in k-separable networks is also asymptotically optimal. The upper bound for mesh networks also shows that the lower bound presented in this paper for the all-to-all problem in planar networks is asymptotically tight.

Key Words: ATM networks, Parallel and Distibuted Computation, Virtual Path Layout

1 Introduction

Asynchronous Transfer Mode (ATM) ([16], [4], [8]) is the most common network protocol under consideration for the very fast Broadband Integrated Service

*Work partially performed in the framework of Esprit Long Term Research Program project "ALCOM-IT" and of Italian MURST 40% project "Algoritmi, Modelli di Calcolo e Strutture Informative"

[†]Istituto di Analisi dei Sistemi ed Informatica del CNR, Viale Manzoni 30, 00185 Roma, Italy, and Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza", Via Salaria 113, 00184 Rome, Italy. (becchett@dis.uniroma1.it)

[‡]Istituto di Analisi dei Sistemi ed Informatica del CNR, Viale Manzoni 30, 00185 Roma, Italy. (bertola@iasi.rm.cnr.it)

[§]Istituto di Analisi dei Sistemi ed Informatica del CNR, Viale Manzoni 30, 00185 Roma, Italy. (gaibisso@iasi.rm.cnr.it)

[¶]Dipartimento di Matematica, Università di Roma "Tor Vergata", via della Ricerca Scientifica, 00133 Rome, Italy. (gambosi@mat.uniroma2.it)

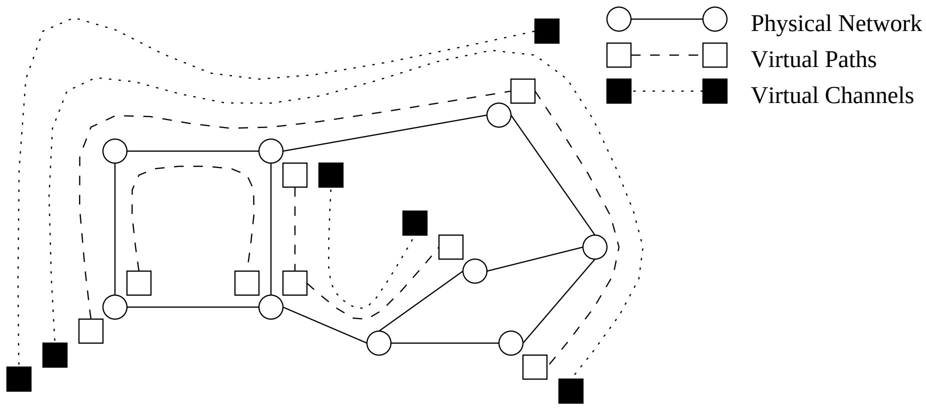

Figure 1: a virtual path layout

Digital Networks of the future (B-ISDN). Due to this fact it has been studied intensively in the recent past.

In ATM, the transfer of data is based on packets of fixed length, called *cells*; in order to achieve the stringent transfer rate requirements, cell routing must be accomplished by dedicated hardware implementing very simple algorithms.

If we describe the physical network through a graph $G = (V, E)$, where V represents the set of switches and end-users, and E represents that of the physical links, an ATM routing scheme is based on two types of predetermined routes in G: *virtual paths* and *virtual channels*. Virtual paths are simple paths in the network, while virtual channels are obtained from the concatenation of a certain number of virtual paths, so that each of them may bundle together several virtual channels sharing parts of their routes. A *connection pattern* is a set of pairs of nodes in the network; each node of the pair represents an end-user, the two nodes of the pair have to be connected by a virtual channel.

In order to transfer data along a virtual channel, each cell header contains two routing labels, respectively named *virtual path identifier* (VPI) and *virtual channel identifier* (VCI), while each node in the network contains two routing tables, the *virtual path table* (VP table) and *virtual channel table* (VC table) respectively. The VPI and the VCI are used as indexes to the corresponding tables in order to identify the next link of the physical network the cell has to be routed on. In particular, routing is hierarchical, in the sense that it is performed according to the VPI, in an essentially hardware fashion, except when this has a conventional NULL value: in this case, the VCI is used to index the VC table; this operation can be much slower, having to be performed in a software fashion in realistic networks. The idea behind is that routing is performed according to the VPI in most of the nodes a cell traverses, thus being faster then in conventional networks, where more elaborate processing, involving software, is normally needed. An example of virtual path layout is presented in fig. 1.

Given a network and a connection pattern, the general Virtual Path Layout problem is that of defining the set of virtual paths in such a way that certain properties are achieved:

1. the number of virtual paths using the same physical link, denoted as *load* in what follows, should be kept as low as possible, so as not to exceed the capacity of the routing tables, which is fixed by the ATM protocol, and in order to allow a possibly fast recovery from faults;

2. the maximum number of virtual paths in a virtual channel, denoted as *hop count* in what follows, should be kept as low as possible, so as to guarantee low set up times for the virtual channels and low end-to-end latencies.

Particular cases of this problem have been extensively studied. They are the *one-to-all* virtual path layout, VPL^{1-a} in the following, where a single node has to be connected to all other nodes in the network, and the *all-to-all* virtual path layout, VPL^{a-a} in what follows, where a connection has to be established between each pair of nodes in the network.

Both these problems are of great practical relevance. In particular, VPL^{1-a} models the problem of setting up virtual path layouts in client-server networks, where data flows from a single source, the server, to multiple destinations, the clients, and viceversa [17]. As to VPL^{a-a}, it represents the worst possible connection pattern in terms of the number of connections to serve and therefore the most critical with respect to the load.

The corresponding decision problem, which asks whether a system of virtual paths with given maximum hop count and load exists, has been shown to be NP-complete for arbitrary network topologies already in the one-to-all case (see [6] and [12] for a complete characterization).

Polynomial algorithms have been proposed for some common networks in [3], [2], [10] and [11].

In particular, VPL^{1-a} is studied in chain networks in [10], while VPL^{a-a} in the same network is considered in [3], [2], [10] and [11]. Both problems are also studied for tree networks and two-dimensional meshes in [2], [3] and [10]. In these papers both upper and lower bounds for the above problems are proposed; upper bounds are asymptotically optimal for chains and for tree networks, when the maximum degree and the hop-count are bounded by constants.

In [2] and [10], an algorithm for VPL^{a-a} in k-separable networks is proposed. We prove in Section 4 that this algorithm is asymptotically optimal when the hop-count and the network degree are bounded by constants.

In [9], the problem of minimizing the load while preserving a good fault tolerance is addressed with respect to complete graphs.

In [14] and [21] the problem is investigated from a different point of view: the load is fixed, while the hop count has to be minimized. In particular, in [14] the authors derive a lower bound for the hop count with respect to arbitrary network topologies and design virtual path layouts for mesh and chain networks; the proposed upper bounds are asymptotically optimal when the network degree is bounded by a constant.

It is worthwhile noting that our approach and the one presented in [14] and [21] are equivalent. In particular, the general lower bound obtained in [14] can be reverted to yield a lower bound for the case in which the hop-count is fixed and the load has to be minimized. However, applied to the setting we are considering, this lower bound is not so sharp as the one obtained in this paper.

As to the model we consider and as far as lower bounds are concerned, the only results known to the authors so far concern chain, ring and tree networks (see [23] for a recent survey about the state of the art).

This paper is organized as follows: in Section 2 the problem is modelled from the graph-theoretical point of view.

In Section 3 a general lower bound for the virtual path layout problem, with respect to arbitrary connection patterns and networks is derived. This result depends on both the network topology and the connection pattern. The general result is then applied to derive lower bounds for both VPL^{1-a} and VPL^{a-a} in some cases of interest (Section 4). In particular, we derive a lower bound for VPL^{1-a} in arbitrary networks (which is also a lower bound for VPL^{a-a}) and we obtain specific lower bounds for VPL^{a-a} in some important classes of networks, among which are planar and k-separable networks and networks of bounded genus. It follows from our result that, if the hop-count and the network degree are bounded by constants, an asymptotically optimal algorithm exists for VPL^{1-a} in general networks. This algorithm is easily derivable from the one proposed in [3] and [10] for tree networks.

In Section 5 we first characterize the cost of VPL^{a-a} in two-dimensional mesh networks and we then design a layout for this problem, which is asymptotically optimal when the hop-count is bounded by a constant. This proves that the lower bound obtained for planar networks is asymptotically tight.

Finally, in Section 6, we draw some conclusions by summarizing and commenting the results obtained in the paper.

2 The mathematical model

An instance of the virtual path layout problem can be described by a triple (G, U, h), where $G = (V, E)$ is a connected graph representing the physical network, with V representing the set of network switches and end-users and E modelling the set of physical links; $U \subseteq V \times V$ is a set connection requests; h is a positive integer denoting the maximum hop count.

The following definitions formalize the concept of virtual path layout:

Definition 1 *A virtual path layout Φ for (G, U, h) is a pair (G_Φ, I_Φ), where: G_Φ is a graph (V, E_Φ) such that for each $(p, q) \in U$ there exists at least one simple path of length at most h in G_Φ connecting p and q and $I_\Phi : E_\Phi \to \mathcal{S}(G)$, where $\mathcal{S}(G)$ is the set of simple paths in G, is an injective function assigning to each edge $(p, q) \in E_\Phi$ a simple path in G connecting p to q.*

Definition 2 *For any virtual path layout Φ for (G, U, h), a simple path π in G is called a virtual path, if and only if an edge $e \in E_\Phi$ exists such that $\pi = I_\Phi(e)$.*

Definition 3 *For any virtual path layout Φ for (G, U, h), a simple path γ in G_Φ is called a virtual channel if and only if it connects any pair of nodes p and q such that $(p, q) \in U$.*

Notice that any virtual channel connecting p and q can also be considered as a, not necessarily simple, path from p to q in G.

Definition 4 *For any link $l \in E$, the load of l with respect to any virtual path layout Φ for (G, U, h) is $L_\Phi(l) = |\{e \in E_\Phi | l \in I_\Phi(e)\}|$.*

Definition 5 *The load of any virtual path layout Φ for (G, U, h) is $L_\Phi = \max_{l \in E_\Phi}\{L_\Phi(l)\}$.*

As stated above, in this paper we consider the specific problem which, given (G, U, h), asks for a virtual path layout Φ with minimum load. In particular, we concentrate on the following subproblems:

- the *one-to-all* virtual path layout problem, VPL^{1-a}, where a single node has to be connected to all other nodes in the graph, that is $U = \{p\} \times V$, for some $p \in V$. In what follows an instance of the VPL^{1-a} will be simply denoted by $(G, \{v\}, h)$;
- the *all-to-all* virtual path layout problem, VPL^{a-a}, where all pairs of nodes in the graph have to be connected, that is $U = V \times V$. In what follows an instance of the VPL^{a-a} will be simply denoted by (G, h).

3 Lower bounds for general networks

In this section we derive a general lower bound for the load in the virtual path layout problem, which holds for arbitrary networks and connection patterns.

In what follows, given any graph $G = (V, E)$, $\Delta(G)$ denotes its maximum degree[1].

For any instance (G, U, h) of the virtual path layout problem, let $C \subseteq E$ be an edge-cut which induces a partition (V_1, V_2) of V, i.e. $(u, v) \in C$ if and only if $(u, v) \in E$, with $u \in V_1$ and $v \in V_2$. Let N_C be the number of pairs $(p, q) \in U$ such that $p \in V_1$ and $q \in V_2$.

Lemma 3.1 *For any (G, U, h) and any virtual path layout Φ for (G, U, h) with load L, at most $(h(\Delta L)^{h+1} + 1)/(\Delta L - 1)^2$ virtual channels of Φ may include the same virtual path.*

Proof. Let π_j be a virtual path of Φ. The maximum number of virtual channels including π_j can be derived as follows:

[1] In order to simplify the notation, L and Δ are used instead of L_Φ and $\Delta(G)$ whenever possible.

1. for any k and for any i, $k \leq h$, $1 \leq i \leq k$, π_j appears as the i-th virtual path in at most $(\Delta L)^{i-1} \cdot (\Delta L)^{k-i} = (\Delta L)^{k-1}$ virtual channels of length k. In fact, let p_j and q_j be the endpoints of π_j. Node p_j is the endpoint of at most ΔL virtual paths of Φ different from π_j. By iterating this consideration $i-1$ times, at most $(\Delta L)^{i-1}$ different simple paths of length $i-1$ exist in G_Φ which are rooted at p_j. In the same way it is possible to see that at most $(\Delta L)^{k-i}$ different simple paths of length $k-i$ exist in G_Φ which are rooted at q_j, thus proving the assumption;

2. consequently, the maximum number of virtual channels of length equal to k including π_j is $\mathcal{C}_k \leq \sum_{i=1}^{k} (\Delta L)^{k-1} = k(\Delta L)^{k-1}$;

3. finally, the maximum number of virtual channels of length at most h including π_j is

$$\overline{\mathcal{C}_h} \leq \sum_{k=1}^{h} \mathcal{C}_k = \sum_{k=1}^{h} k(\Delta L)^{k-1} = \sum_{k=0}^{h} k(\Delta L)^{k-1} =$$
$$< \frac{h(\Delta L)^{h+1}}{(\Delta L - 1)^2},$$

which proves the lemma. □

Let us now derive the general lower bound:

Theorem 3.1 *For any instance (G, U, h) of the virtual path layout problem and any virtual path layout Φ for (G, U, h), if C is an edge-cut that separates N_C pairs in U, then $L > \frac{0.7}{\Delta} \sqrt[h]{\frac{\Delta}{4} \frac{N_C}{|C|}}$.*

Proof. Let us first derive the maximum number $\overline{\mathcal{C}_h}(e)$ of virtual channels of length at most h in G_Φ that include the same physical edge e in G. By lemma 3.1 and since e is included in at most L virtual paths

$$\overline{\mathcal{C}_h}(e) \leq L\overline{\mathcal{C}_h} < L\frac{h(\Delta L)^{h+1}}{(\Delta L - 1)^2} = \frac{\Delta L}{\Delta} \frac{h(\Delta L)^{h+1}}{(\Delta L - 1)^2}.$$

Since $\Delta L \geq 2$ in all cases of practical relevance, we have $\Delta L - 1 \geq \frac{\Delta L}{2}$ and

$$\overline{\mathcal{C}_h}(e) \leq \frac{4}{\Delta} \frac{h(\Delta L)^{h+1}}{\Delta L} < \frac{4}{\Delta} h(\Delta L)^h = 4h\Delta^{h-1}L^h.$$

Given any edge-cut C for G, the number of virtual channels which are disconnected by C is then bounded by $\sum_{e \in C} \overline{\mathcal{C}_h}(e) < |C| 4h\Delta^{h-1}L^h$. Since N_C pairs in U are separated by C, the following inequality must hold: $|C|4h\Delta^{h-1}L^h > N_C$, which implies that for at least one edge e of G, $L(e) > \frac{1}{\Delta} \sqrt[h]{\frac{\Delta}{4h} \frac{N_C}{|C|}}$. Finally, since $0.7 < \sqrt[h]{\frac{1}{h}} \leq 1$, the thesis follows. □

This result suggests a technique for deriving lower bounds for the virtual path layout problem. In fact, using the general lower bound presented in this section, the problem becomes the one of finding an edge-cut that disconnects the graph into two parts and such that the ratio of the number of edges removed to that of the disconnected pairs is minimized. This approach is particularly interesting for VPL^{1-a} and VPL^{a-a}. In the former case, it allows us to derive a lower bound which holds for arbitrary networks (see Subsection 4.1), while in the latter we can derive lower bounds for many classes of networks, including networks of practical relevance, using edge-separators (see Subsection 4.2). Finally, we notice that $N_C/|C|$ is also known as an edge expanding factor [19, 20].

4 Some consequences of the general lower bound

In this subsection, the general lower bound derived in Section 3 is applied to derive more specific lower bounds for VPL^{1-a} in arbitrary networks and for VPL^{a-a} in some important classes of networks. To the latter purpose we use some results on edge separators. As to the former result, it allows us to state that a straightforward algorithm for VPL^{1-a} in arbitrary networks is asymptotically optimal when the network degree and the hop-count are bounded by constants.

4.1 One-to-all in general networks

As an immediate corollary of Theorem 3.1 it is possible to derive the following lower bound for VPL^{1-a}, which obviously applies to VPL^{a-a} as well.

Corollary 4.1 *For any instance $(G, \{v\}, h)$ of VPL^{1-a} and for any virtual path layout Φ for $(G, \{v\}, h)$, $L_\Phi > \frac{0.7}{\Delta}\sqrt[h]{\frac{|V|-1}{4}}$*

Proof. Take C as the set of edges incident to v. In this case $|C| \leq \Delta$ and $N_C = |V| - 1$. □

Let us now derive a simple algorithm which solves any instance $(G, \{v\}, h)$ of VPL^{1-a}. To this purpose, we use the algorithm proposed in [3] for optimal VPL^{1-a} in tree networks as a subroutine. For any n-vertex tree network of maximum degree Δ and for any fixed value h of the hop-count, this algorithm yields a virtual path layout with load at most $h\sqrt[h]{n}$, as shown in [10].

Our algorithm works as follows:

Algorithm $GenVPL^{1-a}$

Input: G, v, h

1. Find a spanning tree in G rooted at v
2. Solve the instance $(T_G, \{v\}, h)$ of VPL^{1-a} using the optimal algorithm presented in [3]

Corollary 4.2 *For any n-vertex network G of degree Δ, any root node v and for any fixed value h of the hop-count, algorithm $GenVPL^{1-a}$ solves the instance $(G, \{v\}, h)$ of VPL^{1-a} with load at most $h\sqrt[h]{n}$; when Δ and h are bounded by constants, the upper bound is asymptotically tight.*

Proof.

The load is exactly the same as that guaranteed by the algorithm proposed in [3] for tree networks. This fact and corollary 4.1 imply that the load is asymptotically optimal when Δ and h are bounded by constants.

□

Observe that the lower bound presented in Corollary 4.1 is obviously also a lower bound for VPL^{a-a} in arbitrary networks. The only other lower bound known to the authors for VPL^{a-a} in general networks is the one presented in [14], which refers to the case when the load is fixed while the hop-count has to be minimized. This bound is asymptotically worse than the one derived in Corollary 4.1 in the case we are considering, when h is fixed and L has to be minimized.

4.2 Lower bounds for VPL^{a-a} in some common classes of networks

We first give the following definitions.

Definition 6 **([5])** *Given an n-vertex graph G, G has an $f(n)$-*edge separator *if there exists a set C of edges and a partition of the vertices into two sets A and B, such that $|C| \leq f(n)$, $|A|$, $|B| \leq 2n/3$ and each edge joining a vertex in A to a vertex in B belongs to C.*

Definition 7 **[15]** *The bisection width of a graph $G = (V, E)$ is the minimum number of edges which have to be removed in order to disconnect G into two halves with same (within one) number of vertices.*

Observe that the definition of edge-separator implies that $|A|$, $|B| \geq n/3$.

Planar graphs

In deriving a lower bound for the load of VPL^{a-a} in planar networks, we rely on the following result

Theorem 4.1 **([5])** *Let G be an n-vertex planar graph of maximum degree Δ. Then G has an edge separator of size bounded by $2\sqrt{2\Delta n}$.*

The next theorem is an immediate consequence of theorem 4.1 and of theorem 3.1.

Theorem 4.2 *For any instance (G, h) of VPL^{a-a}, where $G = (V, E)$ is an n-vertex planar graph, and for any virtual path layout for (G, h),*
$L = \Omega\left(\frac{1}{\Delta}\sqrt[h]{\Delta^{1/2}n^{3/2}}\right)$. *In particular,* $L = \Omega\left(\sqrt[2h]{n^3}\right)$ *when Δ is bounded by a constant.*

Proof. The theorem follows directly from the application of theorems 3.1 and 4.1. In fact, in the case of planar graphs theorem 4.1 implies $|C| \leq 2\sqrt{2\Delta n}$, while the definition of separator implies that $N_C \geq \frac{n^2}{9}$. □

A tighter bound holds when the graph is *outerplanar*. We recall that a graph is outerplanar if it has a planar embedding with all vertices on the exterior face. The result of Heath for the two-page embedding of outerplanar graphs [13] implies that there exist outerplanar graphs with bisection width $O(1)$. This allows us to state

Theorem 4.3 *There exist instances (G, h) of VPL^{a-a}, where G is outerplanar, such that, for any virtual path layout for (G, h),*
$L = \Omega\left(\frac{1}{\Delta}\sqrt[h]{\Delta n^2}\right)$.

Graphs with bounded genus

A result similar to that obtained for planar graphs can be obtained for graphs of bounded genus (see [22] and [20]). The *genus* of a graph is the minimum number of handles that must be added to a sphere so that the graph can be embedded with no crossing edges. Let g be the genus of an n-vertex graph $G = (V, E)$ and let Δ be its maximum degree. Furthermore, let the vertices of G have weights summing to 1 and such that no weight exceeds $2/3$. Finally, for any $X \subseteq V$, let $weight(X)$ denote the sum of the weights of vertices to X.

The following result holds

Theorem 4.4 **([22])** *Given an n-vertex graph $G = (V, E)$ of genus g, there exist a partition of V into sets A, B and a set D of edges such that $weight(A)$, $weight(B) \leq 2/3$, $|D| \leq 5\sqrt{3g\Delta n}$ and every edge between nodes in A and B belongs to D.*

If we assign each vertex in G the same weight $1/n$, theorem 4.4 can be restated this way:

Proposition 4.1 *Given an n-vertex graph $G = (V, E)$ of genus g, there exist a partition of V into sets A, B and a set D of edges such that $|A|$, $|B| \leq 2/3n$, $|D| \leq 5\sqrt{3g\Delta n}$ and every edge between nodes in A and B belongs to D.*

Proof.
Since, from theorem 4.4, $weight(A), weight(B) \leq 2/3$ and each vertex in V has weight $1/n$, the thesis follows immediately.

□

Theorems 3.1 and 4.1 straightforwardly imply the following result

Theorem 4.5 *For any instance* (G, h) *of* VPL^{a-a}*, where* G *is an* n*-vertex graph of genus* g *and for any virtual path layout* Φ *for* (G, h)*,*
$L = \Omega\left(\frac{1}{\Delta}\sqrt[h]{\frac{\Delta^{1/2}n^{3/2}}{g^{1/2}}}\right)$. *In particular,* $L = \Omega\left(\sqrt[2h]{n^3}\right)$ *when* Δ *and* g *are bounded by constants.*

Proof. The proof goes the same way as for planar graphs, the separator C is in this case that of theorem 4.1 and has therefore size $|C| \leq 5\sqrt{3g\Delta n}$, while $|A|, |B| \leq 2/3n$ imply $|A|, |B| \geq n/3$ and hence $N_C \geq \frac{n^2}{9}$. □

The same (asymptotic) bound can be obtained using the result for the edge forwarding index in graphs of bounded genus contained in [20].

Graphs with bounded tree-width

We consider the important class of the graphs with bounded tree-width, also said k-separable. This class includes many popular topologies, such as arrays, rings, trees and trees of rings, and is consequently of great theoretical and practical interest. Let us first recall a few basic definitions ([1]):

Definition 8 *A tree decomposition of a graph* $G = (V, E)$ *is a pair* $(\mathcal{X}, T)$*, where* $\mathcal{X} = \{X_p | p \in V_T\}$ *is a family of subsets of* V*, one for each node of the tree* $T = (V_T, E_T)$*, such that:*

- $\cup_{p \in V_T} X_p = V$*;*
- *for all edges* $(u, w) \in E$*, there exists a node* $p \in V_T$ *with* $v, w \in X_p$*;*
- *for all* $p, q, r \in V_T$*: if* q *is on the path from* p *to* r *in* T*, then* $X_p \cap X_r \subseteq X_q$*.*

Definition 9 *Given a graph* G *and a tree-decomposition* $(\mathcal{X}, T)$ *for* G*, the width of* $(\mathcal{X}, T)$ *is defined as* $\max_{p \in V_T}\{|X_p| - 1\}$*. The tree-width of* G *is the minimum width over all tree-decompositions of* G*.*

The following theorem holds for graphs with bounded tree-width (see [1]):

Theorem 4.6 *For any* n*-vertex graph* $G = (V, E)$ *with tree-width at most* k*, there exists a node separator* F *for* G*, such that* $|F| \leq k + 1$ *and* F *breaks* G *into connected components whose size is not greater than* $\frac{1}{2}(n - k)$*.*

It is now possible to state the following lemma:

Lemma 4.1 *For any* n*-vertex graph* $G = (V, E)$ *with tree-width at most* k *and degree* Δ*, there exists an edge-cut* C *for* G*, such that* $|C| \leq (k + 1)\Delta$ *and* C *separates at least* $\geq \frac{1}{12}(n - k)^2$ *pairs of nodes in* G*.*

Proof. By theorem 4.6 and since the tree-width of G is bounded by k, there exists a node separator F such that $|F| \leq k + 1$. Let us assume that G is broken by F into l connected components $G_i = (V_i, E_i), i = 1, \cdots, l$. Let $m = \max_{i=1,\ldots,l}\{s \in \mathcal{N} \mid |V_i| \leq \frac{1}{s}(n - k)\}$. By theorem 4.6, $s \geq 2$. We consider two different cases:

1. $m = 2$, then for at least one $i \in \{1, \cdots, l\}$, $|V_i| > \frac{1}{3}(n-k)$. Choose $A = V_i$ and $B = V - A$, thus $\frac{1}{3}(n-k) < |A| \leq \frac{1}{2}(n-k)$ and therefore $\frac{1}{2}(n-k) \leq |B| \leq \frac{2}{3}(n-k)$.

   Consequently at least one edge cut C will exist of size at most $\Delta|F| \leq (k+1)\Delta$, such that C separates at least $\frac{1}{3}(n-k) \cdot \frac{1}{2}(n-k) = \frac{1}{6}(n-k)^2$ pairs (p, q) in G with $p \in A$ and $q \in B$.

2. $m \geq 3$, then choose A as the smallest subset of V obtained by the union of subsets in $\{V_1, \cdots, V_l\}$ and such that $|A| \geq \frac{1}{2}(n-k)$.

   Note that, since in this case $|V_i| \leq \frac{1}{3}(n-k)$, $\forall\ 1 \leq i \leq l$, $|A| \leq \frac{1}{2}(n-k) + \frac{1}{3}(n-k) = \frac{5}{6}(n-k)$. Thus, if we set $B = V - A$, $|A| \geq \frac{1}{2}(n-k)$ and $|B| \geq \frac{1}{6}(n-k)$ for all $m \geq 3$.

   By the same reasoning followed for the case $m = 2$, it is possible to conclude that there exists at least an edge-cut C for G, such that $|C| = \Delta|F| \leq (k+1)\Delta$ and C separates at least $\frac{1}{12}(n-k)^2$ pairs (p, q) in G with $p \in A$ and $q \in B$, thus proving the lemma.

□

The next theorem is an immediate consequence of lemma 4.1 and of theorem 3.1:

Theorem 4.7 *Given any instance (G, h) of VPL^{a-a}, where $G = (V, E)$ is an n-vertex graph with bounded tree-width at most k, and for any virtual path layout for (G, h),*

$$L = \Omega\left(\frac{1}{\Delta}\sqrt[h]{n^2}\right).$$

Proof. From lemma 4.1, we can apply theorem 3.1 with $|C| \leq k+1$ and $N_C \geq \frac{1}{12}(n-k)^2$. The claim then follows immediately. □

Theorem 4.7 also proves that the algorithm proposed in [2] and [10] for VPL^{a-a} in networks with bounded tree-width is asymptotically optimal when the hop-count and the degree are bounded by constants.

Some common interconnection networks

We consider here some popular topologies that are of interest in parallel and distributed computing. We only give the formal definition of the r-dimensional N-sided array, since this is necessary to the presentation of our results. The reader can refer to [18], [15] and [15] for the definitions of the other graphs considered in this subsection. Since the results presented here are straightforward applications of theorem 3.1 we only give a proof for the first one, the others being proved exactly the same way.

An *r-dimensional N-sided array* [15] has $n = N^r$ vertices. Each vertex corresponds to an N-ary r-vector, $(v_1, \ldots, v_r)$, where $1 \leq i_j \leq N$, for every

$j = 1, \ldots, r$. Two vertices share an edge if they differ in exactly one coordinate and the absolute value in the difference in that coordinate is 1.

It is well known [15] that the bisection width of the r-dimensional N-sided array is $O(N^{r-1})$ in general (exactly N^{r-1} when N is even) and that the degree of any vertex is between r and $2r$. This allows us to state the following

Theorem 4.8 *Given any instance (G,h) of VPL^{a-a}, where G is the r-dimensional, N-sided array and for any virtual path layout for (G,h),*

$$L = \Omega\left(\frac{1}{r}\sqrt[h]{rN^{r-1}}\right).$$

Proof. The result follows from theorem 3.1, recalling that in this case we have $N_C = \Omega(N^{2r-2})$, $r \leq \Delta \leq 2r$ and $|C| = O(N^{r-1})$. □

In the particular case $N = 2$ we have the *r-dimensional hypercube*

Corollary 4.3 *Given any instance (G,h) of VPL^{a-a}, where G is the r-dimensional hypercube and for any virtual path layout for (G,h),*

$$L = \Omega\left(\frac{1}{r}\sqrt[h]{r2^{r-1}}\right).$$

A slightly better bound ($\Omega\left(\frac{1}{r}\sqrt[h]{\sqrt{r}2^r}\right)$) can be obtained from the results of [19] for the vertex expanding factor in the hypercube ([19] and [20]), considering its relationship to the edge expanding factor. It is interesting to notice that the large bisection width of these networks causes the lower bound to decrease, so that it eventually becomes $O(1)$ as h becomes $\Omega(r \log_r N)$.

Let us now turn to the *Butterfly*, the *Cube-Connected-Cycles* and the *Shuffle-Exchange* graphs. These networks are all derived from the hypercube and all have bisection width $\Theta(n/\log n)$, if n denotes the number of vertices. Furthermore, $\Delta = 4$ for the Butterfly and the Cube-Connected-Cycles, $\Delta = 3$ in the Shuffle-Exchange. These graphs, their definitions and properties are fully addressed in [15].

Theorem 4.9 *Given any instance (G,h) of VPL^{a-a}, where G is either the n-vertex Butterfly, Cube-Connected-Cycles or Shuffle-Exchange network and for any virtual path layout for (G,h),*

$$L = \Omega\left(\sqrt[h]{n \log n}\right).$$

The best bisection width of the n-vertex De Bruijn graph [18] is at most $4n/\log n$ and tends asymptotically to $2\ln(2)n/\log n$ [7]. For the Kautz graph [18] we have that it is $O(n/\log n)$. For both graphs we have $\Delta = 4$. This allows us to state:

Theorem 4.10 *Given any instance (G,h) of VPL^{a-a}, where G is either the n-vertex De Bruijn or Kautz graph and for any virtual path layout for (G,h),*

$$L = \Omega\left(\sqrt[h]{n \log n}\right).$$

5 Virtual Path Layout in mesh networks

In this section we provide a virtual path layout for VPL^{a-a} in two-dimensional mesh networks.

Let us first prove the following lower bound:

Theorem 5.1 *For any instance (M,h) of VPL^{a-a}, where M is an $n_1 \times n_2$ mesh, and for any virtual path layout for (M,h),*

$$L = \Omega\left(\sqrt[h]{\frac{\max\{n_1,n_2\}^2 \cdot \min\{n_1,n_2\}}{h}}\right).$$

In particular, if $n_1 = n_2 = n$ and h is bounded by a constant then $L = \Omega\left(\sqrt[h]{n^3}\right)$.

Proof. W.l.o.g., let us assume $n_1 \leq n_2$. In Theorem 3.1, choose C as the set of edges between the $\lfloor \frac{n_2}{2} \rfloor$-th and the $(\lfloor \frac{n_2}{2} \rfloor + 1)$-th columns. Consequently, $\mid C \mid = n_1$ and $N_C = n_1^2 \lfloor \frac{n_2}{2} \rfloor \lceil \frac{n_2}{2} \rceil$, thus proving the theorem. $\square$

Notice that this lower bound is aysmptotically the same as the general lower bound for planar networks of bounded degree.

Let us now consider the problem of constructing effective virtual path layouts for mesh networks. The layout we propose is asymptotically optimal, with respect to the load, when h is bounded by a constant. This shows that there exist planar graphs for which the lower bound derived in Section 4 is asymptotically tight, when h is bounded by a constant.

For the sake of simplicity, in the following we consider $n \times n$ meshes; the construction can be easily extended to rectangular ones. Furthermore, in what follows, given any two-dimensional mesh M, $v_M(i,j)$ denotes the node at the intersection of the i-th row and the j-th column. For simplicity of notation, we write $v(i,j)$ when M is clear from the context.

The algorithm works in two phases (see figg. 2 (a) and 2 (b)). In the first phase M is partitioned into n submeshes of size $\sqrt{n} \times \sqrt{n}$, on each of which a VPL^{1-a} from a certain node (termed *representative* in the following) is defined, with hop-count at most $\frac{h}{3}$ and load bounded above from $\frac{h}{3}\sqrt[h]{n^3}$. This can be done using algorithm 4.1 of Section 4. In the second phase a VPL^{a-a} restricted to the set of representatives is defined, with hop-count at most $\frac{h}{3}$ and load bounded by $\left(\frac{h}{3}+2\right)\sqrt[h]{n^3}$. From the combination of these two layouts a VPL^{a-a} on M with hop-count at most h and load bounded by $\leq 2\left(\frac{h}{3}+1\right)\sqrt[h]{n^3}$ results.

VPL^{a-a} restricted to the set of representatives is solved by reduction to the instance $(M,U,h/3)$ of the virtual path layout problem, where $U = \{v(i,1)|1 \leq i \leq n\} \times \{v(j,n)|1 \leq j \leq n\}$, that is, to the problem of connecting each node in the leftmost column of M to all nodes in the rightmost column in at most $h/3$ hops. A virtual path layout for the above problem with hop-count h is described in the next lemma.

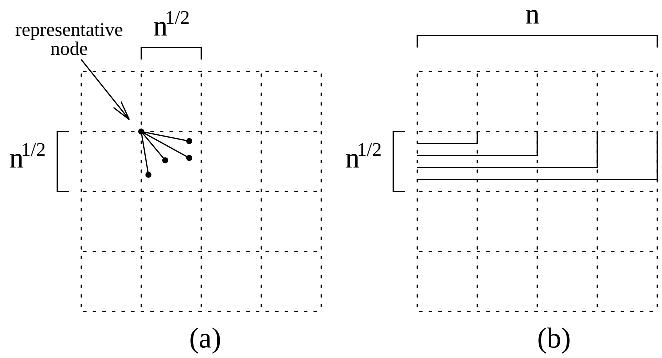

Figure 2: one-to-all from representative nodes (a) and projection of representative nodes onto the leftmost (or rightmost) column of M

Lemma 5.1 *For any instance* (M, U, h)*, where* M *is an* $n \times n$ *mesh and* $U = \{v_M(i,1) | 1 \leq i \leq n\} \times \{v_M(j,n) | 1 \leq j \leq n\}$*, there is a virtual path layout such that* $L \leq h \sqrt[h]{n}$.

Proof.

For the sake of simplicity, we assume that $n^{\frac{1}{h}}$ and n/h are integers[2]. It is possible to remove this assumption at the expense of an increased complexity in the notation.

The virtual path layout is organized in a butterfly-like structure, according to the scheme illustrated below.

M is first divided into partially overlapping submeshes, denoted as *superblocks* in what follows. Each superblock $S(r,t)$, for $t = 1, \dots h$ and $r = 1, \dots, n^{(t-1)/h}$, is described as the intersection of a subset of the columns of M with a subset of its rows. More precisely, nodes in $S(r,t)$ belong to columns $(t-1)n/h + 1, \dots, tn/h + 1$ if $t < h$, $(t-1)n/h + 1, \dots, tn/h$ otherwise; furthermore, they belong to rows $(r-1)n^{1-(t-1)/h} + 1, \dots, rn^{1-(t-1)/h}$. In the following $w(t)$ denotes the width of $S(r,t)$, that is, $w(t) = n/h + 1$ for $t < h$, $w(t) = n/h$ otherwise.

Each superblock $S(r,t)$ is uniquely partitioned into $n^{1/h}$ non-overlapping submeshes, named *blocks*. Each block has size $n^{1-t/h} \times w(t)$ and is denoted by $B(r,t,k)$, $k = 1, \dots, n^{1/h}$. It is worth noting that, for any $t < h$ and $k = 1, \dots, n^{1/h}$, $B(r,t,k)$ contains exactly the same rows as $S(\ell(r,k), t+1)$, where $\ell(r,k) = (r-1)n^{1/h} + k$. Furthermore, the rightmost column of $B(r,t,k)$ and the leftmost column of $S(\ell(r,k), t+1)$ overlap.

Sample decompositions of a 16×16 mesh in blocks and superblocks are shown in figures 3 and 4 respectively.

Considered any superblock $S(r,t)$ and one of its blocks $B(r,t,k)$, $v_{B(r,t,k)}(i,1)$,

[2] It is sufficient to assume that $n = (kh)^{x/h}$ for integers k and x. This does not prevent h from being constant, since x may be variable.

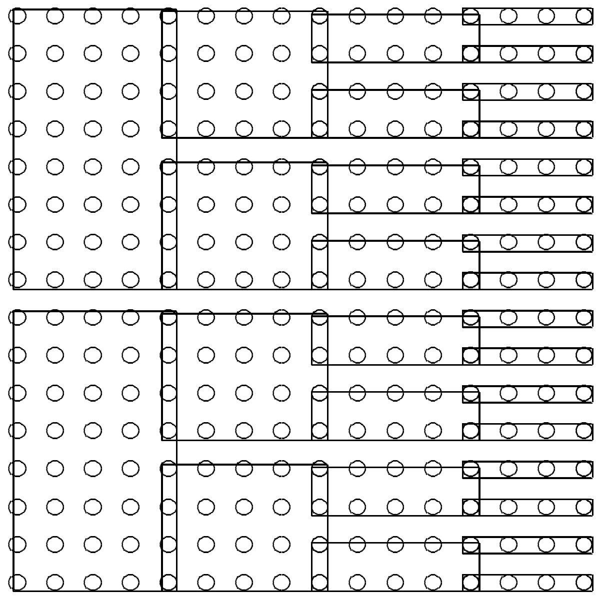

Figure 3: Block decomposition on a 16×16 mesh

$1 \leq i \leq n^{1-\frac{t}{h}}$, is the generical node belonging to the leftmost column of $B(r,t,k)$, while $v_{B(r,t,k)}(i,w(t))$ is any node on the rightmost column.

For each $v_{B(r,t,k)}(i,1)$ we define the set of its *partners* as $\{v_{B(r,t,l)}(i,w(t)), 1 \leq l \leq n^{\frac{1}{h}}\}$, that is, as the set of nodes $v(i,w(t))$, for each block $B(r,t,l)$ in $S(r,t)$.

For each block $B(r,t,k)$ we introduce a virtual path between node $v(i,1)$ and each of its partners. Virtual paths are spread uniformly in $S(r,t)$ as follows: the virtual path connecting $v_{B(r,t,k)}(i,1)$ to its partner $v_{B(r,t,l)}(i,w(t))$ is the concatenation of a horizontal path from $v_{B(r,t,k)}(i,1)$ to node $v(i,((k-1)n^{1-\frac{t}{h}} + i-1) \bmod w(t))$ of $B(r,t,k)$, of a vertical path from this to node $v(i,((k-1)(n^{1-\frac{t}{h}}) + (i-1)) \bmod w(t))$ of $B(r,t,l)$ and, finally, of a horizontal path from this node to $v_{B(r,t,l)}(i,w(t))$. In this way the virtual path is entirely contained in $S(r,t)$.

We now prove that, by means of such a virtual path layout, it is possible to connect any node $v(p,1)$ in the letmost column of M to any node $v(q,n)$ in its rightmost column in at most h hops, i.e. we prove that there exists a sequence $m_1, m_2, \ldots, m_{h+1}$ of nodes of M, such that $m_1 = v(p,1)$, $m_{h+1} = v(q,n)$, and, for each $1 \leq j \leq h$, m_j and m_{j+1} are connected by a virtual path.

Assume, by induction, that node m_j is on the leftmost column of some superblock $S(r,t)$ which is traversed by row q. This is clearly true for $j=1$. In

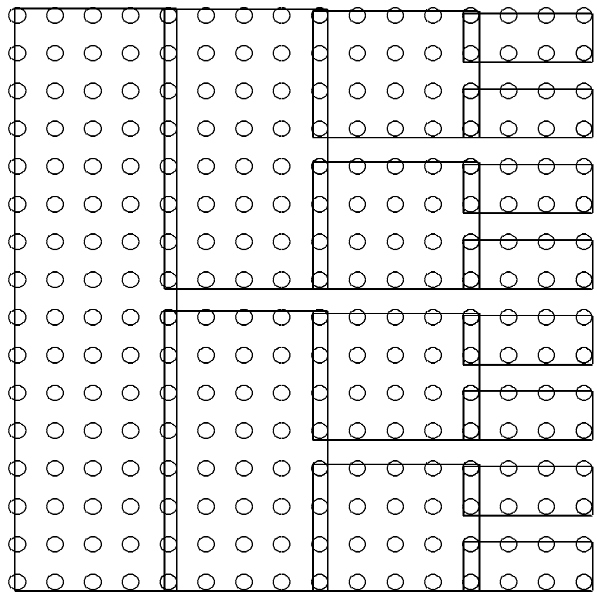

Figure 4: Superblock decomposition on a 16×16 mesh

fact, in this case row q certainly intersects the unique superblock $S(1,1)$. From the induction hypothesis, there must exist a block $B(r,t,k)$ in $S(r,t)$, which is traversed by row q. Let m_{j+1} be the unique partner of m_j in $B(r,t,k)$; then, by construction, there exists a virtual path connecting m_j and m_{j+1}.

Notice that, in the final step, the unique block $B(r,h,k)$ traversed by row q has size $1 \times \frac{n}{h}$ by construction. This implies that its rightmost node, i.e. the selected partner of m_h must indeed be $m_{h+1} = v(q,n)$.

In figure 5 the set of virtual channels resulting from the above construction and connecting node $v(1,1)$ with all nodes on the rightmost column on a 16×16 mesh are shown.

As to the load, observe that, since by construction no virtual path spans different superblocks, the maximum load can be derived by separately considering the sets of virtual paths in each superblock. Let us consider a generical superblock $S(r,t)$.

The load on the edges of $S(r,t)$ is distributed as follows:

1. **Horizontal edges:** the maximum number of virtual paths containing the same horizontal edge is bounded by $2 \cdot n^{\frac{1}{h}}$, where $n^{\frac{1}{h}}$ is the number of blocks in $S(r,t)$.

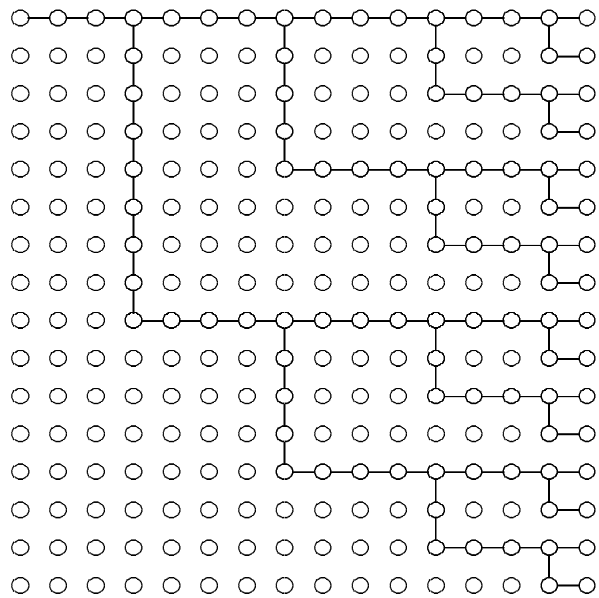

Figure 5: Virtual channels from $m_{1,1}$ on a 16×16 mesh

2. **Vertical edges:** since virtual paths are spread uniformly onto the columns of the $S(r,t)$, the maximum number of virtual paths containing the same vertical edge is bounded by

$$\frac{n^{\frac{h-t+1}{h}} \cdot n^{\frac{1}{h}}}{w(t)} \leq h\frac{n^{\frac{h-t+1}{h}} \cdot n^{\frac{1}{h}}}{n} = hn^{-\frac{t-2}{h}}.$$

Consequently, the maximum load may be on some vertical edge in $S(1,1)$ and is bounded by $h\sqrt[h]{n}$, thus proving the lemma. □

We can now prove the following result:

Theorem 5.2 *For any instance (M,h) of VPL^{a-a}, where M is a $n \times n$ mesh, there is a virtual path layout for (M,h) such that $L \leq 2(\frac{h}{3}+1)\sqrt[h]{n^3}$. When h is constant the load is asymptotically optimal.*

Proof. We first describe the characteristics of the virtual path layout, we then prove its correctness and we finally analyze its performance in terms of load. For the sake of simplicity, we assume that $\sqrt{n}$ is an integer.

M is first partitioned into n square submeshes of size $\sqrt{n} \times \sqrt{n}$. A *representative* node is then chosen in each submesh and locally connected to all other nodes in at most $h/3$ hops. Finally, all representatives are connected to each other by at most $h/3$ hops.

More in detail, M is partitioned into a chessboard, where each square is a submesh $M_{i,j}$ of size $\sqrt{n} \times \sqrt{n}$. Nodes in $M_{i,j}$ are those at the intersection of rows $(i-1)\sqrt{n}+1$ through $i\sqrt{n}$ with columns $(j-1)\sqrt{n}+1$ through $j\sqrt{n}+1$. Node $v_{M_{i,j}}(1,1)$ is chosen as the representative of $M_{i,j}$ and is denoted as $r_{i,j}$. For each submesh $M_{i,j}$ VPL^{1-a} $(M_{i,j}, v(1,1), h/3)$ is then solved using algorithm 4.1 of Section 4. From Corollary 4.2, the resulting virtual path layout has load not exceeding $\frac{h}{3}\sqrt[h]{n^3}$ and is therefore asymptotically optimal when h is bounded by a constant.

Let us now consider the problem of connecting all representatives to each other in at most $h/3$ hops.

This problem can be reduced to that of connecting each node in the leftmost column of M to all nodes in its rigthmost column by the same number of hops, that is to the instance $(M, U, h/3)$ of the virtual path layout problem, where $U = \{v_M(i,1) | 1 \leq i \leq n\} \times \{v_M(j,n) | 1 \leq j \leq n\}$, by simply connecting each $r_{i,j}$ to a pair of nodes, respectively on the rightmost and leftmost column of M. This can be done in the following way:

we first connect the representative $r_{i,j}$ of each $M_{i,j}$ to the node $v((i-1)\sqrt{n}+j, 1)$ on the leftmost column of M by a simple path, in such a way that the chosen paths are mutually edge-disjoint, in the way illustrated in figure 2 (b); we then do the same with respect to the rightmost column of M (the construction is only given for the former case, since it is symmetrical for the latter).

$r_{i,j}$ is connected to $v((i-1)\sqrt{n}+j, 1)$ through a simple path obtained as the concatenation of the vertical path from $r_{i,j}$ to node $v((i-1)\sqrt{n}+j, (j-1)\sqrt{n})$ of M and of the horizontal path from this to $v((i-1)\sqrt{n}+j, 1)$. Since any two consecutive representatives in the same column are separated by $\sqrt{n}-1$ rows and exactly $\sqrt{n}$ representatives appear in the same row of M, the chosen paths are, by construction, mutually edge-disjoint.

In a quite similar way it is possible to connect each representative node $r_{i,j}$ to a different node on the rightmost column of M by a set of mutually edge-disjoint simple paths.

Let Φ be any layout solving $(M, U, h/3)$, with U defined as above: Φ can be modified as follows: for any node u in the leftmost (resp., rightmost) column of M, let $r_{i,j}$ be the corresponding representative and let γ be the simple path connecting them, according to the construction given above. For each virtual path π in Φ incident to u, π is replaced by the concatenation of γ with π. It is worth noting that this way the hop-count of the resulting virtual path layout for $(M, U, h/3)$ is still h.

In this way we have a VPL^{a-a} restricted to the representatives, while the load of Φ is increased by a quantity which is bounded by $2\sqrt[h]{n^3}$. To see this, observe that the virtual paths leaving each representative are exactly $2\sqrt[h]{n^3}$ and that the simple paths connecting different representatives to the leftmost (resp.

rightmost) column of M are edge-disjoint.

As proved in lemma 5.1, there is a virtual path layout Φ for (M, U, $h/3$) whose load is bounded by $(h/3)\sqrt[h]{n^3}$. Consequently the load of the virtual path layout connecting all representatives to each other is bounded by $\left(\frac{h}{3}+2\right)\sqrt[h]{n^3}$.

We now prove that, given any pair (p, q) of nodes in M, p can be connected to q by at most h hops. In fact, let p and q belong to submeshes $M_{i,j}$ and $M_{h,k}$ respectively. p is first connected to its representative $r_{i,j}$ by at most $h/3$ hops, then, by at most $h/3$ additional hops, $r_{i,j}$ is connected to the representative $r_{h,k}$ of q and by $h/3$ further hops $r_{h,k}$ is connected to q.

Let us finally analyze the load of this virtual path layout. It is the result of two contributions: the load of the one-to-all layouts in the submeshes and that of the virtual path layout connecting all representatives to each other. Note that, from our construction, virtual path layouts in different submeshes do not overlap. Since we already observed that the first contribution is bounded by $\frac{h}{3}\sqrt[h]{n^3}$ and the second by $\left(\frac{h}{3}+2\right)\sqrt[h]{n^3}$, the total load does not exceed $2(\frac{h}{3}+1)\sqrt[h]{n^3}$, thus proving the theorem. □

6 Conclusions and future work

In this paper we proved that, given any instance (G, U, h) of the VPL problem and any virtual path layout Φ for (G, U, h), $\frac{1}{\Delta(G)}\sqrt[h]{\frac{\Delta(G)}{4h}\frac{N_C}{|C|}}$ is a lower bound to the load of Φ. This lower bound, for any fixed hop count h, is obtained with respect to the degree $\Delta(G)$ of G, the cardinality of any edge-cut C on G, the number N_C of pairs in U separated by C.

This result has been then applied to derive a lower bound for the one-to-all communication pattern, which is asymptotically tight when $\Delta(G)$ and h are bounded by constants. Finally, the cost of the the all-to-all communication pattern on two-dimensional meshes has been characterized and an asymptotically optimal virtual path layout (when h is bounded by a constant) has been designed.

The general lower bound we derived can be applied to any graph for which a suitable edge-cut can be found. As to open problems, a first question is to consider different virtual path layout problems. Another very interesting open question concerns the case where different connections have different bandwidth requirements: this corresponds to a weighted version of the virtual path layout problem, in which each element in U has an associated value, representing its contribution to the load of any link used by the corresponding virtual channel.

References

[1] H. Bodlaender. A partial k-arboretum of graphs with bounded tree-width. *Theoretical Computer Science*, 209, 1–2:1–45, 1998.

[2] I. Cidon, O. Gerstel, and S. Zaks. A scalable approach to routing in ATM networks. In *Proc. of the 8th International Workshop on Distributed Algorithms, Lecture Notes in Computer Science 857 (Springer Verlag)*, pages 209–221, 1994.

[3] I. Cidon, O. Gerstel, and S. Zaks. The layout of virtual paths in ATM networks. *IEEE/ACM Transactions on Networking*, 4(6):873–884, 1996.

[4] M. de Prycker. *Asynchronous Transfer Mode: Solutions for Broadband ISDN*. Ellis-Horwood, New York, NY, 1993.

[5] K. Diks, H.N. Djidjev, O. Sykora, and I. Vrto. Edge Separators of Planar and Outerplanar Graphs with Applications. *Journal of Algorithms*, 14:258–279, 1993.

[6] T. Eilam, M. Flammini, and S. Zaks. Optimal layouts on a chain ATM network. In *Proc. of the 24th International Colloquium on Algorithms, Languages and Programming - ICALP 97, Lecture Notes in Computer Science 1256 (Springer Verlag)*, pages 527–537, 1997.

[7] R. Feldmann, B. Monien, P. Mysliwietz, and S. Tschöke. A better upper bound on the bisection width of the de Bruijn network. In *Proc. of the 14th Symposium Theoretical Aspects of Computer Science, Lectures Notes in Computer Science 1200 (Springer Verlag)*, pages 511–522, 1997.

[8] ATM Forum. *ATM, User-Network Interface Specification. Version 3.0.* Prentice-Hall, 1993.

[9] L. Gasienic, E. Kranakis, D. Krizanc, and A. Pelc. Minimizing congestion of layouts for ATM networks with faulty links. In *Proc. of the 21th Symposium on Mathematical Foundations of Computer Science, Lectures Notes in Computer Science 1112 (Springer Verlag)*, pages 372–382, 1996.

[10] O. Gerstel. *Virtual Path Design in ATM Networks.* PhD thesis, Technion-Haifa, Israel, 1995.

[11] O. Gerstel, A. Wool, and S. Zaks. Optimal layouts on a chain ATM network. In *Proc. of the 3rd Annual European Symposium on Algorithms, Lectures Notes in Computer Science 979 (Springer Verlag)*, pages 508–522, 1995.

[12] O. Gerstel and S. Zaks. The virtual path layout problem in fast networks. In *Proc. of 13th ACM Conf. on Principles of Distributed Computing*, pages 235–243, 1994.

[13] L. S. Heath. Embedding Outerplanar Graphs in Small Books. *SIAM Journal on Algorithms and Discrete Methods*, 8(2):198–218, 1987.

[14] E. Kranakis, D. Krizanc, and A. Pelc. Hop-congestion tradeoffs for ATM networks. *International Journal of Foundations of Computer Science*, 8(2):117–126, 1997.

[15] F. T. Leighton. *Introduction to Parallel Algorithms and Architectures.* Morgan-Kaufmann, 1992.

[16] D.E. McDysan and D.L. Spohn. *ATM: Theory and Applications.* McGraw-Hill, New York, NY, 1995.

[17] S. Ramanathan and P. V. Rangan. Feedback techniques for intra-media continuity and inter-media synchronization in distributed multimedia systems. *The Computer Journal*, 36(1):19–31, 1993.

[18] J. Rolim, P. Tvrdík, J. Trdlicka, and I. Vrt'o. Bisecting de Bruijn and Kautz Graphs. *Discrete Applied Mathematics*, 85:87–97, 1998.

[19] P. Solé. Expanding and forwarding. *Discrete Applied Mathematics*, 58:67–78, 1995.

[20] P. Solé and I. Vrt'o. Two remarks on "Expanding and forwarding" by P. Solé. *Discrete Applied Mathematics*, 58:85–89, 1995.

[21] L. Stacho and I. Vrt'o. Virtual path layouts for some bounded degree networks. In *Proc. of 3rd International on Structural Information and Communication Complexity (SIROCCO)*, 1996. To appear in SIAM Journal of Computing.

[22] O. Sykora and I. Vrto. Edge separators for graphs of bounded genus with applications. *Theoretical Computer Science*, 122:419–429, 1993.

[23] S. Zaks. Path Layout in ATM Networks. In *Proc. of SOFSEM 97, Lecture Notes in Computer Science 1338 (Springer Verlag)*, pages 144–160, 1997.