Network reliability in hamiltonian graphs

Pol Llagostera, Nacho López, Carles Comas.
Dep. de Matemàtica, Universitat de Lleida
Lleida, Spain
{pol.llagostera,nacho.lopez,carles.comas}@udl.cat

May 10, 2021

Abstract

The reliability polynomial of a graph gives the probability that a graph remains operational when all its edges could fail independently with a certain fixed probability. In general, the problem of finding uniformly most reliable graphs inside a family of graphs, that is, one graph whose reliability is at least as large as any other graph inside the family, is very difficult. In this paper, we study this problem in the family of graphs containing a hamiltonian cycle.

Keywords: Network reliability, network design, reliability polynomial, hamiltonian graph.

1 Introduction

Notation and terminology

In the reliability context, networks are modeled by graphs. We recall that a graph is an ordered pair $G = (V, E)$, where V is a non empty set of *vertices* or *vertices*, and E is a set of unordered pairs of different elements of V, called *links* or *edges*. The degree of a vertex $v \in V$, denoted by $d(v)$, is the number of incident edges at v. A *walk* of length $\ell \geq 0$ from a vertex u to a vertex v is a sequence of $\ell+1$ vertices, $u_0u_1 \dots u_{\ell-1}u_\ell$, such that $u = u_0$, $v = u_\ell$ and each pair $u_{i-1}u_i$, for $i = 1, \dots, \ell$, is an edge of G. A *connected* graph has always a walk between any pair of vertices. Otherwise, the graph is *not connected*. Our model is a stochastic network with perfect vertices but with edge failures: each edge remains operational independently with probability $0 \leq p \leq 1$ (every edge has the same probability of being operational). Moreover, no repair is allowed after an edge fails.

Through this paper, we consider the problem of computing the probability that a network remains connected. More precisely, we focus on what is called, the *all-terminal network reliability problem*: given the probability p of an edge being operational in a network G, what is the probability that there exists an operational path between every pair of vertices u and v of G (see [9]).

Hamiltonian graphs

A graph G is a *hamiltonian graph* if it contains a spanning cycle, that is, a cycle passing through all the vertices of the graph. This cycle is called a *hamiltonian cycle*. The problem of finding a hamiltonian cycle takes back to the 1850s when Sir William Rowan Hamilton presented the problem of finding a hamiltonian cycle in a dodecahedron. The complexity of finding a hamiltonian cycle in a graph is in general NP-complete and much research in this area is devoted to finding necessary and/or sufficient conditions for a graph to be hamiltonian. One of such conditions is Ore's Theorem, which is a well-known result in the field.

Theorem 1.1 (Ore)**.** *Let G be a connected graph of order n such that $d(u) + d(v) \geq n$ for any two pair of non adjacent vertices u and v. Then G is a hamiltonian graph.*

Recent developments and many results regarding hamiltonian graphs can be found in [14].

The reliability polynomial

Let $|G|$ denote the number of edges of a network $G = (V, E)$. Let us consider the set $\mathcal{G}$ of connected spanning subgraphs of G. Then, the probability that G is connected, as a function of p, is

$$\sum_{G' \in \mathcal{G}} p^{|G'|}(1-p)^{m-|G'|} \tag{1}$$

This formula is known as *the reliability polynomial of* G, and it is denoted as $\mathrm{Rel}(G, p)$. There are several methods for computing $\mathrm{Rel}(G, p)$, but in general this problem is NP-complete (see [18]). A *pathset* of a graph $G = (V, E)$ is a subset $N \subseteq E$ of edges that makes the graph (V, N) connected. Hence, an alternative definition for the reliability polynomial is,

$$\mathrm{Rel}(G, p) = \sum_{i=0}^{m} N_i p^i (1-p)^{m-i} \tag{2}$$

where N_i denotes the number of pathsets of cardinality i. From this point of view, some of the coefficients of the reliability polynomial are 'easy' to compute. For instance, $N_i = 0$ for all $i < m-n+1$. Also $N_i = \binom{m}{m-i}$ for all $i > m - \lambda$, where λ denotes the edge-connectivity of G. The edge-connectivity can be found with a network flow algorithm in polynomial time (see [10]). Moreover, any spanning tree of G contain $m - n + 1$ edges, so $N_{m-n+1} = \tau$ where τ is the number of spanning trees of G (also known as the tree-number). Again, τ is computed in polynomial time (see [12]) using the Kirchoff's matrix tree theorem. Further, from [9], section 5.2, $N_{m-\lambda}$ can also be computed in polynomial time.

Uniformly most reliable graphs

A main problem in the reliability context is concerned with the design of networks with 'high' reliability. To this end, let $\mathcal{G}(n, m)$ be the set of all simple connected graphs with n vertices and m edges. Given two graphs $G, G' \in \mathcal{G}(n, m)$, we say that G is *uniformly more reliable* than G' if $\mathrm{Rel}(G, p) \geq \mathrm{Rel}(G', p)$ for all $p \in [0, 1]$. This means that, for any value of an edge to being operational p, graph G has equal or higher probability to remain connected than graph G'. If there exists a graph G such that $\mathrm{Rel}(G, p) \geq \mathrm{Rel}(H, p)$ for all $p \in [0, 1]$ and for all $H \in \mathcal{G}(n, m)$, then G is known as a *uniformly most reliable* graph in the set $\mathcal{G}(n, m)$. We point out that uniformly most reliable graphs have been also known as *uniformly optimally reliable* graphs (see [1, 2, 13, 24]) and *uniformly optimal* digraphs for the directed case (see [6]).

However, the reliability polynomial does not define a total ordering in $\mathcal{G}(n, m)$, because there are graphs whose corresponding reliability polynomial have a crossing point in the interval $(0, 1)$ (see [9] p.48). In order to prove that a graph is uniformly most reliable graph, the following observation is wide used.

Observation 1.1. *Let G, G' be graphs such that $N_i(G) \leq N_i(G')$, for all $0 \leq i \leq m$. Then Rel$(G, p) \leq$ Rel(G', p) for all $p \in [0, 1]$.*

Hence, maximizing the number of pathsets N_i has been a classical method to obtain uniformly most reliable graphs. We point out that the converse assertion of Observation 1.1 is not known to be true.

There exist uniformly most reliable graphs for $m \leq n+3$, $m \geq \binom{n}{2} - n$, and other sporadic values (see Table 1). The uniformly most reliable graphs for $m \leq n+3$ are *subdivisions* of certain small graphs. The *subdivision* operation for an edge uv is the deletion of uv from the graph and the addition of two edges uw and wv along with the new vertex w. A graph which has been derived from G by a sequence of subdivision operations is called a *subdivision* of G. For instance, every uniformly most reliable graph in the case $m = n+1$ ($n \geq 5$) is a particular *θ-graph*, constructed taking a graph of order 5 as a basis and subdividing edges and adding vertices one by one by following the sequence $A, B, C, A, B, C, \dots$ (see Fig. 1). The characteristics of this construction are that the resulting graphs have nearly or equal path lengths. This graph is also called the *Monma graph* [26]. Uniformly most reliable graphs can be also obtained as subdivisions of small graphs for $m = n+2$ and $m = n+3$. We will denote these extremal graphs as UMR(n, m). It has been conjectured that UMR$(n, n+4)$ are particular subdivisions of the Wagner graph for $n \geq 8$ (see [22]). Despite these general constructions, uniformly most reliable graphs have been found only for $(n, m) = (8, 12)$ (Wagner), $(10, 15)$ (Petersen) and $(n, m) = (12, 18)$ (Yutsis 18j-symbol label F). Besides, candidates for uniformly most reliable graphs have been found using heuristics in [3] for even orders between 14 and 20.

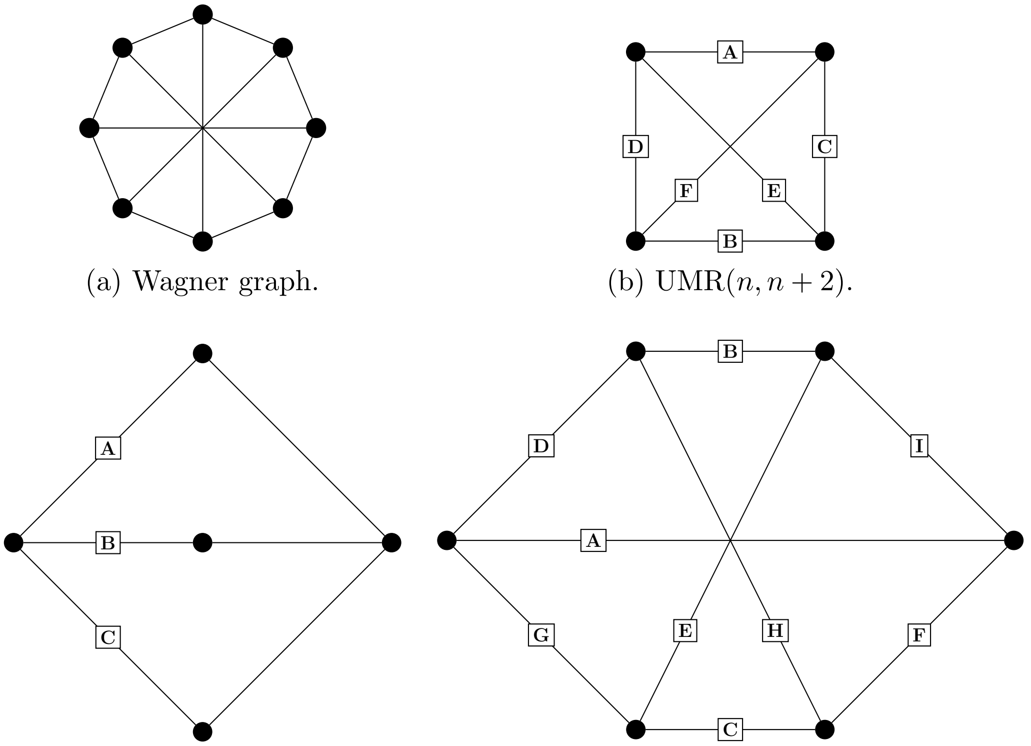
(n, m)	UMR graphs for $m \leq n+3$ and other sporadic values	Reference
$(n, n-1)$	UMR$(n, n-1)$ (Tree graphs T_n)	
(n, n)	UMR(n, n) (Cycle graphs C_n)	
$(n, n+1)$	UMR$(n, n+1)$ (θ-graphs)	[26]
$(n, n+2)$	UMR$(n, n+2)$ (subdivision of K_4)	[23]
$(n, n+3)$	UMR$(n, n+3)$ (subdivision of $K_{3,3}$)	[25]
$(8, 12)$	Wagner graph	[22]
$(10, 15)$	Petersen graph	[21]
$(12, 18)$	Yutsis 18j-symbol label F graph	[7]

Table 1: Uniformly most reliable graphs known in $\mathcal{G}(n, m)$, where n denotes the number of vertices and m the number of edges, for $m \leq n+3$ and other sporadic values.

Besides, a graph G with $m \geq \binom{n}{2} - \frac{n}{2}$ edges is uniformly most reliable if its complement graph has a matching (see [15]). The remaining values in the range $\binom{n}{2} - n \leq m < \binom{n}{2} - \frac{n}{2}$ have also uniformly most reliable graphs characterized by their corresponding complementary graph (see [19, 11]). On the other hand, there are infinitely many values of n and m where uniformly most reliable graphs do not exist (see [17, 5]).

Main contributions of the paper

We study uniformly most optimal graphs inside the family of hamiltonian graphs with n vertices and m edges. First, in section 2, we see that most uniformly most optimal graphs are not hamiltonian, so we perform a deep study in cases $m \leq n+2$. To this end, we compute the reliability polynomial in terms of what we call the chord type graph and the chord-path lengths vector, obtaining the reliability polynomial of any hamiltonian graph. We present an algorithmic approach for the construction of a family of graphs $FCG_{n,m}$, based in a 'fair cutting cake process', that provides uniformly most reliable hamiltonian graphs in some cases. In Section 2.3 we take advantage of the computational tools developed for the construction of $FCG_{n,m}$ graphs to give some light for the case $m = n+3$ and beyond. In particular, a modified version of the Factoring theorem has been developed and therefore we have been able to compute every uniformly most reliable hamiltonian graph with diametrical chords for all values of $n \leq 34$. We also obtain a few uniformly most reliable hamiltonian graphs for $m = n+4$ and $m = n+5$. Section 2.4 is devoted to prove the non-existence of uniformly most reliable hamiltonian graphs for infinitely many values of n and m.



(a) Wagner graph. (b) UMR$(n, n+2)$.

(c) UMR$(n, n+1)$. (d) UMR$(n, n+3)$.

Figure 1: Uniformly most reliable graphs.

2 Uniformly most reliable hamiltonian graphs

Interconnection networks are usually modeled by graphs. In this context, the design of optimal topologies is based on finding graphs satisfying some requirements of their properties. The study of connectivity, fault tolerance and/or reliability is of utmost importance in this area (see [20]). Also hamiltonicity is a common requirement in the design of some network topologies (also called 'ring embedding' problem, see [27]). However the study of uniformly most reliable hamiltonian graphs is a challenging problem in designing suitable topologies under certain requirements.

Uniformly most reliable graphs have been classically studied on the set $\mathcal{G}(n, m)$ of (simple) graphs on n vertices and m edges. A restricted version of the problem is the study of these extremal graphs inside the family of hamiltonian graphs. Let us denote as $\mathcal{H}(n, m)$ the set of (non-isomorphic) hamiltonian graphs with n vertices and m edges. Of course, $\mathcal{H}(n, m)$ is a subset of $\mathcal{G}(n, m)$ and since every hamiltonian graph on n vertices and m edges contains a spanning cycle, then we may assume that $m \geq n$. If a particular graph G is uniformly most reliable in $\mathcal{G}(n, m)$ and it is in addition hamiltonian, then G is also uniformly most reliable in $\mathcal{H}(n, m)$. This happens trivially in case (n, n) for all $n \geq 3$, where C_n are uniformly most reliable graphs, and also in $(8, 12)$ and $(12, 18)$, where the Wagner and Yutsis graphs are uniformly most reliable graphs, respectively, and also hamiltonian. It is well known that the Petersen graph is not hamiltonian (see [14]) and in UMR$(n, m \leq n+3)$, when n is large enough, uniformly most reliable graphs are not hamiltonian, as next result shows.

Proposition 2.1. *The following uniformly most reliable graphs in* $\mathcal{G}(n, m)$*,*

(a) UMR$(n, n+1)$ *is hamiltonian if and only if* $n = 4$*;*

(b) UMR$(n, n+2)$ *is hamiltonian if and only if* $n \leq 8$*;*

(c) UMR$(n, n+3)$ *is hamiltonian if and only if* $n \le 13$*;*

Proof. For any graph G containing a vertex v of degree $d(v) \ge 3$ and such that there are at least three different neighbors w_1, w_2, w_3 of v such that $d(w_1) = d(w_2) = d(w_3) = 2$, then G is not hamiltonian. Indeed, any vertex of G uses exactly two edges in any hamiltonian cycle, hence the only two incident edges of w_1, w_2 and w_3 belong to any hamiltonian cycle. But one of these two pairs of edges is incident also to v, and hence v would use three edges in any hamiltonian cycle, which is impossible. The graph UMR$(n, n+1)$ ($n \ge 5$) given in Figure 1 has a vertex v of degree 3 and that all its neighbors have degree 2. Hence it is not hamiltonian. Besides UMR$(4, 5)$ exists and is $K_4 - e$ (a complete graph of order 4 where an edge has been removed) which is a hamiltonian graph. It is easy to find a hamiltonian cycle in UMR$(n, n+2)$ for $4 \le n \le 8$, when the four subdivisions A, B, C, D are performed at most. But for $n \ge 9$ there exist a vertex of degree 3, such that all its neighbors have degree 2, and hence it is no longer hamiltonian. The argument for case (c) is similar: the sequence of edges (subdivided or not) A, D, B, E, C, F gives a hamiltonian cycle for every $n \le 12$ and the next subdivision (edge G) produces a vertex of degree 3 such that all its neighbors have degree 2. □

So the natural question about which hamiltonian graphs are uniformly most reliable (whenever they do exist) arises. We partially answer this question in the following sections.

2.1 The case $m = n + 1$

Let us consider G as a hamiltonian graph with n vertices $\{0, 1, \ldots, n-1\}$ and $m = n+1$ edges where the hamiltonian cycle is given by the sequence $0, 1, \ldots, n-1, 0$. Without loss of generality, we assume that the remaining edge of the graph joins vertex 0 and vertex x_1. We may consider that $2 \le x_1 \le \lfloor \frac{n}{2} \rfloor$ (the remaining cases $\lfloor \frac{n}{2} \rfloor < x_1 < n-2$ produce an isomorphic graph taking $x_1' = n - x_1$). Then, the reliability polynomial of G is given by

$$\mathrm{Rel}(G, p) = p^m + mp^{m-1}(1-p) + \tau p^{m-2}(1-p)^2, \tag{3}$$

where $\tau = n + x_1(n - x_1)$. Indeed, the removal of one edge or less guarantee the connectivity of the graph (the edge-connectivity of G is precisely $\lambda = 2$ for $n \ge 4$). Besides, deleting any set of three edges or more disconnects the graph ($N_i = 0$ for all $i < m-2$). Hence, just the coefficient N_{m-2} is unknown. This is precisely the tree number τ. The number of spanning trees of G is $n + x_1(n - x_1)$ since we have two cases: if edge $(0, x_1)$ is removed, then we can remove any of the n remaining edges. Otherwise we may delete two edges belonging to the hamiltonian cycle. In order to guarantee the connection of the resulting graph, we must choose one edge from the cycle $0, 1, \ldots, x_1, 0$ and the other from the cycle $0, x_1, x_1+1, \ldots, n, 0$. There are x_1 edges in one cycle and $n - x_1$ in the other. This gives the total number $n + x_1(n - x_1)$.

Proposition 2.2. *The set of hamiltonian graphs* $\mathcal{H}(n, n+1)$*,* $n \ge 4$*, is totally ordered by the reliability polynomial and the uniformly most reliable graph is given by joining any of two vertices of* C_n *at maximum distance.*

Proof. For any graph G in $\mathcal{H}(n, n+1)$, there exist x_1, with $2 \le x_1 \le \lfloor \frac{n}{2} \rfloor$ such that G is isomorphic to a graph with hamiltonian cycle $0, 1, \ldots, n-1, 0$ plus an edge $(0, x_1)$. According to (3), for any p, the value of $\tau(G) = -x_1^2 + nx_1 + n$ attains the maximum at $x_1 = \lfloor \frac{n}{2} \rfloor$. This graph G is isomorphic to the graph constructed from C_n by joining two vertices at maximum distance. □

Although the most reliable graph constructed from C_n and adding one single edge was previously known (see [22]) it is good to notice that $\mathcal{H}(n, n+1)$ is totally ordered by the reliability polynomial for all $n \ge 4$, since in general, this is not true for $m \ge n+2$ where reliability polynomials can cross for $p \in (0, 1)$.

2.2 The case $m = n+2$

Here we will present uniformly most reliable graphs in $\mathcal{H}(n, n+2)$. The reliability polynomial for any $G \in \mathcal{H}(n, n+2)$ is,

$$\mathrm{Rel}(G, p) = p^m + mp^{m-1}(1-p) + N_{m-2}p^{m-2}(1-p)^2 + \tau p^{m-3}(1-p)^3. \tag{4}$$

Any hamiltonian graph G of order n and size $n+2$ can be graphically depicted in a circular embedding where every vertex of G is in a hamiltonian cycle C of G together with 2 more edges (*chords*) through the hamiltonian cycle C. Then, the hamiltonian cycle is split by the chords in four paths of lengths x_1, x_2, x_3 and x_4, where $x_1 + x_2 + x_3 + x_4 = n$ and $x_i \in \mathbb{Z}^+$, for all $1 \le i \le 4$. In fact, we have two more situations regarding the relative position between the chords (see Figure 2). When $x_i = 0$ for an specific $1 \le i \le 4$ then we fall in what we call a *degenerated* case. However, for any chord-type graph $G \in \mathcal{H}(n, n+2)$ we associate its corresponding vector (x_1, x_2, x_3, x_4). We will refer this vector as the vector of *chord-path lengths*. In the other way around, given any positive integer vector (x_1, x_2, x_3, x_4), such that $x_1 + x_2 + x_3 + x_4 = n$, we can construct any graph G of $\mathcal{H}(n, n+2)$ of any type. Moreover, since any cyclic permutation of (x_1, x_2, x_3, x_4) produces an isomorphic graph, we can just consider those vectors modulo cyclic permutations.

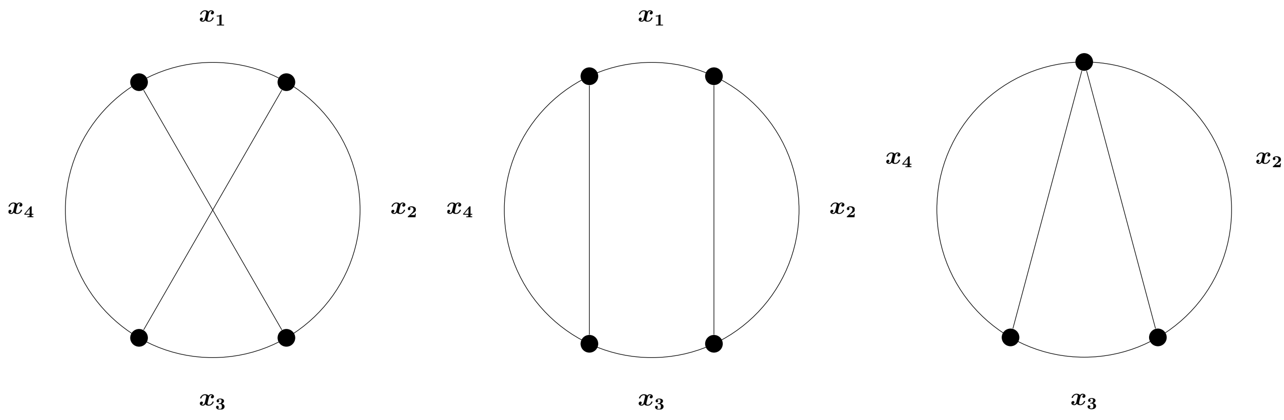


Figure 2: Different chord-types of hamiltonian graphs with two chords. The degenerated case is a chord-type A (or B) graph with $x_1 = 0$.

Next we will compute coefficients N_{m-2} and τ in Eq. (4) depending on (x_1, x_2, x_3, x_4).

Proposition 2.3. *Let G be a chord-type A graph in $\mathcal{H}(n, n+2)$ with vector of chord-path lengths (x_1, x_2, x_3, x_4). Then,*

$$\begin{aligned} N_{m-2} &= 1 + 2n + \sum_{1 \le i < j \le 4} x_i x_j. \\ \tau &= n + (x_1 + x_2)(x_3 + x_4) + (x_1 + x_4)(x_2 + x_3) + \sum_{1 \le i < j < k \le 4} x_i x_j x_k. \end{aligned} \tag{5}$$

Proof. The coefficient N_{m-2} counts the number of connected graphs after the deletion of two edges of G. If these two edges are the chords, then we have just one graph (the hamiltonian cycle). Besides, if the removed edges are one chord and one edge of the cycle, then, we can choose between 2 chords and n edges of the hamiltonian cycle. This gives $2n$ graphs. Finally, if the removed edges belong to the hamiltonian cycle, then both edges must be from different paths defined by the chords. This gives the number $\sum_{1 \le i < j \le 4} x_i x_j$. The computation of τ is similar, we need to remove three edges of G in order to obtain a spanning tree of G. We proceed depending of which edges we are dealing with:

1. *Two chords plus one edge of the cycle*: Any edge of the cycle can be removed after the removal of the two chords, so we count n spanning trees of this type.

2. *One chord plus two edges of the cycle*: After the removal of one chord, say c_1, we must remove one edge of each path defined by the other chord. Since these paths have lengths $x_1 + x_2$ and $x_3 + x_4$, the number of spanning trees is $(x_1 + x_2)(x_3 + x_4)$. The same applies for the other chord, say c_2, where now the paths lengths are $x_1 + x_4$ and $x_2 + x_3$.

3. *Three edges of the cycle*: The graph remain connected after the deletion of three edges of the cycle only if these three edges belong to different paths defined by the chords. Hence the number of spanning trees in this case is given by the all different three products of x_1, x_2, x_3, x_4.

□

Chord-type B graphs are not interesting for the study of uniformly most-reliable hamiltonian graphs, as next result states:

Proposition 2.4. *Let (x_1, x_2, x_3, x_4) be a chord-path lengths vector of G_A and G_B, which are chord-type graphs of types A and B, respectively. Then, $\tau(G_A) > \tau(G_B)$ and $N_{m-2}(G_A) \geq N_{m-2}(G_B)$.*

Proof. Following the ideas behind the proof of proposition 2.3 one can see that

$$\begin{aligned} N_{m-2}(G_B) &= 1 + 2n + \Big(\sum_{1\leq i<j\leq 4} x_i x_j\Big) - x_1 x_3, \\ \tau(G_B) &= n + (x_1 + x_2 + x_4)x_3 + (x_2 + x_4 + x_3)x_1 + \sum_{1\leq i<j<k\leq 4} x_i x_j x_k - (x_1 x_2 x_3 + x_1 x_3 x_4). \end{aligned} \tag{6}$$

A simple comparison of both formulas with the ones given in Proposition 2.3 gives the desired result,

$$\begin{aligned} &N_{m-2}(G_A) = N_{m-2}(G_B) + x_1 x_3 \geq N_{m-2}(G_B) \text{ and} \\ &\tau(G_A) = \tau(G_B) + x_1 x_3 x_4 + (x_1 x_2 + 2x_1)x_3 > \tau(G_B). \end{aligned}$$

□

Notice that equations (5) and (6) coincide when $x_1 = 0$, since they represent the same 'degenerated' graph (the one depicted in Figure 2). In order to obtain those integer vectors (x_1, x_2, x_3, x_4) such that produce uniformly most-reliable hamiltonian graphs, we should maximize both N_{m-2} and τ in equation (4). To this end, let us consider the set of chord-path length vectors $X_n = \{(x_1, x_2, x_3, x_4) \in \mathbb{Z}_+^4 \mid x_1 + x_2 + x_3 + x_4 = n\}$ and the functions

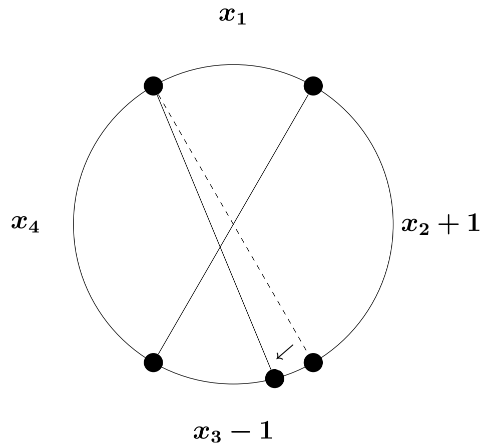
$$\begin{array}{rccc} f\colon & X_n & \longrightarrow & \mathbb{Z} \\ & (x_1, x_2, x_3, x_4) & \longmapsto & \tau \end{array} \quad \text{and} \quad \begin{array}{rccc} g\colon & X_n & \longrightarrow & \mathbb{Z} \\ & (x_1, x_2, x_3, x_4) & \longmapsto & N_{m-2} \end{array} \tag{7}$$

where τ and N_{m-2} are given in Prop. 2.3. Let $n = 4k+\alpha$, $\alpha \in \{0, 1, 2, 3\}$. For any $(x_1, x_2, x_3, x_4) \in X_n$ we define

$$D(x_1, x_2, x_3, x_4) = \sum_{i=1}^{4} |x_i - k|$$

as a measure of closeness to the constant vector (k, k, k, k). Then the elements of X_n can be measured according to its closeness to this constant vector. For instance, when $\alpha = 1$, there is just one vector in X_n with $D = 1$, which is $(k + 1, k, k, k)$ (any cyclic permutation of this vector of chord-path lengths gives an isomorphic graph, so we do not take them into account). Notice that D must be odd when $\alpha = 1$, so next D is 3 and the set of vectors of chord-path lengths with $D = 3$ is $\{(k+1, k+1, k-1, k), (k, k+2, k-1, k), (k, k+2, k, k-1)\}$.

We also define the following graphical operators in X_n: given a graph with vector of chord-path lengths $(x_1, x_2, x_3, x_4) \in X_n$, the operator σ moves just one vertex of a chord in such a way that two contiguous path lengths are modified by one unity each (see Fig. 3), that is, $\sigma(x_1, x_2, x_3, x_4) = (x_1, x_2 + 1, x_3 - 1, x_4)$. Besides the operator r defined as $r(x_1, x_2, x_3, x_4) = (x_2, x_3, x_4, x_1)$ is simply a rotation of the graph. For instance, in the case $\alpha = 1$, starting from the single vector with $D = 1$ one can obtain all the vectors of chord-path lengths for $D = 3$: $\sigma(k+1, k, k, k) = (k+1, k+1, k-1, k)$, $\sigma \circ r(k+1, k, k, k) = \sigma(k, k+1, k, k) = (k, k+2, k-1, k)$ and $r^{-1} \circ (\sigma \circ r)^2(k+1, k, k, k) = (k, k+2, k, k-1)$.



$\sigma(x_1, x_2, x_3, x_4) = (x_1, x_2 + 1, x_3 - 1, x_4)$

Figure 3: Graphical representation of operator σ.

Clearly, functions f, g and D are invariants under operator r. This is no longer true for operator σ. In contrast, we have the following result.

Lemma 2.1. *Let* $\mathbf{x} = (x_1, x_2, x_3, x_4) \in X_n$ *and* $\mathbf{y} = (y_1, y_2, y_3, y_4) \in X_n$ *such that* $D(\mathbf{y}) = D(\mathbf{x}) + 2$ *and* $\mathbf{y} = \sigma(\mathbf{x})$*. Then* $f(\mathbf{x}) \geq f(\mathbf{y})$ *and* $g(\mathbf{x}) \geq g(\mathbf{y})$*.*

Proof. Suppose that $\mathbf{y} = \sigma(\mathbf{x})$, that is, $y_1 = x_1$, $y_2 = x_2 + 1$, $y_3 = x_3 - 1$ and $y_4 = x_4$. From $D(\mathbf{y}) = D(\mathbf{x}) + 2$ we have that

$$|(x_2 + 1) - k| - |x_2 - k| + |(x_3 - 1) - k| - |x_3 - k| = 2.$$

Which is equivalent to

$$\big(|(x_2 - k) + 1| - |x_2 - k|\big) + \big(|(x_3 - k) - 1| - |x_3 - k|\big) = 2. \tag{8}$$

Taking into account that $|a+1| - |a| \leq 1$ and $|b-1| - |b| \leq 1$ and both equalities hold if and only if $a \geq 0$ and $b \leq 0$, respectively. Then equation (8) is equivalent to $x_2 \geq k$ and $x_3 \leq k$. Besides, from equation (5) we have $f(\mathbf{x}) - f(\mathbf{y}) = (x_2 - x_3)(x_1 + x_4 + 1)$. From $x_2 \geq k$ and $x_3 \leq k$ we have $(x_2 - x_3) \geq 0$ and hence $f(\mathbf{x}) - f(\mathbf{y}) = (x_2 - x_3)(x_1 + x_4 + 1) \geq 0$ since both factors are positive. Besides, again from equation (5), $g(\mathbf{x}) - g(\mathbf{y}) = x_2 - x_3 + 1$ which is also positive since $(x_2 - x_3) \geq 0$. □

Theorem 2.1. *A uniformly most reliable graph* G *exists in* $\mathcal{H}(n, n+2)$*. Moreover,* G *is a chord-type* A *graph with vector* (x_1, x_2, x_3, x_4) *of chord-path lengths equal to*

- (k, k, k, k) *if* $n = 4k$ *for some positive integer* k*,*
- $(k+1, k, k, k)$ *if* $n = 4k + 1$ *for some positive integer* k*,*
- $(k+1, k, k+1, k)$ *if* $n = 4k + 2$ *for some positive integer* k*,*

- $(k+1, k+1, k+1, k)$ *if* $n = 4k+3$ *for some positive integer* k.

Proof. By proposition 2.4 we have to take into account only graphs of type A. In order to maximize the coefficients of the reliability polynomial, let us consider the discrete functions $f : X_n \to \mathbb{Z}$ and $g : X_n \to \mathbb{Z}$ defined in equation (5). We want to find the maximum of the discrete functions f and g in X_n. By Lemma 2.1, the elements of X_n with minimum D are precisely the candidates for being a maximum of the functions f and g. Hence we just have to look at those chord path length vectors with minimum D. If $n = 4k$, then the constant vector $(k, k, k, k) \in X_n$ has minimum D and hence this is precisely the chord path length vector giving the maximum value for τ and N_{m-2} in (5). For the remaining cases, that is, when $n = 4k + \alpha$, $\alpha \in \{1, 2, 3\}$. Then $(\frac{n}{4}, \frac{n}{4}, \frac{n}{4}, \frac{n}{4}) = (k + \frac{\alpha}{4}, k + \frac{\alpha}{4}, k + \frac{\alpha}{4}, k + \frac{\alpha}{4}) \notin X_n$ and we want to find the chord path length vectors that give the maximum value of f and g in X_n. When $\alpha = 1$, there is only one vector of chord-path lengths with minimum D, which is $(k+1, k, k, k)$.

For $\alpha = 2$, the set of vectors with minimum D is $\{(k+2, k, k, k), (k+1, k+1, k, k), (k+1, k, k+1, k)\}$. According to equation (5), the last two vectors are maximum with the same value for g, which is $6k^2 + 14k + 6$, meanwhile for f we have that $(k+1, k, k+1, k)$ has maximum value since $f(k+1, k, k+1, k) = 4k^3 + 14k^2 + 14k + 4 = f(k+2, k, k, k) + 2k + 2 = f(k+1, k+1, k, k) + 1$.

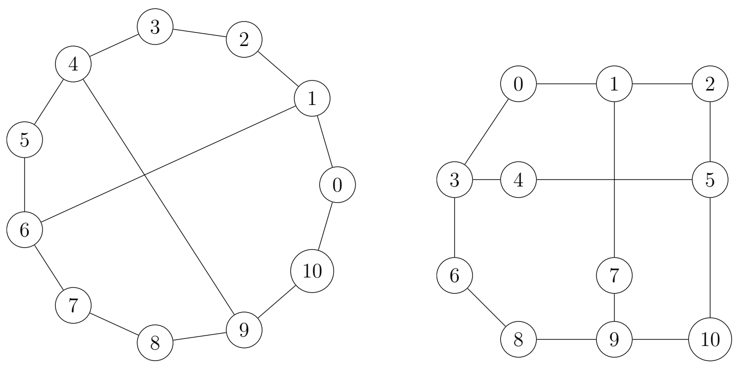
Finally, for $\alpha = 3$, the set of vectors $\{(k+3, k, k, k), (k+2, k+1, k, k), (k+2, k, k+1, k), (k+1, k+1, k+1, k)\}$ has minimum D. The vector of chord-path lengths $(k+1, k+1, k+1, k)$ achieves the maximum both for g and f, where $g(k+1, k+1, k+1, k) = 6k^2 + 17k + 10$ and $f(k+1, k+1, k+1, k) = 4k^3 + 17k^2 + 22k + 8 = f(k+3, k, k, k) + 6k + 5 = f(k+2, k+1, k, k) + 2k + 3 = f(k+2, k, k+1, k) + 2k + 1$. □

Example: For $n = 11$ and $m = 13$, the uniformly most reliable hamiltonian graph must have vector of chord-path lengths $(3, 3, 3, 2)$ and $N_{m-2} = 68$ and $\tau = 152$ (according to Theorem 2.1, case $k = 2$ and $\alpha = 3$). We also used *Nauty*[1] to generate all non-isomorphic graphs with 11 vertices and 13 edges. There are 33851 of such graphs. Then we filtered the hamiltonian ones using a function from a *Python* library called *Graphx*. There are 56 hamiltonian graphs in this case and the uniformly most reliable hamiltonian graph is the expected one. We also computed the uniformly most reliable graph among the total set of graphs (which corresponds to a particular subdivision of K_4, as expected). We have depicted both graphs in Figure 4. The computational method to obtain the reliability polynomial is explained in Section 2.3.

The fair cake-cutting graph $FCG_{n,c}$

In this section we present an algorithmic construction of a family of graphs $FCG_{n,c}$ that produces uniformly most reliable graphs in some cases. We introduce this family of graphs inspired by the results of Theorem 2.1, where uniformly most reliable graphs are those with chord-path length vectors with almost equal components. From a geometrical point of view, these optimal graphs can be seen as 'fair cuts of a cake', where every slice is as similar as possible to the other slices. As far as we know, there are different definitions for this *fair cake-cutting process* in the literature (see [4, 22]). For our purposes, we have to cut a cake (modeled as a circular drawing of the cycle C_n) performing a given number of cuts c. The idea is to deliver each part as equal (fair) as possible for every guest. Each cut of the cake is represented in the graph by a diametrical chord, that is, one edge joining two opposite vertices. Given the order n of the hamiltonian cycle and the number of chords c ('cuts' in terms of cake cutting), we present an algorithm to construct what we call *The fair cake-cutting graph* $FCG_{n,c}$. Since the cuts can be performed only over the lines joining two opposite vertices of C_n, the idea behind this algorithm is that the slices have 'almost' equal areas in the plane.

[1] http://users.cecs.anu.edu.au/ bdm/nauty



$N_{m-2} = 68$ and $\tau = 152$. $N_{m-2} = 70$ and $\tau = 160$.

Figure 4: Uniformly most realiable graphs in $\mathcal{H}(11, 13)$ and $\mathcal{G}(11, 13)$, respectively, and the corresponding coefficients N_{m-2} and τ of their reliability polynomials.

```
   input  : n ← Number of desired vertices for the FCG
            c ← Number of desired chords for the FCG
   output: FCG
1  graph fcg ← Cycle with edges (0, 1), (1, 2), . . . , (n − 1, 0).
2  float separation = n/(2c)
3  float position = separation
4  int placedch = 0
5  while placedch ≤ c do
6      Add edge (int(position) − 1, int(position + n/2) − 1)
7      //-1 due that the vertices begin with 0
8      position = position + separation //note that the variable is still a float
9      placedch = placedh + 1
10 end
11 return graph
```

Algorithm 1: Fair Cake construction

A detailed explanation of the algorithm is the following: With the given number of vertices, the algorithm constructs its corresponding cycle. After, the vertices of its hamiltonian cycle are stored into a list. Then, the 'separation' between vertices of degree 3 (corresponding to the endpoints of the chords) is calculated. This 'separation' is also stored in the variable 'position' that in the first iteration of the loop will be one of the first chord endpoints. The loop in line 5 places all the cuts to the previously generated cycle. Line 6-9: Each edge chord is added to the graph using the vertices stored inside the hamiltonian cycle list in the corresponding positions. Each position is calculated by adding the previous position plus the separation. The last endpoint of each cut must have maximum distance from its first endpoint, this distance is equal to the half of the vertices of the graph. Notice that the positions of the list must be a natural number but its decimal part is not overlooked. This part is added when the next position is calculated. Finally, the cycle graph with all the chords placed is returned.

Example: For $n = 16$ and $c = 3$, we start from a cycle graph C_{16} with edge set $\{(0, 1), (1, 2), \ldots, (15, 0)\}$. This table summarizes the procedure that the algorithm follows to calculate the positions (p_1, p_2) of the hamiltonian path list for each chord (v_1, v_2): First the separation of the

# cut	Separation	Accumulated	Total	p_1	p_2	(v_1, v_2)
1st	$16/6 = 2.\overline{6}$	0	$2.\overline{6}$	2	$2 + 8 = 10$	$(2-1, 10-1)$
2nd	$16/6 = 2.\overline{6}$	$2.\overline{6}$	$5.\overline{3}$	5	$5 + 8 = 13$	$(5-1, 13-1)$
3rd	$16/6 = 2.\overline{6}$	$5.\overline{3}$	8	8	$8 + 8 = 16$	$(8-1, 16-1)$

Table 2: Algorithm of construction of $FCG_{16,3}$

cuts is calculated, $\frac{n}{2c} = \frac{16}{6} = 2.\overline{6}$ and then the accumulated part is added. This part is the summation of all previously calculated values (Totals). With this information, the positions of the chord endpoints can be obtained: p_1 is equal to the decimal part from the 'Total'. $p_2 = p_1 + \frac{n}{2}$. Finally, the vertices of each chord are extracted from the hamiltonian path list using the previous positions: For instance, since the vertices starts with 0, at position 5 we have the vertex 4, and the position 13 the vertex 12, therefore we add edge $(4, 12)$.

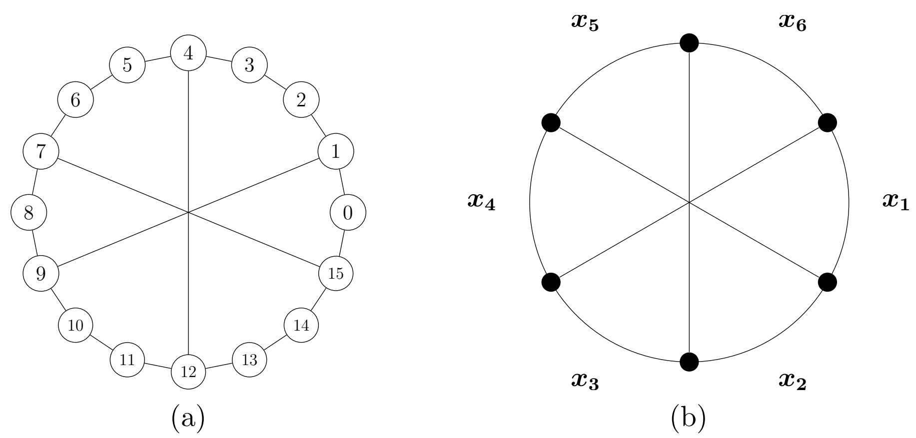


Figure 5:
(a) The fair cake-cutting graph $FCG_{16,3}$. It has vector of chord-path lengths $(2, 3, 3, 2, 3, 3)$.
(b) A chord-type A graph in $\mathcal{H}(n, n + 3)$ with diametrical chords.

The fair cake-cutting graphs $FCG_{n,c}$ are similar to the family of graphs that are a solution of the following augmentation problem: Starting from the cycle graph C_n, add a single edge at each step, in order to maximize the reliability of the resulting graph. Romero (see [22, 21]) finds the sequence of graphs $\{G^{(i)}\}_{i=0,\ldots,\lfloor \frac{n}{2} \rfloor}$ with $G^{(0)} = C_n$ such that $G^{(i+1)} = G^{(i)} \cup \{e_{i+1}\}$ gives the best augmentation. This process (called also the fair cake-cutting process) ends with the circulant graph with steps 1 and $\frac{n}{2}$, that is, a cubic hamiltonian graph where every vertex is joined to its opposite vertex in the hamiltonian cycle. For $n = 8$, this is the Wagner graph depicted in Figure 1, which is a uniformly most reliable graph for $(8, 12)$. The main difference between both families is that in our case the number of cuts c is previously known, meanwhile, in this other family, the cuts are performed as the guest arrives. In fact, both families differ when three or more cuts are performed. As a consequence of Theorem 2.1 and Proposition 2.2 we have that $FCG_{n,c}$ produces uniformly most reliable hamiltonian graphs for $c = 1$ and $c = 2$.

Corollary 2.1. *$FCG_{n,1}$ and $FCG_{n,2}$ are uniformly most reliable hamiltonian graphs for $m = n + 1$ and $m = n + 2$, respectively.*

2.3 Computational approach for $m = n + 3$ and beyond.

Uniformly most reliable hamiltonian graphs have been totally characterized when $m \leq n+2$ in section 2. We also give a construction of these optimal graphs in section 2.2. Beyond this point, we have performed some computational tools in order to generate uniformly most reliable hamiltonian graphs for $m \geq n + 3$. First, we discuss the computation of the reliability polynomial of a graph.

The computation of the reliability polynomial

The factoring theorem is a recursive method for computing $\mathrm{Rel}(G, p)$ based on the combination of two graph operations: edge deletion $G - e$, and edge contraction G/e (for further details see, for instance, [18]):

$$\mathrm{Rel}(G,p) = \begin{cases} \mathrm{Rel}(G-e,p) & \text{if } e \text{ is a loop,} \\ p\mathrm{Rel}(G/e,p) & \text{if } e \text{ is a cut-edge,} \\ (1-p)\mathrm{Rel}(G-e,p) + p\mathrm{Rel}(G/e,p) & \text{otherwise.} \end{cases} \tag{9}$$

The algorithm 2 shows the implementation of the theorem done in our code.

```
input  : g ← Graph
output: Reliability Polynomial
// If the graph is not connected, then it has a reliability polynomial of 0
if g is not connected then
    return 0
end
// if the number of edges > 0, then we perform the two sub-cases of the Factoring Theorem
if number of edges of g > 0 then
    edge e = g.random_edge(e)
    graph contracted = g.contract_edge(e)
    graph deleted = g.delete_edge(e)
    polynomial rec_contracted = recursion wih the graph contracted
    polynomial rec_deleted = recursion wih the graph deleted
    polynomial s = p · rec_contracted + (1 − p) · rec_deleted
    return s
end
// Otherwise, we only have 0 edges and 1 vertex, which is connected, so we return 1.
return 1
```

Algorithm 2: Reliability Polynomial Factoring Theorem

We perform a modified version of this *Factoring Theorem* which works slightly different: In each recursion, if there exist some method that can directly retrieve the reliability polynomial or with less cost than another recursion, then, the method will retrieve it and the recursion will stop in that generated subgraph. In other words the main idea to improve this algorithm is to prevent it to 'dismantle' the graph to its very basic components (trivial graphs) by giving the reliability of the subgraphs before becoming basic components. We developed a series of fast formulas for specific families of graphs such as multi-tree, multi-cycle and glued cycles that when one of the subgraphs matches one of the families of our formulas then, the reliability is directly returned. With this modified method we have been able to compute the reliability polynomial of all graphs in $\mathcal{H}(n, n+3)$ for any $n \leq 11$. All (non-isomorphic) graphs have been generated first using *Nauty* and we get the hamiltonian ones using the library *Graphx* from *Python*. The drawback of this function is that, at the worst case, runs in linear time ($O(n)$). The coefficients list N_i of the reliability polynomial of those uniformly most reliable hamiltonian graphs is presented in Table 3.

Study case	$Rel(H,p)$ coefficients vector (N_i)
$\mathcal{H}(6,9)$	$[1, 9, 36, 78, 81]$
$\mathcal{H}(7,10)$	$[1, 10, 44, 104, 117]$
$\mathcal{H}(8,11)$	$[1, 11, 53, 137, 168]$
$\mathcal{H}(9,12)$	$[1, 12, 63, 178, 240]$
$\mathcal{H}(10,13)$	$[1, 13, 74, 226, 328]$
$\mathcal{H}(11,14)$	$[1, 14, 86, 284, 445]$

Table 3: Coefficients list of uniformly most reliable hamiltonian graphs for $6 \leq n \leq 11$ and $m = n+3$.

Every graph listed in Table 3 is of type A (in the sense explained in section 2.2) and it has some diametrical chords (chords joining two vertices at maximum distance in the hamiltonian cycle, see figure 5). Although we do not have a proof that uniformly most reliable hamiltonian graphs must be of type A with diametrical chords for $m = n+3$, this experimental result encourage us to look optimal graphs in this subset of hamiltonian graphs that we denote as $H_D(n, m)$. So, we designed a direct way to construct all graphs in $H_D(n, n+3)$ that lead us to go beyond $n = 11$ vertices: starting with a cycle, draws a diametrical chord dividing the cycle in two parts. Then, makes a set of 2-element combinations between each group of vertices from each part. Notice that all the edges in this set cross the first diametrical chord. With this set it makes another 2-element combinations, but this time with the elements inside the created edge set. Finally, with each combination of 2 edges creates a new graph by adding them into the graph with the diametrical chord. The algorithm 3 shows in detail this process.

The generated list of graphs contains many isomorphic graphs. Then we use *nauty* to remove isomorphic graphs from the list. We compute the reliability polynomial of each graph of this shorter list with the method explained at the first part of this section. Table 4 shows the uniformly most reliable graphs of chord-type A with at least one diametrical chord for $n \leq 34$ and $m = n + 3$.

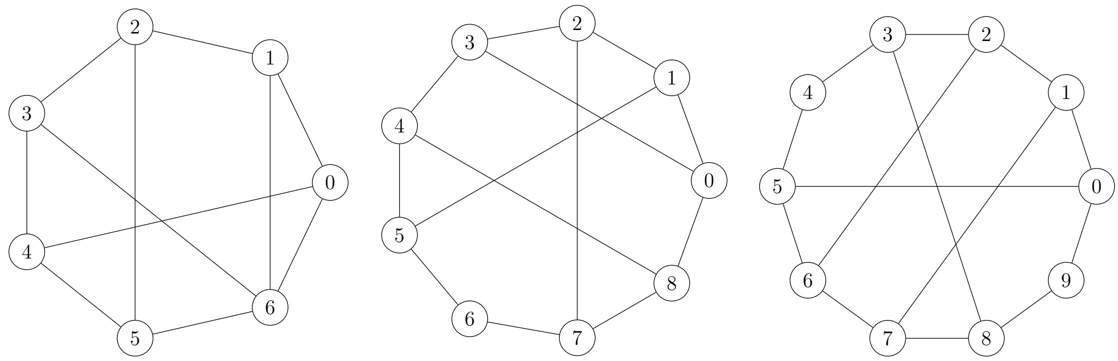
Our conjecture about the uniformly most reliable hamiltonian graph in $H(n, m)$ must be in $H_D(n, m)$ is no longer true for $m \geq n + 4$. In Figure 6 we present some uniformly most reliable hamiltonian graphs for $m = n + 4$ found by computer. Despite the case $n = 8, m = 12$ (see Figure 1 (a)), the remaining cases are not of chord-type A. The situation for $m = n + 5$ is similar (see Figure 7).

We have been able to find all uniformly most reliable hamiltonian graphs up to $n = 11$ vertices and $m = 16$ edges using the general method described at the beginning of the section: First we generate a list containing all non-isomorphic graphs of a given order and size using *Nauty*. Afterwards, the reliability polynomial of every graph in the list is computed by using our improved version of the *factoring theorem*.

```
input  : n ← Number of vertices
output: List of type A hamiltonian graphs with at least one diametrical chord

graph cycle ← Cycle with edges (0,1),(1,2),...,(n − 1,0).
list vertices ← cycle.vertices
// Set the diametrical chord
cycle ← add edge {0, floor(n/2)}
// Remove the vertices of the added chord from vertices
vertices ← remove {0, floor(n/2)}
// Get possible chords
list hvertices1 ← 1st half of the list vertices
list hvertices2 ← 2nd half of the list vertices
list possibleVertices ← all combinations of elements between hvertices1 and hvertices2
// From all non-existent possible edges, get combinations of 2 chords
// Notice that we already added 1 chord (the diametrical one)
list edgeCombinations ← all combinations of 2 elements from the list possibleVertices
// For each combination create a graph and save it into a list of graphs
list hamiltonians
for combination in edgeCombinations do
    graph tmp ← copy(cycle)
    tmp ← add edges in combination
    hamiltonians ← add tmp
end
return hamiltonians
```

Algorithm 3: Generation of chord-type A hamiltonian graphs with three diametrical chords.



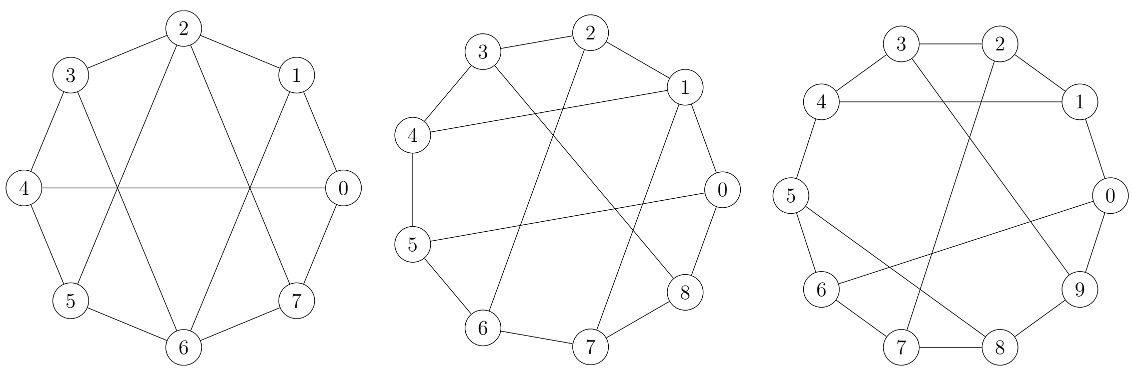
(a) $n = 7, m = 11$ (b) $n = 9, m = 13$ (c) $n = 10, m = 14$

Figure 6: Some uniformly most reliable hamiltonian graphs for $m = n + 4$.

Study case	$Rel(H,p)$ coefficients vector (N_i)
$\mathcal{H}_\mathcal{D}(12,15)$	$[1,15,99,353,600]$
$\mathcal{H}_\mathcal{D}(13,16)$	$[1,16,112,422,755]$
$\mathcal{H}_\mathcal{D}(14,17)$	$[1,17,126,502,948]$
$\mathcal{H}_\mathcal{D}(15,18)$	$[1,18,141,594,1188]$
$\mathcal{H}_\mathcal{D}(16,19)$	$[1,19,157,697,1464]$
$\mathcal{H}_\mathcal{D}(17,20)$	$[1,20,174,814,1799]$
$\mathcal{H}_\mathcal{D}(18,21)$	$[1,21,192,946,2205]$
$\mathcal{H}_\mathcal{D}(19,22)$	$[1,22,210,1078,2611]$
$\mathcal{H}_\mathcal{D}(20,23)$	$[1,23,229,1225,3088]$
$\mathcal{H}_\mathcal{D}(21,24)$	$[1,24,249,1388,3648]$
$\mathcal{H}_\mathcal{D}(22,25)$	$[1,25,270,1566,4272]$
$\mathcal{H}_\mathcal{D}(23,26)$	$[1,26,292,1762,4995]$

Study case	$Rel(H,p)$ coefficients vector (N_i)
$\mathcal{H}_\mathcal{D}(24,27)$	$[1,27,315,1977,5832]$
$\mathcal{H}_\mathcal{D}(25,28)$	$[1,28,338,2192,6669]$
$\mathcal{H}_\mathcal{D}(26,29)$	$[1,29,362,2426,7620]$
$\mathcal{H}_\mathcal{D}(27,30)$	$[1,30,387,2680,8700]$
$\mathcal{H}_\mathcal{D}(28,31)$	$[1,31,413,2953,9880]$
$\mathcal{H}_\mathcal{D}(29,32)$	$[1,32,440,3248,11209]$
$\mathcal{H}_\mathcal{D}(30,33)$	$[1,33,468,3566,12705]$
$\mathcal{H}_\mathcal{D}(31,34)$	$[1,34,496,3884,14201]$
$\mathcal{H}_\mathcal{D}(32,35)$	$[1,35,525,4225,15864]$
$\mathcal{H}_\mathcal{D}(33,36)$	$[1,36,555,4590,17712]$
$\mathcal{H}_\mathcal{D}(34,37)$	$[1,37,586,4978,19704]$

Table 4: Coefficients list of uniformly most reliable hamiltonian graphs of type A with at least one diametrical chord.



(a) $n = 8, m = 13$ (b) $n = 9, m = 14$ (c) $n = 10, m = 15$

Figure 7: Some uniformly most reliable hamiltonian graphs for $m = n + 5$.

A (simple) graph of order n has at most $m = \binom{n}{2}$ edges. There is only one of such graphs with maximum number of edges, which is the complete graph K_n and hence it is uniformly most reliable. There is also only one graph (up to isomorphisms) with $m = \binom{n}{2} - 1$ edges, but for $m = \binom{n}{2} - 2$ we have two different graphs: we can eliminate from K_n either a path of length two or two independent edges. This latter case gives the uniformly most reliable graph. More in general, it is already known that when $m \geq \binom{n}{2} - \frac{n}{2}$ then there exist a uniformly most reliable graph which is the graph whose complement graph have a set of independent edges (see [16]). Any graph with a number of edges large enough is hamiltonian, and hence uniformly most reliable graphs are hamiltonian in this case. For instance, it is well known that every graph satisfying $m \geq 1/2(n^2 - 3n + 6)$ is hamiltonian (see [8]). Every graph with $m \geq \binom{n}{2} - \frac{n}{2}$ satisfies also $m \geq 1/2(n^2 - 3n + 6)$ whenever $n \geq 6$ and hence uniformly most reliable graphs in this case are also uniformly most reliable hamiltonian.

Proposition 2.5. *Given an integer $n \geq 6$, there is a uniformly most reliable graph in $\mathcal{H}(n,m)$ for any $m \geq \binom{n}{2} - \frac{n}{2}$.*

2.4 Non existence of uniformly most reliable hamiltonian graphs for some cases

There are some cases where uniformly most reliable graphs do not exist: it is shown in [17] that the graph G_2 of order $n \geq 6$ even, which is defined as the complement of the graph $P_4 \cup K_2 \cup \frac{n-6}{2} K_2$ satisfies

$\mathrm{Rel}(G_2,p) > \mathrm{Rel}(G',p)$ for all $G' \in \mathcal{G}(n,m)$, $m = \binom{n}{2} - \frac{n+2}{2}$ and p sufficiently close to one. Besides, the complement of $2P_3 \cup \frac{n-6}{2}K_2$, defined as G_1, satisfies $\mathrm{Rel}(G_1,p) > \mathrm{Rel}(G_2,p)$ for p sufficiently close to zero. As a consequence, there is no uniformly most reliable graphs in $\mathcal{G}(n,m)$ for $m = \binom{n}{2} - \frac{n+2}{2}$. We use this result to prove that there are infinitely many pairs (n,m) where uniformly most reliable hamiltonian graphs do not exist.

Proposition 2.6. *There are no uniformly most reliable graphs in $\mathcal{H}(n,m)$ for*

- $m = \binom{n}{2} - \frac{n+2}{2}$ *for all $n \geq 6$ even;*
- $m = \binom{n}{2} - \frac{n+5}{2}$ *for all $n \geq 7$ odd.*

Proof. For the case $m = \binom{n}{2} - \frac{n+2}{2}$ it is suffice to show that the graphs G_1 and G_2 defined above as the complements of the graphs $2P_3 \cup \frac{n-6}{2}K_2$ and $P_4 \cup K_2 \cup \frac{n-6}{2}K_2$ for $n \geq 6$ even, respectively, are hamiltonian graphs. To this end notice that the minimum degree for a vertex in G_1 is $n-3$. Hence for any two given vertices u and v, $d(u) + d(v) \geq 2n - 6 \geq n$ for all $n \geq 6$. Applying Theorem 1.1 we deduce that G_1 is hamiltonian. The same argument applies for G_2.

For $n \geq 7$ odd, define G_3 as the complement of $C_3 \cup P_4 \cup \frac{n-7}{2}K_2$ and G_4 as the complement of $C_5 \cup K_2 \cup \frac{n-7}{2}K_2$ as in [17] and apply again Theorem 1.1 to proof that they belong to $\mathcal{H}(n,m)$, where $m = \binom{n}{2} - \frac{n+5}{2}$. In [17] it is shown that $\mathrm{Rel}(G_4,p) > \mathrm{Rel}(G',p)$ for all $G' \in \mathcal{G}(n,m)$ for p sufficiently close to one. Besides $\mathrm{Rel}(G_3,p) > \mathrm{Rel}(G_4,p)$ for p sufficiently close to zero, and the proof is completed. □

3 Conclusions and open problems

In this paper uniformly most reliable hamiltonian graphs have been characterized for $m \leq n+2$ and we give some light for the case $m = n+3$ which somehow follow the previous cases. The situation is different for $m \geq n+4$, where the problem is totally open except for some small cases found by computer. Nevertheless, one can use these results to obtain uniformly most reliable hamiltonian graphs for a fixed value of n. For instance, when $n = 6$ we have nine cases to analyze ($6 \leq m \leq 15$): For $m = 6$, C_6 is the uniformly most reliable hamiltonian graph. For $7 \leq m \leq 9$ we have that $FCG_{6,c}$ are uniformly most reliable hamiltonian graphs for $c = 1, 2, 3$, respectively (also $K_{3,3}$ for $m = 9$, which is isomorphic to $FCG_{6,3}$). For $m = 11$ there is no uniformly most reliable hamiltonian graph (Proposition 2.6) and for $m \geq 12$ subgraphs of K_6 whose complement graph has a set of independent edges are uniformly most reliable (Proposition 2.5). It remains the case $m = 10$. We have found by computer that adding any edge to the graph $K_{3,3}$ produces the uniformly most reliable graph in this case. We point out that in every case, the uniformly most reliable graph found in this paper has the largest coefficient vector.

Problem 3.1. *Characterize uniformly most reliable hamiltonian graphs for other values of n and m.*

The fair cake-cutting graph $FCG_{n,c}$ presented in section 2.2 is a uniformly most reliable graph for many values, but we believe that is also optimal for other values.

Conjecture 3.1. *Let $n \equiv 0 \pmod{2c}$. Then $FCG_{n,c}$ is a uniformly most reliable hamiltonian graph for $m = n + c$.*

The problem of finding uniformly most reliable graphs seems difficult, even for restricted versions of the problem, like the one we present here for hamiltonian graphs. It would be worth to study this problem for other classes of graphs.

Problem 3.2. *Study the problem of finding uniformly most reliable graphs for a named class of graphs, such as bipartite graphs, Cayley graphs, etc.*

Acknowledgments

The authors would like to thank the anonymous referees whose useful comments lead to an improvement of the manuscript. Research of the authors was supported in part by grant MTM2017-86767-R (Spanish Ministerio de Ciencia e Innovacion).

References

[1] F. T. Boesch, *On the synthesis of optimally reliable networks having unreliable nodes but reliable edges*, in INFOCOM '88. Networks: Evolution or Revolution, Proceedings. Seventh Annual Joint Conference of the IEEE Computer and Communcations Societies, IEEE, Mar 1988, pp. 829–834.

[2] F. T. Boesch, X. Li, and C. Suffel, *On the existence of uniformly optimally reliable networks*, Networks, 21 (1991), pp. 181–194.

[3] M. Bourel, E. Canale, F. Robledo, P. Romero, and L. Stábile, *A hybrid grasp/vnd heuristic for the design of highly reliable networks*, in Hybrid Metaheuristics, Cham, 2019, Springer International Publishing, pp. 78–92.

[4] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia, *Handbook of Computational Social Choice*, Cambridge University Press, 2016.

[5] J. Brown, D. Cox, and R. Ehrenborg, *The average reliability of a graph*, Discrete Applied Mathematics, 177 (2014), pp. 19 – 33.

[6] J. Brown and X. Li, *Uniformly optimal digraphs for strongly connected reliability*, Networks, 49 (2007), pp. 145–151.

[7] E. Canale, F. Robledo, P. Romero, and J. Viera, *Building reliability-improving network transformations*, in 2019 15th International Conference on the Design of Reliable Communication Networks (DRCN), 2019, pp. 107–113.

[8] G. Chartrand, L. Lesniak, and P. Zhang, *Graphs & Digraphs*, Chapman & Hall, 6th ed., 2015.

[9] C. J. Colbourn, *The Combinatorics of Network Reliability*, Oxford University Press, New York, NY, USA, 1987.

[10] S. Evenand and R. Tarjan, *Network flow and testing graph connectivity*, SIAM Journal of Computing, 4 (1975), pp. 507–518.

[11] B. Gilbert and W. Myrvold, *Maximizing spanning trees in almost complete graphs*, Networks, 30 (1997), pp. 97–104.

[12] D. Gross, J. Saccoman, and C. Suffel, *Spanning Tree Results for Graphs and Multigraphs: A Matrix-Theoretic Approach*, World Scientific Publishing Company, 2014.

[13] D. Gross and J. T. Saccoman, *Uniformly optimally reliable graphs*, Networks, 31 (1998), pp. 217–225.

[14] R. J. Gould, *Advances on the hamiltonian problem - a survey*, Graphs and Combinatorics, 19 (2003), pp. 7–52.

[15] A. Kelmans and V. Chelnokov, *A certain polynomial of a graph and graphs with an extremal number of trees*, Journal of Combinatorial Theory, Series B, 16 (1974), pp. 197–214.

[16] A. K. Kelmans, *On graphs with randomly deleted edges*, Acta Mathematica Academiae Scientiarum Hungarica, 37 (1981), pp. 77–88.

[17] W. Myrvold, K. H. Cheung, L. B. Page, and J. E. Perry, *Uniformly-most reliable networks do not always exist*, Networks, 21 (1991), pp. 417–419.

[18] H. Perez-Roses, *Sixty years of network reliability*, Mathematics in Computer Science, 12 (2018), pp. 275–293.

[19] L. Petingi, F. Boesch, and C. Suffel, *On the characterization of graphs with maximum number of spanning trees*, Discrete Mathematics, 179 (1998), pp. 155–166.

[20] J. Rak, M. Pickavet, K. Trivedi, and et al, *Future research directions in design of reliable communication systems*, Telecommun Syst, 60 (2015), pp. 423–450.

[21] G. Rela, F. Robledo, and P. Romero, *Petersen graph is uniformly most-reliable*, in Machine Learning, Optimization, and Big Data, Cham, 2018, Springer International Publishing, pp. 426–435.

[22] P. Romero, *Building uniformly most-reliable networks by iterative augmentation*, in 2017 9th International Workshop on Resilient Networks Design and Modeling (RNDM), Sep. 2017, pp. 1–7.

[23] A. Satyanarayana, L. Schoppmann, and C. L. Suffel, *A reliability-improving graph transformation with applications to network reliability*, Networks, 22 (1992), pp. 209–216.

[24] D. H. Smith and L. L. Doty, *On the construction of optimally reliable graphs*, Networks, 20 (1990), pp. 723–729.

[25] G. Wang, *A proof of boesch's conjecture*, Networks, 24 (1994), pp. 277–284.

[26] J. F. Wang and M. H. Wu, *Network reliability analysis: on maximizing the number of spanning trees*, Proceeding of the National Science Council, Republic of China, Part A, Physical Science and Engineering, 11 (1987), pp. 193–196.

[27] B. Yener, *Virtual embeddings on regular topology networks*, Proceedings of SPDP '96: 8th IEEE Symposium on Parallel and Distributed Processing, (1996), pp. 562–565.