Applying 4-regular grid structures in large-scale access networks

Pedersen, Jens Myrup; Knudsen, Thomas P.; Patel, Ahmed; Madsen, Ole Brun

Published in:
Computer Communications

Publication date:
2006

Document Version
Publisher's PDF, also known as Version of record

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computer Communications 29 (2006) 1350–1362

computer communications

www.elsevier.com/locate/comcom

Applying 4-regular grid structures in large-scale access networks

Jens Myrup Pedersen[a,*], Ahmed Patel[b], Thomas Phillip Knudsen[a], Ole Brun Madsen[a]

[a]Department of Control Engineering, Aalborg University, DK-9220 Aalborg East, Denmark

[b]Computer Networks & Distributed Systems Research Group, Department of Computer Science, University College Dublin, Belfield, Dublin 4, Ireland

Received 10 August 2004; revised 26 July 2005; accepted 29 July 2005
Available online 24 August 2005

Abstract

4-Regular grid structures have been used in multiprocessor systems for decades due to a number of nice properties with regard to routing, protection, and restoration, together with a straightforward planar layout. These qualities are to an increasing extent demanded also in large-scale access networks, but concerning protection and restoration these demands have been met only to a limited extent by the commonly used ring and tree structures. To deal with the fact that classical 4-regular grid structures are not directly applicable in such networks, this paper proposes a number of extensions concerning restoration, protection, scalability, embeddability, flexibility, and cost. The extensions are presented as a tool case, which can be used for implementing semi-automatic and in the longer term full automatic network planning tools.

Keywords: Communication system planning; Network; Routing; Topology; WAN

1. Introduction

The Internet is to an ever-increasing extent becoming a part of every day life for people all over the world. While it was traditionally used for best-effort services such as email, news, FTP, and to some extent WWW, a large variety of applications have now been developed that demand higher levels of QoS and reliability. In order to support this, new protocols such as Intserv and Diffserv have been developed together with different protection and restoration schemes.

While most households today receive telephony, television, and radio by dedicated technologies for each medium, this is expected to change as virtually all media are becoming able to communicate via Internet protocols and consequently make use of the same physical connections [1]. Together with an increasing use of telerobotics [2,3], tele operations [4], and other critical applications, the access networks are becoming a critical part of the whole communication infrastructure. Even though protocols are being developed to ensure reliability, the physical network structures limit what level of reliability can be offered: communication between two nodes is only possible if there is a physical connection between them. In a tree-based network structure there exists only one path between any pair of nodes, making it vulnerable to attacks and failures. Obvious and commonly used alternatives to tree structures are ring structures, which offer connectivity in case of any single failure. However, given the expected demands of reliability, this is likely to become insufficient in near future.

The study of structural and topological properties of networks is highly relevant because Fiber To The Home is about to replace the old copper-based telephony infrastructure in many countries worldwide. This is a unique opportunity to implement a structure almost from scratch. At the same time, it is a huge task, especially because it requires a huge amount of duct digging. This together with the fact that fibre infrastructures are expected to have a long lifetime because they are upgradeable by changing end equipment only is a major argument for choosing network structures with good and predictable properties.

It has been shown that node symmetry, maximal connectivity, and regularity are important properties to satisfy for robust network structures [5]. A regular structure is a structure where all nodes are connected by the same number of lines n, and given n, such a structure is said to be

* Corresponding author. Tel.: +45 96358700; fax: +45 98151739.
E-mail addresses: jens@control.aau.dk (J.M. Pedersen), a.patel@cnds.ucd.ie (A. Patel), thomas@control.aau.dk (T.P. Knudsen), obm@control.aau.dk (O.B. Madsen).

0140-3664/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2005.07.020

n-regular. 2-Regular structures are equivalent to rings, and inherently node symmetric and maximally connected. The next step, discussed in this paper, is to consider 3-regular or 4-regular structures, which are preferably node symmetric and maximally connected, but at least sufficiently regular and symmetric to benefit from these properties.

Among the 3-regular structures, the group of $N2R(p; q)$ structures has been introduced as a generalization of double rings. Another set of 3-regular structures, the honeycomb structures, has been introduced for multiprocessor systems [6], but most of these results are directly transferable to large-scale networks. However, some hierarchical extensions must be developed for these structures to perform reasonably well with regard to average distances and diameters, and this challenge has not yet been met.

4-Regular grid structures have been used for multiprocessor systems in decades due to their nice properties with respect to routing and restoration, and they were recently suggested used as a base for access network structures as well [7,8]. This paper is an extension of these two conference papers. In addition to the results previously published, it contains a more clear and in-depth presentation of the theory, as well as new results obtained by pruning the structures as described in Section 6. A few of the previously published results are not presented in this paper, in order to make it more straightforward to read.

The main contribution of the work presented, is a number of extensions of the 4-regular grid structures. While the basic structures are suitable for multiprocessor systems, the extensions presented are required in order to be able to apply the structures in large-scale networks and planning hereof. This work forms the base for developing semi-automatic and in the longer term full automatic network planning tools.

The remainder of the paper is organized as follows: Section 2 introduces preliminaries, background, and notation. Section 3 introduces the basic 4-regular grid structure, which forms a base for the extensions provided in Sections 4–6. Section 4 introduces restoration and protection schemes. The two other important extensions introduced are the hierarchical extension (Section 5) and the pruning (Section 6). Section 7 provides some tools for embedding the structures in real-world networks, leading to the discussion on applying the theory in real-world network planning given in Section 8. Section 9 concludes the paper.

2. Preliminaries

Throughout this paper, network structures are studied. The definition of a structure is similar to the definition of a graph, and can be used for modelling a network, abstracting away from specific physical conditions. Node equipment, transmission technologies, wiring, and bandwidth are not taken into consideration. The terminology used is well known from graph theory, so in the following we shall simply provide the usual conventions.

A structure consists of a set of nodes and a set of lines, such that each line interconnects two nodes. Lines are bi-directional: if a pair of nodes (u, v) is connected by a line, so is (v, u). A path from a source node u to a destination node v is a sequence of nodes and lines $(u=u_0)$, e_1, u_1, e_2, $u_2,\ldots$, e_{n-1}, u_{n-1}, e_n, $(u_n=v)$, where every line e_i connects the nodes u_{i-1} and u_i. It is assumed that $u_i \neq u_j$ whenever $i \neq j$. Only connected structures are dealt with, i.e. for each pair of nodes (u, v) in the structure, there exists at least one path between u and v. The length of a path is determined by the number of lines it contains; in the previous case, the path is of length n. The distance between two nodes u and v is written $d(u, v)$ and is determined by the length of the shortest path between them.

Two different paths between a pair of nodes (u, v) are said to be line independent if they share no lines, and a set of paths between u and v are said to be line independent if they are pair wise line independent. Similarly, two different paths between a pair of nodes (u, v) are said to be node independent if they share no nodes except for u and v, and a set of paths between u and v are said to be node independent if they are pair wise node independent. It is easy to see that two node independent paths (of length more than one) must be line independent, but that the converse is not in general true.

For a node u, the set of nodes v such that $d(u, v)=1$ are said to be the neighbours of u, and two nodes are said to be connected if and only if they are neighbours. The degree of a node corresponds to the number of neighbours it has. If all nodes of a structure have the same degree n, the structure is said to be n-regular. The size of a structure is given by the number of nodes it contains.

A number of parameters for evaluation of structures [9] are referred to throughout the paper, and defined in the following. Let S be a network structure consisting of a set of nodes N and a set of lines L:

- Average average distance: the average average distance is obtained by taking the average of $d(u, v)$ over all pairs of nodes $u \neq v$, where $u, v \in N$.
- Worst-case average distance: the worst-case average distance is obtained by taking the maximum over all nodes u of the average of $d(u, v)$ for all nodes v, $u \neq v$, where $u, v \in N$.
- Diameter: the diameter is obtained by taking the maximum of $d(u, v)$ over all pairs of nodes $u \neq v$, where $u, v \in N$.
- Cost: since the representations of structures abstract from specific physical conditions such as node equipment, transmission technologies, bandwidth, and line ducts and lengths, it is hard to estimate the cost of a structure as such. We use either the number of lines or the average node degree to indicate the cost.

3. The basic structure

The basic structure is well known from multiprocessor systems (e.g. [10,11]). In order to define and discuss various extensions later in the paper, it is necessary to provide precise definitions and notation. Notation and definitions are chosen to make the paper fairly easy to read.

Let dim_x and dim_y be positive integers. They define a 4-regular grid structure S with node set N and line set L as follows. Every node in N is associated to a pair of integer coordinates (x, y) such that $0 \leq x \leq dim_x$ and $0 \leq y \leq dim_y$, and every such coordinate pair is associated to a node. Furthermore, no two nodes are associated to the same pair of coordinates. Consequently, there are exactly $(dim_x+1)(dim_y+1)$ nodes in S. If a node u is associated to a coordinate pair (x_u, y_u), we write $u=(x_u, y_u)$ to ease notation. The definition of lines depends on whether the mesh or torus is dealt with.

The lines of the 4-regular mesh are given as follows: two nodes (x_u, y_u) and (x_v, y_v) are connected by a line if and only if $|x_u-x_v|+|y_u-y_v|=1$. Despite not being regular, we still refer to it as the 4-regular mesh. A torus is obtained by adding a set of lines, such that two nodes (x_u, y_u) and (x_v, y_v) are connected also if either $|x_u-x_v|=dim_x$ and $y_u=y_v$ or $|y_u-y_v|=dim_y$ and $x_u=x_v$. The torus is regular, node symmetric, and maximally connected, which is not the case for the mesh. The mesh nevertheless possesses most of the qualities related to these properties, except for the unevenly distributed traffic and the less robust nodes on the edges. Since the mesh is planar and thus easily embedded on a surface, this paper focuses on the mesh, and when referring to the 4-regular grid structure, the mesh is implicitly assumed. However, most of the results are easily extended to the torus, even though certain parts, such as routing, become more complicated.

A node (x, y) such that either $x=0$, $x=dim_x$, $y=0$ or $y=dim_y$ is said to be an edge node, and the four nodes (0, 0), $(dim_x, 0)$, $(0, dim_y)$, and (dim_x, dim_y) are said to be corner nodes. While in general there exist four node independent paths between any pair of nodes, there exist no more than three or two node independent paths between a pair of nodes if one of the nodes is an edge or corner node, respectively. It is also easy to see that the distances to and from edge and corner nodes are generally larger than those to and from nodes in the middle of S.

3.1. Routing

The purpose of the routing scheme proposed in the following is to make it possible to send packets from one node to another using a shortest path. This is done using hop-by-hop routing, such that when a node receives a packet of which it is not the intended destination, it is forwarded to a neighbour. Doing this without tables, relying only on node addresses, is known as Topological Routing. Topological Routing is especially beneficial in large-scale networks, because maintaining tables of the complete network topology is a resource-consuming task.

For the basic structures, standard XY-routing (see e.g. [12]) is used. Let p be a packet with destination (x_v, y_v). Whenever p is received by a node (x_u, y_u) it is determined if it has reached its destination. If this is not the case, it is forwarded using the following algorithm, which also applies if (x_u, y_u) is the source node:

- Let $\Delta x=x_v-x_u$ and $\Delta y=y_v-y_u$.
- If $\Delta y<0$, p can be forwarded to (x_u, y_u-1), and if $\Delta y>0$ to (x_u, y_u+1).
- If $\Delta x<0$, p can be forwarded to (x_u-1, y_u), and if $\Delta x>0$ to (x_u+1, y_u).

If $\Delta y=0$ or $\Delta x=0$, the path is uniquely determined. Otherwise two possibilities exist and a choice must be made. A random choice can be made, one direction can be given highest priority such that it is followed whenever possible, or the packet can be sent in the direction with the highest value of Δ. The advantage of the latter approach is that the number of potential paths is the highest possible in every intermediate node, i.e. in case of an arbitrary failure the risk of having to route along a longer path is minimized. This choice can also be used by protection schemes.

4. Restoration and protection

The routing scheme introduced always results in a shortest path, given that the structure is as defined, without failing nodes or lines. However, both lines and nodes do fail from time to time, and therefore it must be possible to route even in case of one or more failures: in any case where a path exists between two nodes, the routing scheme should be able to find it. Some applications tolerate a certain delay or jitter, which allows time for establishing a new path, while others are more critical with respect to delays and must be able to communicate smoothly, even in case of failures. This is handled by restoration and protection schemes, which we propose as follows.

The restoration scheme allows for restoration and for choosing paths in networks with failures. This is handled by lake algorithms. Let S be a 4-regular grid structure and assume that a set of nodes N' and a set of lines L' are missing or out of order. Furthermore, any line connected to a node in N' is considered to belong to L'. Let S' denote the structure without failing nodes and lines (that is $S'=S-L'-N'$).

A set of nodes $N'' \subseteq N'$ and lines $L'' \subseteq L'$ are said to form a lake A if in a standard (x, y) planar representation of S as shown in Fig. 1, it is possible to draw a (not necessarily straight) line from any element (node or line) in A to any other element in A without crossing any nodes or lines not in A. Furthermore, in this planar representation, it must not be possible to draw a line from an element in A to an element in N' or L', which is not in A, without crossing an element in S'.

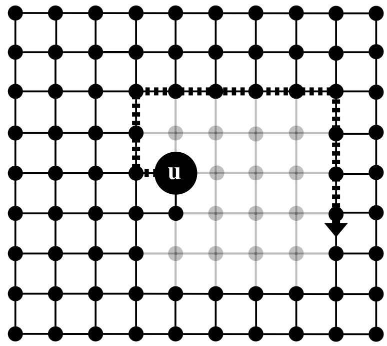

Fig. 1. The routing of a left packet (dashed line) from u ensures it traverses around the lake (shaded). For any lake A, a line can be drawn between any two elements of it without crossing any element not in A.

A node in S' is said to be a border node of A, if it is in S connected to a line in A. In the following it is assumed that only one lake A exists in S', but this is easily generalized, keeping in mind the fact that no node in S' is a border node to more than two lakes.

When a node u detects that a line or node connected to it experiences a failure, it becomes aware that a lake has appeared, and that consequently it has become a border node. The next step is to collect the information necessary to be able to route packets around the lake. This is done by using left control packets, right control packets or both. Consider S mapped onto a standard (x, y) coordinate system as shown in Fig. 1. From u a left control packet q_{left} is sent along the first available line on the left-hand side of the detected unavailable line/node. Initially it contains only information stating that it is a left control packet with origin u. When a node v receives q_{left}, it first checks if v is the origin of q_{left}. If this is not the case, v is added to a list carried by q_{left}, keeping record of all nodes passed, as well as their order. q_{left} is then forwarded along the first available line on the left-hand side of the line from which it was received. When q_{left} is received by its origin, the list of nodes passed is stored in a table T_u, called a lake table, and q_{left} is killed. The nodes stored in this table define the border of A seen from u (there may exist several borders of A such that no path exists between any pair of nodes from different borders). Note that not all nodes on the border are actually border nodes. Right control packets are defined in a similar manner, replacing the occurrences of 'left' in the above definition by 'right'. For every border node u of A, such a table T_u is kept updated by sending right and/or left control packets within specified intervals. When it is determined that the failing link(s) has recovered from the failure, T_u is deleted and routing again done as usual. If u is a border node of two lakes, a table is maintained for each lake.

When a packet p with destination w is received in u, the following happens: if in S' there exists a neighbour u' of u such that $d(u', w) < d(u, w)$, p is forwarded from u to u' as usual. If, however, there is no such node, a lookup is made in T_u, and a node v in T_u is chosen such that $d(v, w) < d(u, w)$, and such that $d(v, w)$ is smallest possible. A shortest path from u to v using the nodes of T_u is now determined, and v is sent to the first node on this path along with the path specification. In any node of this explicitly defined path, it is forwarded simply to the next node of the path. In case a line of this path is failing, the path is discarded and p treated like any other packet. Given that the lake is or becomes stable during the routing process and that a path to the destination exists, it is ensured that the packet reaches its destination in a finite number of hops. If T_u contains no node v such that $d(v, w) < d(u, w)$, the packet is either treated as if no table exists (described below), or it is discarded. If the table is updated and contains no such node v, w is either in A, or it is unreachable from u, possibly because the removal of A has disconnected S.

Different schemes can be used to optimize the set-up of paths given a table T_u. The simplest solution is to define the path as the list of nodes traversed by a right or left control packet, but in some cases gains can be obtained by discarding loops, or even by using nodes not listed in T_u. Maintaining tables of a larger part of the structure than just the border of A may be useful in order to determine shorter paths, and it may be possible to improve performance by storing tables in a larger set of nodes around a lake, such that alternative routing can be done before a packet reaches the border. However, there is a trade-off between different factors including path lengths, set-up times in case of failures, restoration time, and resource usage in terms of storage capacity and control traffic.

In case only one line or node fails, and even in case a few lines/nodes are failing, lake tables can be generated fast since the left and right control packets only need to traverse a few nodes. However, it can happen that u receives a packet p with destination w, which could in S only be sent along a line in A, before a table T_u has been created. In this case, a chance is taken to send the packet right or left (or both) around the lake (using a scheme similar to that of left and right control packets) for example until it reaches a node v such that $d(v, w) < d(u, w)$, from where it is routed either normally or by using lake algorithms. It is also possible to simply discard the packet.

The protection scheme allows for choosing paths with protection. In general up to four node independent paths can be set-up between any pair of nodes, but if one of the nodes is an edge or corner node, fewer paths exist. Assume that (x_u, y_u) and (x_v, y_v) are nodes in S, that none of them are edge nodes, and that a packet p is to be sent from (x_u, y_u) to (x_v, y_v).

Clearly, either $x_u \neq x_v$ or $y_u \neq y_v$ (or both). Without loss of generality, it is necessary only to consider two cases: in the first case $x_u \neq x_v$ and $y_u \neq y_v$, and in the second case $x_u \neq x_v$ and $y_u = y_v$. In both cases it can be assumed without loss of generality that $x_u < x_v$ and $y_u < y_v$.

In the first case, four node independent paths are established by duplicating the packet and routing each copy as follows:

(1) (x_u, y_u), (x_u, y_u+1),…,(x_u, y_v), (x_u+1, y_v),…,(x_v, y_v). This path has length $y_v-y_u+x_v-x_u$ and is a shortest path.
(2) (x_u, y_u), (x_u+1, y_u),…,(x_v, y_u), (x_v, y_u+1),…,(x_v, y_v). This path has length $y_v-y_u+x_v-x_u$ and is a shortest path.
(3) (x_u, y_u), (x_u, y_u-1), (x_u+1, y_u-1),…,(x_v+1, y_u-1), (x_v+1, y_u),…,(x_v+1, y_v), (x_v, y_v). This path has length $y_v-y_u+x_v-x_u+4$.
(4) (x_u, y_u), (x_u-1, y_u), (x_u-1, y_u+1),…,(x_u-1, y_v+1), (x_u, y_v+1),…,(x_v, y_v+1), (x_v, y_v). This path has length $y_v-y_u+x_v-x_u+4$.

In the second case, the first three node independent paths are established by sending copies of the packet as follows:

(1) (x_u, y_u), (x_u+1, y_u),…,$(x_v, y_u=y_v)$. This path has length x_v-x_u and is a shortest path.
(2) (x_u, y_u), (x_u, y_u+1), (x_u+1, y_u+1),…,(x_v, y_u+1), $(x_v, y_u=y_v)$. This path has length x_v-x_u+2.
(3) (x_u, y_u), (x_u, y_u-1), (x_u+1, y_u-1),…,(x_v, y_u-1), $(x_v, y_u=y_v)$. This path has length x_v-x_u+2.

The fourth node independent path can be established in two ways. In some cases where either (x_u, y_u), (x_v, y_v) or both are neighbours to an edge node, it is possible that only one of them exists:

- (x_u, y_u), (x_u-1, y_u), (x_u-1, y_u+1), (x_u-1, y_u+2), (x_u, y_u+2),…,(x_v+1, y_u+2), (x_v+1, y_u+1), (x_v+1, y_u), $(x_v, y_u=y_v)$.
- (x_u, y_u), (x_u-1, y_u), (x_u-1, y_u-1), (x_u-1, y_u-2), (x_u, y_u-2),…,(x_v+1, y_u-2), (x_v+1, y_u-1), (x_v+1, y_u), $(x_v, y_u=y_v)$.

In both cases the path length is x_v-x_u+8.

5. Hierarchical extension

We propose a hierarchical extension to deal with the fact that distances in 4-regular grid structures increase as shown in Fig. 2, which results in considerably larger distances than in today's Internet. For example, the average distance in a square mesh structure of 10 000 nodes is 66.67, while in 1998 average path lengths of the Internet were measured, and it was shown that the Internet at that time had an average path length of around 11–24, depending on location [13]. In 1999 appr. 88 000 different nodes (routers) were found in the Internet [14].

The extension is introduced in two steps: first an additional set of lines, hierarchical lines, are defined, and next a revised routing scheme is presented in order to make use of these new lines. At the end of the section, the performance of the hierarchical extension is evaluated.

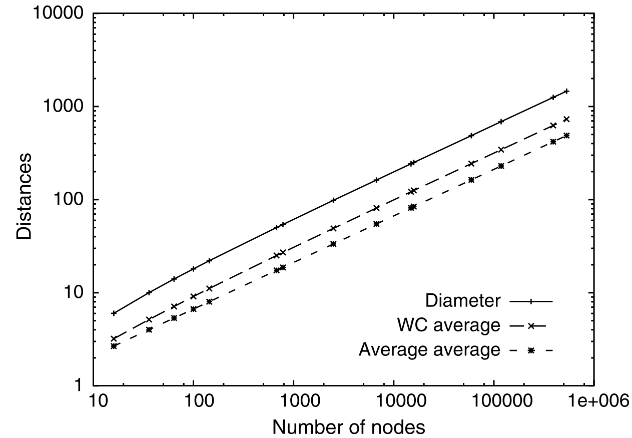

Fig. 2. Distances in 4-regular grid mesh with no hierarchical extension.

5.1. Physical extension

Let S be a 4-regular grid structure, and let g_x and g_y be positive integers defining the granularity in x respectively y directions. Furthermore, n_H must be a non-negative integer defining the number of hierarchies. If $n_H=0$, there are no hierarchical lines added to the structure. While the definitions in the following are valid for all $g_x>0$ and $g_y>0$, we assume that both g_x and g_y are chosen odd such that $g_x\geq 3$ and $g_y\geq 3$ in order to support the revised routing schemes.

Let (x, y) be a node of S. As given by the definition of the 4-regular grid structure, it is connected by basic lines to the nodes $(x+1, y)$, $(x-1, y)$,$(x, y+1)$, $(x, y-1)$ that exist in S. For any i, $0\leq i\leq n_H$ such that $x\equiv 0 \pmod{g_x^i}$, and $y\equiv 0 \pmod{g_y^i}$ a node (x, y) is said to belong to layer i. For $i>0$ this means that it is connected by hierarchical lines to the nodes $(x+g_x^i, y)$, $(x-g_x^i, y)$, $(x, y+g_y^i)$, and $(x, y-g_y^i)$ that exist in the structure. These lines are also said to be lines of layer i. As such, the introduction of hierarchies is the introduction of an additional set of lines. The main model used in this paper is the Perfect Square Mesh. In addition to the conditions above, S must satisfy that $dim_x=g_x^{n_H}$, $dim_y=g_y^{n_H}$, and $g_x=g_y$. For short, we write $g=g_x=g_y$. An example of such a structure, illustrating how performance gains are obtained, is shown in Fig. 3.

5.2. Revised routing scheme

Before introducing the revised routing scheme, some notation is necessary. For any pair of nodes $u=(x_u, y_u)$ and $v=(x_v, y_v)$, both belonging to layer i $(0\leq i\leq n_H)$ of a structure with hierarchical extension, let

$$d_i(u,v)=\frac{|x_u-x_v|}{g_x^i}+\frac{|y_u-y_v|}{g_y^i}.$$

Furthermore, let $u_j=(x_{u_j}, y_{u_j})$ and $v_j=(x_{v_j}, y_{v_j})$ be the nodes of layer j $(0\leq j\leq n_H)$ such that $d_0(u, u_j)$ and $d_0(v, v_j)$ are smallest possible.

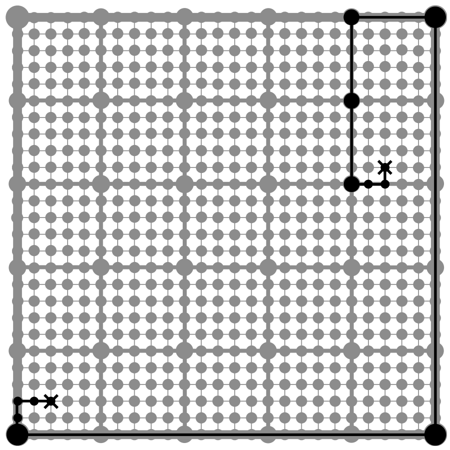

Fig. 3. A Perfect Square Mesh with $g=5$ and $n_H=2$. The distance between the marked nodes is reduced from 34 to 12 by the addition of hierarchical lines.

The revised routing scheme makes use of a number of easily established properties. Assume that S is a 4-regular grid structure with g_x, g_y, and n_H chosen as above. The scheme is designed for the Perfect Square Mesh with $g>3$, but even if $g_x \neq g_y$ and/or $g=3$, it can be easily modified to always result in a shortest path. In these cases it may be necessary to perform more calculations because path lengths in higher layers must be calculated.

(1) Let $u=(x, y)$ be a node of S. Then the node u_i is easily determined by $(g_x^i \text{round}(x/g_x^i), g_y^i \text{round}(y/g_y^i))$, where round($a$) determines the integer I such that $|a-I|$ is smallest possible. Since g_x and g_y are odd, such rounding is always unique.
(2) Let u and v be nodes of S, and assume for some value of $i<n_H$ that no path p_{i+1} between u and v using at least one line of layer $i+1$ exists, which is shorter than the shortest path using only lines of layer i or lower, p_i. Then there exists no path using lines of layer $i+2$ or higher, which is shorter than p_i.
(3) Let u and v be nodes of S, and assume for some i that a shortest path between u and v uses some line of layer i. This implies that there exists a shortest path between u and v that contains all nodes u_j and v_j, for all $j \leq i$.

The routing scheme works as follows, assuming that a packet p is to be sent from u to v, u belonging to layer i but not layer $i+1$. p can be sent either through a node of layer $i+1$ or through nodes of layer i and below only. If it is sent through layer $i+1$, it is forwarded to a node u' among $(x_u+g_x^i, y_u)$, $(x_u-g_x^i, y_u)$, $(x_u, y_u+g_y^i)$ and $(x_u, y_u-g_y^i)$ that minimizes $d_0(u_{i+1}, u')$. If it is not sent through layer $i+1$, two schemes can be used. In case $g_x \neq g_y$, only the first scheme guarantees that a shortest path is always chosen:

(1) Let u' be a neighbour of u that belongs to the highest possible layer while still satisfying $d_0(u', v)<d_0(u, v)$. Then p is sent to u'.
(2) p is forwarded to any neighbour u' of u that minimizes $d_0(u', v)$.

If p is sent through layer i, a shortest path from u to v that passes at least one node of layer i can be constructed passing both u_i and v_i due to property (3), and the same holds for any shortest path from u to v that passes at least one node of layer $i+1$: such a shortest path exists passing both u_i, u_{i+1}, v_{i+1}, and v_i.

Therefore, when deciding if routing should be done through layer $i+1$, it is sufficient to compare the distance from $u=u_i$ (u belongs to layer i) to v_i using layer i to the distance using layer $i+1$. Note that routing through layer i does not necessarily imply that any line of this layer is actually used. The distance between u_i and v_i using layer i is given by $d_i(u, v_i)$ while the distance using layer $i+1$ is given by $d_i(u,u_{i+1})+d_{i+1}(u_{i+1},v_{i+1})+d_i(v_i,v_{i+1})$. All these nodes are easily determined due to property (1).

Cases can occur where the distances using layer i and layer $i+1$ are similar, and similarly if routing is done using layer i, it may happen that more than one neighbour of u can be selected for forwarding. In this case, routing can be done using any such layer and line.

Routing can also be done using layer $i+2$ or higher. This decision is made whenever a packet reaches a node of layer $i+1$. This is sufficient due to properties (2) and (3).

The presented scheme always determines a shortest path, and selects the appropriate hierarchical layer, ensuring maximum benefit from the hierarchical extension.

The restoration and protection schemes introduced in Section 4 have not yet been extended to deal with the problems, which occur in hierarchical structures. The protection scheme is easily extended to provide four node independent paths between any pair of nodes within each layer, making it possible to establish line independent paths from source to destination even in a hierarchical structure. However, the same nodes may be used when shifting from one layer to another. This may be managed by improving the reliability of nodes belonging to first or higher layers, but increasing distances are still a problem: the additional hops as specified in the scheme are added at each layer. The restoration scheme must also be extended, especially in order to handle situations where a packet is to be routed from a layer i to layer $i+1$, but where the nearest layer $i+1$ node is out of order or not reachable.

5.3. Performance

The performance of the hierarchical extension in terms of average distances, worst-case average distances, and diameters were measured by calculating those values for a number of hierarchical structures. The results are shown for the basic structures (Fig. 2) and for the Perfect Square Mesh (Figs. 4–6). While the basic structures have power-law dependencies between the number of nodes and

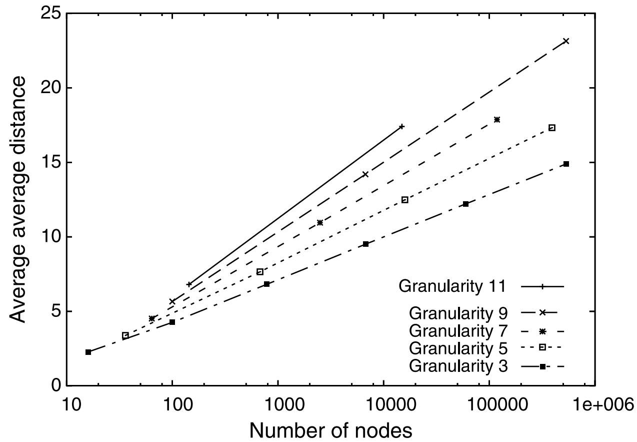

Fig. 4. Average average distances in Perfect Square Mesh.

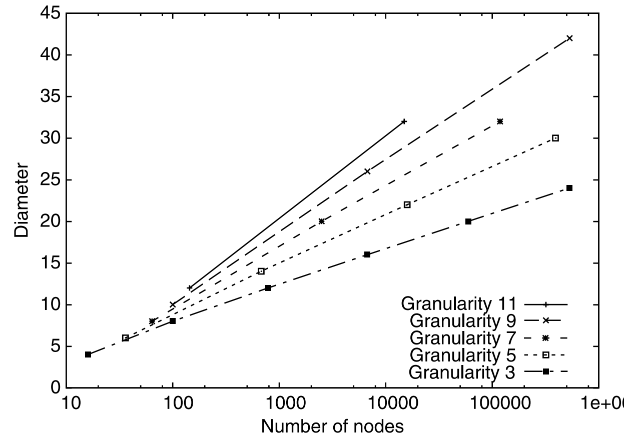

Fig. 6. Diameters in Perfect Square Mesh.

the distances, the dependencies are logarithmic for the Perfect Square Mesh.

6. Cost reduction

Since ring structures have average degree 2, and double ring structures as well as Generalized Petersen Graphs [15] have average degree 3, the 4-regular grid structure is more expensive: the torus is 4-regular and consequently the average degree is 4, while the mesh has a slightly lower average degree due to edge and corner nodes. For a structure of for example 26×26 nodes, the average degree is 3.85, approaching 4 for larger structures. The hierarchical extensions generally add to the number of lines and node degrees, but their contribution to the total number of lines and average degree is limited, as can be seen from Table 1.

The high costs of the 4-regular grid structures make it difficult to apply them in real-world networks since rings offer a level of protection, which has been satisfactory so far. However, cheaper variants of the 4-regular grid structure may form better solutions than the rings even in the short term, while at the same time allowing for gradual extensions. In the following, we propose and discuss two ways of pruning.

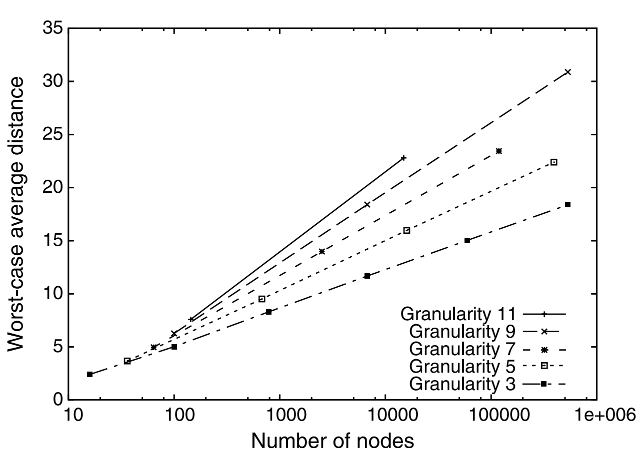

Fig. 5. Worst-case average distances in Perfect Square Mesh.

The first approach is to remove every second line in one of the directions as shown in Fig. 7, leading to a honeycomb structure. This approach seems feasible: it is cheaper than the 4-regular grid, it can be extended/upgraded to a 4-regular grid structure, and addressing and routing have been described, with properties similar to those of the 4-regular grid structure [6]. However, no hierarchical extension has been developed or proposed so far.

The rest of the section deals with another approach, making use of the fact that the number of hierarchical lines is very small compared to the number of basic lines. Based on this, we propose to reduce only the number of lines in the basic structure, see Fig. 8, by removing all basic lines in the y direction except for those co-located with lines of a higher hierarchy. In other words, two nodes (x_u, y_u) and (x_v, y_v) are connected by a basic line if and only if one of the following is true:

- $y_u = y_v$ and $|x_u - x_v| = 1$.
- $x_u = x_v$, $|y_u - y_v| = 1$ and $x_u \equiv 0 \pmod{g_x}$.

The performances of such a pruned Perfect Square Mesh with $g = 3$ and $g = 9$ are shown in Figs. 9 and 10. It can be seen that while the number of lines, and thus the average degree as well as the cost, is significantly reduced, the average and worst-case average distances are only slightly larger. For the structures with 6724 nodes, the number of lines is reduced by 29.2% for the Perfect Square Mesh with

Table 1
Average degree and the contribution to this from basic lines in Perfect Square Mesh with $g = 5$

Size	Avg. degree	% of lines basic
$n_H = 1$, 36 nodes	3.56	93.8
$n_H = 2$, 676 nodes	4.04	95.3
$n_H = 3$, 15 876 nodes	4.14	95.8
$n_H = 4$, 391 876 nodes	4.16	96.0

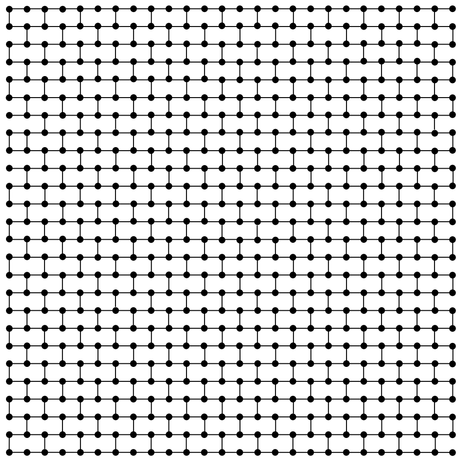

Fig. 7. Pruning a basic structure to obtain a honeycomb.

$g=3$, and by 43.3% for the Perfect Square Mesh with $g=9$. The average distances are increased by 0.024% respectively 0.35%. The corresponding values for the structures with 532 900 nodes are reductions in the number of lines of 29.6% respectively 43.8%, and an increase in average distance of $<0.001\%$ respectively 0.003%. The calculations were done with a precision of three decimals, but for $g=3$ with 532 900 nodes, the calculations returned identical values. Due to the fact that only distances between pairs of nodes for which all shortest paths use no hierarchical lines are affected, the differences are smallest for large structures and structures with small granularity. The diameters are the same for pruned and non-pruned structures. While the distances are only slightly affected by this pruning, the reliability is better in the non-pruned structures due to the higher connectivity.

This approach to pruning requires a revised routing scheme. Assume that a packet is send from a node $u=(x_u, y_u)$ to another node $v=(x_v, y_v)$. A revised routing scheme must be used in case all of the following conditions are true:

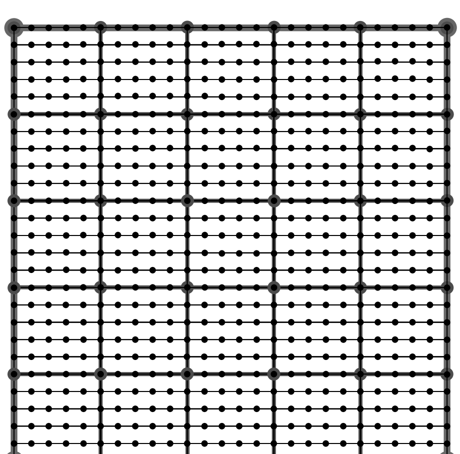

Fig. 8. Pruning of a Perfect Square Mesh with $g=5$ and $n_H=2$ by removing all basic lines in the y-direction, except those co-located with lines of higher hierarchies.

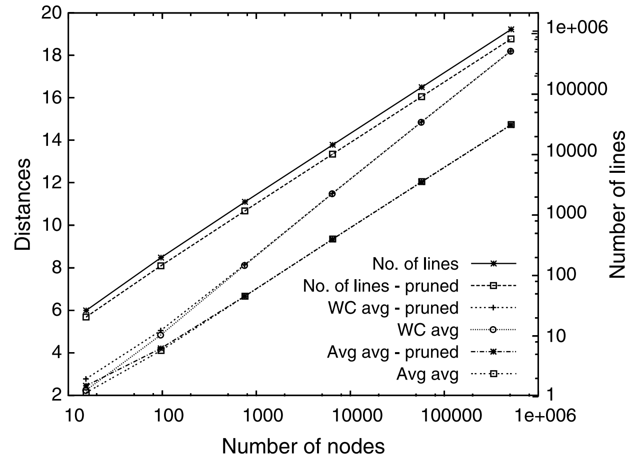

Fig. 9. Comparison of Perfect Square Mesh and pruned Perfect Square Mesh for $g=3$.

- $x_u \neq 0 \pmod{g_x}$.
- $x_v \neq 0 \pmod{g_x}$.
- $y_v \neq y_u$.
- $\lfloor x_u/g_x \rfloor = \lfloor x_v/g_x \rfloor$.

In this case, $d_0(u, v)$ is different from the corresponding value in the non-pruned structure. Two alternative path lengths are calculated, corresponding to using either $(g_x \lfloor x_u/g_x \rfloor, y_u)$ or $(g_x \lceil x_u/g_x \rceil, y_u)$ as intermediate node. The corresponding values of $d_0(u, v)$ are $|y_u - y_v| + (x_u \bmod g_x) + (x_v \bmod g_x)$ and $|y_u - y_v| + 2g_x - (x_u \bmod g_x) - (x_v \bmod g_x)$, respectively. Depending on which one results in the shortest path, the packet is sent towards one of these intermediate nodes in case routing is done using only the basic layer.

In all other cases, a shortest path is ensured by forwarding packets along the x-direction in the basic layer only if routing along the y-direction of the basic layer does not reduce the distance to the destination. It is necessary though to adjust the protection scheme in all cases.

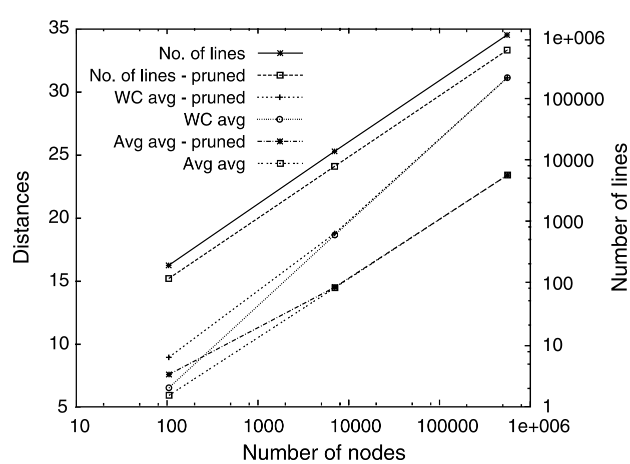

Fig. 10. Comparison of Perfect Square Mesh and pruned Perfect Square Mesh for $g=9$.

7. Embedding of structures

The studies conducted so far have mainly concentrated on topological properties of network structures, and abstractions from real networks have been made to large extents. However, in order for the structures to be applied in real-world networks, it is necessary to devise methods for mapping them onto the physical level.

In order to decide if it is possible to map some structure into a real-world network, it is necessary to consider a number of physical conditions: existing structures and ducts may be preferable to use, location and density of nodes may depend on the density of homes and offices, some places are more suitable for ducts than others, etc. Due to the dependability on such specific conditions, it is not possible to present one single method for embedding the structures. Instead we provide a number of useful tools, which can be combined in order to develop structures that are possible to use in specific cases.

7.1. Varying hierarchical depth

From a Perfect Square Mesh, we propose to omit either one or more of the lowest hierarchical layers in parts of the structure. In the following, we will show that this can be done without affecting the routing scheme, allowing for making a structure denser in some areas than in others. It also makes it possible to extend a structure after it is initially deployed, because omitted layers are easily added later on, and it allows for constructing structures of different sizes. An example of a structure with such omitted areas is shown in Fig. 11.

The areas to be left out must be selected carefully. Removing a single node or an arbitrary set of nodes may result in the routing scheme no longer guaranteeing that a shortest path is always chosen. In the following, areas of the lowest layer that can be left out are described, and a generalization to more layers presented.

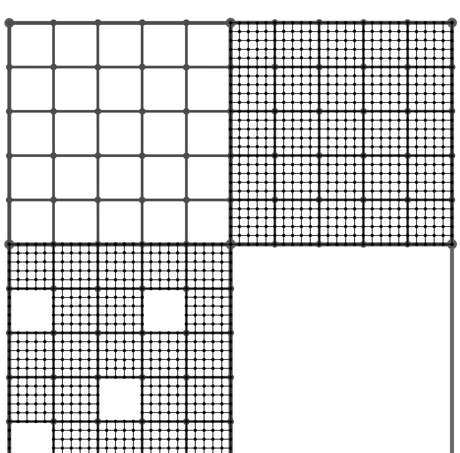

Fig. 11. Flexibility obtained by removing areas of lowest hierarchical layers.

Assume a Perfect Square Mesh, with dim_x, dim_y, and $g = g_x = g_y$, and let a and b be positive integers such that $ag \leq dim_x$ and $bg \leq dim_y$. a and b then define an area A. A node (x, y) belongs to the inner of A if and only if $(a-1)g < x < ag$ and $(b-1)g < y < bg$. A also has a border. It consists of the four corner nodes $((a-1)g, (b-1)g)$, $((a-1)g, bg)$, $(ag, (b-1)g)$, and (ag, bg) as well as four sections, each defined by the nodes with coordinates fulfilling the conditions below, respectively:

- $x = (a-1)g$ and $(b-1)g < y < bg$.
- $x = ag$ and $(b-1)g < y < bg$.
- $y = (b-1)g$ and $(a-1)g < x < ag$.
- $y = bg$ and $(a-1)g < x < ag$.

When A is left out, this is done by removing all nodes in the inner of A as well as all lines connected to at least one node in the inner of A. Nodes belonging to the border of A can be removed as well, but the conditions are slightly more complicated: no node belonging to the inner of A belongs to the inner or border of any other area, but a node belonging to the border of A may belong to the border of up to three other areas.

The four corner nodes of A cannot be removed at this step because they also belong to layer 1, but other border nodes can be removed section-wise. Within each section either none or all are removed, and if a section is removed so are all basic lines connected to at least one node in the section. A section can only be removed if no node in it is connected to a node outside it except for the corner nodes of A.

It is interesting but also rather trivial to realize that for two nodes u and v in a structure where an area of the lowest layer is left out, the distance is the same as in the full structure. Let S be a Perfect Square Mesh, and let S' be identical to S, except that one area A (say, the inner of it) is left out. This implies that there exist integers a and b such that the nodes with x coordinates $(a-1)g < x < ag$ and y coordinates $(b-1)g < y < bg$ are left out.

Assume for contradiction that there are two nodes, u and v, in S' between which a shortest path is longer than a path between u and v in S. This implies that any shortest path between u and v in S uses at least one node from the inner of A, which again implies that there exist two nodes u' and v' on the border of A such that any shortest path between them passes only nodes of the inner of A and thus is shorter in S than in S'. Assume that u' has coordinates (x_1, y_1), and that v' has coordinates (x_2, y_2), and note that neither u' nor v' are corner nodes of A. Without loss of generality it is assumed that $x_1 \leq x_2$ and $y_1 \leq y_2$.

Three cases are considered:

- $x_2 - x_1 < g$ and $y_2 - y_1 < g$. If $x_1 = x_2$ or $y_1 = y_2$, assume without loss of generality that $x_2 = x_1$. $y_2 - y_1 < g$ and thus $x_1 = (a-1)g$ or $x_1 = ag$, and in both cases a path exists: $(x_1, y_1), (x_1, y_1+1), \ldots, (x_1 = x_2, y_2)$. This path has length $y_2 - y_1$, equal to $d_0(u', v')$. So, assume $x_1 \neq x_2$

and $y_1 \neq y_2$. Then $x_1 = (a-1)g$ and $y_2 = bg$ or $x_2 = bg$ and $y_1 = (b-1)g$. In the first case there exists a path (x_1, y_1), $(x_1, y_1+1),\ldots,(x_1, y_2)$, $(x_1+1, y_2),\ldots,(x_2, y_2)$, and in the second case a path (x_1, y_1), $(x_1+1, y_1),\ldots,(x_2, y_1)$, $(x_2, y_1+1),\ldots,(x_2, y_2)$. Both paths are of length $x_2 - x_1 + y_2 - y_1$, and thus equal to $d_o(u', v')$, and a contradiction is obtained.
- $x_2 - x_1 = g$ and $y_2 - y_1 < g$. This implies $x_1 = (a-1)g$ and $x_2 = ag$. Two paths p_1 and p_2 are constructed, where $|p_1|$ and $|p_2|$ denote their respective lengths. p_1 contains the layer 1 line connecting $(x_1, (b-1)g)$ and $(x_2, (b-1)g)$ while p_2 contains the layer 1 line connecting (x_1, bg) and (x_2, bg). Thus, $p_1 = (x_1, y_1), (x_1, y_1-1),\ldots,(x_1, (b-1)g)$, $(x_2, (b-1)g)$, $(x_2, (b-1)g+1),\ldots,(x_2, y_2)$ and $p_2 = (x_1, y_1), (x_1, y_1+1),\ldots,(x_1, bg), (x_2, bg), (x_2, bg-1),\ldots,(x_2, y_2)$. $|p_1| = y_1 + y_2 - 2(b-1)g + 1$ and $|p_2| = 2bg - y_1 - y_2 + 1$. This implies that $|p_1| + |p_2| = 2 + 2\,g$. Since $x_2 - x_1 = g$, $d_0(u', v') \geq g$. It must then be true that $|p_2| > \mathrm{g}$. If $|p_2| \geq g+2$ then $|p_1| \leq g$, and thus $|p_2| = g+1$. Since $|p_2| = 2bg - y_1 - y_2 + 1$ this implies that $2bg - y_1 - y_2 = g$. Since g is odd, $y_1 - y_2$ is odd, but then $y_1 \neq y_2$, implying that $d_0(u', v') = g + y_2 - y_1 \geq g + 1$, but then the distance is not shorter in S than in S', a contradiction.
- $x_2 - x_1 < g$ and $y_2 - y_1 = g$. This case is similar to the case above, and a final contradiction obtained.

The property also holds if one or more sections of border nodes of A are removed: these nodes are only connected to each other and to nodes of the first layer, and their removal does therefore not increase any distances.

Since the distance between no pair of nodes is increased in S' compared to S, the existing routing scheme only needs to be changed in nodes of borders of left-out areas. While initially algorithms similar to the lake algorithms seem necessary, it can actually be solved by simpler means.

According to the proof stated, two approaches will both result in a shortest path being chosen in all cases:

- Routing always using the highest hierarchical layer in which a shortest path exists.
- Whenever a packet is forwarded from a node of the border of a left-out area, and it should normally be forwarded to a node of the inner of this area, it is instead forwarded towards the nearest higher-hierarchy node.

The principle described can be used for other layers than the basic layer. However, no node of layer i can be removed if it is connected by any lines of a layer $<i$.

For a layer i, and positive integers a and b such that $ag^i \leq dim_x$ and $bg^i \leq dim_y$, an area A of layer i is defined by the nodes such that $(a-1)g^i \leq x \leq ag^i$ and $(b-1)g^i \leq y \leq bg^i$. A node (x, y) belongs to the inner of A if and only if $(a-1)g^i < x < ag^i$ and $(b-1)g^i < y < bg^i$. The inner of A can be removed only if all nodes not belonging to layer i within the inner of A have been removed. Sections of border nodes of A are defined as in the basic layer, and can be removed under the same conditions with the addition that a node cannot be removed if it is connected by lines of a layer $<i$. Nodes on the corner of A cannot be removed at this step because they are also nodes of layer $i+1$. The nodes of layer n_{H} are all corner nodes of S, and it makes no sense to remove them, since this would imply a removal of the whole structure.

The distances of a structure with various left-out layers may differ from those of a Perfect Square Mesh, depending on which layers a left out, and further analysis is needed in each case to clarify this. However, for each such structure the distance between each pair of nodes equals the distance between the same pair of nodes in a full Perfect Square Mesh. Since the second approach to pruning described in Section 6 only applies to the lowest hierarchical layer, the gains of pruning may be less significant if combined with leaving out layers, depending on the extent to which layers are left out.

It is also possible to merge Perfect Square Meshes side-by-side in order to obtain structures with different sizes and shapes than the Perfect Square Mesh while still maintaining $g_x = g_y$. Actually, this corresponds to leaving out a number of lowest layers as well as removing the highest layer. Leaving out the highest layer does affect performance, and it should also be noted that it is possible to do so only if the structure remains connected. Furthermore, if the highest hierarchical layer is removed, care must be taken as to which areas are left out, unless the routing scheme is revised.

7.2. Varying granularity

Hierarchical structures different from the Perfect Square Mesh (i.e. $g_x \neq g_y$) also result in logarithmic dependencies between the number of nodes and distances in the structure. An example of such a skew structure with $g_x = 11$ and $g_y = 3$ is shown in Fig. 12. Applying this in combination with leaving out layers makes it necessary to refine the routing scheme in order to ensure that a shortest path is always chosen. As shown in Fig. 13, the structures with $g_x = 11$ and $g_y = 3$ have performances close to those of the Perfect Square Mesh with $g = 9$. However, as can be seen in Tables 2 and 3, the skew mesh requires relatively more hierarchical lines and contains more edge nodes than the Perfect Square Mesh, resulting in more expensive and less robust structures.

It is also possible to vary the granularity from layer to layer: instead of choosing one value of g_x and one value of g_y, g_{x_i} and g_{y_i} are chosen for every i, $0 < i \leq n_{\mathrm{H}}$. Now $dim_x = g_{x_1} g_{x_2} \cdots g_{x_{n_{\mathrm{H}}}}$ and $dim_y = g_{y_1} g_{y_2} \cdots g_{y_{n_{\mathrm{H}}}}$. The definition of

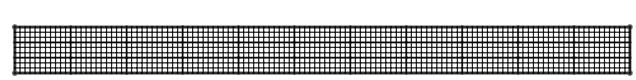

Fig. 12. A structure with $g_x = 11$, $g_y = 3$, and $n_H = 2$.

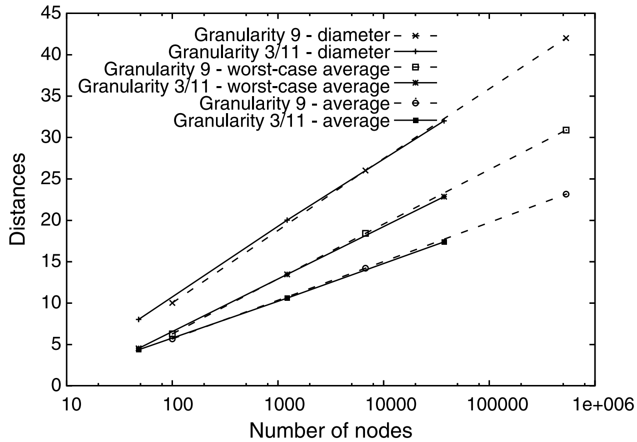

Fig. 13. Performances of skew structures with $g_x = 11$ and $g_y = 3$ compared to performances of Perfect Square Mesh with $g = 9$.

Table 2
Contribution from hierarchical lines to the average node degree and the relative number of edge nodes for structures with $g_x = 11$ and $g_y = 3$

Size	Avg. degree from h-lines	% of nodes on edge
$n_H = 1$, 48 nodes	0.17	58.3
$n_H = 2$, 1220 nodes	0.14	21.3
$n_H = 3$, 37 296 nodes	0.13	7.28

hierarchical nodes and lines as well as the routing scheme must be adjusted accordingly.

We also propose another promising extension of the Perfect Square Mesh, the extended mesh, which maintains the square shape, but the size is redefined, such that $-\lfloor g^{n_H}/2\rfloor \leq x,\ y \leq \lfloor 3g^{n_H}/2\rfloor$ as seen in Fig. 14. It was evaluated for $g = 5$, and performance was close to the Perfect Square Mesh with the same granularity, as can be seen in Fig. 15. Using this approach to various extents makes it possible to construct structures of different sizes than the Perfect Square Mesh definitions allow for.

8. Applying the theory to real-world network planning

Compared to simple ring and tree networks, applying 4-regular grid structures in large-scale communication networks obviously requires more careful planning, even with the proposed extensions and embedding tools. In this section, we state how such a planning process can be carried

Table 3
Contribution from hierarchical lines to the average node degree and the relative number of edge nodes for Perfect Square Mesh with $g = 9$

Size	Avg. degree from h-lines	% of nodes on edge
$n_H = 1$, 100 nodes	0.080	36.0
$n_H = 2$, 6724 nodes	0.055	4.82
$n_H = 3$, 532 900 nodes	0.051	0.55

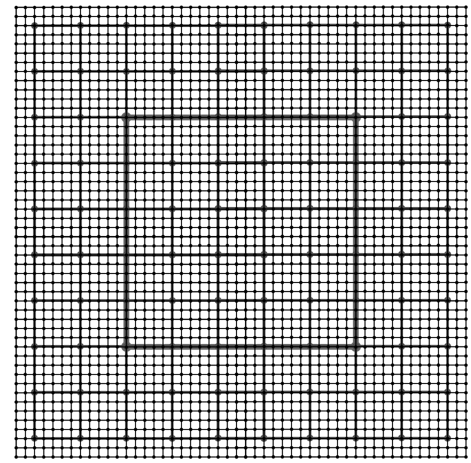

Fig. 14. Extended structure with $g = 5$ and $n_H = 2$.

out, leading to an optimization problem. Solving this problem is beyond the scope of this paper. We are currently working on it [16], and also strongly encourage other researchers to do so.

The first step in the planning process is to identify the network termination points (NTPs). The nature of these points depends on which kind of network is to be planned; in the case of access networks, we include private households, businesses and public institutions as NTPs. In order to perform the required analyses, the placement of NTPs should be available in some digital format, e.g. as GIS data. In many countries including Denmark, GIS data are available at a very high quality.

After having identified the network nodes, the next and more complex step is to identify the potential lines. In most cases, network lines should preferably be placed along existing infrastructures such as roads, paths and railway tracks, which can be identified by GIS data. These usually connect the NTPs in some way, and thus form a set of potential lines. It may be possible to establish lines away from existing infrastructures (i.e. crossing bare fields), but

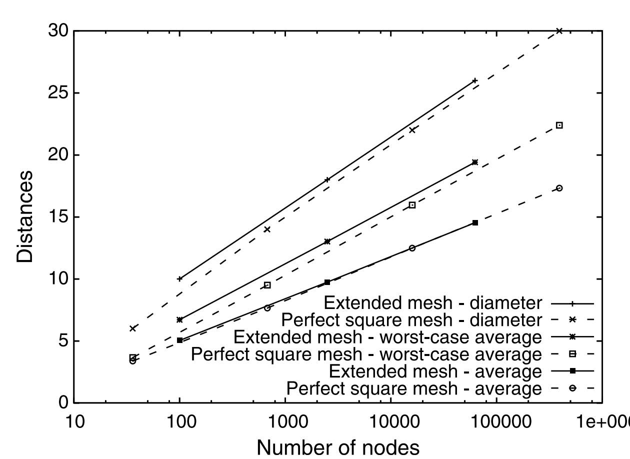

Fig. 15. Performances of extended structures with $g = 5$ compared to performances of Perfect Square Mesh with $g = 5$.

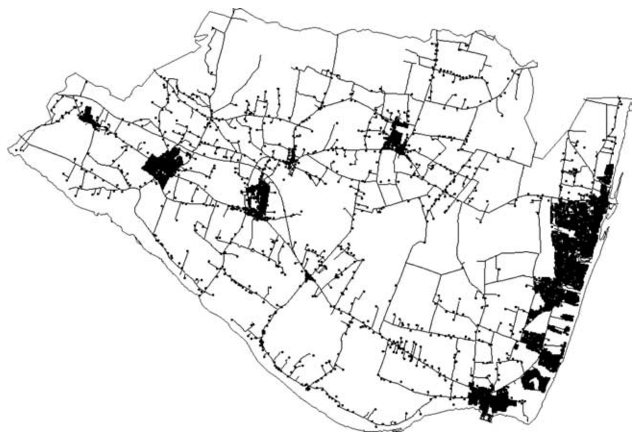

Fig. 16. The Municipality of Hals, with Network Termination Points and roads obtained from GIS data.

some areas may be impracticable and expensive, or even impossible, to establish lines across. A visualization of the available GIS data of roads and NTPs of the Municipality of Hals is shown in Fig. 16. It is a rural Danish municipality with a total of approximately 7500 NTPs, including approximately 3400 summerhouses.

Given the nodes and the potential lines, the remaining problem can be split into two subproblems:

- Find a suitable physical topology, given the 4-regular grid structure with proposed extensions.
- Map this topology onto the potential paths.

These subproblems cannot be considered separately, because the choice of suitable physical topology will be heavily dependent on which potential paths are available.

It adds to the complexity of the problems that the set of potential paths is not strictly defined, since lines may be established without following existing infrastructures, sometimes at significantly greater costs. Furthermore, some techniques easing the mapping can be used with care, such as co-locating two or more lines along a single road segment. However, doing so inevitably compromises the independency of lines, and thus affects the network reliability.

As can be seen from Fig. 16, it is non-trivial to perform this topology selection and mapping. At least some semi-automatic tools must be available. This also implies that such tools are necessary in order to determine how well the proposed extensions perform with respect to real-world network planning.

9. Conclusion and further work

This paper has discussed a number of aspects on how to apply the 4-regular grid structures in large-scale communication access networks. Due to the nice properties of these structures, including simple routing and restoration schemes, they have been deployed in multiprocessor systems for decades. However, they are not directly applicable in large-scale networks. In this paper, four major barriers have been dealt with, namely restoration/-protection, scalability, cost, and embeddability. For each barrier, extensions and schemes have been proposed to facilitate the use of 4-regular grid structures in large-scale access networks. These form a tool-case, which can be used as a base for developing semi- or full automatic tools for network planning.

With regard to restoration, a simple scheme was introduced, facilitating fast and easy restoration by the concept of lake algorithms: if only a small part of the network is failing, only small tables are needed, and consequently only little communication overhead is necessary in order to create and maintain these tables.

A simple way of establishing up to four independent paths was introduced for protection. In the basic structure (and generally within each hierarchical layer) the paths are node independent, but in the hierarchical extension, nodes are shared wherever a packet moves from one hierarchical layer to another. In a few cases only two or three independent paths can be set up. This is the case if the source or destination nodes are edge/corner nodes, or if the routing is done in a pruned structure and either source or destination node belongs only to the basic layer.

A hierarchical extension was proposed in order to deal with the scalability problem. This approach results in logarithmic dependencies between the number of nodes and the distances in the structure. Depending on the chosen granularity, the number of hierarchical lines was small compared to the number of lines in the basic structure.

To deal with the cost problem, an approach was suggested where in a hierarchical structure some lines of the basic layer are left out. As with any other way of reducing the number of lines, it reduces the connectivity, but the distances in the structure were hardly affected.

Due to the many physical and demographical conditions and constraints, it is not possible to provide one general method for embedding these structures in real-world networks. Instead, the different schemes and extensions proposed in the paper were presented as a tool-case, allowing for various approaches to be combined. Further studies should deal with developing this into semi-automatic and in the longer term full automatic tools for embedding the structures in real-world networks. Such tools can be based on GIS data, and their performance should be verified through case studies under different geographical and demographic conditions.

Currently, the method proposed in this paper applies only to wired networks. Based on GIS information, and taking the coverage of wireless networks into account, the method could be extended to cover also wireless and combined wired and wireless networks. Following this track, a significant contribution would be to develop models and tools supporting planning of networks for 4G and beyond.

We expect development of automatic tools for network planning to become an increasingly important research area over the coming years. In addition to properties strictly related to physical topologies and embeddings, such tools must take into account a number of other aspects including security, traffic models, and advanced financial models. The latter is particularly important, as it can be used for fitting network planning strategies with business models and investment strategies.

Acknowledgements

We would like to thank University College Dublin for hosting Jens Myrup Pedersen and Thomas Phillip Knudsen as visiting researchers during the work of this paper, as well as Nikita Schmidt for his helpful comments.

References

[1] O.B. Madsen, J.D. Nielsen, H. Schiøler, Convergence, Proceedings of RTLIA 2002, the First International Workshop on Real-Time LANs in the Internet Age, Vienna, Austria, June 2002 pp. 101–104.

[2] K.-H. Han, S. Kim, Y.-J. Kim, J.-H. Kim, Internet control architecture for internet-based personal robot, Autonomous Robots 10 (2) (2001) 135–147.

[3] L. Xiao-ming, Y. Can-Jun, C. Ying, H. Xu-dong, Hybrid event based control architecture for tele-robotic systems controlled through internet, Journal of Zhejiang University Science 5 (3) (2004) 296–302.

[4] N. Xi, T.-J. Tarn, Planning and control of internet-based teleoperation, Proceedings of the SPIE—The International Society for Optical Engineering 3524 (1998) 189–195.

[5] A.H. Dekker, B.D. Colbert, Network robustness and graph topology, Proceedings of ACSC04, the 27th Australasian Computer Science Conference, Dunedin, New Zealand, January 2004 pp. 359–368.

[6] I. Stojmenovic, Honeycomb networks: topological properties and communication algorithms, IEEE Transactions on Parallel and Distributed Systems 8 (10) (1997) 1036–1042.

[7] J.M. Pedersen, T.P. Knudsen, O.B. Madsen, Topological routing in large-scale networks, Proceedings of IEEE/ICACT 2004, The Sixth International Conference on Advanced Communication Technology, Korea, February, vol. 2, 2004 pp. 911–916.

[8] J.M. Pedersen, A. Patel, T.P. Knudsen, O.B. Madsen, On Hierarchical extensions of 4-regular grid network structures, Proceedings of PDPTA '04, The International Conference on Parallel and Distributed Processing Techniques and Applications, Las Vegas, USA, June, vol. 2, 2004 pp. 738–743.

[9] J.M. Pedersen, T.P. Knudsen, O.B. Madsen, An evaluation framework for large-scale network structures, Proceedings of IT&T Conference 2004, Information Technology and Telecommunications Annual Conference 2004, Limerick, Ireland, October 2004 pp. 191–199.

[10] Z. Jiang, J. Wu, Fault-tolerant broadcasting in 2-D wormhole-routed meshes, The Journal of Supercomputing 25 (2003) 255–275.

[11] S.-D. Wang, P.-H. Sui, Fault-tolerant routing in two-dimensional mesh networks with less-restricted fault patterns, Proceedings of the 2001 Pacific Rim International Symposium on Dependable Computing, December 2001 pp. 111–118.

[12] J. Wu, A fault-tolerant and deadlock-free routing protocol in 2D meshes based on odd-even turn model, IEEE Transactions on Computers 52 (9) (2003) 1154–1158.

[13] A. Fei, G. Pei, R. Liu, L. Zhang, Measurements on delay and hop-count of the internet, Proceedings of Globecom 1998, Sydney, Australia 1998.

[14] H. Burch, B. Cheswick, Mapping the internet, IEEE Computer 32 (4) (1999) 97–98 +102.

[15] O.B. Madsen, T.P. Knudsen, J.M. Pedersen, SQoS as the base for the next generation global infrastructure, Proceedings of IT&T Conference 2003, Information Technology and Telecommunications Annual Conference 2003, Letterkenny, Ireland, October 2003 pp. 127–136.

[16] T.M. Riaz, J.M. Pedersen, O.B. Madsen, SQoS based planning using 4-regular grid for optical fiber networks, Proceedings of IEEE/ICACT 2005, The Seventh International Conference on Advanced Communication Technology, Korea, February, vol. 2, 2005 pp. 1287–1291.