When Should You Wait Before Updating?

Toward a Robustness Refinement

Swan Dubois
Sorbonne Université, CNRS, LIP6, DELYS, France

Laurent Feuilloley
Univ Lyon, CNRS, INSA Lyon, UCBL, LIRIS, UMR5205, Villeurbanne, France

Franck Petit
Sorbonne Université, CNRS, LIP6, DELYS, France

Mikaël Rabie
Université Paris Cité, CNRS, IRIF, Paris, France

Abstract

Consider a dynamic network and a given distributed problem. At any point in time, there might exist several solutions that are equally good with respect to the problem specification, but that are different from an algorithmic perspective, because some could be easier to update than others when the network changes. In other words, one would prefer to have a solution that is more robust to topological changes in the network; and in this direction the best scenario would be that the solution remains correct despite the dynamic of the network.

In [6], the authors introduced a very strong robustness criterion: they required that for any removal of edges that maintain the network connected, the solution remains valid. They focus on the maximal independent set problem, and their approach consists in characterizing the graphs in which there exists a robust solution (the existential problem), or even stronger, where any solution is robust (the universal problem). As the robustness criteria is very demanding, few graphs have a robust solution, and even fewer are such that all of their solutions are robust. In this paper, we ask the following question: *Can we have robustness for a larger class of networks, if we bound the number k of edge removals allowed*?

To answer this question, we consider three classic problems: maximal independent set, minimal dominating set and maximal matching. For the universal problem, the answers for the three cases are surprisingly different. For minimal dominating set, the class does not depend on the number of edges removed. For maximal matching, removing only one edge defines a robust class related to perfect matchings, but for all other bounds k, the class is the same as for an arbitrary number of edge removals. Finally, for maximal independent set, there is a strict hierarchy of classes: the class for the bound k is strictly larger than the class for bound $k+1$.

For the robustness notion of [6], no characterization of the class for the existential problem is known, only a polynomial-time recognition algorithm. We show that the situation is even worse for bounded k: even for $k = 1$, it is NP-hard to decide whether a graph has a robust maximal independent set.

2012 ACM Subject Classification Theory of Computation → Design and analysis of algorithms; Mathematics of computing → Discrete mathematics

Keywords and phrases Robustness, dynamic network, temporal graphs, edge removal, connectivity, footprint, packing/covering problems, maximal independent set, maximal matching, minimum dominating set, perfect matching, NP-hardness

Digital Object Identifier 10.4230/LIPIcs.SAND.2023.7

Funding ANR SKYDATA (ANR-22-CE25-0008-02) ANR GrR (ANR-18-CE40-0032)

2nd Symposium on Algorithmic Foundations of Dynamic Networks (SAND 2023).
Editors: David Doty and Paul Spirakis; Article No. 7; pp. 7:1–7:15
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

In the field of computer networks, the phrase "*dynamic networks*" refers to many different realities, ranging from static wired networks in which links can be unstable, up to wireless ad hoc networks in which entities directly communicate with each other by radio. In the latter case, entities may join, leave, or even move inside the network at any time in completely unpredictable ways. A common feature of all these networks is that communication links keep changing over time. Because of this aspect, algorithmic engineering is far more difficult than in fixed static networks. Indeed, solutions must be able to adapt to incessant topological changes. This becomes particularly challenging when it comes to maintaining a single leader [3] or a (supposed to be) "static" covering data structure, for instance, a spanning tree, a node coloring, a Maximal Independant Set (MIS), a Minimal Dominating Set (MDS), or a Maximal Matching (MM). Most of the time, to overcome such topological changes, algorithms compute and recompute their solution to try to be as close as possible to a correct solution in all circumstances.

Of course, when the network dynamics is high, meaning that topological changes are extremely frequent, it sometimes becomes impossible to obtain an acceptable solution. In practice, the correctness requirements of the algorithm are most often relaxed in order to approach the desired behavior, while amortizing the recomputation cost. Actually, this sometimes leads to reconsider the very nature of the problems, for example: looking for a "moving leader", a leader or a spanning tree per connected component, a temporal dominated set, an evolving MIS, a best-effort broadcast, *etc.*— we refer to [3, 4] for more examples.

In this paper, we address the problem of network dynamics under an approach similar to the one introduced in [1, 6]: *To what extent of network dynamics can a computation be performed without relaxing its specification?* Before going any further into our motivation, let us review related work on which our study relies.

Numerous models for dynamic networks have been proposed during the last decades–refer to [3] for a comprehensive list of models– some of them aiming at unifying previous modeling approaches, mainly [4, 11]. As is often the case, in this work, the network is modeled as a graph, where the set of vertices (also called nodes) is fixed, while the communication links are represented by a set of edges appearing and disappearing unexpectedly over the time. Without extra assumptions, this modeling includes all possibilities that can occur over the time, for example, the network topology may include no edges at some instant, or it may also happen that some edge present at some time disappears definitively after that. According to different assumptions on the appearance and disappearance (frequency, synchrony, duration, etc.), the dynamics of temporal networks can be classified in many classes [4].

One of these classes, Class $\mathcal{TC}^{\mathcal{R}}$, is particularly important. In this class, a temporal path between any two vertices appears infinitely often. This class is arguably the most natural and versatile generalization of the notion of connectivity from static networks to dynamic networks: every vertex is able to send (not necessarily directly) a message to any other vertex at any time.

For a dynamic network of the class $\mathcal{TC}^{\mathcal{R}}$ on a vertex set V, one can partition $V \times V$ into three sets: the edges that are present infinitely often over the time –called *recurrent* edges–, the edges that are present only a finite number of times –called *eventually absent* edges–, and the edges that are never present. The union of the first two sets defines a graph called the *footprint* of the network [4], while its restriction to the edges that are infinitely often present is called the *eventual footprint* [2]. In [2], the authors prove that Class $\mathcal{TC}^{\mathcal{R}}$ is actually the set of dynamic networks whose *eventual footprint* is connected.

In conclusion, from a distributed computing point of view, it is more than reasonable to consider only dynamic networks such that some of their edges are recurrent and their union does form a *connected* spanning subgraph of their footprint.

Unfortunately, it is impossible for a node to distinguish between a recurrent and an eventually absent edge. Therefore, the best the nodes can do is to compute a solution relative to the footprint, hoping that this solution still makes sense in the eventual footprint, whatever it is. In [6], the authors introduce the concept of *robustness* to capture this intuition, defined as follows:

▶ **Definition 1** (Robustness). *A property P is robust over a graph G if and only if P is satisfied in every connected spanning subgraph of G (including G itself).*

Another way to phrase this definition is to say that *a property P is robust if it is still satisfied when we remove any number of edges, as long as the graph stays connected.*

In [6], the authors focus on the problem of maximal independent set (MIS). That is, they study the cases where a set of vertices can keep being an MIS even if we remove edges. They structure their results around two questions:

Universal question: For which networks are *all the solutions* robust against any edge removals that do not disconnect the graph?

Existential question: For which networks does *there exist a solution* that is robust against any edge removals that do not disconnect the graph?

The authors in [6] establish a characterization of the networks that answer the first questions for the MIS problem. Still for the same problem, they provide a polynomial-time algorithm to decide whether a network answers the second question.

Note that the study of robustness was also very recently addressed for the case of metric properties in [5]. In that paper, the authors show that deciding whether the distance between two given vertices is robust can be done in linear time. However, they also show that deciding whether the diameter is robust or not is coNP-complete.

1.1 Our approach

Our goal is to go beyond [6], and to get both a more fine-grained and a broader understanding of the notion of robustness.

Let us start with the fine-grain dimension. In [6], a solution had to be robust against any number of edge removals as long as the graph remains connected. In this paper, we want to understand what are the structures that are robust against k edge removals while keeping the connectivity constraint, for any specific k, adding granularity to the notion. We call this concept k-robustness (see formal definition below) and we focus on the universal and the existential question of [6] for this fined-grain version of the robustness.

Now for the broader dimension, let us discuss the problems studied. In [6], the problem studied is MIS, which is a good choice in the sense that it leads to an interesting landscape. Indeed, robustness being a very demanding property, one has to find problems to which it can apply without leading to trivial answers. In this direction, one wants to look at local problems, because a modification will only have consequences in some neighborhood and not on the whole graph, which leaves the hope that it actually does not affect the correctness at all. Among the classic local problems, as studied in the LOCAL model (see [9] for the original definition and [8] for a recent book), there are mainly coloring problems and packing/covering problems. The coloring problems (with a fixed number of colors) are not meaningful in our context: an edge removal can only help. But the packing/covering problems are all interesting, thus we widen the scope to cover three classic problems in this paper: maximal independent set (MIS) as before, but also maximal matching (MM) and minimal dominating set (MDS).

To help the reader grasp some intuition on our approach, let us illustrate the 1-robustness for the maximal matching, *i.e.* a set of edges that do not share vertices and that is maximal in the sense that no edge can be added. To be 1-robust, a matching must still be maximal after the removal of *one arbitrary* edge that does not disconnect the graph. Let us go over various configurations illustrated in Figure 1 (the matched edges are bold ones).

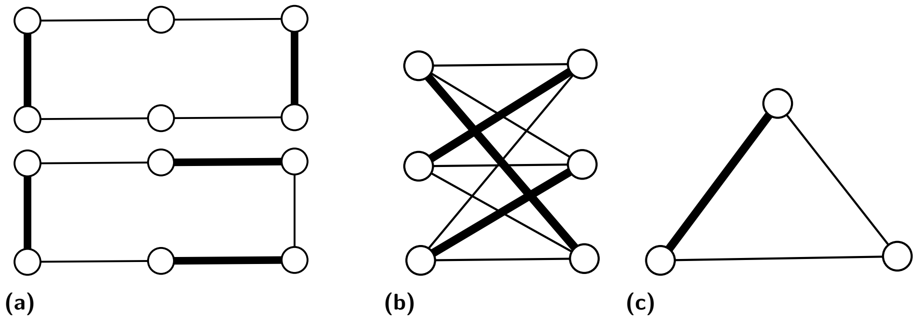


Figure 1 Three examples of MMs in various graphs.

For the two graphs in Figure 1a, that are cycles of 6 vertices, we can observe that two instances of maximal matching can have different behaviors. Indeed, in the top one, if we remove one matched edge, we are left with a matching that is not maximal in the new graph: the two edges adjacent to the removed one could be added. By contrast, in the bottom graph, any edge removal leaves a graph that is still a maximal matching. Now, in the graph of Figure 1b, a complete balanced bipartite graph, all the maximal matchings are identical up to isomorphism. After one arbitrary edge removal, we are left with a graph where no new edge can be matched. Therefore in this graph, any matching is robust to one edge removal. Note that this is not true for any number of edge removals, illustrating the fact that k-robustness and robustness are not equivalent. Finally, in Figure 1c, all the maximal matchings consists of only one edge, and they are not robust to an edge removal. Indeed, after the matched edge is removed, one can choose any of the two remaining ones.

To summarize, Figure 1 illustrates the effect of 1-robustness in three different cases: one where *some* matchings are 1-robust, one where *all* matchings are 1-robust, and one where *no* matching is 1-robust.

1.2 Our results

Our first contribution is to introduce the fine-grained version of robustness in Section 2. After that, every technical section of this paper is devoted to provide some answer to the fine-grained version of one of the two questions highlighted above (existential *vs.* universal) for one of the problems we study. Our focus is actually in understanding how do the different settings compare, in terms of both problems and number of removable edges.

Let us start with the universal question. Here, we prove that the three problems have three different behaviors.

For minimal dominating set, the class of the graphs for which any solution is k-robust is exactly the same for every k (a class that already appeared in [6] under the name of *sputnik graphs*) as proved in Section 3.

For maximal matching, the case of $k = 1$, which we used previously as an example, is special and draws an interesting connection with perfect matchings, but then the class is identical for every $k \geq 2$. These results are presented in Section 4.

Finally, for maximal independent set, we show in Section 5 that there is a strict hierarchy: the class for k edge removals is strictly smaller than the one for $k - 1$. For this case, we do not pinpoint the exact characterization, but give some additional structural results on the classes.

The existential question is much more challenging. Section 6 presents some preliminary results on the study of this question. For maximal independent set, we show that for any k, deciding whether a graph has a maximal independent set that is robust to k edge removals is NP-hard. This is the first NP-hardness result for this type of question.

2 Model, definitions, and basic properties

In the paper, except when stated otherwise, the graph is named G, the vertex set V and the edge set E.

2.1 Robustness and graph problems

The key notion of this paper is the one of k-robustness.

▶ **Definition 2.** *Given a graph problem and a graph, a solution is* k-robust *if after the removal of* at most *k edges, either the graph is disconnected, or the solution is still valid.*

Note that k-robustness is about removing at most k edges, not exactly k edges.

We will abuse notation and write ∞-robust when mentioning the notion of robustness from [6], with an unbounded number of removals. Hence k is a parameter in $\mathbb{N} \cup \infty$.

▶ **Notation 1.** *We define $\mathcal{U}_P^k$ and $\mathcal{E}_P^k$ the following way:*

- *Let $\mathcal{U}_P^k$ be the class of graphs such that any solution to the problem P is k-robust.*
- *Let $\mathcal{E}_P^k$ be the class of graphs such that there exists a solution to the problem P that is k-robust*

Note that to easily incorporate the parameter k, we decided to not follow the exact same notations as in [6].

Graph problems.

We consider three graph problems:

1. Minimal dominating set (MDS): Select a minimal set of vertices such that every vertex of the graph is either in the set or has a neighbor in the set.
2. Maximal matching (MM): Select a maximal set of edges such that no two selected edges share endpoint.
3. Maximal independent set (MIS): Select a maximal set of vertices such that no two selected vertices share an edge.

A *perfect matching* is a matching where every vertex is matched. We will also use the notion of *k-dominating set*, which is a set of selected vertices such that every vertex is either selected or is adjacent to two selected vertices. Note that k-dominating set sometimes refer to another notion related to the distance to the selected vertices, but this is not our definition.

The case of robust maximal matching.

For maximal matching, the definition of robustness may vary. The definition we take is the following. A maximal matching M of a graph G is k-robust if after removing any set of at most k edges such that the graph G is still connected, what remains of M is a maximal matching of what remains of G.

2.2 Graph notions

We list a few graph theory definitions that we will need.

▶ **Definition 3.** *The* neighborhood *of a node v, denoted $N(v)$, is the set of nodes that are adjacent to v. The* closed neighborhood *of a node v, denoted $N[v]$, is the neighborhood of v, plus v itself.*

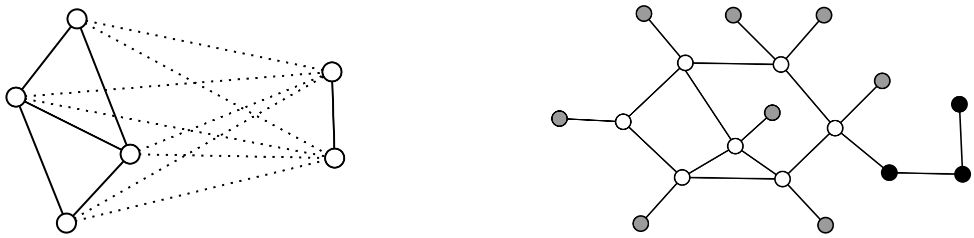
▶ **Definition 4.** *A graph is t-(edge)-connected if, after the removal of any set of $(t-1)$ edges, the graph is still connected. A t-(edge)-connected component is a subgraph that is t-(edge)-connected.*

In the following we are only interested in edge connectivity thus we will simply write *t-connectivity* to refer to *t-edge-connectivity*. In our proofs, we will use the following easy observation multiple times : in a 2-connected graph every vertex belongs to a cycle.

▶ **Definition 5.** *In a connected graph, a* bridge *is an edge whose removal disconnects the graph.*

▶ **Definition 6.** *Given two graphs G and H, the* join *of these two graphs, join(G, H), is the graph made by taking the union of G and H, and adding all the possible edges (u, v), with $u \in G$ and $v \in H$. See Figure 2a.*

▶ **Definition 7.** *A* sputnik graph *([6]) is a graph where every node that is part of a cycle has an* antenna*, that is a neighbor with degree 1. See Figure 2b.*



(a) The join of two graphs: the black edges are the original edges, the doted edges are the one added by the join operation.

(b) A sputnik graph. The white vertices are part a cycles, the grey vertices are their antennas, and the black vertices do not belong to any cycle, nor are antennas.

Figure 2 Illustration of the definitions of Subsection 2.2.

2.3 Basic properties

The following properties follow from the definitions.

▶ **Property 1.** *For any problem P, for any k, $\mathcal{U}_P^{k+1} \subseteq \mathcal{U}_P^k$ and $\mathcal{E}_P^{k+1} \subseteq \mathcal{E}_P^k$.*

In particular, $\mathcal{U}_P^\infty \subseteq \mathcal{U}_P^k \subseteq \mathcal{U}_P^1$ and $\mathcal{E}_P^\infty \subseteq \mathcal{E}_P^k \subseteq \mathcal{E}_P^1$, for all k.

▶ **Property 2.** *If a graph is* $(k+1)$-connected *then a solution is k-robust if and only if after the removal of any set of k edges the solution is still correct.*

3 Minimal dominating set

▶ **Theorem 8.** *For all k in $\mathbb{N} \cup \infty$, $\mathcal{U}_{MDS}^k$ is the set of sputnik graphs.*

Proof. We know from [6] that the theorem holds for $k = \infty$. Hence, thanks to Property 1, it is sufficient to prove that the theorem is true for $k = 1$. For the sake of contradiction, consider a graph G in $\mathcal{U}_{MDS}^1$ that is not a sputnik graph. Then there is a node u that belongs to a cycle, and that has no antenna. Let S be the closed neighborhood of u, $S = N[u]$. We say that a node of S, different from u, is an *inside node* if it is only connected to nodes in S. We now consider two cases depending on whether there is an inside node or not. See Figure 3.

1. Suppose there exists an inside node v. Note that v has at least one neighbor different from u because otherwise it would be an antenna. Let the set W be the closed neighborhood of v, except u. The set $D = V \setminus W$ is a dominating set of the graph, because all the nodes either belong to D or are neighbors of u (which belongs to D). Now, we transform D into a *minimal* dominating set greedily: we remove nodes from D in an arbitrary order, until no more nodes can be removed without making D non-dominating. We claim that this minimal dominating set is not 1-robust. Indeed, if we remove the edge (u, v), v is not covered any more (none of its current neighbors belongs to D), and the graph is still connected (because v has a neighbor different from u).
2. Suppose there is no inside vertex. Let a be a neighbor of u in the cycle. Let W be the set $S \setminus a$. Again we claim that $V \setminus W$ is a dominating set. Indeed, because there is no inside node, every node in S different from u is covered by node outside W, and u is covered by a, which belongs to $V \setminus W$. As before we can make this set an MDS by removing nodes greedily, and again we claim it is not 1-robust. Indeed, if we remove the edge (u, a), we do not disconnect the graph (because of the cycle containing u), and u is left uncovered.

◀

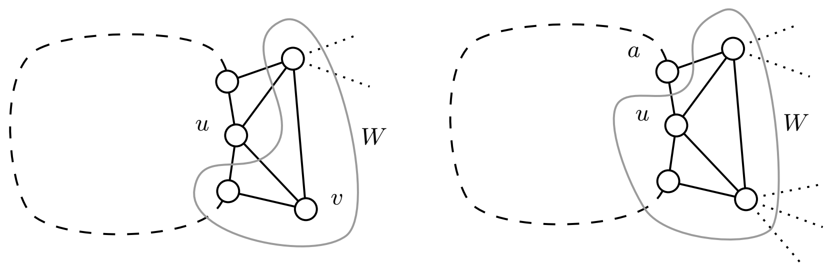


Figure 3 The two cases of the proof of Theorem 8: with an inside node, on the left, and without an inside node on the right. The cycle is represented by the dashed line, and the dotted lines represent outgoing edges of non-inside nodes.

4 Maximal matching

We now turn our attention to the problem of maximal matching, and get the following theorem.

▶ **Theorem 9.** *The class $\mathcal{U}^1_{MM}$ is composed of the set of trees, of balanced complete bipartite graphs, and of cliques with an even number of nodes. For any $k \geq 2$, the class $\mathcal{U}^k_{MM}$ is composed of the cycle on four nodes and of the set of trees.*

The core of this part is the study of the case where only one edge is removed. At the end of the section we consider the more general technically less interesting case of multiple edges removal.

4.1 One edge removal

In this subsection we characterize the class of graphs where every maximal matching is 1-robust.

▶ **Lemma 10.** *$\mathcal{U}^1_{MM}$ is composed of the set of trees, of balanced complete bipartite graphs, and of cliques with an even number of nodes.*

The rest of this subsection is devoted to the proof of Lemma 10.

A result about perfect matchings

The core of the proof is to show a connection to perfect matchings. Once this is done, we can use the following theorem from [10].

▶ **Theorem 11** ([10])**.** *The class of graphs such that any maximal matching is perfect is the union of the balanced complete bipartite graphs and of the cliques of even size.*

First inclusion

We start with the easy direction of the theorem, which is to prove that the graphs we mentioned are in $\mathcal{U}^1_{MM}$. In trees, any property is robust, since no edge can be removed without disconnecting the graph. For the two other types, we will use the following claim.

▷ Claim 12. Perfect matchings are 1-robust maximal matchings.

Consider a perfect matching in a graph, and remove an arbitrary edge (that does not disconnect the graph). If this edge was not in the matching, and then we still have a perfect matching, thus a maximal matching. If this edge was in the matching, then there are only two non-matched nodes in the graph (the ones that were adjacent to the edge), and all their neighbours are matched, thus the matching is still maximal. This proves the claim. ◁

In balanced complete bipartite graphs and cliques of even size, any maximal matching is perfect (Theorem 11), and since perfect matchings are 1-robust maximal matchings, we get the first direction of Lemma 10.

Second inclusion: three useful claims

We now tackle the other direction. The following lemma establishes a local condition that 1-robust matchings must satisfy. See Figure 4 for an illustration.

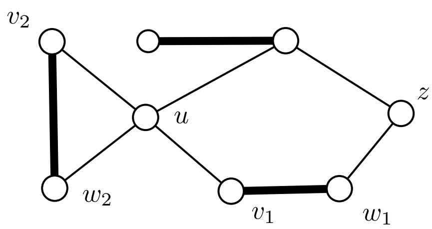


Figure 4 Illustration of Claim 13. Here we have a maximal matching, and in particular all the neighbors of u are matched, but it is not a 1-robust matching. Indeed, removing (v_1, w_1) gives the possibility of adding (u, v_1) and (w_1, z). Also, having a triangle with a matched edge and an unmatched node, like $(u, v2, w_2)$ is impossible (Claim 14), since removing (v_2, w_2) gives the possibility of adding either (u, v_2) or (u, w_2) to the matching, contradicting the maximality. Hence we need the bridge condition.

▷ Claim 13. In a 1-robust maximal matching M, if a node u is not matched, then all the nodes of $N(u)$ are matched, and their matched edges are bridges of the graph.

The fact that all the nodes in $N(u)$ are matched follows from M being a maximal matching. Now, suppose that there exists $(v, w) \in M$, such that $v \in N(u)$ and (v, w) is not a bridge. In other words, the removal of (v, w) does not disconnect the graph. After this removal, both u and v are unmatched, and since (u, v) is an edge of the graph, the matching in the new graph cannot be maximal. This contradicts the 1-robustness of M, and proves the lemma. ◁

The following claim follows directly from Claim 13.

▷ Claim 14. In a 1-robust maximal matching M, if there is an unmatched node u, two nodes $a, b \in N(u)$ with $(a, b) \in E$, then $(a, b) \notin M$.

We now study the shape of 1-robust maximal matchings in cycles.

▷ Claim 15. In every maximal matching of a graph in $\mathcal{U}^1_{MM}$, if a node belongs to a cycle, then it is matched.

Our proof of Claim 15 consists in proving that if a maximal matching does not satisfy the condition, then either it is not 1-robust, or we can use it to build another maximal matching that is not 1-robust. In both cases this means the graph was not in $\mathcal{U}^1_{MM}$.

Consider a node u in a cycle. Let a and b be its direct neighbors in the cycle, and let its other neighbors be $(c_i)_i$. There can be several configurations, with a adjacent to b or not, etc. The proof is generic to all these cases, but Figure 5 illustrates different cases. Consider a 1-robust maximal matching M where u is unmatched. Because of Claim 13, we know that there exists nodes a', b', and c'_i for all i, such that respectively (a, a'), (b, b') and (c_i, c'_i) (for all i) are bridges of the graph. Because of the bridge condition, these nodes a', b' and c'_i (for all i) are all different, and are different from a, b, u and the c_i's. Let us also denote d the neighbor of a in the cycle that is not u. Note that d can be a c_i or b, but no other named node. (See Figure 5 for an illustration.) Now we create a new matching M' from M in the following way. First remove all the edge of the matching that are not adjacent to one of the nodes above. Then, remove (a, a') and any edge matching d (if it exists). Note that this last edge matching d could be a (c_j, c'_j) or (b, b'). Add (a, d) to the matching (note that both nodes are unmatched before this operation). In this matching, all the neighbors of u are matched. We complete this matching into a maximal matching M'. The edge (a, d) is in M'

and u is unmatched, which is a contradiction with Claim 13, thus M' cannot be 1-robust, and this proves the claim. ◁

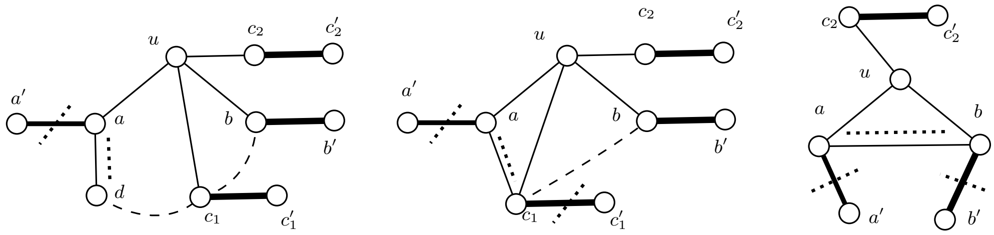


Figure 5 Illustration of the proof of Claim 16, in three cases: d is not b nor a c_i, d is one of the c_i, d is b. The dashed lines represent paths with at least one edge. The dotted lines represent the change we operate: the edges that are crossed out are removed from the matching, the edges that are have a dotted double are added to the matching.

Second inclusion: putting pieces together

▷ Claim 16. A graph in the class $\mathcal{U}^1_{MM}$ is either a tree or is 2-connected.

Consider a graph that is neither a tree nor a 2-connected graph. There necessarily exists a bridge (u, v) such that u belongs to a cycle. We distinguish two cases.

1. Node v is linked only to u, that is, v is a pendant node. Then we build a maximal matching M by first forcing u to be matched to a node that is not v, and then completing it greedily. Now, if we remove the edge that matches u, we do not disconnect the graph since u was part of a cycle, but neither u nor v is matched, thus the matching is not maximal ((u, v) could be added). Thus the matching M was not 1-robustness.
2. Node v is linked to another node w. Let consider $(v_i)_i$ the set of nodes such that $v_i \neq v$ and (u, v_i) is a bridge. By the previous point, we know that there exists some $w_i \neq u$ in $N(v_i)$. Moreover, those (w_i) must be distinct pairwise and from all the other named nodes, otherwise (u, v_i) would not have been a bridge. The node w and the nodes $(w_i)_i$ cannot be part of the 2-connected component of u, otherwise (u, v) and (u, v_i) would not be a bridge. We build a maximal matching M by first forcing (u, v) and (v_i, w_i) for all i, and then completing it greedily. As observed earlier, in the 2-connected component of u every node must belong to a cycle, thus by Claim 15, we get that every node of this component must be matched. We now build a second matching M'. We start from M and remove from the matching (u, v) and every edge that is in v's side of the bridge. Then we force (v, w) in the matching, and complete it greedily. The matching M' is maximal and u is unmatched, since all of its neighbors are matched, hence by Claim 15 it is not 1-robust, since it belongs to a 2-connected components thus to a cycle.

This concludes the proof of the claim. ◁

To conclude a graph in the class is either a tree, or is 2-connected, and in this last case because of Claim 15, every node must be matched in every maximal matching. Then Lemma 10 follows from Theorem 11.

4.2 More than one edge removal

▶ **Lemma 17.** *For any $k \geq 2$, $\mathcal{U}^k_{MM}$ is composed of the cycle on four nodes and of the set of trees.*

Proof. We first prove the reverse inclusion. As before, trees are in $\mathcal{U}^k_{MM}$ for any k because any edge removal disconnects the graph. Then for C_4, note that it belongs to $\mathcal{U}^1_{MM}$, and that the removal of more than one edge disconnects the graph.

For the other direction, we can restrict to $\mathcal{U}^2_{MM}$, and by definition it is included in $\mathcal{U}^1_{MM}$. Thus we can simply study the case of the balanced complete bipartite graphs and of the cliques on an even number of nodes. Consider first a complete bipartite graph $B_{k,k}$ with $k > 2$ (that is any $B_{k,k}$ larger than C_4), and a maximal matching M. Take two arbitrary edges (a_1, b_1) and (a_2, b_2) from the matching and remove them from the graph. The graph is still connected. Now the nodes a_1 and b_2 are unmatched and there is an edge between them, thus the resulting matching is not maximal and M is not 2-robust. Thus the only $B_{k,k}$ left in the class $\mathcal{U}^2_{MM}$ is C_4. For the cliques on an even number of nodes, consider one that has strictly more than two vertices. A maximal matching M contains at least two edges (u_1, v_1) and (u_2, v_2). When we remove these edges from the graph, we still have a connected graph, u_1 and u_2 are unmatched, but (u_1, u_2) still exists, thus the resulting matching is not maximal and M was not 2-robust. ◀

5 Maximal independent set

Maximal independent set illustrates yet another behavior for the classes $(\mathcal{U}^k_{MIS})_k$: they form an infinite strict hierarchy.

5.1 An infinite hierarchy

▶ **Theorem 18.** *For every $k \geq 1$, $\mathcal{U}^{k+1}_{MIS}$ is strictly included in $\mathcal{U}^k_{MIS}$.*

Proof. Let $k \geq 1$. We will define a graph G_k, and prove that it belongs to $\mathcal{U}^k_{MIS}$ but not to $\mathcal{U}^{k+1}_{MIS}$.

To build G_k, consider a bipartite graph with $k + 2$ nodes on each of the sides A and B, and add a pendant neighbor v to a node u on the side A. See Figure 6. This graph has only three MIS: A, $v \cup B$, and $v \cup (A \setminus u)$. Indeed: (1) if the MIS contains u, then it cannot contain vertices outside of A, and to be maximal it contains all of A, (2) if it contains a vertex of B, it cannot contain a vertex of A, and by maximality it contains all of B and v, and (3) if it contains v, and no vertex of B, then by maximality it is $v \cup (A \setminus u)$.

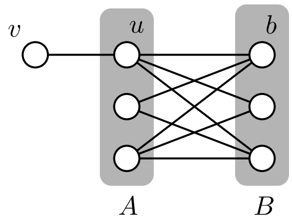


Figure 6 . Illustration of the graph G_k in the proof of Theorem 18.

We claim that these three MIS are k-robust, therefore G_k is in $\mathcal{U}^k_{MIS}$. Suppose an MIS is not k-robust. Then there exists a vertex w that is not part of the MIS, such that after at most k edge removals, it has no neighbor in the MIS anymore. Let us make a quick case analysis depending on who is this vertex w. It cannot be v, since removing the edge (u, v) would disconnect the graph. It cannot be a vertex of A, nor of B, because in all MIS mentioned, all non selected nodes (except v) have at least $k + 1$ selected neighbors.

Now we claim that $v \cup (A \setminus u)$ is not $(k+1)$-robust, thus G_k does not belong to $\mathcal{U}_{MIS}^{k+1}$. We choose a vertex b on the B side, and remove all the edges (a, b) for $a \in A \setminus u$. This is a set of $k+1$ edges whose removal does not disconnect the graph, but leaves b without selected neighbors. This $v \cup (A \setminus u)$ is not $(k+1)$-robust. ◀

5.2 A structure theorem for $\mathcal{U}_{MIS}^{k}$

The construction used in the proof of Theorem 18 is very specific and does not really inform about the nature of the graphs in $\mathcal{U}_{MIS}^{k}$. It can be generalized, with antennas on both sides and arbitrarily large (unbalanced) bipartite graphs with arbitrary number of antennas per nodes, but it is still specific. Moreover these construction heavily rely on pendant nodes, that are in some sense abusing the fact that we do not worry about the correctness of the solution if the graph gets disconnected.

In order to better understand these classes, and to give a more flexible way to build such graphs, we prove a theorem about how the class behaves with respect to the join operation (Definition 6).

We denote by $\mathcal{G}_p$ the class of graphs where every maximal independent set has size *at least* p. We say that a graph class is *stable by an operation* if, by applying this operation to any (set of) graph(s) from the class, the resulting graph is also in the class.

▶ **Theorem 19.** *For all k, the class $\mathcal{U}_{MIS}^{k} \cap \mathcal{G}_{k+1}$ is stable by join operation. Also, if either G or H is not in $\mathcal{U}_{MIS}^{k+1}$, then $join(G, H)$ is not in $\mathcal{U}_{MIS}^{k+1}$ either.*

Proof. Let us start with the first statement of the theorem. Consider two graphs G and H in $\mathcal{U}_{MIS}^{k} \cap \mathcal{G}_{k+1}$. We prove that $J = join(G, H)$ is also in $\mathcal{U}_{MIS}^{k} \cap \mathcal{G}_{k+1}$.

▷ Claim 20. Any MIS of J is either completely contained in the vertex set of G, and is an MIS of G, or contained in the vertex set of H, and is an MIS of H.

Consider an independent set in J. If it has a node u in G, then it has no node in H, as by construction, all nodes of H are linked to u. The analogue holds if the independent set has a node in H. Thus any independent set is either completely contained in G or completely contained in H. Now, a set is maximal independent in G (resp. H) alone if and only if it is maximal independent in G (resp. H) inside J. Indeed the only edges that we have added are between nodes of G and nodes of H. This proves the claim. ◁

Therefore, the resulting graph is in $\mathcal{G}_{k+1}$. Now for the k-robustness, consider without loss of generality an MIS of J that is in part G, and suppose it is not k-robust. In this case there must exists a non-selected vertex v, that has no more selected neighbors after the removal of k edges (while the graph stays connected). This node cannot be in the part G, otherwise the same independent set in the graph G would not be k-robust. And it cannot be in the part H, since every node of H is linked to all the vertices of the MIS, and this set has size at least $k+1$ since $G \in \mathcal{G}_{k+1}$.

Now, let us move on to the second statement of the theorem. Let's assume that G has an MIS S and $k+1$ edges such that their removal makes that S is not longer maximal (i.e. there exists some u that can be added to the set). Then, S is also an MIS of $join(G, H)$, and the removal of the same edges will allow to add u to the set, as the only new neighbors of u are from H that does not contain any node of the chosen MIS ◀

6 The existence of a robust MIS is NP-hard

Remember that we have defined two types of graph classes related to robustness. For a given problem, and a parameter k, the universal class is the class where every solution is k-robust. This is the version we have explored so far. For this version, recognizing the graphs of the class is easy since these have simple explicit characterization. The second type of class is the existential type, where we want that there exists a solution that is k-robust. And here the landscape is much more complex. Indeed, in [6] in the simpler case of robustness without parameter, there is no explicit characterization of the existential class, only a rather involved algorithm. In this section we show that, when we add the parameter k the situation becomes even more challenging: the algorithm of [6] runs in polynomial time, and here we show that the recognition of $\mathcal{E}^1_{MIS}$ is NP-hard.

▶ **Theorem 21.** *For every odd integer k, it is NP-hard to decide whether a graph belongs to $\mathcal{E}^k_{MIS}$.*

The rest of this section is devoted to the proof of this theorem. It is based on the NP-completeness of the following problem.

Perfect stable
Input: A graph $G = (V, E)$.
Question: Does there exists a subset of vertices $S \subset V$ that is independent 2-dominating?

Remember that a set is independent 2-dominating if no two neighbors can be selected, and every non-selected vertex should have at least two selected neighbors. Just to get some intuition about why we are interested in this problem, note that with independent 2-dominating after removing an edge between a selected and a non-selected vertex, the non-selected vertex is still dominated. It was proved in [7] that Perfect stable is NP-hard in general. We will need the following strengthening of this hardness result.

▶ **Lemma 22.** *Deciding whether a 2-connected graph has an independent 2-dominating set is NP-complete.*

Note that this lemma does not follow directly from [7] because the reduction there does use some non-2-connected graphs.

Proof. Let G be an arbitrary connected graph with at least one edge. Consider G' to be the same as G but with a universal vertex, that is, G with an additional vertex u that is adjacent to all the vertices of G. This graph is 2-edge connected. Indeed, since G is connected and has at least two vertices, removing any edge (u, v) with $v \in V(G)$ cannot disconnect the graph, and removing an edge from G does not disconnect the graph because all nodes are linked through v.

We claim that G' has an independent 2-dominating set if and only if G has one. First, suppose that G has such a set S. Note that the set S has at least two selected vertices. Indeed, G has at least one edge, which implies that at least one vertex is not selected (by independence), and such a vertex should be dominated by at least two selected vertices. Now we claim that S is also a solution for G'. Indeed, the addition of u to the graph does not impact the independence of S, nor the 2-domination of the nodes of G, and v is covered at least twice, since there are at least two selected vertices in G. Second, if G' has independent 2-dominating set S', it cannot contain v. Indeed, because of the independence condition, if v

is selected, then no other node can be selected, and then the 2-domination condition is not satisfied. Then S' is contained in G and is an independent 2-dominating set of G. ◀

Now, let us formalise the connection between robustness and independent 2-domination.

▶ **Lemma 23.** *In a 2-connected graph, the 1-robust maximal independent sets are exactly the independent 2-dominating sets.*

Proof. As a consequence of Property 2, in a 2-connected graph, a 1-robust MIS is an MIS that is robust against the removal of any edge (that is, we can forget about the preserved connectivity in the robustness definition). This means that every node not in the MIS is covered twice, otherwise one could break the maximality by removing the edge between the node covered only once and the node that covers it. In other words, the independent set must be 2-dominating. For the other direction it suffices to note that any independent dominating set is a maximal independent set. ◀

At that point, plugging Lemma 22 and Lemma 23 we get that deciding whether there exists a 1-robust MIS in a graph is NP-hard, even if we assume 2-connectivity. This last lemma is the final step to prove Theorem 21.

▶ **Lemma 24.** *For any 2-connected graph G and any integer $k > 1$, we can build in polynomial-time a graph G', such that: G has a 1-robust MIS if and only if G' has a $2k-1$-robust MIS.*

Proof. We build G' in the following way. Take k copies of G, denoted $G_1, ..., G_k$, with the notation that u_x is the copy of vertex u in the x-th copy. For every edge (u, v) of G, we add the edge (u_x, v_y) for any pair $x, y \in 1, ..., k$.

Let us first establish the following claim. An MIS in the graph G' necessarily has the following form: it is the union of the exact same set repeated on each copy. Indeed, let u_i be in the MIS. For any $j \neq i$, all the neighbors of u_j in the copy G_j are a neighbor of u_i, which implies that they are not in the MIS. Hence, no neighbor of u in any copy can be in the MIS. As those nodes are the only neighbors of u_j, it implies that u_j is also in the MIS.

Now suppose that G has a 1-robust MIS. We can select the clones of this MIS in each copy, and build an MIS for G' (the independence and maximality are easy to check). In this MIS of G', every non-selected vertex has at least $2k$ selected neighbors, therefore this MIS is $2k - 1$ robust.

Finally, suppose that G' has a $2k-1$ robust MIS. Thanks to the claim above, we know that this MIS is the same set of vertices repeated on each copy. We claim that when restricted to a given copy, this MIS is 1-robust. Indeed, if it were not, then there would be one non-selected vertex with at most one selected neighbor, and this would mean that in G' this vertex would have only k selected neighbors, which contradicts the $2k - 1$ robust (given the connectivity). ◀

7 Conclusions

In this paper we have developed the theory of robustness in several ways: adding granularity and studying new natural problems to explore its diversity. The next step is to fill in the gaps in our set of results: characterizing exactly the classes $\mathcal{U}^k_{MIS}$, and understanding the complexity of answering the existential question for maximal matching and minimum dominating set. We believe that a polynomial-time algorithm can be designed to answer the existential question in the case of maximal matching with $k = 1$, with an approach similar to the one of [6] for MDS (that is, via a careful dynamic programming work on a tree-like decomposition of the graphs). A more long-term goal is to reuse the insights gathered by studying robustness to help the design of dynamic algorithms.

Acknowledgements.

We thank Nicolas El Maalouly for fruitful discussions about perfect matchings and Dennis Olivetti for pointing out reference [10].

References

1. Nicolas Braud Santoni, Swan Dubois, Mohamed Hamza Kaaouachi, and Franck Petit. A generic framework for impossibility results in time-varying graphs. In *IEEE 29nd International Symposium on Parallel and Distributed Processing (IPDPS 2015), 17th Workshop on Advances in Parallel and Distributed Computational Models (APDCM 2015)*, pages 483–489. IEEE, 2015.
2. Nicolas Braud-Santoni, Swan Dubois, Mohamed-Hamza Kaaouachi, and Franck Petit. The next 700 impossibility results in time-varying graphs. *International Journal of Networking and Computing*, 6(1):27–41, 2016.
3. A. Casteigts and P. Flocchini. Deterministic algorithms in dynamic networks: Problems, analysis, and algorithmic tools. Technical report, Defence Research and Development Canada, 2013-020, 2013.
4. A. Casteigts, P. Flocchini, W. Quattrociocchi, and N. Santoro. Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408, 2012.
5. Arnaud Casteigts, Timothée Corsini, Hervé Hocquard, and Arnaud Labourel. Robustness of distances and diameter in a fragile network. In *1st Symposium on Algorithmic Foundations of Dynamic Networks, SAND 2022*, volume 221 of *LIPIcs*, pages 9:1–9:16, 2022. doi:10.4230/LIPIcs.SAND.2022.9.
6. Arnaud Casteigts, Swan Dubois, Franck Petit, and John Michael Robson. Robustness: A new form of heredity motivated by dynamic networks. *Theor. Comput. Sci.*, 806:429–445, 2020. doi:10.1016/j.tcs.2019.08.008.
7. Cornelius Croitoru and Emilian Suditu. Perfect stables in graphs. *Inf. Process. Lett.*, 17(1):53–56, 1983. doi:10.1016/0020-0190(83)90091-1.
8. Juho Hirvonen and Jukka Suomela. Distributed algorithms, 2020. https://jukkasuomela.fi/da2020/da2020.pdf.
9. David Peleg. *Distributed computing: a locality-sensitive approach.* SIAM, 2000.
10. David P Summer. Randomly matchable graphs. *Journal of Graph Theory*, 3(2):183–186, 1979. doi:10.1002/jgt.3190030209.
11. B. Xuan, A. Ferreira, and A. Jarry. Computing shortest, fastest, and foremost journeys in dynamic networks. *International Journal of Foundations of Computer Science*, 14(02):267–285, 2003.