Available online at www.sciencedirect.com
SCIENCE DIRECT®

Discrete Applied Mathematics 150 (2005) 99–120

DISCRETE APPLIED MATHEMATICS

www.elsevier.com/locate/dam

Self-spanner graphs☆

Serafino Cicerone[a], Gabriele Di Stefano[a], Dagmar Handke[b]

[a] *Dipartmento di Ingegneria Elettrica, Università dell'Aquila, I-67040 L'Aquila, Italy*
[b] *Fakultät für Mathematik und Informatik, University of Konstanz,78457 Konstanz, Germany*

Received 30 September 2002; received in revised form 13 September 2004; accepted 26 April 2005
Available online 24 June 2005

Abstract

We introduce the (k, ℓ)-*self-spanners* graphs to model non-reliable interconnection networks. Such networks can be informally characterized as follows: if at most ℓ edges have failed, as long as two vertices remain connected, the distance between these vertices in the faulty graph is at most k times the distance in the non-faulty graph. By fixing the values k and ℓ (called *stretch factor* and *fault-tolerance*, respectively), we obtain specific new graph classes. We first provide characterizational, structural, and computational results for these classes. Then, we study relationships between the introduced classes and special graphs classes (distance-hereditary graphs, cographs, and chordal graphs), and common network topologies (grids, tori, hypercubes, butterflies, and cube-connected cycles) as well.

Keywords: Special graph classes; Spanners; Stretch number; Interconnection networks; Fault tolerance

1. Introduction

The main function of a network is to provide connectivity between the sites. In many cases it is crucial that connectivity is preserved even in the case of faults in either sites or links. Accordingly, a major concern in network design is fault-tolerance and reliability. The large amount of research dedicated to fault-tolerant network design is basically based on two approaches. The first approach consists of techniques that add redundancy to the

☆ Preliminary results of this paper has been presented at the 10th International Symposium on Algorithms and Computation, December 16–18, Madras, India, 1999 [10].

E-mail addresses: cicerone@ing.univaq.it (S. Cicerone), gabriele@ing.univaq.it (G. Di Stefano), dagmar.handke@web.de (D. Handke).

0166-218X/$ - see front matter
doi:10.1016/j.dam.2005.04.004

desired architecture by introducing new network components (e.g., see [6,17,26]). In the second approach, the fault-tolerance is achieved not by adding redundancy to the network, but by using the non-faulty part of the network to simulate the desired architecture (e.g., see [2,11,21]).

Following a different approach, in this work we are interested in networks in which distances between sites remain small even in the case of faulty links or sites. Hence, we do not start with a fixed target graph, nor do we allow a re-structuring of the graph; we keep the identification of each vertex fixed. As the underlying model, we use unweighted graphs, and measure the distance in a network in which faults have occurred by a shortest path in the subnetwork that is induced by the non-faulty components.

To study such networks, we introduce new classes of graphs that guarantee constant stretch factors k even when a multiple number of *edges* have failed. In a first step, we do not limit the number of edge faults at all, that is we allow for *unlimited* edge faults. The graphs that model this case are called *k-self-spanners* and the corresponding class is denoted by $\mathrm{SS}(k)$. Secondly, we examine the case where the number of edge faults is *bounded* by a constant ℓ. For this, we introduce the class $\mathrm{SS}(k,\ell)$ of (k,ℓ)-*self-spanner* graphs. In both cases, the name is motivated by strong relationships to the concept of k-*spanners* [23].

A network modeled as a (k,ℓ)-self-spanner graph can be informally characterized as follows: *if at most ℓ edges have failed, as long as two vertices remain connected, the distance between these vertices in the faulty graph is at most k times the distance in the non-faulty graph*. By fixing the values k and ℓ (called *stretch factor* and *fault-tolerance*, respectively), we obtain a specific new graph class. The goal of this work is twofold: (1) to provide characterizational, structural and computational results for the new classes, and (2) to study relationships between the introduced classes and common network topologies, and special graphs classes as well.

Related works: As observed above, several papers present results about classical fault-tolerant network design. Recently, some papers introduced and analyzed networks according to the approach followed in this work. In [1,7–9], authors have considered networks that guarantee constant delay factors even when an *unlimited* number of *vertices* fail. In particular, in [7,9] they study graphs in which the induced distance function is bounded by a *multiplicative constant*, while in [1,8] the induced distance function is bounded by an *additive constant*. In [13], author gives characterizations for graphs in which *no delay* occurs in the case that a *single* vertex fails. These graphs are called *self-repairing*. Unfortunately, in all cases these results do not carry over to the dual case of *edge faults*. In [15], a different notion of fault-tolerance and reliability is considered. There, the goal was to find subgraphs with a certain structure in a given graph such that a constant distance guarantee can be given.

Results: As a preliminary step, we first introduce and investigate *k-self-spanners*, providing different strict characterizations. Such results prove that the recognition problem for the class $\mathrm{SS}(k)$ is polynomially solvable for $k \leqslant 3$, and that it is hard in general (for k not fixed).

As main contribution, we introduce and investigate the (k,ℓ)-self-spanners graphs. Characterizational and structural results are used to tackle the main problem: deciding whether a given graph is a (k,ℓ)-self-spanner. This problem is $\mathcal{NP}$-complete for the general case where k and ℓ are part of the input and remains $\mathcal{NP}$-complete if $k \geqslant 5$ is fixed.

However, if $k \leqslant 2$ is fixed or if $\ell \geqslant 0$ is fixed, then there are polynomial time algorithms to solve it. For $k = 3$ the problem is polynomial for $(\ell + 1)$-edge-connected graphs, $\ell > 0$. In conclusion, it remains to be settled for general graphs when $2 < k \leqslant 4$.

At a second phase, we define some sufficient conditions to guarantee that a given graph belongs to $\mathrm{SS}(k, \ell)$ for some k and ℓ. These conditions are used to show that some well known graph classes such as distance-hereditary, cographs, and chordal graphs (e.g., see [5]) exhibit strong self-spanner properties, by providing upper bounds on the trade-off between stretch factor and fault-tolerance.

Finally we show how the new graph classes of (k, ℓ)-self-spanners fit into the context of some popular network topologies. To this end, we first study self-spanner properties of graphs built by means of Cartesian product. Then, we use these properties to show that *grids*, *tori*, and *hypercubes* exhibit strong self-spanner properties, in particular for small fault-tolerance values. Bounded-degree approximations of the hypercube such as *connected cycles* and *butterflies*, however, result in big stretch factors even in the case of small fault-tolerance values.

The remainder of this paper is organized as follows. Notation and basic concepts used in this work are given in Section 2. Sections 3 and 4 introduce and investigate k-self-spanners and (k, ℓ)-self-spanners, respectively. In Section 5, we provide self-spanner properties of special graph classes. Section 6 shows how Cartesian product affects self-spanner properties of graphs; this result is used to investigate relations between (k, ℓ)-self-spanners and popular network topologies. Finally, in Section 7, we give some final remarks.

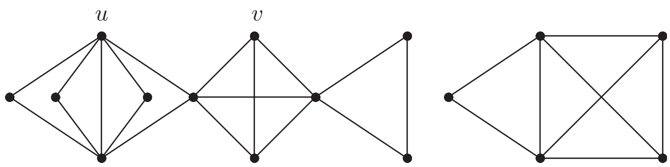
2. Basic notions

In this work, we use standard notation for graphs (cf. [16]). Let $G = (V, E)$ be a simple (i.e. without multiple edges or loops), unweighted, and undirected graph. Let n denote the number of vertices, and let m denote the number of edges. The *set of vertices* (and *set of edges*, resp.) of G is denoted by $V(G)$ (and $E(G)$, resp.). A subgraph $H = (V', E')$ of $G = (V, E)$ (with $V' \subseteq V$ and $E' \subseteq E$) is called *spanning* if $V = V'$. If $R \subseteq V(G)$, then $G[R]$ denotes the subgraph of G induced by R. $G - e$ where $e \in E(G)$ is the graph obtained from G by deleting edge e. The neighborhood $N_G(v)$ of a vertex v in G is the set of all vertices that are adjacent to v in G.

The *distance* between two vertices u and v in G is denoted by $d_G(u, v)$, and corresponds to the number of edges in a shortest path between u and v. If we consider *cycles*, we always mean *simple* cycles, i.e. cycles in which each vertex appears at most once. The *length of a cycle* is the number of its vertices or its edges, resp. An edge is a *chord* of a cycle C if it connects two non-adjacent vertices of C. A cycle C in G is called *induced* if $G[V(C)] = C$, i.e. if C does not contain chords.

C_n denotes the *induced cycle graph* (also called ring) with n vertices. Conversely, $\mathbb{C}_n$ denotes a cycle on n vertices that may contain an arbitrary number of chords. Moreover, P_n is the *path graph* on n vertices. K_n is the *complete graph* (or *clique*) on n vertices, and $K_{n,m}$ is the *complete bipartite graph* with a bipartition on n and m vertices.

For a connected graph, an *articulation vertex* is a vertex whose deletion disconnects the graph. A graph is called *biconnected* (or 2-*vertex-connected*) if it has no articulation



(a) (b)

Fig. 1. (a) A 3-self-spanner graph and (b) a 4-self-spanner graph.

vertex. It is called ℓ*-vertex-connected* if there is no subset of vertices S of size $\ell - 1$ such that $G[V \setminus S]$ is disconnected. A graph is ℓ*-edge-connected* if no deletion of $\ell - 1$ edges disconnects it. An edge e of G is called *bridge* if $G - e$ is disconnected. Observe that an ℓ-edge-connected graph does not contain a bridge if $\ell \geqslant 2$. A *block* of a graph is a maximal biconnected subgraph.

A *diamond* is a biconnected graph formed by two possibly adjacent vertices u and v, which are connected by $K \geqslant 2$ disjoint paths of length 2 (see for example the leftmost block in Fig. 1(a)).

For any fixed rational $k \geqslant 1$, a k*-spanner* of an unweighted graph G is a spanning subgraph S in G such that the distance between every pair of vertices in S is at most k times their distance in G. The parameter k is called *stretch factor*. We say that an edge e is *covered* if in S there exists a path of length at most k that connects the endpoints of e. Such a path is called a *covering path*. Since in particular each edge has to be covered in a k-spanner, it is clear that in unweighted graphs S is a k-spanner of G if and only if S is a $\lfloor k \rfloor$-spanner of G. Thus it suffices to consider integer stretch factors k.

Moreover, in order to prove that a given spanning subgraph is a k-spanner, we do not have to consider all pairwise distances of the vertices. It suffices to look only at edges of the graph that are not part of the spanning subgraph.

Lemma 2.1 (*Peleg and Schaeffer [23]*). *A subgraph $S = (V, E')$ of a graph $G = (V, E)$ is a k-*spanner *of G if and only if all edges that do not belong to S are covered, i.e.,*

$$d_S(u, v) \leqslant k \quad \text{for every edge } e = \{u, v\} \in E \setminus E'. \tag{1}$$

The concept of spanners has been introduced by Peleg and Ullman in [24], where they used spanners to synchronize asynchronous networks. One of the many other applications for spanners are communication networks, where one is interested in finding a sparse subnetwork that nevertheless guarantees a constant delay factor. Further results on k-spanners and variants thereof can be found for example in [18].

3. *k*-self-spanner

In this section, we examine a class of graphs that guarantees constant delays even in the case of an *unlimited* number of *edge faults*.

Definition 3.1. For any fixed integer $k \geqslant 1$, a graph $G=(V,E)$ is a *k-self-spanner* if for every subgraph $G'=(V,E')$ of G:

$$d_{G'}(u,v) \leqslant k \cdot d_G(u,v) \quad \text{for all } u,v \in V \text{ that are connected in } G'. \tag{2}$$

The class of all k-self-spanners is denoted by $\mathrm{SS}(k)$. The parameter k is called *stretch factor*. For a graph G, $minS(G)$ denotes the smallest k such that $G \in \mathrm{SS(k)}$.

For instance, the graph G in Fig. 1(a) belongs to SS(3), but as $minS(G)=3$, it does not belong to $\mathrm{SS}(2)$. If G' is achieved from G by adding the edge $\{u,v\}$, then $minS(G')=6$, and thus G' does not belong to $\mathrm{SS}(3)$ anymore. The graph in Fig. 1(b) belongs to $\mathrm{SS}(4)$, but not to $\mathrm{SS}(3)$. The previous definition works equally well for connected and disconnected graphs; but it is obvious that we can restrict our analysis to connected graphs in the following.

Notice that k-self-spanner graphs form a hierarchy of graph classes: if $1 \leqslant k \leqslant k'$, then $\mathrm{SS(k)} \subseteq \mathrm{SS(k')}$. A network modeled as a graph $G \in \mathrm{SS(k)}$ is characterized as follows: if G' is the graph resulting by removing from G an arbitrary number of faulty edges, then the distance between two connected vertices in G' is at most k times their distance in G. By replacing 'edges' by 'vertices' in this characterization we get the class of *k-bounded induced distance graphs*, which have been introduced in [7] and deeply investigated in [7,9].

The following lemma motivates the name *k-self-spanner* (by showing a strong relationship with the concept of k-spanners) and provide useful characterizations.

Lemma 3.2. *Let $G=(V,E)$, and $k \geqslant 1$. The following statements are equivalent*:

1. $G \in \mathrm{SS(k)}$;
2. *every* connected spanning *subgraph $G'=(V,E')$ of G is a k-spanner of G*;
3. *every* connected *subgraph $G'=(V',E')$ of G is a k-spanner of $G[V']$*;
4. *every simple cycle of G has at most $k+1$ edges*;
5. *for every edge $e=\{u,v\} \in E$, a longest simple path between u and v in G has length at most k.*

Proof. [1 ⇒ 2] and [4 ⇒ 5] Trivial.

[2 ⇒ 3] Assume that every connected spanning subgraph of G is a k-spanner of G and there is a connected (not necessarily spanning) subgraph $G'=(V',E')$ of G such that $d_{G'}(u,v) > k \cdot d_{G[V']}(u,v)$ for two vertices $u,v \in V'$. Expand G' to a connected spanning subgraph $G''=(V,E'')$ by linking missing vertices of G to V' such that these vertices do not lie on a cycle (this is always possible because G is connected). Then, G'' is a spanning subgraph of G and $d_{G''}(u,v) > k \cdot d_G(u,v)$, a contradiction.

[3 ⇒ 4] By contradiction, let us assume that there exists a simple cycle C in G with at least $k+2$ edges. Let $\{u,v\}$ be an edge of C, and let G' be the subgraph of G induced by the edges of C except $\{u,v\}$. Hence, $d_{G'}(u,v) \geqslant k+1$. This inequality implies that G' is not a k-spanner of $G[V(G')]$, a contradiction.

[5 ⇒ 1] By contradiction, let us assume that $G \notin \mathrm{SS(k)}$. By Part 3, there exists a connected subgraph $G'=(V',E')$ of G such that G' is not a k-spanner of $G[V']$. By Lemma 2.1, there exists an edge $e=\{u,v\}$ in $G[V']$ that does not belong to E' such that $d_{G'}(u,v) > k$. This results in a simple path of length at least $k+1$, a contradiction. □

Part 5 of the lemma above implies that the class of k-self-spanners is closed under taking subgraphs.

3.1. Complexity results

Since $\mathrm{SS(k)} \subseteq \mathrm{SS(k')}$, $1 \leqslant k \leqslant k'$, and since there always exists an integer k'' such that $G \in \mathrm{SS(k'')}$ for a given graph G, the problem of determining the smallest class which a graph belongs to naturally arises. This recognition problem can be formally defined as follows:

Problem 1. MINIMUM SELF-SPANNER: *Given a graph G and an integer* $k \geqslant 1$, *does G belong to* $\mathrm{SS}(k)$, *i.e.,* $minS(G) \leqslant k$?

In what follows we prove that: (1) MINIMUM SELF-SPANNER is hard in general, and (2) there exist strict characterizations for $\mathrm{SS}(k)$ for small k that lead to efficient recognition algorithms. These results are based on Lemma 3.2 and on the following lemma, respectively.

Lemma 3.3. *Let G be a graph. Then following characterizations hold*:

1. $G \in \mathrm{SS}(1)$ *if and only if every block of G is a* K_2 (*i.e., G is a tree*);
2. $G \in \mathrm{SS}(2)$ *if and only if every block of G is a* K_3 *or* K_2;
3. $G \in \mathrm{SS}(3)$ *if and only if every block of G is a diamond,* K_4, K_3, *or* K_2.

Proof. The characterizations of $\mathrm{SS}(1)$ and $\mathrm{SS}(2)$ can be derived from Definition 3.1.

Concerning $\mathrm{SS}(3)$, notice that $minS(K_4) = 3$ and $minS(D) = 3$ for any diamond D. For the other direction, consider a block G' of G. If G' contains at most 4 vertices we are done, so assume that G' contains at least 5 vertices. Since G' is biconnected, then it contains a cycle C; according to Part 4 of Lemma 3.2, C has at most 4 vertices. So, assume $C = (a, b, c, d)$. To avoid to generate cycles with 5 vertices, a vertex u such that $u \in G'$ and $u \notin C$ has to be adjacent to 2 non-adjacent vertices of C (w.l.o.g., assume u adjacent to a and c). At this point, other vertices can be adjacent to a and c only. Finally, C may have one chord only, and such a chord joins a and c. It is easy to see that the component G' is a diamond. □

Theorem 3.4. MINIMUM SELF-SPANNER *is co-$\mathcal{NP}$-complete. Moreover, testing whether a graph G belongs to* $\mathrm{SS}(k)$, *for each fixed* $k \leqslant 3$, *can be performed in polynomial time*.

Proof. As mentioned in [14] (ND28), the following LONGEST CIRCUIT Problem is $\mathcal{NP}$-complete: Given a graph $G = (V, E)$ and a positive integer $K \leqslant |V|$, is there a simple cycle in G of length K or more? By Part 4 of Lemma 3.2 this is exactly the complementary problem of MINIMUM SELF-SPANNER, and hence MINIMUM SELF-SPANNER is *co*-$\mathcal{NP}$-complete. The last part of the statement is a consequence of Lemma 3.3. □

It could be interesting to study MINIMUM SELF-SPANNER for $k \geqslant 4$ fixed. Observe that Lemmas 3.2 and 3.3 show that, if we ask for a class $\mathrm{SS}(k)$ that contains non-trivial networks, we have to pay for a large stretch factor k. This fact is due to the strong constraint for the fault-tolerance that we have used in the definition of k-self-spanners: a k-self-spanner has

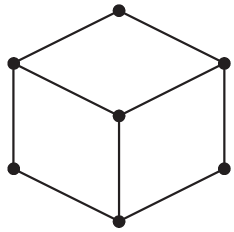


Fig. 2. The opaque cube OC.

to guarantee for a fixed bounded stretch factor even in case of an *unlimited* number of edge faults. In the light of applicability, this assumption is overly pessimistic; usually a *limited* number of edge faults is sufficient. Thus, the model of (k, ℓ)-self-spanners as treated in the following section is much more realistic.

4. $(\boldsymbol{k}, \ell)$-self-spanners

In this section, we consider limited fault-tolerance, that is we study networks in which at most ℓ edges may fail. To model these networks, we introduce the following graphs:

Definition 4.1.

1. For any fixed integer $k \geqslant 1$ and fixed integer $\ell \geqslant 0$, a graph $G = (V, E)$ is a (k, ℓ)-*self-spanner* if for every subgraph $G' = (V, E')$ of G with $|E'| \geqslant |E| - \ell$ and $E' \subseteq E$:

$$d_{G'}(u, v) \leqslant k \cdot d_G(u, v) \quad \text{for all } u, v \in V \text{ that are connected in } G'.$$

   The class of all (k, ℓ)-self-spanners is denoted by $\mathrm{SS}(k, \ell)$. The parameter k is called *stretch factor*, and the parameter ℓ is called *fault-tolerance* of the class $\mathrm{SS}(k, \ell)$.
2. For a graph G, $minS_\ell(G)$ denotes the smallest k such that $G \in \mathrm{SS}(k, \ell)$ (i.e., ℓ is fixed), whereas $maxT_k(G)$ denotes the largest ℓ such that $G \in \mathrm{SS}(k, \ell)$ (i.e., k is fixed).

For example, consider again Fig. 1. If G is the graph in Fig. 1(a), then $minS_1(G) = 2$, $minS_2(G)=3$, $\max T_2(G)=1$, and $\max T_3(G)=2$. Thus, G is in $\mathrm{SS}(2, 1)$ and in $\mathrm{SS}(3, 2)$, but not in $\mathrm{SS}(2, 2)$. The 'opaque cube' OC (see Fig. 2) has $minS_1(OC)=3$ and $\max T_3(OC)=1$. Thus, OC belongs to $\mathrm{SS}(3, 1)$ but not to $\mathrm{SS}(3, 2)$.

As for k-self-spanners, we restrict our analysis to *connected* graphs. Note that the definition of (k, ℓ)-self-spanners does *not* imply that G remains connected when at most ℓ edges are removed. If this is necessary, then we can restrict our attention to graphs belonging to the intersection of the classes of $(\ell + 1)$-edge-connected graphs and (k, ℓ)-self-spanners.

Remark 4.2. By similar arguments as in Lemma 2.1, to check whether a graph $G=(V, E)$ belongs to $\mathrm{SS}(k, \ell)$ it is sufficient to check that for each subgraph $G' = (V, E')$ of G, with $|E'| \geqslant |E| - \ell$ and $E' \subseteq E$, the following holds:

$$d_{G'}(u, v) \leqslant k \quad \text{for every } e = \{u, v\} \in E \backslash E'. \tag{3}$$

The following lemma shows that, in order to check whether a graph belongs to a class $\mathrm{SS}(k, \ell)$, we do not have to consider all (possibly disconnected) subgraphs but only connected subgraphs.

Lemma 4.3. *For fixed integers $k \geqslant 1$ and $\ell \geqslant 0$, $G \in \mathrm{SS}(k, \ell)$ if and only if every* connected *and* spanning *subgraph $G' = (V, E')$ with $|E'| \geqslant |E| - \ell$ and $E' \subseteq E$ is a k-spanner of G.*

Proof. It suffices to show the 'if'-part: suppose every connected spanning subgraph $G' = (V, E')$ with $|E'| \geqslant |E| - \ell$ and $E' \subseteq E$ is a k-spanner of G, and, by contradiction, assume that G is not a (k, ℓ)-self-spanner. By definition, there is a subgraph $G'' = (V, E'')$ with $|E''| \geqslant |E| - \ell$ and $E'' \subseteq E$ (not necessarily connected) such that there is a pair of vertices u and v (within one connected component of G'') and $d_{G''}(u, v) > kd_G(u, v)$. This also implies $E'' \subset E$.

Since G is connected, there is also a connected subgraph $\widetilde{G} = (V, \widetilde{E})$ with $E'' \subset \widetilde{E} \subseteq E$ (and thus $|\widetilde{E}| \geqslant |E| - \ell$) constructed as follows: let $\mathscr{C}$ be the set of connected components of G''. Obtain $\widetilde{G}$ from G'' by adding $|\mathscr{C}| - 1$ bridge edges such that $\widetilde{G}$ is connected. Then $d_{\widetilde{G}}(u, v) > kd_G(u, v)$ and hence $\widetilde{G}$ is not a k-spanner of G, a contradiction. □

In the sequel, we use Lemma 4.3 as a characterization for the class of (k, ℓ)-self-spanners.

4.1. Characterization results

It is clear that for every connected graph G there are some parameters k and ℓ such that G belongs to $\mathrm{SS}(k, \ell)$. Analogously, if we fix one of the parameters we can always find a feasible value for the other parameter. Furthermore, it is easy to see that (k, ℓ)-self-spanners have inductive properties with respect to the parameters as stated below.

Lemma 4.4. *The following properties trivially hold*:

1. *If $1 \leqslant k \leqslant k'$, then $\mathrm{SS}(k, \ell) \subseteq \mathrm{SS}(k', \ell)$ for each $\ell > 0$;*
2. *if $0 < \ell \leqslant \ell'$, then $\mathrm{SS}(k, \ell) \supseteq \mathrm{SS}(k, \ell')$ for each $k \geqslant 1$;*
3. *if $k \geqslant 1$, then $\mathrm{SS}(\mathrm{k}) \subseteq \mathrm{SS}(\mathrm{k}, \ell)$ for each $\ell \geqslant 0$.*

The class of (k, ℓ)-self-spanners is *not* closed under subgraphs. For example, the 'opaque cube' is in $\mathrm{SS}(3, 1)$, but the graph G' obtained from removing the internal vertex is not (in fact, it has a stretch factor $minS_1(G') = 5$, and thus is in $\mathrm{SS}(5, 1)$). Also (k, ℓ)-self-spanners is not closed under supergraphs in the following sense: if a graph G is in $\mathrm{SS}(k, \ell)$ for some fixed parameters k and ℓ then there may be a supergraph of G on the same vertex set (i.e., a graph with additional edges) that does *not* belong to $\mathrm{SS}(k, \ell)$. The same remains true if we consider only $(\ell + 1)$-edge-connected graphs. As a consequence, the self-spanner properties of a graph cannot be inferred directly from the self-spanner properties of sub- or supergraphs.

As examples of standard graphs that exhibit some particular self-spanner properties, it is easy to see that $P_n \in \mathrm{SS}(1, \ell)$ for every $\ell \geqslant 1$. Furthermore $C_n \in \mathrm{SS}(n-1, \ell)$ but $C_n \notin \mathrm{SS}(n-2, \ell)$ for every $\ell \geqslant 1$, since $minS_\ell(C_n) = n-1$ for every $\ell \geqslant 1$ (i.e., the fault of one edge results in a path of length $n-1$). Starting from these observations, we are interested in finding non-trivial parameters such that a graph is a (k, ℓ)-self-spanner. This includes the problem of deciding for given parameters k and ℓ whether a given graph belongs to $\mathrm{SS}(k, \ell)$ as well as the more general recognition problems where we fix one of the parameters and try to optimize the other. To analyze the complexity of these problems, let us first consider the special case where we allow for single edge faults only, i.e., $\ell = 1$. The following lemma can be easily derived.

Lemma 4.5. *$G \in \mathrm{SS}(k, 1)$ if and only if every induced cycle of G has at most $k+1$ edges.*

Unfortunately, we cannot extend this characterization in a straightforward way to the case $\ell > 1$. But, if we restrict ourselves to $(\ell+1)$-edge-connected graphs we get the following lemma:

Lemma 4.6. *Let $G = (V, E)$ be $(\ell+1)$-edge-connected. Then $G \in \mathrm{SS}(k, \ell)$ if and only if for every edge $e = \{u, v\}$ of G there are at least ℓ edge disjoint paths (not involving e) of length at most k connecting u and v.*

Proof. For the ‘if’-part, let $G' = (V, E')$ be a subgraph with $E' \subseteq E$ and $|E'| \geqslant |E| - \ell$, and let $e = \{u, v\}$ be an edge that does not belong to E'. Assume that there are ℓ edge disjoint paths (not involving e) of length at most k connecting u and v. Thus, even if the remaining $\ell - 1$ edge faults happen to appear in one of these paths each, at least one covering path for e in G' remains.

We show the opposite direction by contradiction: assume $G \in \mathrm{SS}(k, \ell)$, and there is an edge $e = \{u, v\}$ such that there are at most $j < \ell$ edge disjoint paths (not involving e) $p_1, p_2, \ldots, p_j$ of length at most k connecting u and v. It is possible to construct a subgraph G' as follows: delete from G the edge e along with one edge in p_i, for each $1 \leqslant i \leqslant j$. G' remains connected (since G is $(\ell+1)$-edge-connected) but $d_{G'}(u, v) > k$, a contradiction to $G \in \mathrm{SS}(k, \ell)$. $\square$

Observe that we cannot relax on the edge-connectivity constraint in this lemma. Consider for example the diamond consisting of a C_4 and one chord: this graph is 2-edge-connected and belongs to $\mathrm{SS}(3, 2)$, but it does not fulfill the constraints of Lemma 4.6.

Lemma 4.7. *The following properties hold:*

1. $\mathrm{SS}(1) \equiv \mathrm{SS}(1, \ell)$ *for each* $\ell > 0$;
2. $\mathrm{SS}(2) \equiv \mathrm{SS}(2, \ell)$ *for each* $\ell > 0$;
3. *if* $k \geqslant 3$, *then* $\mathrm{SS}(k, \ell) \supsetneq \mathrm{SS}(k, \ell+1)$ *for each* $\ell > 0$.

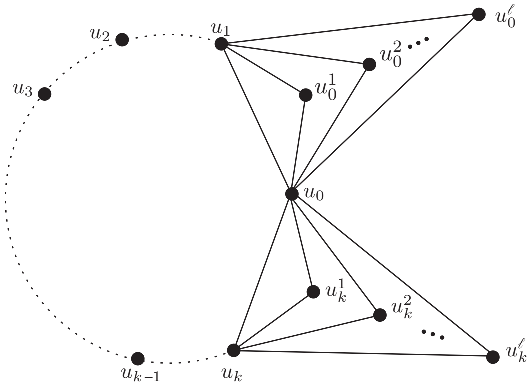


Fig. 3. The graph $G_{k,\ell}$ used in the proof of Lemma 4.7. $G_{k,\ell}$ is composed by an induced cycle of $k+1$ vertices; moreover, for each edge e of the cycle, ℓ disjoint paths of length 2 connect the endpoints of e.

Proof.

1. It directly follows from Definition 4.1. Moreover, as noted in Lemma 3.3, $\mathrm{SS}(1)$ coincides with the class of trees.
2. According to Item 2 of Lemma 4.4, it is sufficient to show that $\mathrm{SS}(2) \equiv \mathrm{SS}(2, 1)$. By Lemma 3.3, a graph G belongs to $\mathrm{SS}(2)$ if and only if every block of G is a K_3 or K_2. By Lemma 4.5, G belongs to $\mathrm{SS}(2, 1)$ if and only if every induced cycle of G has at most 3 edges. Since these two characterizations are equivalent, the statement follows.
3. We show that, for $k \geqslant 3$ and $\ell > 0$, there exists a graph $G_{k,\ell}$ such that $\in \mathrm{SS}(k, \ell)$ and $G_{k,\ell} \notin \mathrm{SS}(k, \ell+1)$. $G_{k,\ell}$ is composed by an induced cycle of $k+1$ vertices $u_0, u_1, \ldots, u_k$; moreover, for each vertex u_i of the cycle, $G_{k,\ell}$ contains the ℓ vertices $u_i^1, u_i^2, \ldots, u_i^\ell$, each connected to both u_i and $u_{(i+1) \bmod (k+1)}$ (see Fig. 3).

To prove that $G_{k,\ell} \notin \mathrm{SS}(k, \ell+1)$, it is sufficient to consider the subgraph obtained by removing the ℓ edges $\{u_0, u_0^i\}$, $1 \leqslant i \leqslant \ell$, along with $\{u_0, u_1\}$. In this subgraph the distance between u_0^ℓ and u_0 is given by the path $(u_0^\ell, u_1, u_2, \ldots, u_k, u_0)$. Since the length of this path is $k+1$, then $G_{k,\ell} \notin \mathrm{SS}(k, \ell+1)$.

To prove that $G_{k,\ell} \in \mathrm{SS}(k, \ell)$, we now show that $G_{k,\ell} \in \mathrm{SS}(3, \ell)$. By symmetrical properties of graph $G_{k,\ell}$, it is sufficient to test Property 3 of Remark 4.2 for edges $\{u_0, u_0^\ell\}$ (case (a) below) and $\{u_0, u_1\}$ (case (b) below) only.

(a) Let us consider G' obtained from $G_{k,\ell}$ by removing $\{u_0, u_0^\ell\}$ and at most other $\ell-1$ edges. The edge $\{u_0^\ell, u_1\}$ belongs to G', otherwise u_0 and u_0^ℓ are not connected in G'. If $\{u_0, u_1\}$ is in G', then $d_{G'}(u_0, u_0^\ell) = 2 < k$. If $\{u_0, u_1\}$ is not in G', then the removal of $\{u_0, u_0^\ell\}$, $\{u_0, u_1\}$, and at most other $\ell-2$ edges from $G_{k,\ell}$ cannot destroy all the remaining $\ell-1$ paths of length 2 from u_0 to u_1 passing through u_0^i, $1 \leqslant i \leqslant \ell-1$. As a consequence, assume that the edges $\{u_0, u_0^j\}$ and $\{u_0^j, u_1\}$ for some j, $1 \leqslant j \leqslant \ell-1$, are in G': then the covering path $(u_0^\ell, u_1, u_0^j, u_0)$ implies $d_{G'}(u_0, u_0^\ell) = 3 \leqslant k$.

(b) Let us consider that G' is obtained from $G_{k,\ell}$ by removing $\{u_0, u_1\}$ and at most other $\ell-1$ edges. This removal cannot destroy all the ℓ paths of length 2 from u_0 to u_1 passing through u_0^i, $1 \leqslant i \leqslant \ell$. As a consequence, $d_{G'}(u_0, u_1) = 2 \leqslant k$. $\square$

4.2. Complexity results

In this section, we consider the problem of recognizing graphs that belong to a given class and investigate characterization problems by finding the optimal stretch factor or fault-tolerance value of a given graph. As our main results, we establish an almost complete set of complexity results for these problems, that are formally stated as follows.

Problem 2. MINIMUM ℓ-STRETCH-FACTOR: *Given a graph G and an integer* $k \geqslant 1$, *does G belong to* $\mathrm{SS}(k, \ell)$, *i.e.*, $min S_\ell(G) \leqslant k$?

Problem 3. MAXIMUM k-FAULT-TOLERANCE: *Given a graph G and an integer* $\ell \geqslant 0$, *does G belong to* $\mathrm{SS}(k, \ell)$, *i.e.*, $max\ T_k(G) \geqslant \ell$?

Problem 4. GENERAL SELF-SPANNER: *Given a graph G and two integers* $k \geqslant 1$, $\ell \geqslant 0$, *does G belong to* $\mathrm{SS}(k, \ell)$?

Thus, in MINIMUM ℓ-STRETCH-FACTOR we consider ℓ as a fixed parameter, whereas in MAXIMUM k-FAULT-TOLERANCE k is a fixed parameter.

Now, if we fix the fault-tolerance value ℓ, we can determine the smallest stretch factor of a given graph $G = (V, E)$ in polynomial time. This trivially results by observing that the cardinality of the set $\{G' = (V, E') \mid |E'| \geqslant |E| - \ell\}$ is bounded by $|V|^{2(\ell+1)}$. Hence:

Theorem 4.8. MINIMUM ℓ-STRETCH-FACTOR *is in* $\mathscr{P}$ *for all* $\ell \geqslant 0$.

As a consequence, the problem of deciding whether a graph is a (k, ℓ)-self-spanner for fixed $k \geqslant 1$ and $\ell \geqslant 0$ is in $\mathscr{P}$. If we consider the dual problem where we fix the stretch factor and we want to find the largest fault-tolerance value of a given graph, the situation is different. To this aim, we introduce the following problem.

Problem 5. *Given an integer* $\ell \geqslant 0$, *a* $(\ell + 1)$*-edge-connected graph* $G = (V, E)$, *and an edge* $e = \{s, t\} \in E$, *does G contains* ℓ *or more mutually edge disjoint paths* (*not involving edge e*) *from s to t*, *which all have length at most* 5?

Theorem 4.9. *Problem* 5 *is* $\mathscr{NP}$*-complete*.

Proof. Consider the following problem:

- Given a connected graph $G = (V, E)$, two vertices $s, t \in V$, and integers $0 < K, L \leqslant |V|$, we have to decide whether G contains L or more mutually edge disjoint paths from s to t, which all have length at most K.

Such a problem is known as MAXIMUM LENGTH-BOUNDED DISJOINT PATHS (cf. [14] (ND41)). As shown in [20], this problem is $\mathscr{NP}$-complete for all fixed $K \geqslant 5$, it is polynomially solvable for $K \leqslant 3$, and it is open for $K = 4$. We show that MAXIMUM 5-BOUNDED DISJOINT PATHS (that is, the same problem when $K = 5$) is polynomially reducible to Problem 5.

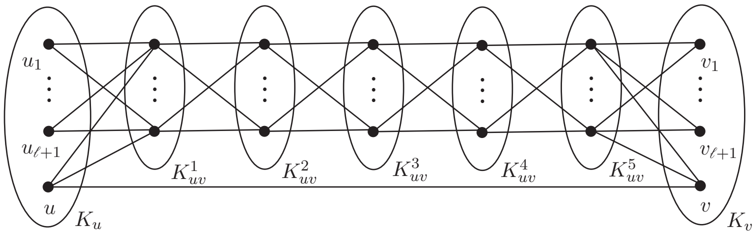


Fig. 4. The subgraph G_{uv} used to built the graph G' in the proof of Theorem 4.9. Each oval represents a clique and all the cliques have the same size.

Let $G=(V,E)$, $s,t \in V$, and $0<L \leqslant |V|$ be an instance of Maximum 5-Bounded Disjoint Paths. We construct a $(\ell+1)$-edge-connected graph $G'=(V',E')$ with an edge $e'=\{s',t'\} \in E'$ such that G contains the requested paths from s to t if and only if G' contains the requested paths from s' to t'.

First of all, let $\ell = \begin{cases} L-1 & \text{if } \{s,t\} \in E \\ L & \text{if } \{s,t\} \notin E \end{cases}$.

If $\{s,t\} \in E$, then G' is formed by $m=|E|$ subgraphs, one subgraph G_{uv} for each edge $\{u,v\} \in E$. If $\{s,t\} \notin E$, then G' is formed by $m+1$ subgraphs, one subgraph G_{uv} for each edge $\{u,v\} \in E$ along with the subgraph G_{st}. G_{uv} is composed by 7 cliques (see Fig. 4), each containing $\ell+2$ vertices. These 7 cliques are denoted by K_u and K_v (the *basic* cliques), and by $K_{uv}^1, K_{uv}^2, \ldots, K_{uv}^5$. A basic clique K_w contains vertices $w, w_1, \ldots, w_{\ell+1}$. The only edges in G_{uv} are the edges in each clique along with the following ones:

1. $\{u,v\}$;
2. $\{x,y\}$, for each $x \in K_{uv}^i$ and for each $y \in K_{uv}^{i+1}$, $1 \leqslant i < 5$;
3. $\{x,y\}$, for each $x \in K_u$ and for each $y \in K_{uv}^1$;
4. $\{x,y\}$, for each $x \in K_{uv}^5$ and for each $y \in K_v$.

Edges at Item 1 are called *basic* edges, while edges at Items 2, 3, and 4 are called *additional* edges. Two (basic or additional) cliques are *adjacent* if there exists an additional edge $\{w_1,w_2\}$ such that w_1 belongs to the first clique and w_2 to the second one. Consider $s' \equiv s$ and $t' \equiv t$, and notice that, by construction, $\{s',t'\} \in E'$. The union of vertices and edges of G_{uv}, for each edge $\{u,v\} \in E$ (along with vertices and edges of G_{st} if $\{s,t\} \notin E$), forms the requested graph G'. G' enjoys the following property:

P: If a path in G' between vertices u and v, with $u,v \in V$, contains an additional edge, then such path has length at least 6.

We first show that G' is $(\ell+1)$-edge-connected. By contradiction, assume that there is a subset $X \subseteq E'$ containing at most ℓ edges such that $G''=(V,E' \backslash X)$ is not connected; moreover, assume that G_1'' and G_2'' are two connected components of G''. Let H be a basic or additional clique in G': if both G_1'' and G_2'' contain vertices of H, then the removal of edges in X cannot disconnect G_1'' from G_2'' (since there are at least $\ell+1>|X|$ edges between G_1'' and G_2''). Then, assume that each clique is entirely contained either in

G_1'' or G_2''. Since G' is connected, G_1'' contains a clique which is adjacent to a clique of G_2''; again, this implies that there are at least $\ell + 1$ edges between G_1'' and G_2'', a contradiction.

Now assume that G contains L or more mutually edge disjoint paths from s to t, each one having length at most 5. If $\{s, t\} \in E$ ($\{s, t\} \notin E$, resp.) then G contains $L - 1 = \ell$ ($L = \ell$, resp.) or more of such paths. Since all these paths are also in G', then G' contains the requested paths.

Conversely, assume that G' contains ℓ or more mutually edge disjoint paths from s' to t' (not involving e'), which all have length at most 5. According to Property P, all such paths are formed by basic edges. Hence, there are L or more mutually edge disjoint paths from s to t in G, which all have length at most 5. □

Corollary 4.10.

1. Maximum k-Fault-Tolerance *is $\mathcal{NP}$-complete for all fixed* $k \geqslant 5$;
2. Maximum k-Fault-Tolerance, $k = 1, 2$, *is in* $\mathcal{P}$;
3. Maximum 3-Fault-Tolerance *is in $\mathcal{P}$ for the class of* $(\ell + 1)$*-edge-connected*, $\ell > 0$, *graphs*;
4. General Self-Spanner *is $\mathcal{NP}$-complete*.

Proof.

1. We first prove that the statement holds for $k = 5$.

   According to the characterization provided by Lemma 4.6, Maximum 5-Fault-Tolerance for the class of $(\ell + 1)$-edge-connected graphs, $\ell \geqslant 0$, can be reformulated as follows:

   - Given a graph $G = (V, E)$ and an integer $0 \leqslant \ell \leqslant |V|$ such that G is $(\ell + 1)$-edge-connected, we have to decide whether for every edge $e = \{u, v\}$ of G there are at least ℓ edge disjoint paths (not involving e) of length at most 5 connecting u and v.

   To solve Maximum 5-Fault-Tolerance for the class of $(\ell+1)$-edge-connected graphs we have to solve Problem 5 for each pair of adjacent vertices of the input graph. Then, Maximum 5-Fault-Tolerance is $\mathcal{NP}$-complete for the class of $(\ell + 1)$-edge-connected graphs. To show that the same result holds for each fixed $k > 5$, it is sufficient to observe that the proof of Theorem 4.9 can be extended to each fixed $k > 5$ by suitably setting the number of additional cliques, that is, from 5 to k.

   As a consequence, Maximum k-Fault-Tolerance is $\mathcal{NP}$-complete, for all fixed $k \geqslant 5$, also for the general graphs.
2. According to Items 1 and 2 of Lemma 4.7, solving Maximum k-Fault-Tolerance for $k = 1$ ($k = 2$, resp.) corresponds to test the membership of G to the class SS(1) (SS(2), resp.). By Theorem 3.4, these membership problems can be solved efficiently.
3. By the formulation of the Maximum k-Fault-Tolerance for the class of $(\ell + 1)$-edge-connected graphs given in the proof of Item 1, it is immediate to note that Maximum 3-Fault-Tolerance can be solved by running an algorithm that solves Maximum Length-Bounded Disjoint Paths when $K = 3$ for each pair of adjacent vertices.

Since MAXIMUM LENGTH-BOUNDED DISJOINT PATHS is in $\mathscr{P}$ when $K = 3$, then this approach leads to the required efficient solution for MAXIMUM 3-FAULT-TOLERANCE.

4. This is a consequence of Item 1. □

The problem MAXIMUM k-FAULT-TOLERANCE, $2 < k \leqslant 4$, remains to be settled for general graphs, while MAXIMUM 4-FAULT-TOLERANCE is open even for the class of $(\ell + 1)$-edge-connected graphs. Observe that it does not suffice to look for a maximum number of edge disjoint paths from s to t under *no length constraint*. This problem is solvable in polynomial time [14]. But in our case, the distance guarantee for every path is crucial.

5. Self-spanner properties of special graph classes

We now consider some sufficient conditions that guarantee that a given graph is a (k, ℓ)-self-spanner for some k and ℓ. The main idea here is the following: if a graph contains a long cycle that has only few chords, then this graph is likely to have bad self-spanner properties. In other words, if we can guarantee that a graph does not contain such a long cycle with only few chords, then the self-spanner properties are good. This fact is expressed in the following lemma. In the sequel, we denote by $\mathbb{C}_n$ a cycle on n vertices that may contain an arbitrary number of chords (in contrast to C_n denoting an *induced* cycle).

Lemma 5.1. *Given a graph $G = (V, E)$ and two fixed positive integers k and ℓ, let $\mathbb{C}_t$ be a cycle of G with at most $\ell - 1$ chords having maximum length. If $t \leqslant k + 1$, then G belongs to* $\mathrm{SS}(k, \ell)$.

Proof. By contradiction, suppose that $t \leqslant k + 1$ and $G \notin \mathrm{SS}(k, \ell)$. By Lemma 4.3, there exists a subgraph $G' = (V, E')$ of G with $|E'| \geqslant |E| - \ell$ such that G' is not a k-spanner of G. By Lemma 2.1, this implies that there exists an edge $e = \{u, v\} \in E \backslash E'$ such that $d_{G'}(u, v) > k$. The path P giving the distance $d_{G'}(u, v)$ together with edge e forms a cycle $\mathbb{C}_{t'}$ of G. Since P is obtained from G by removing e and at most $\ell - 1$ other edges of E, then $t' > k + 1$ and $\mathbb{C}_{t'}$ contains at most $\ell - 1$ chords. This is a contradiction, since $\mathbb{C}_t$ is a maximum cycle of G with at most $\ell - 1$ chords. □

We call a condition as given in the previous lemma a *cycle-chord condition*. Observe that this lemma does not provide a strict characterization for the class $\mathrm{SS}(k, \ell)$: there are (k, ℓ)-self-spanners that do not fulfill the cycle-chord condition. We can extract some further cycle-chord condition from Lemma 5.1 resulting in an upper bound on the trade-off between stretch factor and fault-tolerance.

Corollary 5.2. *Let $G = (V, E)$ be a graph, $t \geqslant 3$ an integer, and $f : \mathbb{N} \to \mathbb{N}$ a monotone increasing function. If every cycle of G on t vertices has at least $f(t)$ chords, then G belongs to* $\mathrm{SS}(t, f(t + 2))$.

Proof. If every cycle of G on t vertices has at least $f(t)$ chords, then, by monotonicity of f, also every cycle on t or more vertices has at least $f(t)$ chords. Let $\mathbb{C}_{t'}$ be a cycle of G with at most $f(t) - 1$ chords and having maximum length. Then, the number $c(\mathbb{C}_{t'})$ of chords of $\mathbb{C}_{t'}$ fulfills the following inequality:

$$f(t') \leqslant c(\mathbb{C}_{t'}) \leqslant f(t) - 1.$$

By the monotonicity of f, it follows that $t' \leqslant t - 1$. Hence, by Lemma 5.1, G belongs to $\mathrm{SS}(t-2, f(t))$, and, by the generality of t, also to $\mathrm{SS}(t, f(t+2))$. □

The cycle-chord conditions also support the intuition that graphs in which every vertex has a large degree are likely to have good self-spanner properties.

In the remainder of this section, we use the previous corollary to investigate the self-spanner properties of widely studied graph classes, namely, *distance-hereditary graphs*, *cographs*, and *chordal graphs* [5]. A graph is *distance-hereditary* if every two vertices have the same distance in every connected induced subgraph containing both. A graph is a *cograph* that does not contain any induced path of length 3. A graph is chordal if every cycle of length at least 4 possesses a chord. Equivalently, a chordal graph does not contain an induced subgraph isomorphic to C_n for any $n \geqslant 4$.

Both distance-hereditary graphs and cographs can be characterized by means of *one-vertex extension* operations. These operations can be used to enlarge a graph of the respective graph class to another graph of the same class containing more vertices. Let G be a graph, u be any vertex of G, and v be a new vertex. The operations to extend G by adding v are the following:

- $\alpha(u, v)$: v is adjacent only to u (v is a *pendant vertex*);
- $\beta(u, v)$: v is adjacent to u and to every neighbor of u (v is a *true twin* of u);
- $\gamma(u, v)$: v is adjacent to every neighbor of u (v is a *false twin* of u).

Bandelt and Mulder showed in [4] that every distance-hereditary graph is obtained starting from a single vertex by applying a sequence of operations α, β, and γ. Corneil et al. showed in [12] that every cograph is obtained starting from a single vertex by applying a sequence of operations β and γ.

Lemma 5.3. *In a distance-hereditary graph, every cycle $\mathbb{C}_t$, $t \geqslant 3$, has at least $t-4$ chords if t is even, and at least $t-3$ chords if t is odd. In a cograph, every cycle $\mathbb{C}_t$, $t \geqslant 3$, has at least $t(t-4)/4$ chords if t is even, and at least $(t-1)(t-3)/4$ chords if t is odd.*

Proof. We prove the property of distance-hereditary graphs by induction on the number of vertices in a cycle. The induced cycles C_4 and C_3 are distance-hereditary, and thus the base case of the induction is true. Let us consider a distance-hereditary graph G isomorphic to a cycle $\mathbb{C}_t$ with $t \geqslant 5$. Since Howorka [19] showed that H is distance-hereditary if and only if every cycle of H having at least 5 vertices has two crossing chords, then $\mathbb{C}_t$ has at least two crossing chords, say $\{u, v\}$ and $\{u', v'\}$. Chord $\{u, v\}$ divides $\mathbb{C}_t$ into two cycles $\mathbb{C}_{t_1}$ and $\mathbb{C}_{t_2}$ such that $t = t_1 + t_2 - 2$. Let us suppose t odd, and, w.l.o.g, t_1 odd and t_2 even. By induction hypothesis, $\mathbb{C}_{t_1}$ has at least $t_1 - 3$ chords and $\mathbb{C}_{t_2}$ has at least $t_2 - 4$ chords. Thus $\mathbb{C}_t$ has at least the chords belonging to $\mathbb{C}_{t_1}$ and to $\mathbb{C}_{t_2}$ plus the two crossing chords$\{u, v\}$ and $\{u', v'\}$, that is $t_1 - 3 + t_2 - 4 + 2 = t_1 + t_2 - 5 = t - 3$ chords. When t is even, t_1 and t_2 are either both even or both odd. By repeating the previous arguments, the total number of chords of $\mathbb{C}_t$ is $t - 4$ in the first case and $t - 2$ in the second one.

We now prove the property about cographs. Let us assume t even. First notice that every connected distance-hereditary graph having at least three vertices is generated by a sequence of extension operations that starts with a β-operation, i.e., G is an extension of K_2. Moreover, the following properties are straightforward:

- A γ-operation introduces one edge less than a β-operation; so, if G' is generated by a sequence of $t - 2$ γ-operations starting from K_2 and if G' is isomorphic to a cycle $\mathbb{C}_t$, then G' has the minimum number of chords.
- The extension of K_2 by a sequence of γ-operations gives a complete bipartite graph $K_{p,q}$.
- A complete bipartite graph $K_{p,q}$ is isomorphic to a cycle if and only if $p = q$ and $p, q \geqslant 2$.

The properties above imply that if $t \geqslant 4$ is even, then a cograph isomorphic to a cycle $\mathbb{C}_t$ has the minimum number of chords if and only if it is isomorphic to $K_{t/2,t/2}$. This cycle has $t(t - 4)/4$ chords.

Now let us assume t odd. The statement is trivially true for $t = 3$. According to the three properties stated in the even case, a cograph G that is isomorphic to a cycle $\mathbb{C}_t$ with t odd and $t > 3$, cannot be obtained from K_2 by using γ-operations only. This means that G has the minimum number of chords if it is obtained from K_2 by using the minimum number of β-operations, and all the β-operations used in the sequence are applied after all the γ-operations.

Now, let G be a cograph that is isomorphic to $\mathbb{C}_t$ with $t > 3$. G can be generated from K_2 by applying first $t - 3$ γ-operations, and then only one β-operation to an arbitrary vertex. Since G is isomorphic to a cycle $\mathbb{C}_t$, the first $t - 3$ γ-operations produce a cograph G that is isomorphic to $\mathbb{C}_{t-1}$ where $t - 1$ is even. By the result from the even case, $\mathbb{C}_{t-1}$ is isomorphic to $K_{(t-1)/2,(t-1)/2}$ and contains $(t - 1)(t - 5)/4$ chords. The last β-operation results in the creation of $(t - 1)/2$ new chords. Thus, G has $(t - 1)(t - 5)/4 + (t - 1)/2 = (t - 1)(t - 3)/4$ chords. □

From the basic characterization of chordal graphs, the following lemma can be derived.

Lemma 5.4. *Every cycle $\mathbb{C}_t$, $t \geqslant 4$, of a* chordal graph *G has at least $t - 3$ chords.*

By using Corollary 5.2 together with Lemmas 5.3 and 5.4, we get the following self-spanner properties for the three graph classes:

Theorem 5.5.

1. *Every distance-hereditary graph is in* $\mathrm{SS}(n, n - 2)$ *for every even* $n \geqslant 4$; *for odd* $n \geqslant 3$, *distance-hereditary graphs even belong to* $\mathrm{SS}(n, n - 1)$.
2. *Every cograph is in* $\mathrm{SS}(n, (n^2 - 4)/4)$ *for every even* $n \geqslant 4$; *for odd* $n \geqslant 3$, *cographs even belong to* $\mathrm{SS}(n, (n^2 - 1)/4)$.
3. *Every chordal graph is in* $\mathrm{SS}(n, n - 1)$ *for every* $n \geqslant 4$.

To summarize this subsection, distance-hereditary and chordal graphs exhibit strong self-spanner properties: the stretch factor does not grow faster than the number of edge faults. In particular, if the number of edge faults is bounded by a constant then also the stretch factor is bounded by more or less the same constant. For cographs, the result is even stronger: the stretch factor only grows in the order of the square root of the number of edge faults.

6. Self-spanner properties of common network topologies

In this section, we study how the new graph classes of (k, ℓ)-self-spanners fit into the context of some popular network topologies. Since the graphs used for modeling most of such topologies can be defined by composing simpler graphs, we first study self-spanner properties of graphs built by means of Cartesian product. The obtained results are then used to examine some mesh-like networks (namely *grid*, *torus*, and *hypercube*) with respect to their self-spanner properties. In a second phase, we also investigate some hypercube derived networks (*cube connected cycles* and *butterflies*).

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two nontrivial graphs; the Cartesian product $G := G_1 \times G_2$ is the graph with vertex set V and edge set E as follows:

- $V = \{(x_1, x_2) \mid x_1 \in V_1, x_2 \in V_2\}$,
- $E = \{\{(x_1, x_2), (y_1, y_2)\} \mid (x_1 = y_1 \text{ and } \{x_2, y_2\} \in E_2) \text{ or } (x_2 = y_2 \text{ and } \{x_1, y_1\} \in E_1)\}$.

Consequently, two vertices of $G_1 \times G_2$ are adjacent if and only if the first components are equal and the second components form an edge in G_2 or vice versa. Moreover, for any $x_1 \in V_1$, $G[\{(x_1, x_2) \mid x_2 \in V_2\}]$ is isomorphic to G_2, and for any $x_2 \in V_2$, $G[\{(x_1, x_2) \mid x_1 \in V_1\}]$ is isomorphic to G_1. W.l.o.g., we do not consider the case where G_1 or G_2 is a graph having no edge.

The next lemma shows that graphs that arise from the Cartesian product of two graphs have strong self-spanner properties. In particular, it indicates that a stretch factor of 3 plays an important role.

Lemma 6.1. *Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two connected graphs, $G = (V, E) = G_1 \times G_2$, and $i \in \{1, 2\}$.*

1. *If $G_i \in \mathrm{SS}(k_i, \ell_i)$ and $(\ell_i + 1)$-edge-connected then $G \in \mathrm{SS}(\max\{k_1, k_2\}, \min\{\ell_1, \ell_2\})$.*
2. *Let δ be the minimum vertex degree of vertices in $V_1 \cup V_2$. Then $G \in \mathrm{SS}(3, \delta)$.*
3. *$G \in \mathrm{SS}(2, \ell)$ if and only if each edge in G_i belongs to at least ℓ disjoint triangles in G_i.*
4. *If G_1 or G_2 contains a bridge then $max\, T_2(G) = 0$, i.e., there is no $\ell > 0$ such that $G \in \mathrm{SS}(2, \ell)$. In particular, if G_1 or G_2 contains a bridge and $G \in \mathrm{SS}(k, \ell)$ for some $\ell > 0$, then $k \geqslant 3$.*

Proof.

1. Consider the edge $e = \{(x_1, x_2), (y_1, y_2)\}$ in G. By Remark 4.2, it suffices to show that the distance between (x_1, x_2) and (y_1, y_2) is at most $\max\{k_1, k_2\}$ after the removal of

e and $\min\{\ell_1, \ell_2\} - 1$ other arbitrary edges from G. By definition of Cartesian product, e belongs to an induced subgraph G'' of G that is isomorphic either to G_1 or to G_2. By assumption, $G_i \in \mathrm{SS}(k_i, \ell_i)$ and G_i is $(\ell_i + 1)$-edge-connected. Hence, even if all the removed edges from G belong to G'', the distance between (x_1, x_2) and (y_1, y_2) is at most $\max\{k_1, k_2\}$ (because such a distance can be thought as computed in G'' after the removal of edges from G).

2. W.l.o.g., assume that $x_1 \in G_1$ is the vertex with minimum degree. Then there are δ vertices x_1^j adjacent to x_1 in $V_1, 1 \leqslant j \leqslant \delta$. Assuming that $\{x_2, y_2\}$ is and edge in G_2, then $e = \{(x_1, x_2), (x_1, y_2)\}$is an edge in G. By definition of Cartesian product there are δ edge disjoint paths $((x_1, x_2), (x_1^j, x_2), (x_1^j, y_2), (x_1, y_2))$ of length 3 connecting (x_1, x_2) to (x_1, y_2) in G. The removal of δ edges from G including e, cannot destroy all these paths and the statement follows. By the generality of e and according to Remark 4.2, this proves that $G \in \mathrm{SS}(3, \delta)$.
3. We have to show the 'only if'-part: consider edge $e = \{(x_1, x_2), (y_1, y_2)\}$ in G and, w.l.o.g., assume that $x_1 = y_1$ and $\{x_2, y_2\} \in E_2$. Since $G \in \mathrm{SS}(2, \ell)$, there are ℓ edge disjoint paths from (x_1, x_2) to (y_1, y_2) of length at most 2 in G not using e. According to the proof of Part 2, any path from (x_1, x_2) to $(y_1, y_2) \equiv (x_1, y_2)$ via a vertex (v, w) with $v \neq x_1$ has length at least 3. Thus, there are vertices $z_j \in V_2$ such that $\{(x_1, x_2), (x_1, z_j)\}, \{(x_1, z_j), (x_1, y_2)\} \in E$, and $\{x_2, z_j\}, \{z_j, y_2\} \in E_2$for $1 \leqslant j \leqslant \ell$. Hence, e belongs to ℓ disjoint triangles in G_2. The same arguments hold for G_1.
4. Part 4 is a special case of Part 3. □

Observe that, for Part 1 of the previous lemma, it is really necessary to claim the respective edge connectivity. Otherwise, we cannot guarantee that the graph considered in the proof remains connected. Also, for Part 3 of that lemma, it does not suffice to claim that $G_1 \in \mathrm{SS}(2, \ell)$ (and $G_2 \in \mathrm{SS}(2, \ell)$, respectively): we again need that both graphs are $(\ell+1)$-edge-connected. For smaller stretch factors, i.e., $k = 1$, we already know that $G_1 \times G_2$ has a stretch factor smaller than 2 if and only if it is a tree.

Remark 6.2. Part 2 of Lemma 6.1 is tight in the following sense: if $G_i \notin \mathrm{SS}(2, 1)$ and G_i has minimum degree δ for $i \in \{1, 2\}$, then $min S_\delta(G_1 \times G_2) = 3$ and $\max T_3(G_1 \times G_2) = \delta$. Thus $G_1 \times G_2 \in \mathrm{SS}(3, \delta)$, but $G_1 \times G_2 \notin \mathrm{SS}(2, \delta)$ and $G_1 \times G_2 \notin \mathrm{SS}(3, \delta + 1)$.

6.1. Mesh-like networks

In this section, we study self-spanner properties of mesh-like networks. In particular, we consider *grids*, *tori*, and *hypercubes*:

- the grid $G_{n,m}$ is the Cartesian product $P_n \times P_m$ for $n, m \geqslant 2$;
- the torus $T_{n,m}$ is the Cartesian product $C_n \times C_m$ for $n, m \geqslant 3$;
- the hypercube H_d is recursively defined from P_2 by $H_d = P_2 \times H_{d-1} = \underbrace{P_2 \times \cdots \times P_2}_{d \text{ times}}$.

The following lemma indicates the self-spanner properties of these topologies.

Theorem 6.3.

1. *$G_{n,m}$ belongs to* $\mathrm{SS}(3, 1)$*, but not to* $\mathrm{SS}(2, 1)$.
   If $n > 2$ or $m > 2$ then $G_{n,m}$ does not belong to $\mathrm{SS}(3, 2)$.
   If $n, m > 2$ then $G_{n,m}$ belongs to $\mathrm{SS}(5, 2)$*, but not to* $\mathrm{SS}(4, 2)$ *or* $\mathrm{SS}(5, 3)$.
2. *$T_{n,m}$ belongs to* $\mathrm{SS}(3, 2)$*, but not to* $\mathrm{SS}(2, 2)$.
   If $n > 3$ or $m > 3$ then $T_{n,m}$ does not belong to $\mathrm{SS}(3, 3)$.
   $T_{n,m}$ belongs to $\mathrm{SS}(\min\{5, \max\{n, m\} - 1\}, 3)$.
   If $n, m \geqslant 5$ then $T_{n,m}$ belongs to $\mathrm{SS}(5, 4)$*, but not to* $\mathrm{SS}(4, 4)$.
   If $n, m > 5$ then $T_{n,m}$ does not belong to $\mathrm{SS}(5, 5)$.
3. *H_d belongs to* $\mathrm{SS}(3, d - 1)$*, but not to* $\mathrm{SS}(3, d)$ *or to* $\mathrm{SS}(2, 1)$.

Proof.

1. $G_{n,m} \in \mathrm{SS}(3, 1)$ and $G_{n,m} \notin \mathrm{SS}(2, 1)$ are immediate consequences of Parts 2 and 4 of Lemma 6.1. To see the other self-spanner properties, observe that, for any edge on the boundary of the grid, there is only one path of length 3 connecting the end-vertices of that edge, all other paths have length 5 or longer. This 3-path (and the edge itself) may be broken by a double edge fault such that the end-vertices still remain connected (if n, m are large enough). Accordingly, $G_{n,m} \in \mathrm{SS}(5, 2)$. If $G_{n,m} \neq C_4$ then $G_{n,m} \notin \mathrm{SS}(4, 2)$ and if $n, m > 2$, $G_{n,m} \notin \mathrm{SS}(5, 3)$.
2. Parts 2 and 3 of Lemma 6.1 directly imply that $T_{n,m} \in \mathrm{SS}(3, 2)$ and $T_{n,m} \notin \mathrm{SS}(2, 2)$. From Remark 6.2 it follows that $T_{n,m} \notin \mathrm{SS}(3, 3)$, if $m > 3$ or $n > 3$. Observe that $T_{3,3} \in \mathrm{SS}(3, 3)$.
   For every edge $\{x, y\}$ in $T_{n,m}$ there are two edge disjoint paths of length 3 connecting x and y and one (also disjoint) path of length at most $\max\{n, m\} - 1$. If n and m are at least 5, then there are six different paths of length 5 connecting x and y, but only two of length at most 4. It is easy to see that at least one of these paths of length 5 remains complete if $\{x, y\}$ and three further edges are removed. If n and m are at least 6, consider the case of fault of five direct parallel edges in $T_{n,m}$: $T_{n,m}$ remains connected and the middle failing edge has a stretch factor that is greater than 5. Consequently, $T_{n,m} \in \mathrm{SS}(\min\{5, \max\{n, m\} - 1\}, 3)$. For m, n large enough, $T_{n,m} \in \mathrm{SS}(5, 4)$, but $T_{n,m} \notin \mathrm{SS}(4, 4)$ and also $T_{n,m} \notin \mathrm{SS}(5, 5)$.
3. To show that H_d belongs to $\mathrm{SS}(3, d - 1)$, but not to $\mathrm{SS}(3, d)$, it is sufficient to observe that every edge e of H_d belongs to $d-1$ induced cycles of length 4 that are edge disjoint apart from e. By Part 4 of Lemma 6.1, H_d does not belong to $\mathrm{SS}(2, 1)$. □

Observe that the fault-tolerance value of the torus is higher than that of the grid, due to the additional wrap-around connections, which make the topology symmetric. But note that the addition of edges does not result in higher fault-tolerance values in general.

Furthermore, note that the hypercube H_d still guarantees a constant stretch factor 3, even if $d - 1$ edges fail, i.e., if the number of edge faults is in the order of the dimension of H_d. Consequently, this topology expresses especially strong self-spanner properties.

6.2. Hypercube derived networks

In this section, we study self-spanner properties of two different types of bounded-degree approximations of the hypercube; in particular, we consider *cube-connected cycles graph* and *butterfly* (e.g., see [22] and the references therein). Here we use the following alternative definition of hypercube [18]: the *d-dimensional binary hypercube* H_d, $d \geqslant 1$, has 2^d vertices, which are labeled with the binary strings of length d. Two vertices in H_d are adjacent if their labels differ in exactly one bit.

The *cube-connected cycles graph* of dimension d, denoted CCC_d, is derived from H_d by replacing each vertex of H_d by a *fundamental cycle* of length d. Each vertex of such a cycle is labeled by a tuple (i, x) for $0 \leqslant i \leqslant d-1$, and i is called the *level* of the vertex. Apart from the *cycle edges* of the fundamental cycles, every vertex (i, x) is connected to vertex $(i, x(i))$, where $x(i)$ denotes the vertex of H_d that is labeled by the same string as vertex x but with bit i flipped. These edges are called *hypercube edges*.

The *butterfly graph* (with wrap-around) of dimension d, denoted B_d, is derived from H_d similarly as CCC_d: B_d consists of the same vertices (i, x) for $0 \leqslant i \leqslant d-1$ as CCC_d, and the same *fundamental cycles* of length d. But now every vertex (i, x) is connected by two *hypercube edges* to vertices $(i+1, x(i))$ and $(i-1, x(i-1))$.

CCC_d can be obtained from B_d by replacing every pair of hypercube edges $\{(i, x), (i+1, x(i))\}$ and $\{(i, x), (i-1, x(i-1))\}$ by one edge $\{(i, x), (i, x(i))\}$. Thus, CCC_d can be viewed as a spanning subgraph of B_d.

In [3], it is shown that different hypercube-derived topologies can be embedded within other such topologies with small slowdown. Results on the existence of cycles and the construction of k-spanners can be found in [25,18], respectively. But all these results do not imply on the self-spanner properties of the topologies studied here. We get the following results concerning the self-spanner properties of the topologies above:

Theorem 6.4. *B_d belongs to* $\mathrm{SS}(3, 1)$ *and to* $\mathrm{SS}(d+1, 2)$, *but not to* $\mathrm{SS}(2, 1)$, $\mathrm{SS}(d, 2)$, *or* $\mathrm{SS}(d+1, 3)$. *CCC_d belongs to* $\mathrm{SS}(7, 1)$ *and to* $\mathrm{SS}(\max\{7, d-1\}, 2)$, *but not to* $\mathrm{SS}(6, 1)$.

Proof. Any edge of B_d belongs to exactly one induced cycle of length 4 consisting of two cycle edges and two hypercube edges. Thus, $B_d \in \mathrm{SS}(3, 1)$. From [25], we know that B_d does not contain a cycle of length 3 if $d > 3$. For smaller d, no cycle of length 3 contains a hypercube edge. Hence, $B_d \notin \mathrm{SS}(2, 1)$. Now consider the case when two edges fail in B_d: if two edges of the same fundamental cycle fail, there still remains a path of length 3 connecting the end-vertices of the faulty edges each. If both cycle edges of a 4-cycle as mentioned above fail then there remains a path of length $d-1$ via a fundamental cycle, but no shorter one. If a cycle edge and a hypercube edge within such a 4-cycle fail then a shortest path of length $d+1$ remains but not two such paths.

CCC_d consists of the same fundamental cycles as B_d, but contains only half of the hypercube edges. This results in longer cycles: for every *hypercube edge*, there are two (shortest) edge disjoint paths of length 7 that connect the end-vertices. For every *cycle edge*, there is a path of length $d-1$ (via the fundamental cycle) and another (disjoint) path of length 7 using hypercube edges. Consequently, $CCC_d \in \mathrm{SS}(7, 1)$ and $CCC_d \in \mathrm{SS}(\max\{7, d-1\}, 2)$, but $CCC_d \notin \mathrm{SS}(6, 1)$. □

The previous theorem shows that bounded-degree approximations of the hypercube like CCC_d and B_d perform poorly with respect to their self-spanner properties: in the case of single edge faults the stretch factor is still a constant (though much larger than for the hypercube), but for double edge faults the stretch factor grows linearly with the dimension d. Thus, the guarantees for delays in case of faults are really weak for these kinds of topologies. The big differences between the self-spanner properties of H_d on the one side, and CCC_d and B_d on the other are due to the bounded degree.

7. Further remarks

In this work, we have introduced the classes of k-self spanners and (k, ℓ)-self-spanners. Such graphs model networks that guarantee constant stretch factors even in the case of multiple edges faults. We have considered both the cases of unlimited and limited number of edge faults. We have given characterizational, structural and computational results, and we have shown that some popular network topologies and special graph classes exhibit (more or less) strong self-spanner properties.

We consider this work as a first step towards a more general approach to the design of networks that guarantee constant stretch factors in case of edge faults, and naturally many problems remain open. On the one hand, it would be interesting to know how well MAXIMUM k-FAULT-TOLERANCE can be approximated for the cases where it is $\mathcal{NP}$-complete. Another further goal in this context is to design sparse (k, ℓ)-self-spanner networks for given parameters k and ℓ such that specific connectivity requirements are fulfilled. On the other hand, we are interested in further investigating the self-spanner properties of other known topologies.

Acknowledgements

Work partially supported by the Human Potential Programme of the European Union under contract No. HPRN-CT-1999-00104 (“AMORE”), by the Italian MURST Project “Teoria dei Grafi ed Applicazioni”, and by the *Deutsche Forschungsgemeinschaft* under grant Wa 654/10-2.

References

[1] M. Aïder, Almost distance-hereditary graphs, Discrete Math. 242 (2002) 1–16.
[2] F.S. Annexstein, Fault-tolerance in hypercube-derivative networks, Comput. Architecture News 19 (1) (1991) 25–34.
[3] F.S. Annexstein, M. Baumslag, A.L. Rosenberg, Group action graphs and parallel architectures, SIAM J. Comput. 19 (3) (1990) 544–569.
[4] H.J. Bandelt, H.M. Mulder, Distance-hereditary graphs, J. Combinat. Theory Series B 41 (1986) 182–208.

[5] A. Brandstädt, V.B. Le, J. Spinrad, Graph Classes: A Survey, SIAM Monographs on Discrete Mathematics and Applications, 1999.
[6] J. Bruck, R. Cypher, C.-T. Ho, Fault-tolerant meshes with small degree, SIAM J. Comput. 26 (6) (1997) 1764–1784.
[7] S. Cicerone, G. Di Stefano, Graphs with bounded induced distance, Discrete Appl. Math. 108 (1–2) (2001) 3–21.
[8] S. Cicerone, G. Di Stefano, $(k, +)$-distance-hereditary graphs, J. Discrete Algorithms 1 (3–4) (2003) 281–302.
[9] S. Cicerone, G. Di Stefano, Networks with small stretch number, J. Discrete Algorithms 2 (4) (2004) 383–405.
[10] S. Cicerone, G. Di Stefano, D. Handke, Survivable networks with bounded delay: the edge failure case (extended abstract), in: Proceedings of the 10th Annual International Symposium on Algorithms and Computation, ISAAC'99, Lecture Notes in Computer Science, vol. 1741, Springer, Berlin, 1999, pp. 205–214.
[11] R. Cole, B.M. Maggs, R.K. Sitaraman, Reconfiguring arrays with faults part I: worst-case faults, SIAM J. Comput. 26 (6) (1997) 1581–1611.
[12] D.G. Corneil, H. Lerchs, L. Stewart-Burlingham, Complement reducible graphs, Discrete Appl. Math. 3 (1981) 163–174.
[13] A.M. Farley, A. Proskurowski, Self-repairing networks, Parallel Process. Lett. 3 (4) (1993) 381–391.
[14] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of $\mathcal{NP}$-Completeness, W H Freeman & Co Ltd., New York, 1979.
[15] D. Handke, Independent tree spanners: fault-tolerant spanning trees with constant distance guarantees, Discrete Appl. Math. 108 (1–2) (2001) 105–127.
[16] F. Harary, Graph Theory, Addison–Wesley, Reading, MA, 1969.
[17] J.P. Hayes, A graph model for fault-tolerant computing systems, IEEE Trans. Comput. C-25 (9) (1976) 875–884.
[18] M.-C. Heydemann, J. G. Peters, D. Sotteau, Spanners of hypercube-derived networks, Technical Report CMPT TR 94-02, School of Computer Science, Simon Fraser University, Burnaby, Canada, 1994.
[19] E. Howorka, A characterization of distance-hereditary graphs, Quart. J. Math., Oxford Univ. 2 (28) (1977) 417–420.
[20] A. Itai, Y. Perl, Y. Shiloach, The complexity of finding maximum disjoint paths with length constraints, Networks 12 (1982) 277–286.
[21] F.T. Leighton, B.M. Maggs, R.K. Sitaraman, On the fault tolerance of some popular bounded-degree networks, SIAM J. Comput. 27 (6) (1998) 1303–1333.
[22] F.T. Leighton, Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes, Morgan Kaufman Publishers, Los Altos, CA, 1992.
[23] D. Peleg, A.A. Schaeffer, Graph spanners, J. Graph Theory 13 (1989) 99–116.
[24] D. Peleg, J.D. Ullman, An optimal synchronizer for the hypercube, in: Proceedings of the 6th ACM Symposium on Principles of Distributed Computing, Vancouver, 1987, pp. 77–85.
[25] A.L. Rosenberg, Cycles in networks, Technical Report UM-CS-1991-020, Department of Computer and Information Science, University of Massachusetts, 1991.
[26] T.-Y. Sung, M.-Y. Lin, T.-Y. Ho, Multiple-edge-fault tolerance with respect to hypercubes, IEEE Trans. Parallel Distributed Systems 8 (2) (1997) 187–192.