The Complexity of a Reliable Distributed System

Rachid Guerraoui
EPFL

Alexandre Maurer
EPFL

Abstract

Studying the complexity of distributed algorithms typically boils down to evaluating how the number of messages exchanged (resp. communication steps performed or shared memory operations executed) by nodes to reliably achieve some common task, evolves with the number n of these nodes. But what about the complexity of building the distributed system itself? How does the number of physical network components (e.g., channels and intermediary nodes acting as routers), needed for building a system of n nodes to ensure some global reliable connectivity property, evolves with n? Addressing such a question lies at the heart of achieving the dream of *elasticity* in so-called *cloud computing*.

In this paper, we show for the first time how to construct a distributed system of which any two of the n nodes, for any n, remain connected (i.e., able to communicate) with probability at least μ, despite the very fact that (a) every other node or channel has an independent probability λ of failing, and (b) the number of channels connected to every node is physically bounded by a constant. We show however that if we also require any two of the n nodes to maintain a balanced message throughput with a constant probability, then $O(n \log^{1+\epsilon} n)$ additional intermediary nodes are necessary and sufficient, where ϵ is an arbitrarily small constant.

Our distributed system constructions, based on the composition of fractal and tree-like graphs, are not claimed to be simple and cost-effective enough to constitute the architectural blueprints of the next generation cloud data centers with millions of computers. Yet, they might constitute their theoretical backbone.

1 Introduction

The growth of modern networks seems to be exceeding Moore's Law [30]. More and more computers are getting connected in cloud computing centers handling massive data storage [6, 4]. We talk for example about 60,000 cores for the Human Brain Project [1] and over 100,000 for the CERN data center [2]. Companies like Google and Microsoft have data centers with millions of servers [3]. Not surprisingly, the problem of how to achieve the dream of *cloud elasticity* and effectively connect a very large number of computers has been extensively studied [20, 7, 28, 14, 15, 27, 8, 10]. In particular, a lot of attention has been devoted to maintaining a *reliable* message *throughput* (i.e., avoid traffic congestion), even when the size of the network increases [15, 18, 12, 14, 23, 31, 24]. A major difficulty that hinders such elasticity is the *bounded* (by a physical constant) capacity of network components (computers and channels): there is a maximal number of messages per second that a channel can transmit, and a maximal number of channels that a node (computer) can connect. A closer look at all existing cloud constructions [20, 7, 28, 14, 15, 27, 8, 10] reveals in fact that, strictly speaking, traffic congestion increases when the size of the network increases. This is without even accounting for *failures*: when the size of the network increases, the probability that several components of the network *fail* also increases [32, 17, 14, 28], making it even more difficult to maintain any stable throughput. This paper asks the question of the "theoretical price of elasticity". We seek to determine the complexity, in terms of the number of networks components required, of constructing a distributed system that preserves a stable message throughput despite failures, even when the number n of nodes of the system increases significantly.

We proceed incrementally. We first address what we call the RBD (**R**eliable **B**ounded **D**egree) problem, of how to connect a set of nodes so that every pair can communicate with probability at least μ, assuming that any other node or channel has an independent probability at most λ to crash [29, 19].[1] (We leave aside any throughput requirement as well as Byzantine failures in this first step.) Building a "complete graph", connecting any two nodes with a channel is not a solution as the node degree (i.e., the number of channels connected to a given node) explodes.[2] In fact, the RBD problem might actually seem impossible without additional *intermediary* nodes between the n nodes (acting as routers and not necessarily reliably connected to the rest), and this is what we thought for a long while. When n increases, the diameter of the graph also increases: pairs of nodes become more distant from each other, inevitably dragging down the communication probability. Compensating for this loss of reliability by adding redundant paths between any pair of (distant) nodes is infeasible for the number of parallel paths is bounded by the maximal degree whereas the network diameter keeps increasing with n.

We show in this paper how to address the RBD problem (with no additional intermediate nodes). For any number of nodes n, we show how to build a graph of n nodes that ensures arbitrarily high reliability while preserving a bounded degree. We proceed in two substeps. We first solve the **W**eak **RBD** (WRBD) problem, which goal is to reliably connect n nodes with a graph of bounded degree, by allowing to add intermediary nodes between these n nodes, provided that their number is $O(n)$ (at most linear in n). We do so by defining a *fractal* graph that ensures a constant communication probability between any two given nodes (independently of their distance) with a bounded degree, expressing the communication probability as a *convergent sequence*, and then a *tree-like floor* graph reliably connecting n nodes. We then use the solution to the WRBD problem to solve our seemingly stronger RBD problem (i.e., reliably connecting n nodes *without* intermediary nodes).[3] The idea is to combine several instances of a WRBD graph, each instance reliably connecting a smaller number of nodes, and to make their intermediary nodes "disappear" by merging them with other nodes.

We then address the problem of *message throughput*. We model the exchanges of messages by continuous and "fluid" *flows* of messages. Each of the n nodes needs to transmit the same flow of messages to the $n-1$ other nodes.[4] Assuming a bound, independent from n, on (1) the maximal *degree* of the network and (2) the maximal *flow* of the network, i.e., the maximal flow of messages crossing each node and channel, we address the **BDF** (**B**ounded **D**egree and **F**low) problem (first leaving aside the reliability requirement), which consists in finding a graph that enables to maintain the flow of messages between the n nodes. Again, the constraint on the degree prevents a "complete graph" directly connecting each pair of n nodes. Thus, some flows of messages will have to go through *intermediary* nodes. At first glance, one might consider using these intermediary nodes in a tree topology, of which the leaves would be the n nodes. However, a tree network is problematic for all messages would need to cross the root node, making the maximal flow increase with n. In fact, we prove that solving the BDF problem requires at least $\Omega(n \log n)$ intermediary nodes. Basically, the bounded degree implies a distance $\Omega(\log n)$ between most pairs of nodes, and the resulting amount of messages has to be distributed over a minimal number of intermediary nodes, due to the bounded capacity. We then describe a graph solving the BDF problem using $O(n \log n)$ intermediary nodes, which matches the lower bound. Essentially, our solution is again "multi-floor"

[1] Solving our RBD problem should not be confused with requiring the entire graph to remain connected with probability μ, which would clearly be impossible. Indeed, given that the node degree is physically bounded by a constant, when the size of the network increases, the probability that all channels surrounding some node crash approaches 1. There can be no lower bound on the probability that the whole graph remains connected.

[2] In fact, all network topologies that were proposed to reliably connect a large number of nodes with a "reasonable" degree [20, 7, 28, 14, 15, 27, 8, 10] were empirical and have only been experimented through simulations: their performances were evaluated only for a specific number of nodes. If we consider the asymptotic behavior of their proposed graphs (i.e., when the number of nodes grows), either the communication probability approaches zero, or the maximal degree approaches infinity. In [11, 5, 25], the focus was to construct a graph satisfying certain topological properties. In [11] and [5] the node degree is not bounded, whereas in [25], the length of the paths between two given nodes increases with the number of nodes. When each node or channel has a given probability to fail, the probability that the k paths are cut approaches 1.

[3] The construction works with any graph solving the WRBD problem.

[4] Here, "identical" means that any node p sends the same quantity of messages to any two nodes q and r, which does not mean that the messages sent to q and r are the same.

and consists in stacking $O(\log n)$ floors of $O(n)$ nodes each, and then crossing the flow of messages between each floor so that (1) the flow of messages crossing each node remains constant and (2) the flows of messages are uniformly "mixed" when reaching the last floor. We merge the first and the last floor of the graph, enabling each one of the n nodes to exchange messages with the $n-1$ other nodes.

Finally, we combine the RBD and BDF problems and define the **RBDF** (**R**eliable **B**ounded **D**egree and **F**low) problem. As for RDB, we assume that each node and channel has a given probability λ to crash, and that each pair of nodes (among the n initial nodes) must keep exchanging the same flow of messages with probability μ. We also define a fractal graph that ensures reliable communication between any two nodes, at whatever distance they may be (w.r.t the parameters λ and μ). Then, we make a "floor by floor" product of this graph with the BDF "multi-floor" graph, in order to combine this reliability property with the bounded degree and flow properties. The number of intermediary nodes of the resulting graph then goes from $O(n \log n)$ to $O(n \log^{1+\epsilon} n)$, where ϵ is a positive constant that can be as small as wanted. In other words, the additional cost of the reliability property lies in a factor $\log^{\epsilon} n$, where ϵ can be as small as wanted.

Interestingly, all our constructions have an optimal (logarithmic) diameter. Besides, they can be extended to tolerate Byzantine failures (when the failed components, i.e., nodes or channels, behaves arbitrarily), assuming the failure rate λ to be strictly smaller than 0.5, by (1) increasing the level of redundancy (compared to the case of crash failures) and (2) adding several layers of majority votes to eliminate malicious messages.

The rest of the paper is organized as follows. Section 1 presents our model and Section 2 defines the problems we address. The following sections (4-7) present solutions to these problems. In Section 8, we prove the correctness of our 4 graph constructions. In Section 9, we prove the complexity results: (1) solving the RBD problem requires at least $\Omega(n \log n)$ nodes, and our solution actually involves $O(n \log n)$; (2) our solution to the RBDF problem involves $O(n \log^{1+\epsilon} n)$ nodes, where ϵ is an arbitrarily small positive constant. In Section 10, we show that our solutions have an optimal (logarithmic) diameter. In Section 11, we explain how our solutions can be generalized to handle Byzantine failures. We conclude the paper in Section 12.

2 Model

A graph is a tuple $G = (V, E)$ where V is the set of *nodes* and E is the set of *channels*, modeled as a set with repetition of pairs of nodes $\{p, q\} \subseteq V$ (we enable multiple channels between p and q). The *degree* $\delta(v)$ of a node v is the number of channels (p, q) such that $p = v$ or $q = v$ (the number of channels connected to v). The *maximal degree* of graph G is $\max_{v \in V} \delta(v)$. A *path* connecting two nodes p and q is a sequence of nodes $(u_1, \ldots, u_m)$ such that $u_1 = p$, $u_m = q$ and $\forall i \in \{1, \ldots, m-1\}$, u_i and u_{i+1} are neighbors.

A *component* of a graph G is any node or channel of G. Each component of G can be either *correct* (functional) or *crashed* (failed). A *correct path* is a sequence of nodes $(p_1, \ldots, p_m)$ such that, $\forall i \in \{1, \ldots, m\}$, p_i is correct, and $\forall i \in \{1, \ldots, m-1\}$, there exists a correct channel $\{p_i, p_{i+1}\}$. Two nodes p and q are *connected* if there exists a correct path $(p_1, \ldots, p_m)$ such that $p_1 = p$ and $p_m = q$. We denote by $\lambda \in]0, 1[$ and $\mu \in]0, 1[$ two arbitrary constants.

Fluid Message Flow (FMF). Let $S \subseteq V$ be any arbitrary set of n nodes, with $n \geq 2$, representing the computers of the network that need to issue and exchange messages. The rest of the nodes are *intermediary* nodes corresponding to routers that forward the messages sent by the n computers of S: they do not issue messages of their own.

We consider a perfectly balanced distributed (peer-to-peer) system: each of the nodes of S sends the same quantity of messages to every other node. More precisely, we assume that each node

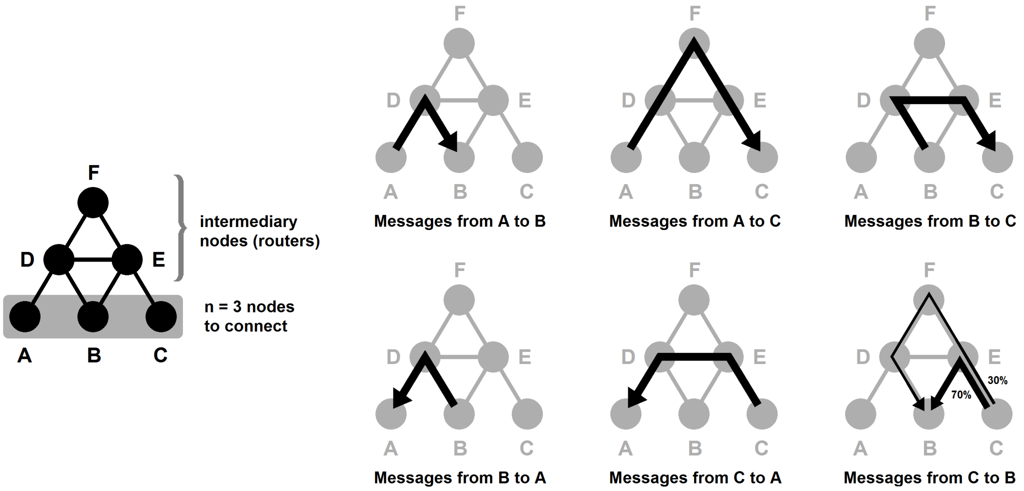

Figure 1: In this arbitrary graph, $n = 3$ nodes A, B and C are connected by 3 intermediary nodes D, E and F ($S = \{A, B, C\}$ here). The pictures describe the (arbitrary) paths used by the flow of messages from any node to any other node. The paths are not necessarily symmetrical: the path from A to C and the path from C to A are different. Besides, the flow of messages can be split into several paths: for the messages from C to B, 70% of the flow goes through (C, E, B), and 30% of the flow goes through (C, E, F, D, B). If we gather the six pictures, the maximal flow of messages is reached for node D.

$p \in S$ sends a flow of messages F, equally distributed between the $n-1$ other nodes of S.[5] Thus, for any two nodes p and q of S, p sends a flow of messages $F/(n-1)$ directed towards q. We now define the paths taken by these messages.

A *weighted path* is a tuple (P, α), where P is a path and α is an arbitrary coefficient. A weighted path represents a continuous flow of messages between two nodes p and q, where P is the path used by the messages, and α is the fraction of messages directed towards q. For any two nodes p and q of S, the flow of messages from p to q uses a set of weighted paths $R(p,q) = \{(P_1, \alpha_1), (P_2, \alpha_2), \dots, (P_m, \alpha_m)\}$. The paths $P_1, P_2, \dots, P_m$ are connecting p to q, and $\alpha_1 + \alpha_2 + \dots + \alpha_m = 1$. For each path P_i, the coefficient α_i corresponds to the fraction of the flow of messages using the path P_i. We illustrate this structure through a simple example in Figure 1.

Thus, the path P_i receives a flow $\alpha_i F/(n-1)$ of messages from p to q. We call the function R the *routing map* of S (which takes two nodes p and q of S as input, and returns a set of weighted paths in output). For instance, in the toy example of Figure 1, $R(C, B) = \{(P_1, 0.7), (P_2, 0.3)\}$, with $P_1 = (C, E, B)$ and $P_2 = (C, E, F, D, B)$.

We say that a path $(u_1, \dots, u_m)$ *crosses* a node p if there exists $i \in \{1, \dots, m\}$ such that $u_i = p$. Similarly, we say that this path *crosses* a channel $\{p, q\}$ if there exists $i \in \{1, \dots, m-1\}$ such that $u_i = p$ and $u_{i+1} = q$. A weighted path (P, α) crosses a node or channel x if the path P crosses x. For a given node or channel x, we now define the *flow* of messages $f(x)$ crossing x. Let $\Omega = \bigcup_{\{p,q\} \subseteq S} R(p, q)$ be the set containing *all* weighted paths used by the nodes of S. Let $W = \{(Q_1, \beta_1), (Q_2, \beta_2), \dots, (Q_k, \beta_k)\}$ be the set of weighted paths of Ω crossing x. Then, $f(x) = (\beta_1 + \beta_2 + \dots + \beta_n)F/(n-1)$ (the sum of the flows of messages crossing x). The *maximal flow* of (G, S, R) is $f_{\max} = \max_{(x \in V) \vee (x \in E)} f(x)$ (the maximal flow crossing a node or channel of G).

[5] We consider a "fluid", continuous flow of messages, to abstract away the granularity of messages. This continuous flow of messages does not represent the network at a given instant, but rather the quantity of messages exchanged in a given time period, which is assumed to be relatively stable.

Generalized Fluid Message Flow (GFMF). We generalize the previous model to take failures into account. Here, R_n now takes two additional parameters $\mathcal{V}$ and $\mathcal{E}$, where $\mathcal{V}$ (resp. $\mathcal{E}$) represents the set of faulty nodes (resp. channels) – that is, the routing map adapts to the failures of nodes and channels in order to find correct paths, when it is possible. Thus, a set of weighted paths $R_n(p,q)$ becomes $R_n^{\mathcal{V},\mathcal{E}}(p,q)$, and the routing map R_n becomes $R_n^{\mathcal{V},\mathcal{E}}$. If this set of paths does not contain any faulty node or channel, we say that p and q are *reliably connected*. We will first consider faults as crashes for simplicity of presentation and then, later, we will discuss Byzantine failures.

3 Problems

The WRBD (Weak Reliable Bounded Degree) problem consists in finding, for any $n \geq 2$, a graph G_n satisfying the three following properties:

1. **Reliability.** Assume each node and channel crashes with probability at most λ (the probabilities being independent). Then, there exists a set S_n of n nodes of G_n such that any two correct nodes of S_n are connected with probability at least μ.
2. **Bounded degree.** There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.
3. **Linear number of nodes.** There exists a constant C such that, $\forall n \geq 2$, the number of nodes of G_n is at most Cn.

The RBD (Reliable Bounded Degree) problem consists in finding, for any $n \geq 2$, a graph G_n containing *exactly* n nodes and satisfying the two following properties:

1. **Reliability.** Assume each node and channel crashes with probability at most λ (the probabilities being independent). Then, any two correct nodes of G_n are connected with probability at least μ.
2. **Bounded degree.** There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.

The BDF (Bounded Degree and Flow) problem considers the FMF model and consists in finding, for any $n \geq 2$, a tuple (G_n, S_n, R_n) – where G_n is a graph, S_n is a set of n nodes of G_n, and R_n is a routing map of S_n – satisfying the two following properties:

1. **Bounded Degree.** There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.
2. **Bounded Flow.** There exists a constant f_0 such that, $\forall n \geq 2$, the maximal flow of (G_n, S_n, R_n) is at most f_0.

The RBDF (Reliable Bounded Degree and Flow) problem considers the GFMF model and consists in finding, for any $n \geq 2$, a tuple $(G_n, S_n, R_n^{\mathcal{V},\mathcal{E}})$ – where $G_n = (V_n, E_n)$ is a graph, S_n is a set of n nodes of G_n, and $R_n^{\mathcal{V},\mathcal{E}}$ is a routing map of S_n – satisfying the three following properties:

1. **Bounded Degree.** There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.
2. **Bounded Flow.** There exists a constant f_0 such that, $\forall n \geq 2$, $\forall \mathcal{V} \subseteq V_n$ and $\forall \mathcal{E} \subseteq E_n$, the maximal flow of $(G_n, S_n, R_n^{\mathcal{V},\mathcal{E}})$ is at most f_0.

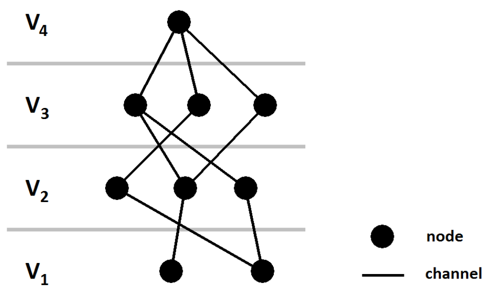

Figure 2: A floor graph of height $H = 4$.

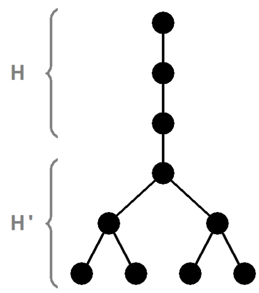

Figure 3: Structure of graph T_m.

3. **Reliability.** Assume each node and channel crashes with probability at most λ (the probabilities being independent). Let $\mathcal{V}$ (resp. $\mathcal{E}$) be the set of crashed nodes (resp. channels). Then, any two correct nodes of S_n are reliably connected in $R_n^{\mathcal{V},\mathcal{E}}$ with probability at least μ.

4 WRBD Graph

We define a graph G_n to solve the WRBD problem. We first give an overview, then the complete definition. The correctness proof is in Section 8.

Overview. We first define the notion of *floor graph*, namely a graph where nodes are separated into several "floors", and where only nodes of two adjacent floors can be connected. Then, we define two floor graphs: T_n, which contains a binary tree connecting at least n nodes, and F_n, which is a "fractal" graph defined by induction. The fractal definition of F_n enables to preserve a constant communication probability between the first and last floor (independently of n) when $\lambda < 0.01$ (Lemma 1).[6] We show how to overcome this "$\lambda < 0.01$" constraint below. Besides, F_n is defined so that the number of nodes doubles at most every 2 floors, which enables to preserve a linear number of nodes, as shown in Theorem 3. The number of floors of T_n is adjusted so that T_n and F_n have the same number of floors H_n.

We consider a graph X_n, which is a "floor by floor" product of T_n and F_n, and a graph Y_n, which puts two graphs X_n in parallel. Doing so ensures a constant communication probability between any two nodes of the first floor.

Finally, we make three transformations in order to reach any communication probability μ with any failure rate λ. First, we connect several graphs Y_n in parallel, in order to achieve any communication probability μ. Second, we replicate each node, in order to simulate a failure rate $\lambda < 0.01$ for each node. Third, we replicate each channel, in order to simulate a failure rate $\lambda < 0.01$ for each channel. The graph thus obtained is G_n.

Definitions. For any $n \geq 2$, let h_n be the smallest integer such that $2^{h_n - 1} \geq n$. Let K_n be the smallest integer such that $2 + 4K_n \geq h_n$, and let $H_n = 2 + 4K_n$. Let α be the smallest integer such that $\alpha \geq 1$ and $0.5^\alpha \leq 1 - \mu$. Let β be the smallest integer such that $\beta \geq 1$ and $\lambda^\beta \leq 0.01$.

A *floor graph* of height H is a tuple $(V_1, \ldots, V_H, E)$ satisfying the three following conditions:

1. (V, E) is a graph with $V = \bigcup_{i \in \{1,\ldots,H\}} V_i$.
2. The sets V_i ("floors") are disjoint: $\forall \{i, j\} \subseteq \{1, \ldots, H\}$, $V_i \cap V_j = \emptyset$.
3. The channels only connect neighbor floors: $\forall \{p, q\} \in E$, if $p \in V_i$ and $q \in V_j$, then $|i - j| = 1$.

[6] Note that this bound "$\lambda < 0.01$" is not supposed to be tight, and is simply small enough to have the desired property.

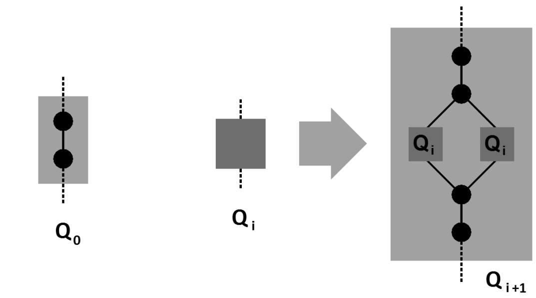

Figure 4: Construction (by induction) of fractal graph Q_i. The graph is defined so that the number of nodes doubles at most every 2 floors, which enables to preserve a linear number of nodes (see Theorem 3).

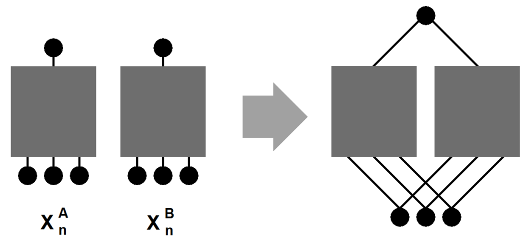

Figure 5: Construction of graph Y_n.

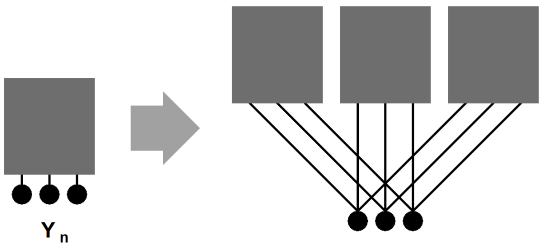

Figure 6: Transformation 1 (Network replication) with $\alpha = 3$.

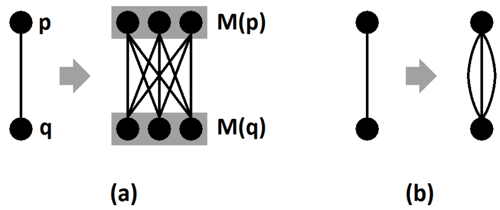

Figure 7: Transformations 2 (Node replication) and 3 (Channel replication) with $\beta = 3$.

An example of a floor graph is given in Figure 2. By convention, in the following figures, V_1 always corresponds to the lower floor on the figure. We call V_1 the "first floor" and V_H the "last floor".

Graph T_n. We first define a tree-like floor graph of height H_n. Consider the floor graph represented in Figure 3: this graph is composed of a line of height $H = 3$ and of a binary tree of height $H' = 3$. In other words, $\forall i \in \{1, \ldots, H'\}$, the floor i contains 2^{i-1} nodes, and the H remaining floors contain each 1 node. Then, $\forall n \geq 2$, we define T_n as a similar graph with $H = H_n - h_n$ and $H' = h_n$.

Graph F_n. $\forall k \geq 0$, we first define a floor graph Q_i by induction. Let Q_0 be a floor graph of height 2 containing 2 nodes and 1 channel, as described in Figure 4. Then, $\forall i \geq 0$, Q_{i+1} is constructed with 2 instances of Q_i in parallel and 4 additional nodes, as described in Figure 4 (Q_{i+1} has 4 more floors than Q_i). We now define F_n as follows: $\forall n \geq 2$, $F_n = Q_{K_n}$.

Graph X_n. $\forall n \geq 2$, T_n is a floor graph of height H_n, and F_n is a floor graph of height $2 + 4K_n = H_n$. As T_n and F_n are floor graphs, let $T_n = (V_1, \ldots, V_{H_n}, E)$ and $F_n = (V'_1, \ldots, V'_{H_n}, E')$. Then, $\forall n \geq 2$, we define the floor graph $X_n = (V^*_1, \ldots, V^*_{H_n}, E^*)$ as follows:

- $\forall i \in \{1, \ldots, H_n\}$, to each pair of nodes $(u, v) \in V_i \times V'_i$, we associate a unique node $p = f(u, v) \in V^*_i$ (thus, $|V^*_i| = |V_i||V'_i|$).
- Let $p = f(u, v)$ and $p' = f(u', v')$. Then, p and p' are neighbors in X_n if and only if u and u' (resp. v and v') are neighbors in T_n (resp. F_n).

Observe that, as the last floors of T_n and F_n contain 1 node, the last floor of X_n also contains 1 node.

Graph Y_n. $\forall n \geq 2$, we define the graph Y_n as follows: we consider two instances of X_n (X_n^A and X_n^B), we merge the nodes of their first floors, and we merge the nodes of their last floors. This is illustrated in Figure 5.

Graph G_n. $\forall n \geq 2$, the graph G_n is finally obtained by applying three successive transformations to Y_n:

1. **Transformation 1 (Network replication).** First, we connect α instances of Y_n by merging the nodes of their first floors. This is illustrated in Figure 6 for $\alpha = 3$.
2. **Transformation 2 (Node replication).** Second, we replace each node p by a set of β nodes $M(p)$. Then, for each channel $\{p, q\}$, we add a channel between each node of $M(p)$ and each node of $M(q)$ (see Figure 7-a).
3. **Transformation 3 (Channel replication).** Third, we replace each channel by β channels in parallel (see Figure 7-b).

5 RBD Graph

We define a graph G_n to solve the RBD problem. We first give an overview, then the complete definition. The correctness proof is in Section 8.

Overview. The idea is to combine several instances of a WRBD graph, each instance reliably connecting a smaller number of nodes, and to make their intermediary nodes "disappear" by merging them with other nodes.

Let W_m be any WRBD graph (for instance, the WRBD graph defined in Section 4). Then, $\forall n \geq 2$, we consider the largest integer m such that the number of nodes of W_m is at most n. If such a m does not exist, we define G_n as a complete graph with redundancy of channels. As it only happens for bounded values of n, it does not break the "Bounded degree" property.

Otherwise, we consider a set V of n nodes, and we split V into subsets of $\lfloor m/2 \rfloor$ nodes. Then, we connect each pair of subsets with an instance of W_m merged with the nodes of V. The resulting graph is G_n. Doing so ensures that any two nodes of V are reliably connected. Besides, according to the "Linear number of nodes" property of W_m, the number of instances of W_m is bounded, and so is the maximal degree of G_n.

Construction of G_n. Let $n \geq 2$, and let V be a set of n nodes.

Let W_m be a WRBD graph. Let N_m be the total number of nodes of W_m ($N_m \geq m$), and let S_m be the set of m nodes reliably connected by W_m.

If there exists no $m \geq 2$ such that $N_m \leq n$, then for any two nodes p and q of V, we add $\lceil \log(1-\mu)/\log(1-\lambda) \rceil$ channels between p and q ("complete graph" case).

Otherwise, let $m \geq 2$ be the largest integer such that $N_m \leq n$. Let M be the smallest integer such that $M\lfloor m/2 \rfloor \geq n$. Let $\{A_1, \ldots, A_M\}$ be a set of M subsets of V such that $\bigcup_{i \in \{1,\ldots,M\}} A_i = V$ and $\forall i \in \{1, \ldots, M\}$, $|A_i| = \lfloor m/2 \rfloor$.

Then, $\forall (i,j) \in \{1, \ldots, M\}^2$, we apply the following transformations. Let $W(i,j)$ be an instance of W_m, let $V(i,j)$ be the set of nodes of $W(i,j)$, and let $S(i,j)$ be the set of m nodes corresponding to S_m. Let $A(i,j)$ and $B(i,j)$ be two disjoint subsets of $S(i,j)$ such that $|A(i,j)| = |B(i,j)| = \lfloor m/2 \rfloor$. We merge the $\lfloor m/2 \rfloor$ nodes of $A(i,j)$ (resp. $B(i,j)$) with the $\lfloor m/2 \rfloor$ nodes of A_i (resp. A_j). Then, we merge the $N_m - 2\lfloor m/2 \rfloor$ nodes of $V(i,j) - A(i,j) - B(i,j)$ with any $N_m - 2\lfloor m/2 \rfloor$ nodes of $V - A_i - A_j$. The graph thus obtained is G_n.

6 BDF Graph

We define a tuple (G_n, S_n, R_n) to solve the BDF problem. We first give an overview, then the complete definition of G_n, S_n and R_n. The correctness proof is in Section 8.

Overview. To construct G_n, the intuitive idea is the following. We define a sequence $(X_1, \ldots, X_H)$ of sets of $O(n)$ nodes. X_1, X_2, …, X_H can be represented as tables of respectively $2^{H-1} \times 1$, $2^{H-2} \times 2$, …, $1 \times 2^{H-1}$ nodes (each time, the "width" is divided by two and the "height" is multiplied by two). Then, each node of X_i is connected to two nodes of X_{i+1} with the same "height" modulo 2 and the same "width" modulo 2^{H-i}.[7] Finally, we merge X_1 and X_H so that the sets of nodes form a cycle. As we show further, this construction enables to mix the flows of messages in a perfectly balanced way. S_n is an arbitrary set of n nodes of the first floor of G_n.

We then define the routing map R_n as follows. The flows of messages between two nodes p and q of S_n take a unique path $r(p, q)$ (p is seen as a node of X_1 and q as a node of X_H). The path is determined by the binary decomposition of the position of q in X_H: at each new step, 0 means "go down" ($v_{k+1} = x(b_k)$) and 1 means "go up" ($v_{k+1} = y(b_k)$). We show that $r(p, q)$ actually reaches q in the correctness proof.

Graph G_n. Let H be the smallest integer such that $2^{H-1} \geq n$ (as $n \geq 2$, $H \geq 2$). We consider H sets of nodes $(X_1, \ldots, X_H)$, containing 2^{H-1} nodes each. $\forall k \in \{1, \ldots, H\}$, we denote each node of X_k by $u_k(i, j)$, with $i \in \{1, \ldots, 2^{H-k}\}$ and $j \in \{1, \ldots, 2^{k-1}\}$ (this is possible as $2^{H-k} \times 2^{k-1} = 2^{H-1}$). We connect these H sets of nodes with communication channels as follows. $\forall k \in \{1, \ldots, H-1\}$, $\forall i \in \{1, \ldots, 2^{H-k-1}\}$ and $\forall j \in \{1, \ldots, 2^{k-1}\}$, let $a = u_k(2i-1, j)$, $b = u_k(2i, j)$, $x = u_{k+1}(i, 2j-1)$ and $y = u_{k+1}(i, 2j)$. Then, we add the following communication channels: $\{a, x\}$, $\{a, y\}$, $\{b, x\}$ and $\{b, y\}$. Finally, $\forall i \in \{1, \ldots, 2^{H-1}\}$, we merge the node $u_1(i, 1)$ with the node $u_H(1, i)$. The graph thus obtained is G_n.

Set of nodes S_n. We define S_n as an arbitrary subset of the set X_1, containing exactly n nodes. This is possible as $2^{H-1} \geq n$.

Routing map R_n. For a given node $v \in X_1 \cup \cdots \cup X_{H-1}$, let k, i and j be such that $v = u_k(i, j)$. Let i_0 be the smallest integer such that $2i_0 \geq i$. Let $x(v) = u_k(i_0, 2j-1)$ and $y(v) = u_k(i_0, 2j)$. Let $p \in X_1$ and $q \in X_H = X_1$. Let j be such that $q = u_H(1, j)$. Let $(b_1, \ldots, b_{H-1})$ be a binary sequence ($\forall k \in \{1, \ldots, H-1\}, b_k \in \{0, 1\}$) such that $j - 1 = \Sigma_{k=1}^{k=H-1} b_k 2^{H-k-1}$ (that is, the binary decomposition of $i-1$).

Let $v_1 = p$. We define v_{k+1} by induction: if $b_k = 0$, $v_{k+1} = x(b_k)$, and if $b_k = 1$, $v_{k+1} = y(b_k)$. Let $r(p, q) = (v_1, \ldots, v_H)$. Then, we define the routing map R_n by $R_n(p, q) = \{(r(p, q), 1)\}$.

7 RBDF Graph

We define a tuple $(G_n, S_n, R_n^{\mathcal{V},\mathcal{E}})$ to solve the RBDF problem. We first give an overview, then the complete definition of G_n, S_n and $R_n^{\mathcal{V},\mathcal{E}}$. The correctness proof is in Section 8.

[7] The "demultiplexing" properties of G_n are similar to those of a butterfly network. However, G_n is defined differently. In the butterfly network, the nodes of each floor are described by an index i. Here, they are described by two indexes i and j ("$u_k(i, j)$").

Overview. Let G_n^0 be the BDF graph defined in Section 6. After introducing preliminary definitions, we first define graph G_n. For this purpose, we define 4 intermediary graphs A_n, F_n, P_n and X_n. All these graphs are *floor graphs*, as introduced in Section 4, and have the same height H'_n. A_n is an variation of the previous graph G_n^0 with additional floors. F_n is a fractal graph designed to satisfy the reliability property. P_n is an adaptation of F_n to the reliability parameters λ and μ. Similarly to Section 4, X_n is a "floor by floor" product of A_n and P_n, in order to combine the properties of the previous graph G_n^0 with the reliability property of P_n. G_n is finally obtained by merging the first and the last floor of X_n, similarly to G_n^0. S_n is an arbitrary set of n nodes of the first floor of G_n.

To define the routing map $R_n^{\mathcal{V},\mathcal{E}}$, the intuitive idea is the following. For any two nodes p and q of S_n, we first define a subgraph $W(p,q)$. Schematically, if p' and q' are the two corresponding nodes in G_n^0, and $r(p',q')$ is the path connecting them, then $W(p,q)$ is the instance of B_n corresponding to $r(p',q')$ in G_n. Then, the routing map connects p and q with a unique path avoiding the crashed nodes and channels in $W(p,q)$ (if it exists).

Definitions. Let $\epsilon > 0$ be any arbitrary positive constant. ϵ is the constant determining the complexity of the graph. Therefore, it impacts many subsequent parameters.

Let K be the smallest integer such that $K \geq 2^{1/\epsilon}$. K is a parameter involved in the definition of graph F_n. $\forall n \geq 2$, let H_n be the smallest integer such that $2^{H_n - 1} \geq n$. We define the following sequence $(h_0, h_1, h_2, \dots)$ by induction: $h_0 = 1$, and $\forall i \geq 0$, $h_{i+1} = 2 + Kh_i$. $\forall n \geq 2$, let M_n be the smallest integer such that $h_{M_n} \geq H_n$. Let $H'_n = h_{M_n}$. H'_n corresponds to the height of the floors graphs A_n, F_n, B_n, X_n and G_n.

Let $g(x) = 2x^K - x^{2K}$. Let z be the smallest integer such that $g(\gamma_z) \geq \gamma_z$, with $\gamma_z = 1 - (1/2^z)$ (we show that such an integer z always exists in Lemma 3 in Section 8), and let $\mu_0 = \gamma_z$. Let $\lambda_0 = \min(1 - \mu_0, 1 - (\mu_0/g(\mu_0))^{1/(4+2K)})$. Let α be the smallest integer such that $\alpha \geq 1$ and $(1-\mu_0)^\alpha \leq 1 - \mu$. Let β be the smallest integer such that $\beta \geq 1$ and $\lambda^\beta \leq \lambda_0$. The parameters α and β impact the redundancy of nodes and channels in the definition of B_n.

Let (G_n^0, S_n^0, R_n^0) be the solution to the BDF problem described in Section 6.

Graph G_n. To define $G_n = (V_n, E_n)$, we first define 4 intermediary graphs A_n, F_n, B_n and X_n.

$\forall n \geq 2$, we define the floor graph A_n as follows. Consider graph G_n^0 and its definition in Section 6. The last step of construction of G_n^0 consists in merging the nodes of X_1 and X_H. Let G'_n be graph G_n^0 just before this last step. Then, G'_n can be seen as a floor graph of height $H = H_n$, where the H floors are $(X_1, \dots, X_H)$. We define graph A_n as a combination of G'_n and of $2^{H_n - 1}$ sequences of $H'_n - H_n$ nodes, such as described in Figure 8. Thus, A_n is a floor graph of height H'_n.

$\forall i \geq 0$, we first define a floor graph Q_i by induction. Let Q_0 be a floor graph of height 1 containing 1 node (see Figure 9). Then, $\forall i \geq 0$, Q_{i+1} is constructed with $2K$ instances of Q_i and 2 additional nodes, as described in Figure 9. We now define F_n as follows: $\forall n \geq 2$, $F_n = Q_{M_n}$.

$\forall n \geq 2$, graph B_n is obtained by applying three successive transformations to F_n. **Transformation 1** consist in connecting α instances of F_n by merging the nodes of their first floors and then of their last floors. **Transformation 2 and 3** are the same as for the WRBD graph.

$\forall n \geq 2$, A_n is a floor graph of height H'_n, and F_n is also a floor graph of height H'_n (by definition of H'_n). Thus, B_n is also a floor graph of height H'_n. As A_n and B_n are floor graphs, let $A_n = (V_1, \dots, V_{H'_n}, E)$ and $B_n = (V'_1, \dots, V'_{H'_n}, E')$. Then, $\forall n \geq 2$, we define the floor graph $X_n = (V''_1, \dots, V''_{H'_n}, E'')$ by the same mechanism as for the WRBD graph.

The first floor V''_1 of X_n contains $m = 2^{H_n - 1}$ nodes, and so does its last floor $V''_{H'_n}$. Let $V''_1 = \{u_1, \dots, u_m\}$ and $V''_{H'_n} = \{v_1, \dots, v_m\}$ (the order of numbering is unimportant here). We finally obtain graph G_n as follows: $\forall i \in \{1, \dots, m\}$, we merge the nodes u_i and v_i.

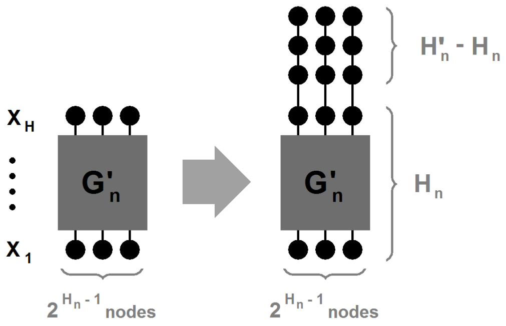

Figure 8: Construction of graph A_n with graph G'_n and 2^{H_n-1} sequences of $h_{M_n} - H_n$ nodes.

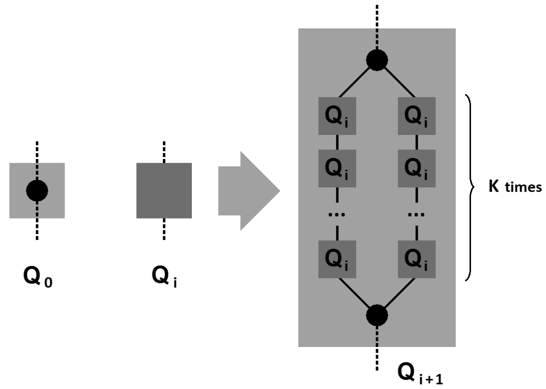

Figure 9: Construction (by induction) of graph Q_i.

Set of nodes S_n. For the set of nodes S_n, let S'_n be a set of any n nodes of the first floor of X_n (such a set exists, as $|V''_1| \geq 2^{H_n-1} \geq n$). We define S_n as the corresponding set of nodes in G_n.

Routing map $R_n^{\mathcal{V},\mathcal{E}}$. Let p and q be two nodes of S_n. As G_n is obtained by merging the nodes of V''_1 and $V''_{H'_n}$ in X_n, let p'' (resp. q'') be the corresponding node if V''_1 (resp. $V''_{H'_n}$). According to the definition of X_n, let p_F (resp. q_F) be the node of A_n such that there exists a node v (resp v') such that $p'' = \pi(p_F, v)$ (resp. $q'' = \pi(q_F, v')$). According to the definition of A_n, let p_G be the node of G'_n corresponding to p_F, and let q_G be the node of the last floor of G'_n which is connected to q_F by a path of $H'_n - H_n$ nodes (according to Figure 8). Finally, let p' (resp. q') be the node corresponding to p_G (resp. q_G) in G_n^0.

Let $r(p', q')$ be the path connecting p' and q' in G_n^0, such as defined in Section 6 (as shown in the proof of Theorem 6, $r(p', q')$ actually connects p' and q'). Let $r_G(p_G, q_G)$ be the corresponding path in G'_n. Let $r_F(p_F, q_F) = (u_1, \dots, u_{H'_n})$ be an extension of $r_G(p_G, q_G)$ connecting p_F and q_F in A_n with $H'_n - H_n$ additional nodes (see Figure 8). $\forall i \in \{1, \dots, H'_n\}$, let W_i be the set of nodes w of X_n such that there exists a node v such that $w = \pi(u_i, v)$. Let $W = \bigcup_{i \in \{1,\dots,H'_n\}} W_i$. Let W' be the corresponding set of nodes in G_n. We define $W(p, q)$ as the subgraph containing the nodes of W (and the channels connecting them) in G_n.

Now, let $\mathcal{V}$ (resp. $\mathcal{E}$) be an arbitrary set of crashed nodes (resp. edges) of G_n. If there exists a path of correct nodes and channels connecting p and q in $W(p, q)$, let $\psi(\mathcal{V}, \mathcal{E}, p, q)$ be this path. Otherwise, let $\psi(\mathcal{V}, \mathcal{E}, p, q)$ be any path connecting p and q in $W(p, q)$. We define the routing map R_n by $R_n^{\mathcal{V},\mathcal{E}}(p, q) = \{(\psi(\mathcal{V}, \mathcal{E}, p, q), 1)\}$ for any two nodes p and q of S_n.

8 Correctness Proofs

WRBD

We prove that graph G_n described in Section 4 solves the WRBD problem. For this purpose, we prove the three properties of the WRBD problem: **Reliability**, **Bounded degree** and **Linear number of nodes**.

In Lemma 1, we show that, for a sufficiently small failure rate ($\lambda \leq 0.01$), the first floor and the last floor of R_n are connected with a constant probability (independently of n). To do so, we call P_i the probability that the first and last floor of Q_i are connected, then express P_{i+1} as a function of P_i (according to the inductive definition of Q_i). Then, we show that if $P_i \geq 0.8$, we also have $P_{i+1} \geq 0.8$. Thus, the first and last floor of Q_i (and thus, R_n) are connected with probability at least 0.8.

In Lemma 2, we show that the first floor of G_n contains at least n nodes. Then, we consider that S_n is a subset of the first floor of G_n to prove the following property.

In Theorem 1, we prove the **Reliability** property. We first consider the case $\lambda \leq 0.01$ and $\mu \leq 0.5$ (in this case, $Y_n = G_n$). According to the definition of X_n and Y_n, any two nodes of S_n are connected to the last floor of Y_n by two graphs R_n. Thus, the result, according to Lemma 2. We then consider that λ and μ can have any value, and show that the 3 final transformations of Section 4 enable to simulate the previous situation where $\lambda \leq 0.01$ and $\mu \leq 0.5$.

In Theorem 2, we prove the **Bounded degree** property. As G_n is intentionally defined as a combination of graphs with a bounded degree, the property follows.

In Theorem 3, we prove the **Linear number of nodes** property. We use the fact that the number of nodes of T_n is divided by 2 every floor (starting from the first floor), while the number of nodes of R_n at most doubles every 2 floors. Therefore, the number of nodes of X_n (which is the combination of T_n and R_n) is at least divided by 2 every 2 floors. Then, as $1 + 1/2 + 1/4 + 1/8 + \cdots \leq 2$, the number of nodes of X_n is linear in n, and so is the number of nodes of G_n.

Lemma 1. *Assume each node and channel crashes with probability at most λ (the probabilities being independent). If $\lambda \leq 0.01$, then $\forall n \geq 2$, the nodes of the first and last floor of R_n are both correct* and *connected with probability at least* 0.8*.*

Proof. $\forall k \geq 0$, let p_i (resp. q_i) be the only node of the first (resp. last) floor of graph Q_i. Let P_i be the probability that p_i and q_i are both correct and connected.

Let $i \geq 0$. Figure 4 shows how Q_{i+1} is constructed with 2 instances of Q_i and 10 additional components. Then, observe that p_{i+1} and q_{i+1} are connected in the following particular situation: the 10 additional components are all correct, and at least one of the two instances of Q_i has the nodes of its first and last floor connected (which happens with probability P_i). Therefore, $P_{i+1} \geq p(P_i)$, with $p(x) = (1-\lambda)^{10}(1-(1-x)^2)$.

The function $p(x)$ is increasing for $x \in [0.8, 1]$, $p(0.8) \in [0.8, 1]$ and $p(1) \in [0.8, 1]$. Therefore, $\forall x \in [0.8, 1]$, $p(x) \in [0.8, 1]$.

As Q_0 contains 3 components, $P_0 \geq (1-\lambda)^3$. Thus, as $\lambda \leq 0.01$, $P_0 \geq 0.8$ and $P_0 \in [0.8, 1]$. Therefore, by induction, $\forall k \geq 0$, $P_i \in [0.8, 1]$: p_i and q_i are both correct and connected with probability 0.8. Thus, as $R_n = Q_{K_n}$, the result follows. □

Lemma 2. $\forall n \geq 2$*, the first floor of G_n contains at least n nodes.*

Proof. Let $n \geq 2$. The first floor of T_n contains $2^{h_n - 1} \geq n$ nodes. Then, by definition of X_n, the first floor of X_n contains at least n nodes, and so does the first floor of Y_n. Thus, as the 3 final transformations of Section 4 can only increase the number of nodes of each floor, the first floor of G_n contains at least n nodes. □

Theorem 1. *Assume each node and channel crashes with probability at most λ (the probabilities being independent). Then, there exists a set S_n of n nodes of G_n such that any two correct nodes of S_n are connected with probability at least μ.*

Proof. According to Lemma 2, $\forall n \geq 2$, let S_n be a set containing n nodes of the first floor of G_n.

Let $n \geq 2$, and let p and q be any two nodes of S_n. First, assume that $\lambda \leq 0.01$ and $\mu \leq 0.5$. Then, $\alpha = 1$ and $\beta = 1$, and according to the 3 final transformations of Section 4, G_n is identical to Y_n. Let a be a node of the first floor of X_n, and let b be the only node of the last floor of X_n. Let P_0 be the probability that a and b are connected in X_n. Then, according to the definition of X_n, P_0 is at least the probability that the nodes of the first and last floor of R_n are connected. Thus, according to Lemma 1, $P_0 \geq 0.8$.

As Y_n is formed by 2 instances of X_n, the probability that p and q are connected is $P_1 \geq P_0^2(1-\lambda) \geq 0.5$ (as $P_0 \geq 0.8$ and $\lambda \leq 0.01$). Thus, as $\mu \leq 0.5$ here, $P_1 \geq \mu$, and p and q are reliably connected.

Now, we only assume that $\lambda \leq 0.01$ (μ can have any value in $]0, 1[$). Then, $\beta = 1$, and transformations 2 and 3 do not change anything. After transformation 1, the probability that p and q are connected is $P_2 = 1 - (1 - P_1)^\alpha \geq 1 - 0.5^\alpha$ (as $P_1 \geq 0.5$). According to the definition of α, $0.5^\alpha \leq 1 - \mu$. Thus, $P_2 \geq \mu$, and p and q are reliably connected.

Finally, we consider that λ and μ can have any value in $]0, 1[$. Let us show that, after transformations 2 and 3, we reach a situation which is equivalent to the previous case where $\lambda \leq 0.01$.

Let Z_n be the graph after transformation 1. After transformation 2, each node u is replaced by a set of β nodes $M(u)$. We consider that $M(u)$ is *crashed* if all its nodes are crashed, which happens with probability $\lambda^\beta \leq 0.01$. Thus, if $M(u)$ is *correct*, at least one node of $M(u)$ is correct.

For two correct sets of nodes $M(u)$ and $M(v)$, let u' (resp. v') be a correct node of $M(u)$ (resp. $M(v)$). Then, after transformation 3, the channel $\{u', v'\}$ is replaced by a set of β channels. We consider that this group of channels is *crashed* if all its channels are crashed, which happens with probability $\lambda^\beta \leq 0.01$. Otherwise, u' and v' are connected by at least one channel.

Let u and v be the two nodes of Z_n such that $p \in M(u)$ and $q \in M(v)$. Then, the probability that p and q are connected in G_n is at least the probability that u and v are connected in Z_n when $\lambda \leq 0.01$. Thus, the situation is equivalent to the previous case, and p and q are connected with probability μ. □

Theorem 2. *There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.*

Proof. Let $n \geq 2$. The maximal degree of T_n and R_n is 3. Thus, the maximal degree of X_n is at most 9, and the maximal degree of Y_n is at most 18. After the 3 final transformations of Section 4, the maximal degree of G_n is at most $\Delta = 18\alpha\beta^2$. Thus, the result, as α and β are independent from n. □

Theorem 3. *There exists a constant C such that, $\forall n \geq 2$, the number of nodes of G_n is at most Cn.*

Proof. Let $n \geq 2$. As T_n, R_n and X_n are 3 floor graphs of height H_n, let $T_n = (V_1, \ldots, V_{H_n}, E)$, $R_n = (V'_1, \ldots, V'_{H_n}, E)$ and $X_n = (V^*_1, \ldots, V^*_{H_n}, E)$.

According to the definition of T_n, $\forall i \in \{1, \ldots, h_n\}$, $|V_i| \leq 2^{h_n - i}$, and $\forall i \in \{h_n + 1, \ldots, H_n\}$, $|V_i| = 1$. According to the definition of R_n, starting from the first floor, $|V'_i|$ at most doubles every 2 floors. This is also true if we start from the last floor. Thus, $\forall i \in \{1, \ldots, H_n\}$, $|V'_i| \leq 2^{i/2}$ and $|V'_i| \leq 2^{(H_n - i)/2}$.

Thus, $\forall i \in \{1, \ldots, h_n\}$, $|V^*_i| = |V_i||V'_i| \leq 2^{h_n - i}2^{i/2} = 2^{h_n - (i/2)}$, and $\forall i \in \{h_n + 1, \ldots, H_n\}$, $|V^*_i| = |V_i||V'_i| \leq 2^{(H_n - i)/2}$. Thus, X_n contains at most $D = A + B$ nodes, with $A = \Sigma_{i=1}^{i=H_n} 2^{h_n - (i/2)}$ and $B = \Sigma_{i=1}^{i=H_n} 2^{(H_n/2) - (i/2)}$.

$A \leq 2\Sigma_{i=0}^{i=H_n} 2^{h_n - i} \leq 2(a + a/2 + a/4 + \ldots) \leq 4a$, with $a = 2^{h_n}$. Thus, $A \leq 2^{h_n + 2}$. $B \leq 2\Sigma_{i=0}^{i=H_n} 2^{(H_n/2) - i} \leq 2(b + b/2 + b/4 + \ldots) \leq 4b$, with $b = 2^{H_n/2}$. Thus, as $h_n \geq H_n/2$, $b \leq 2^{h_n}$ and $B \leq 2^{h_n + 2}$. Therefore, $D \leq 2^{h_n + 3}$.

As h_n is the smallest integer such that $2^{h_n - 1} \geq n$, we have $h_n \leq 2 + \log n$ and $D \leq 2^{5 + \log n} = 2^5 n$. Therefore, the graph Y_n contains at most $2^6 n$ nodes, and the graph G_n contains at most Cn nodes, with $C = 2^6\alpha\beta$. Thus, the result. □

RBD

We prove that graph G_n described in Section 5 solves the RBD problem. For this purpose, we prove the two properties of the WRBD problem: **Reliability** and **Bounded degree**.

In Theorem 4, we prove the **Reliability** property. Let p and q be two nodes of G_n. In the "complete graph" case, the reliability property is ensured by the number of channels between p and q. Otherwise, it is ensured by the fact that p and q belong to the set S_m of at least one instance of W_m.

In Theorem 5, we prove the **Bounded degree** property. We first notice that the "complete graph" case only occurs when $n \leq N_2$. Thus, in this case, the degree is bounded. Otherwise, we show that the number of subsets of $\lfloor m/2 \rfloor$ nodes is bounded (which is a consequence of the linearity property of the WRBD problem). Thus, the number of instances of W_m is bounded, and so is the degree of G_n.

Theorem 4. *Assume each node and channel crashes with probability at most λ (the probabilities being independent). Then, any two correct nodes of G_n are connected with probability at least μ.*

Proof. Let p and q be two correct nodes of G_n. If there exists no $m \geq 2$ such that $N_m \leq n$, then p and q are connected by $k = \lceil \log(1-\mu)/\log(1-\lambda) \rceil$ channels. Thus, the probability that p and q are connected is $P = 1 - (1-\lambda)^k$. As $k \geq \log(1-\mu)/\log(1-\lambda)$, $\log(1-\mu) \geq k \log(1-\lambda)$ (as $\log(1-\lambda) < 0$). Then, $1 - \mu \geq (1-\lambda)^k$, and $P = 1 - (1-\lambda)^k \geq \mu$. Thus, the result.

Otherwise, let i and j be such that $p \in A_i$ and $q \in A_j$. Then, p and q belong to the set of nodes $S(i,j)$ of the graph $W(i,j)$. Thus, according to the reliability property of the WRBD problem, p and q are connected with probability at least μ. □

Theorem 5. *There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.*

Proof. As the graph W_m solves the WRBD problem, there exists two constants Δ_0 and C_0 such that, $\forall m \geq 2$, the maximal degree of W_m is at most Δ_0 ("Bounded degree" property) and $N_m \leq C_0 m$ ("Linear number of nodes" property).

Let $n \geq 2$. If there exists no $m \geq 2$ such that $N_m \leq n$, then $\forall m \geq 2$, $N_m > n$. In particular, $n < N_2$. Thus, each node of S is connected to at most $\Delta_1 = N_2 \lceil \log(1-\mu)/\log(1-\lambda) \rceil$ neighbors. Thus, the result, if we take $\Delta = \Delta_1$.

Otherwise, let $m \geq 2$ be the largest integer such that $N_m \leq n$. Thus, $N_{m+1} > n$, and as $N_{m+1} \leq C_0(m+1)$, $n < C_0(m+1)$. As M is the smallest integer such that $M\lfloor m/2 \rfloor \geq n$, we have $(M-1)\lfloor m/2 \rfloor < n$. Thus, $M < 1 + n/\lfloor m/2 \rfloor < 1 + C_0(m+1)/\lfloor m/2 \rfloor$. Then, as $(m+1)/\lfloor m/2 \rfloor \leq 4$, $M \leq 1 + 4C_0$.

$\forall (i,j) \in \{1, \ldots, M\}^2$, each node of V is merged with at most 2 nodes of $W(i,j)$. As the maximal degree of $W(i,j)$ is at most Δ_0, the maximal degree of G_n is at most $2\Delta_0 M^2 \leq 2\Delta_0(1+4C_0)^2$. Thus, the result, if we take $\Delta = 2\Delta_0(1+4C_0)^2$. □

BDF

We prove that the tuple (G_n, S_n, R_n) described in Section 6 solves the BDF problem. For this purpose, we first prove that R_n is actually a routing map of S_n. Then, we prove the two properties of the BDF problem: **Bounded degree** and **Bounded flow**.

In Theorem 6, we show that R_n is a routing map of S_n. For this purpose, we show that the definition of $r(p,q)$ (with the binary decomposition of the position of q in X_H) is so that the path actually reaches q. To do so, we show by induction that the k first "bits" always reflect the position of the node crossed by $r(p,q)$ in X_k.

In Theorem 7, we prove the **Bounded degree** property: the degree of G_n is at most 4 by construction.

In Theorem 8, we prove the **Bounded flow** property: we show that according to the definition of the routing map, each node of X_k is crossed by $2^{k-1} \times 2^{H-k} = 2^{H-1}$ paths (which is a constant). Hence, the maximal flow is bounded.

Theorem 6. *R_n is a routing map of S_n.*

Proof. Let p and q be any two nodes of S_n. $R_n(p,q)$ contains one weighted path $r(p,q)$ of weight 1. Let us show that $r(p,q) = (v_1, \dots, v_H)$ is indeed a path connecting p and q.

First, note that $\forall k \in \{1, \dots, H-1\}$, the node v_k corresponds to a node of type a or b in the definition of G_n. Then, $x(v_k)$ (resp. $y(v_k)$) corresponds to the node x (resp y). Thus, v_k and v_{k+1} are indeed neighbors, and $r(p,q)$ is actually a path.

Now, let us show that $r(p,q)$ connects p and q. By definition, $p = v_1$. In the following, we show that $q = v_H$.

Let j be such that $q = u_H(1,j)$. According to the definition of R_n, let $(b_1, \dots, b_{H-1})$ be the binary decomposition of $j-1$. $\forall k \in \{1, \dots, H\}$, let i_k and j_k be such that $v_k = u_k(i_k, j_k)$. Let us show the following property $\mathcal{P}_k$ by induction, $\forall k \in \{1, \dots, H\}$: $j_k = 1$ (if $k = 1$) or $j_k = 1 + \Sigma_{x=1}^{x=k-1} b_x 2^{k-x-1}$ (if $k \geq 2$).

$\mathcal{P}_0$ is true, as $j_1 = 1$. Now, suppose that $\mathcal{P}_k$ is true for $k \in \{1, \dots, H-1\}$, and let us show that $\mathcal{P}_{k+1}$ is true. Then, two possible cases:

- **Case 1:** $b_k = 0$. Then, $v_{k+1} = x(b_k)$. Thus, as $v_k = u_k(i_k, j_k)$ and $v_{k+1} = u_{k+1}(i_{k+1}, j_{k+1})$, we have $j_{k+1} = 2j_k - 1 = 2(\Sigma_{x=1}^{x=k-1} b_x 2^{k-x-1} + 1) - 1 = 1 + 2(\Sigma_{x=1}^{x=k-1} b_x 2^{k-x-1}) = 1 + \Sigma_{x=1}^{x=k-1} b_x 2^{k-x} = 1 + \Sigma_{x=1}^{x=k} b_x 2^{k-x}$ (as $b_k = 0$). Thus, $\mathcal{P}_{k+1}$ is true.
- **Case 2:** $b_k = 1$. Then, $v_{k+1} = y(b_k)$. Thus, as $v_k = u_k(i_k, j_k)$ and $v_{k+1} = u_{k+1}(i_{k+1}, j_{k+1})$, we have $j_{k+1} = 2j_k = 2(\Sigma_{x=1}^{x=k-1} b_x 2^{k-x-1} + 1) = 2 + 2(\Sigma_{x=1}^{x=k-1} b_x 2^{k-x-1}) = 2 + \Sigma_{x=1}^{x=k-1} b_x 2^{k-x} = 1 + \Sigma_{x=1}^{x=k} b_x 2^{k-x}$ (as $b_k = 1$). Thus, $\mathcal{P}_{k+1}$ is true.

Hence, by induction, $\mathcal{P}_H$ is true, and $j_H = 1 + \Sigma_{x=1}^{x=H-1} b_x 2^{H-x-1} = j$. Thus, $v_H = u_H(1, j_H) = u_H(1, j) = q$: the path $r(p,q)$ actually connects p and q. Thus, the result. □

Theorem 7. *There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.*

Proof. Consider graph G_n for an arbitrary $n \geq 2$.

Let v be a node of $X_1 = X_H$. Let i be such that $v = u_1(i,1) = u_H(1,i)$. Let i_0 be the smallest integer such that $2i_0 \geq i$. Then, v has two neighbors in X_2 (resp. X_{H-1}): $u_2(i_0, 1)$ and $u_2(i_0, 2)$ (resp. $u_{H-1}(1, i_0)$ and $u_{H-1}(2, i_0)$). Thus, v has 4 neighbors.

If $H \geq 3$, let $k \in \{2, \dots, H-1\}$ and let v be a node of X_k. Let i and j be such that $v = u_k(i,j)$. Let i_0 (resp. j_0) be the smallest integer such that $2i_0 \geq i$ (resp. $2j_0 \geq j$). Then, v has two neighbors in X_{k+1}(resp. X_{k-1}): $u_{k+1}(i_0, 2j)$ and $u_{k+1}(i_0, 2j-1)$ (resp. $u_{k-1}(2i, j_0)$ and $u_{k-1}(2j-1, j_0)$). Thus, v has 4 neighbors.

Therefore, the maximal degree of the graph can be bounded by a constant $\Delta = 4$. □

Theorem 8. *There exists a constant f_0 such that, $\forall n \geq 2$, the maximal flow of (G_n, S_n, R_n) is at most f_0.*

Proof. Let $k \in \{1, \dots, H\}$ and let $v \in X_k$.

According to the definition of the routing map, a path $r(p,q)$ crossing v is described by a unique binary sequence $(b_1, \dots, b_{H-1})$.

- The node p is described by the binary sequence $(b_1, \dots, b_{k-1})$. Thus, there are 2^{k-1} possible nodes p.
- The node q is described by the binary sequence $(b_k, \dots, b_{H-1})$. Thus, there are 2^{H-k} possible nodes q.

Therefore, at most $2^{k-1} \times 2^{H-k} = 2^{H-1}$ paths $r(p,q)$ cross v, and the flow of v is at most 2^{H-1}.

Note that, if the flow of every node is at most f, then the flow of every channel is at most f. Indeed, if a channel $\{u, v\}$ had a flow greater than f, then u and v would also have a flow greater than f, which would be a contradiction.

Thus, as the flow of every node is at most 2^{H-1}, the maximal flow is at most 2^{H-1}. Thus, the result, if we take $f_0 = 2^{H-1}$. □

RBDF

We prove that the tuple $(G_n, S_n, R_n^{\mathcal{V},\mathcal{E}})$ described in Section 7 solves the RBDF problem. For this purpose, we prove the three properties of the RBDF problem: **Bounded degree**, **Bounded flow** and **Reliability**.

In Lemma 3, we prove a small property assumed in the description of the RBDF solution in Section 7.

In Theorem 9, we show the **Bounded degree** property, which follows from the construction of the graph.

In Theorem 10, we show the **Bounded flow** property: the worst case in terms of maximal flow (after merging several nodes) corresponds to our solution to the BDF problem.

In Lemma 4, we show that if the failure rate is at most λ_0, then the communication probability in Q_i (and thus, in F_n) is at least μ_0. This is due to the fractal definition of Q_i, which enables this property to propagate through each recursive step. In Lemma 5, we show that the three transformations between F_n and B_n adapt the result of Lemma 4 to any parameters λ and μ. Then, in Theorem 11, we show the **Reliability** property, which follows from the properties of B_n.

Lemma 3. *Let $\gamma_i = 1 - (1/2^i)$. There exists an integer $i \geq 1$ such that $g(\gamma_i) \geq \gamma_i$.*

Proof. Let $w(x) = g(1-x) + x - 1 = 2(1-x)^K - (1-x)^{2K} + x - 1$. The derivative of w is w', with $w'(x) = -2K(1-x)^K + 2K(1-x)^{2K} + 1$. The functions w and w' are continuous, $w(0) = 0$ and $w'(0) = 1$. Thus, there exists $e > 0$ such that, $\forall x \in]0, e]$, $w(x) > 0$.

Let i be an integer such that $i \geq 1$ and $1/2^i \leq e$. Then, $w(1/2^i) = g(1 - (1/2^i)) + (1/2^i) - 1 = g(\gamma_i) - \gamma_i > 0$, and $g(\gamma_i) \geq \gamma_i$. □

Theorem 9. *There exists a constant Δ such that, $\forall n \geq 2$, the maximal degree of G_n is at most Δ.*

Proof. According to Theorem 7, the degree of G_n^0 is bounded by a constant Δ^*. Therefore, the degree of G_n' is at most Δ^*, and the degree of B_n is at most $\Delta^* + 1$.

$\forall i \geq 0$, the degree of Q_i is at most 3. Thus, after transformations 1, 2 and 3, the degree of F_n is at most $3\alpha\beta^2$. Hence, the degree of X_n is at most $\Delta' = 3\alpha\beta^2(\Delta^* + 1)$. Therefore, the degree of G_n is at most $\Delta = 2\Delta'$. □

Theorem 10. *There exists a constant f_0 such that, $\forall n \geq 2$, $\forall \mathcal{V} \subseteq V_n$ and $\forall \mathcal{E} \subseteq E_n$, the maximal flow of $(G_n, S_n, R_n^{\mathcal{V},\mathcal{E}})$ is at most f_0.*

Proof. Let R_n^0 be the routing map described in Section 6. Let R_n^A (resp. R_n^B) be the routing map corresponding to R_n (resp. R_n^0) in graph X_n (resp. G_n').

For each node p of A_n, let us merge all the nodes $\pi(u, v)$ of X_n such that $u = p$. The graph thus obtained is equivalent to A_n. Then, we merge the $H_n' - H_n$ last floors of A_n. The graph thus obtained is equivalent to G_n'.

Let p and q be two distinct nodes of S_n. Let p_A and q_A be the corresponding nodes in X_n (where p_A belongs to the first floor and q_A to the last floor). Let p_B and q_B be the corresponding nodes in G'_n, according to the previous merging scheme. Finally, let p' and q' be the corresponding nodes in G^0_n.

Observe that $\forall \mathcal{V} \subseteq V_n$ and $\forall \mathcal{E} \subseteq E_n$, the path $R_A^{\mathcal{V},\mathcal{E}}(p_A, q_A)$ corresponds to the path $R_B(p_B, q_B)$ after the previous merging scheme.

According to Theorem 8, the maximal flow of (G^0_n, S^0_n, R^0_n) is bounded by a constant f^*_0, and so is the flow in G'_n. As G'_n and its routing map R^B_n can be obtained by merging nodes of X_n, the maximal flow in X_n is at most f^*_0. Thus, as G_n is obtained by merging the first and the last floor of X_n, the maximal flow of $(G_n, S_n, R_n^{\mathcal{V},\mathcal{E}})$ is at most $f_0 = 2f^*_0$. □

Lemma 4. *$\forall i \geq 0$, let p_i (resp. q_i) be the node of the first (resp. last) floor of Q_i. Suppose that $\lambda \leq \lambda_0$. If each node and channel crashes with probability at most λ, p_i and q_i are connected with probability at least μ_0.*

Proof. Let P_i be the probability that p_i and q_i are correct and connected. $\forall i \geq 0$, let us express P_{i+1} as a function of P_i.

$\forall i \geq 0$, we say that Q_i is *correct* if p_i and q_i are connected. Q_{i+1} is built with 2 nodes and $2 \times K$ instances of Q_i. Then, note that Q_{i+1} is correct in the following situation: (1) the $4 + 2K$ components of Q_{i+1} that are not instances of Q_i are all correct *and* (2) at least one column of K instances of Q_i *only contains* correct instances of Q_i. Event (1) happens with probability $(1 - \lambda)^{4+2K}$. The opposite of event (2) happens with probability $(1 - P_i^K)^2$ (i.e., the probability that both columns do *not* only contain correct instances of Q_i). Thus, $P_{i+1} \geq h(P_i)$, with $h(x) = (1 - \lambda)^{4+2K}(1 - (1 - x^K)^2) = (1 - \lambda)^{4+2K}(2x^K - x^{2K}) = (1 - \lambda)^{4+2K} g(x)$.

$h'(x) = 2K(1 - \lambda)^{4+2K}(x^{K-1} - x^{2K-1}) \geq 0 \ \forall x \in [0, 1]$. Thus, the function f is strictly increasing on the interval $[0, 1]$. Besides, as $\lambda \leq \lambda_0$, $h(\mu_0) = (1 - \lambda)^{4+2K} g(\mu_0) \geq (1 - \lambda_0)^{4+2K} g(\mu_0) \geq (\mu_0/g(\mu_0))g(\mu_0)$. Thus, $h(\mu_0) \geq \mu_0$.

Let us prove the following property by induction, $\forall i \geq 0$: $P_i \geq \mu_0$.

- $P_0 = 1 - \lambda \geq 1 - \lambda_0 \geq \mu_0$.
- Suppose that $P_i \geq \mu_0$ for $i \geq 0$. As $P_i \geq \mu_0$ and h is strictly increasing on $[0, 1]$, $h(P_i) \geq h(\mu_0) \geq \mu_0$. Thus, $P_{i+1} \geq h(P_i) \geq \mu_0$.

Therefore, by induction, $\forall i \geq 0$, $P_i \geq \mu_0$. Thus, the result. □

Lemma 5. *$\forall n \geq 2$, let p_n (resp. q_n) be any node of the first (resp. last) floor of B_n. If p_n and q_n are correct, and each other node and channel crashes with probability at most λ, then p_n and q_n are connected with probability at least μ.*

Proof. As $F_n = Q_{M_n}$, the result of Lemma 4 is also true for F_n. First, assume that $\lambda \leq \lambda_0$ and $\mu \leq \mu_0$. Thus, according to Lemma 4, p_n and q_n are connected with probability at least $\mu \leq \mu_0$.

Now, we only assume that $\lambda \leq \lambda_0$ (μ can have any value in $]0, 1[$). Then, $\beta = 1$, and transformations 2 and 3 do not change anything. After transformation 1, p_n and q_n are connected with probability at least $1 - (1 - \mu_0)^\alpha \leq 1 - (1 - \mu) = \mu$.

Finally, we consider that λ and μ can have any value in $]0, 1[$. Let us show that, after transformations 2 and 3, we reach a situation which is equivalent to the previous case where $\lambda \leq \lambda_0$.

Let Z_n be the graph after transformation 1. After transformation 2, each node u is replaced by a set of β nodes $M(u)$. We consider that $M(u)$ is *crashed* if all its nodes are crashed, which happens with probability $\lambda^\beta \leq \lambda_0$. Otherwise, at least one node of $M(u)$ is correct.

For two correct sets of nodes $M(u)$ and $M(v)$, let u' (resp. v') be a correct node of $M(u)$ (resp. $M(v)$). Then, after transformation 3, the channel $\{u', v'\}$ is replaced by a set of β channels. We consider that this group of channels is *crashed* if all its channels are crashed, which happens with probability $\lambda^\beta \leq \lambda_0$. Otherwise, u' and v' are connected by at least one channel.

Let u and v be the two nodes of Z_n such that $p_n \in M(u)$ and $q_n \in M(v)$. Then, the probability that p_n and q_n are connected in B_n is at least the probability that u and v are connected in Z_n when $\lambda \leq \lambda_0$. Thus, the result, as the situation is equivalent to the previous case. □

Theorem 11. *Assume each node and channel crashes with probability at most λ (the probabilities being independent). Let $\mathcal{V}$ (resp. $\mathcal{E}$) be the set of crashed nodes (resp. channels). Then, any two correct nodes of S_n are reliably connected in $R_n^{\mathcal{V},\mathcal{E}}$ with probability at least μ.*

Proof. Let p and q be two distinct nodes of S_n.

First, note that graph $W(p,q)$ is equivalent to B_n (by definition), where p (resp. q) corresponds to the node of the first (resp. last) floor of B_n.

Suppose that p and q are correct, and that any other node or channel is crashed with probability at most λ. Let $\mathcal{V}$ (resp. $\mathcal{E}$) be the set of crashed nodes (resp. channels). Then, according to Lemma 5, with probability μ, p and q are connected in $W(p,q)$. Therefore, according to the definition of R_n, with probability μ, the path $\psi(\mathcal{V},\mathcal{E},p,q)$ only contains correct nodes and channels. Thus, the result, as $R_n^{\mathcal{V},\mathcal{E}}(p,q) = \{(\psi(\mathcal{V},\mathcal{E},p,q),1)\}$. □

9 Complexity

Lower bound on the BDF problem

In Theorem 12, we show that solving the BDF problem requires at least $\Omega(n \log n)$ nodes.

In broad outline, we assume a solution (G_n, S_n, R_n) of the BDF problem. We first show that there are at least $\Omega(n^2)$ tuples of nodes (p,q) of S_n such that p and q are at distance at least $\Omega(\log n)$ from each other, due to the bounded degree. Therefore, as the flow of messages sent by each node of S_n is divided between the $n-1$ other nodes, the sum of the flows of all nodes is $\Omega(n \log n)$. Thus, for the maximal flow to be bounded, at least $\Omega(n \log n)$ nodes are required.

Theorem 12. *A graph solving the BDF problem, if it exists, contains at least $\Omega(n \log n)$ nodes.*

Proof. Let (G_n, S_n, R_n) be a tuple solving the BDF problem, with $G_n = (V_n, E_n)$. Let $N \geq n$ be the number of nodes of G_n. According to the definition of the BDF problem, let $\Delta \geq 2$ be a constant bounding the maximal degree, and let $f_0 \geq 1$ be a constant bounding the maximal flow. In the following, we assume that $n \geq 4\Delta$.

Let p be a node of G_n. There are at most Δ nodes at distance 1 from p, at most Δ^2 nodes at distance 2 from p, $\ldots$, at most Δ^k nodes at distance k from p. Thus, as $\Delta \geq 2$, at most $1 + \Delta + \Delta^2 + \cdots + \Delta^k \leq 2\Delta^k$ nodes are either p or at distance k or less from p.

Let D be the largest integer such that $2\Delta^D \leq n/2$. As $n \geq 4\Delta$, we have $D \geq 1$. Thus, at least $\lfloor n/2 \rfloor$ nodes are at distance $D+1$ or more from p. As this is true for any node $p \in S$, there exists a set Z of tuples $(p,q) \in S \times S$ such that the distance between p and q is at least $D+1$, with $|Z| = n\lfloor n/2 \rfloor \geq n^2/4$. Let $Y = \bigcup_{(p,q)\in Z} R(p,q)$, and let us denote Y by $\{(P_1,\alpha_1),(P_2,\alpha_2),\ldots,(P_m,\alpha_m)\}$. For any two nodes p and q, the sum of the weights of the weighted paths of $R(p,q)$ is 1. Then, $\Sigma_{i=1}^{i=m}\alpha_i = |Z| \geq n^2/4$.

Let $W = \Sigma_{p \in V_n} f(p)$ be the sum of the flows of the nodes of G_n. Then, the maximal flow is at least W divided by the number of nodes: $f_{max} \geq W/N$. Besides, as Y contains m weighted paths of at least D nodes each, $W \geq D\Sigma_{i=1}^{i=m} F\alpha_i/(n-1) = (\Sigma_{i=1}^{i=m}\alpha_i)DF/(n-1) \geq n^2DF/(4(n-1))$. Thus, $f_{max} \geq n^2DF/(4N(n-1)) \geq nDF/(4N)$. As $f_{max} \leq f_0$, $nDF/(4N) \leq f_0$ and $N \geq nDF/(4f_0)$.

As D is the largest integer such that $2\Delta^D \leq n/2$, $2\Delta^{D+1} \geq n/2$ and $D \geq \log n/\log(4\Delta) - 1$. As $n \mapsto \log n$ is a strictly increasing function, let $n_0 \geq 4\Delta$ be the smallest integer such that $\log n/\log(4\Delta) - 1 \geq \log n/2\log(4\Delta)$. Then, if $n \geq n_0$, $D \geq \log n/2\log(4\Delta)$.

Therefore, if $n \geq n_0$, we have $N \geq \beta n \log n$, with $\beta = F/(8f_0 \log(4\Delta))$. Thus, N is $\Omega(n \log n)$. $\square$

Complexity of our BDF solution

In Theorem 13, we show that graph G_n described in Section 6 contains $O(n \log n)$ nodes: G_n is composed of H sets $(X_1, \ldots, X_H)$ of $O(n)$ nodes each, with $H = O(\log n)$.

Theorem 13. *Graph G_n, described in Section 6, contains $O(n \log n)$ nodes.*

Proof. As H is the smallest integer such that $2^{H-1} \geq n$, we have $2^{H-2} < n$. Thus, $H - 2 < \log n$, and $H < \log n + 2$. Besides, as $2^{H-2} < n$, we have $2^{H-1} < 2n$. Thus, as the graph is entirely covered by H sets $(X_1, \ldots, X_H)$ of 2^{H-1} nodes each, the total number of nodes is $H2^{H-1} \leq (\log n + 2)2n$. As $n \geq 2$, we have $3\log n \geq \log n + 2$. Thus, the total number of nodes is at most $6n \log n = O(n \log n)$. $\square$

Complexity of our RBDF solution

We show that graph G_n described in Section 7 contains $O(n \log^{1+\epsilon} n)$ nodes. In Lemma 6, we show that the floors of F_n contain $O(log^{\epsilon} n)$ nodes. In Lemma 7, we show that the height of G_n is $O(\log n)$. Then, as shown in Theorem 14, G_n contains $O(\log^{\epsilon} n) \times O(\log n) \times O(n) = O(n \log^{1+\epsilon} n)$ nodes.

Lemma 6. *There exists a constant C_1 such that the floors of graph F_n contain at most $C_1 \log^{\epsilon} n$ nodes each.*

Proof. We have $h_0 = 1$, and $\forall i \geq 0$, $h_{i+1} \geq Kh_i$. Hence, by induction, $\forall i \geq 0$, $h_i \geq K^i$. As M_n is the smallest integer such that $h_{M_n} \geq H_n$, $h_{M_n - 1} \leq H_n$. Thus, $K^{M_n - 1} \leq H_n$, $(M_n - 1)\log K \leq \log H_n$ and $M_n \leq 1 + (\log H_n / \log K)$.

According to Figure 9, $\forall i \geq 0$, the floors of Q_i contain at most 2^i nodes each. Thus, the floors of $F_n = Q_{M_n}$ contain at most $\rho = 2^{M_n}$ nodes each. Therefore, $\rho \leq 2^{1+(\log H_n/\log K)} = 2(2^{\log H_n})^{1/\log K} = 2H_n^{1/\log K}$. As K is such that $K \geq 2^{1/\epsilon}$, $\log K \geq 1/\epsilon$ and $(1/\log K) \leq \epsilon$. Thus, $\rho \leq 2H_n^{\epsilon}$.

As H_n is the smallest integer such that $2^{H_n - 1} \geq n$, $2^{H_n - 2} \leq n$, $H_n - 2 \geq \log n$ and $H_n \leq \log n + 2 \leq 3\log n$ (as $n \geq 2$). Thus, $\rho \leq 2(3\log n)^{\epsilon} \leq C_1 \log^{\epsilon} n$, with $C_1 = 2(3^{\epsilon})$. $\square$

Lemma 7. *There exists a constant C_2 such that $H'_n \leq C_2 \log n$.*

Proof. M_n is the smallest integer such that $h_{M_n} \geq H_n$. Thus, $h_{M_n - 1} \leq H_n$. As $\forall i \geq 0$, $h_{i+1} = 2 + Kh_i$, we have $H'_n = h_{M_n} \leq 2 + KH_n$. As H_n is the smallest integer such that $2^{H_n - 1} \geq n$, $2^{H_n - 2} \leq n$. Thus, $H_n - 2 \leq \log n$ and $H_n \leq 2 + \log n$.

Therefore, $H'_n \leq 2 + K(2 + \log n) = 2 + 2K + K\log n \leq C_2 \log n$, with $C_2 = 2 + 3K$ (as $n \geq 2$). $\square$

Theorem 14. *Graph G_n, described in 7, contains $O(n \log^{1+\epsilon} n)$ nodes.*

Proof. By definition of A_n, each floor of A_n contains $2^{H_n - 1}$ nodes. As H_n is the smallest integer such that $2^{H_n - 1} \geq n$, $2^{H_n - 2} \leq n$ and $2^{H_n - 1} \leq 2n$. Thus, each floor of A_n contains at most $2n$ nodes.

According to Lemma 6, each floor of F_n contains at most $C_1 \log^{\epsilon} n$ nodes. Thus, after transformations 1, 2 and 3, each floor of B_n contains at most $C_1 \alpha \beta^2 \log^{\epsilon} n$ nodes.

Therefore, each floor of X_n contains at most $2C_1\alpha\beta^2 n \log^\epsilon n$ nodes. As X_n has H'_n floors, according to Lemma 7, X_n contains at most $2C_1C_2\alpha\beta^2 n \log^{1+\epsilon} n = O(n \log^{1+\epsilon} n)$ nodes, and so does G_n. $\square$

10 Diameter

We show here that the 4 graphs presented in this paper have an optimal (logarithmic) diameter. The diameter of a network (i.e., the maximal distance between two nodes) corresponds to the maximal number of hops that a message has to cross, which directly impacts the communication delays.

In Theorem 15, we first show that any graph solving our problems has a diameter at least $\Omega(\log n)$ (due to the bounded degree). Then, in Theorem 16, we show that our 4 graphs have a $O(\log n)$ diameter.

Theorem 15. *If a graph G_n solves one of the 4 problems (WRBD, RBD, BDF or RBDF), then the diameter of G_n is at least $\Omega(\log n)$.*

Proof. If a graph G_n solves one of the 4 problems, then there exists a constant Δ such that $\forall n \geq 2$, the maximal degree of G_n is at most Δ ("Bounded degree" property).

Let p be any node of G_n. Then, at most Δ nodes are at distance 1 from p, at most Δ^2 nodes are at distance 2 from p, ..., at most Δ^k nodes are at distance k from p. Thus, if D is the diameter of G_n, then G_n contains at most $1 + \Delta + \Delta^2 + \cdots + \Delta^D \leq 2\Delta^D$ nodes (as $\Delta \geq 2$). Thus, $n \leq 2\Delta^D$ and $D \geq (\log n - \log 2)/\log \Delta = \Omega(\log n)$. $\square$

Theorem 16. *The 4 graphs described in the paper have a $O(\log n)$ diameter.*

Proof. **WRBD.** As G_n is a floor graph of height H_n, the diameter of G_n is at most $D = 2H_n$. As K_n is the smallest integer such that $2 + 4K_n \geq h_n$, $2 + 4(K_n - 1) < h_n$ and $H_n = 2 + 4K_n < h_n + 4$. As h_n is the smallest integer such that $2^{h_n - 1} \geq n$, $2^{h_n - 2} < n$ and $h_n < \log n + 2$. Thus, $D = 2H_n < 2(\log n + 6) = O(\log n)$.

RBD. As G_n is the combination of several graphs W_m of diameter $O(\log m)$ with $m \leq n$, the diameter of G_n is also $O(\log n)$.

BDF. As G_n is a floor graph of height at most H, the diameter of G_n is at most $D = 2H$. As H is the smallest integer such that $2^{H-1} \geq n$, $H \geq \log n + 1$ and $D = O(\log n)$.

RBDF. As G_n is a floor graph of height at most H'_n, the diameter of G_n is at most $D = 2H'_n$. As M_n is the smallest integer such that $h_{M_n} \geq H_n$, $h_{M_n - 1} \leq H_n$. As $h_{M_n} = 2 + Kh_{M_n - 1}$, $h_{M_n} \leq 2 + KH_n$. As H_n is the smallest integer such that $2^{H_n - 1} \geq n$, $H_n \geq \log n + 1$ and $H'_n = h_{M_n} \leq 2 + K(\log n + 1) = K \log n + 2 + K = O(\log n)$. Thus, $D = O(\log n)$. $\square$

11 Byzantine Failures

We focused in the paper on *crash* failures, where the failed components (nodes and channels) simply stop functioning. With Byzantine failures [22], the graphs we presented so far reveal insufficient. Indeed, even one single Byzantine failure, if not contained, can lead to potentially broadcast false messages to every other node and deceive the whole network.

A classical strategy to contain Byzantine failures is to perform majority votes [9, 26]: a message is accepted and forwarded only if it is received through a majority of channels. Thus, assuming there is a majority of correct components, the effect of Byzantine components can be masked by the vote. In the following, we explain how our solutions can be adapted to tolerate Byzantine failures by increasing the level of redundancy and adding several layers of majority votes. Essentially, the main ideas behind our solutions remain the same.

Whilst the solutions we presented (assuming only crashes) work for any failure rate $\lambda \in]0,1[$, in order to tolerate Byzantine failures, we assume however $\lambda \in]0,0.5[$. This is necessary because of the classical argument of *indistinguishability* (e.g., [9] and [26]). Indeed, if a solution existed for $\lambda = 0.5$, then with the same probability, correct and Byzantine components could be exchanged. As the correct components can ensure to deliver a given message with probability μ, the Byzantine components also could, which is a contradiction for any $\mu > 0.5$. If $\lambda > 0.5$, then the Byzantine components can simulate the case $\lambda = 0.5$ by acting as correct components with probability $\lambda - 0.5$.

Now, assuming that $\lambda \in]0,0.5[$, our solutions (WRBD, RBD, RBDF) can be modified as follows to handle Byzantine failures. (We exclude the BDF case, that does not consider any node or channel failure.) All these modifications only affect the number of nodes by a linear factor, and thus do not change the complexity of the graphs.

WRDB. First, we modify the construction scheme of the fractal graph described in Figure 4 to contain three instances of Q_i instead of two, with a majority vote at the junction.

In the proof of Lemma 1, we consider the probability that at least one instance of Q_i (out of two) is correct. Here, we consider the probability that at least two instances of Q_i (out of three) are correct. Therefore, the formula $p(x)$ bounding the reliability becomes $(1-\lambda)^{12}(x^3 + 3x^2(1-x))$. If we assume that $\lambda \leq 0.001$, $p(x)$ keeps the same property on the interval $[0.8, 1]$, and the result of Lemma 1 remains correct.

After this modification, the number of nodes of X_m is now multiplied by $4/3$ every two floors (instead of $1/2$). But it is still at least divided by 2 at regular intervals (every 6 floors). Thus, the argument we used in Theorem 3 (i.e., $1 + 1/2 + 1/4 + 1/8 + \cdots \leq 2$) remains applicable.

Second, we adapt the three last transformations of Section 4 to Byzantine failures, by increasing the level of redundancy and adding majority votes:

1. In Transformation 1 (Network replication), the number of replicas α must be large enough so that the probability to have a strict majority of correct instances of Y_m is at least μ. Then, a majority vote must be performed by each node of the first floor.

2. In Transformation 2 (Node replication), the number of replicas β must be large enough so that the probability to have a strict majority of correct nodes is at least 0.999 (according to the hypothesis $\lambda \leq 0.001$ above). Then, a majority vote must be performed by each node over each set of β neighbors.

3. Similarly, in Transformation 3 (Channel replication), the same number β of replicas must be used. Then, a majority vote must be performed by each node over each set of β channels.

RDB. The reliability property is ensured by the WRBD graphs used in our construction of the RBD graph. Thus, no further modification is required.

RBDF. Similarly to the WRBD case, Q_{i+1} (see Figure 9) must contain 3 columns of K instances of Q_i instead of 2. Then, Q_{i+1} is "correct" if at least 2 columns of K instances of Q_i over 2 are "correct". Thus, we redefine the function g by $g(x) = x^{3K} + 3x^{2K}(1 - x^K)$ (Lemma 3 remains valid, as we still have $w(0) = 0$ and $w'(0) = 1$). In the definitions of λ_0 (see Section 7) and $h(x)$ (see Lemma 4), we replace "$4 + 2K$" by "$5 + 3K$". Transformations 1, 2 and 3 are modified in the same way as in the WRBD case.

12 Concluding Remarks

The properties underlying the problems we consider may look similar to those of expander graphs [16, 13, 21]. However, these graphs are not suited for proving the reliability property: as a network is not a continuum, the combinatorial complexity of the problem explodes with the size of the network, making for example any proof by induction impracticable. On a different front, a lot of work in distributed computing has been devoting to tolerating a specific number of failures. A constant failure rate raises different problems when the size of the network is unbounded, e.g., even a very small failure rate can entirely change asymptotic properties.

Our approach suggests several research directions. For instance, instead of considering a "fluid" flow of messages, we could model more accurately the granularity of messages with a probabilistic model. One could also consider the complexity of "physically wiring" the network, and try to bound it.

References

[1] http://bluebrain.epfl.ch/page-58110-en.html.

[2] http://home.cern/about/computing.

[3] Sebastian Anthony. Microsoft now has one million servers – less than Google, but more than Amazon, says Ballmer. http://tinyurl.com/microsoft-now-has-one-million, 2013.

[4] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. A view of cloud computing. *Communications of the ACM*, 53:50–58, 2010.

[5] Roberto Baldoni, Silvia Bonomi, Leonardo Querzoni, and Sara Tucci Piergiovanni. Investigating the existence and the regularity of logarithmic harary graphs. *Theor. Comput. Sci.*, 410(21-23):2110–2121, 2009.

[6] Luiz André Barroso, Jimmy Clidaras, and Urs Hölzle. The datacenter as a computer: An introduction to the design of warehouse-scale machines, second edition. *Synthesis Lectures on Computer Architecture*, 2013.

[7] Paolo Costa, Austin Donnelly, Greg O'Shea, and Antony Rowstron. CamCubeOS: a key-based network stack for 3D torus cluster topologies. *22nd international symposium on High-performance parallel and distributed computing (HPDC 2013).*

[8] Csernai, Ciucu, Florin, Braun, and Gulyas. Towards 48-fold cabling complexity reduction in large flattened butterfly networks. *IEEE Conference on Computer Communications (INFOCOM 2015).*

[9] D. Dolev. The Byzantine generals strike again. *Journal of Algorithms*, 3(1):14–30, 1982.

[10] Jose Duato, Sudhakar Yalamanchili, and Ni Lionel. Interconnection networks: An engineering approach. *Morgan Kaufmann Publishers*, 2002.

[11] Roy Friedman, Shiri Manor, and Katherine Guo. Scalable stability detection using logical hypercube. *IEEE Trans. Parallel Distrib. Syst.*, 13(9):972–984, 2002.

[12] Amlan Ganguly, Kevin Chang, Sujay Deb, Partha Pratim Pande, Benjamin Belzer, and Christof Teuscher. Scalable hybrid wireless network-on-chip architectures for multicore systems. *IEEE Trans. Comput.*, 60(10):1485–1502, October 2011.

[13] David Gillman. A chernoff bound for random walks on expander graphs. *SIAM Journal on Computing*, 27(4):1203–1220, 1998.

[14] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta. VL2: a scalable and flexible data center network. *ACM SIGCOMM 2009 conference on Data communication*, pages 51–62.

[15] C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu. DCell: a scalable and fault-tolerant network structure for data centers. *ACM SIGCOMM 2008 conference on Data communication*, pages 75–86.

[16] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.

[17] C. Huang, M. Li, and A. Srinivasan. A scalable path protection mechanism for guaranteed network reliability under multiple failures. *IEEE Transactions on Reliability*, 56(2):254–267, June 2007.

[18] A. Iwata, Ching-Chuan Chiang, Guangyu Pei, M. Gerla, and Tsu-Wei Chen. Scalable routing strategies for ad hoc wireless networks. *IEEE J.Sel. A. Commun.*, 17(8):1369–1379, September 2006.

[19] Pankaj Jalote. *Fault tolerance in distributed systems*. Prentice-Hall, Inc., 1994.

[20] J. Kim, IL Evanston, W.J. Dally, S. Scott, and D. Abts. Technology-driven, highly-scalable dragonfly topology. *35th International Symposium on Computer Architecture (ISCA 2008).*

[21] Jon Kleinberg and Ronitt Rubinfeld. Short paths in expander graphs. In *Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on*, pages 86–95. IEEE, 1996.

[22] Leslie Lamport, Robert E. Shostak, and Marshall C. Pease. The Byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(3):382–401, 1982.

[23] Jinyang Li, Charles Blake, Douglas S.J. De Couto, Hu Imm Lee, and Robert Morris. Capacity of ad hoc wireless networks. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking*, MobiCom '01, pages 61–69, New York, NY, USA, 2001. ACM.

[24] Y. C. Liang, Y. Zeng, E. C. Y. Peh, and A. T. Hoang. Sensing-throughput tradeoff for cognitive radio networks. *IEEE Transactions on Wireless Communications*, 7(4):1326–1337, April 2008.

[25] Dmitri Loguinov, Anuj Kumar, Vivek Rai, and Sai Ganesh. Graph-theoretic analysis of structured peer-to-peer systems: routing distances and fault resilience. In *Proceedings of the ACM SIGCOMM 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication, August 25-29, 2003, Karlsruhe, Germany*, pages 395–406, 2003.

[26] Mikhail Nesterenko and Sébastien Tixeuil. Discovering network topology in the presence of Byzantine faults. *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, 20(12):1777–1789, December 2009.

[27] L. M. Ni and P. K. McKinley. A survey of wormhole routing techniques in direct networks. *Computer Journal, IEEE Computer Society Press*, 26:63–76, 1993.

[28] R. Niranjan, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat. Portland: a scalable fault-tolerant layer 2 data center network fabric. *ACM SIGCOMM 2009 conference on Data communication*, pages 39–50.

[29] Richard D Schlichting and Fred B Schneider. Fail-stop processors: an approach to designing fault-tolerant computing systems. *ACM Transactions on Computer Systems (TOCS)*, 1(3):222–238, 1983.

[30] J. Snyder. Microsoft: Datacenter Growth Defies Moore's Law. http://www.pcworld.com/article/130921/article.html, 2007.

[31] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, Dec 1992.

[32] Victor C. Zandy and Barton P. Miller. Reliable network connections. In *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*, MobiCom '02, pages 95–106, New York, NY, USA, 2002. ACM.