When Should You Wait Before Updating?

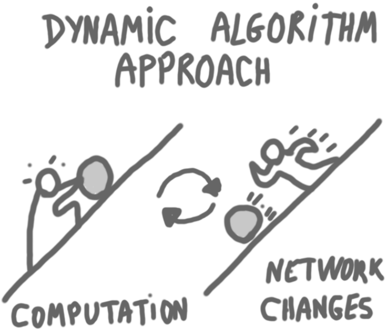
Toward a Robustness Refinement

Swan Dubois, Laurent Feuilloley, Franck Petit, Mikaël Rabie

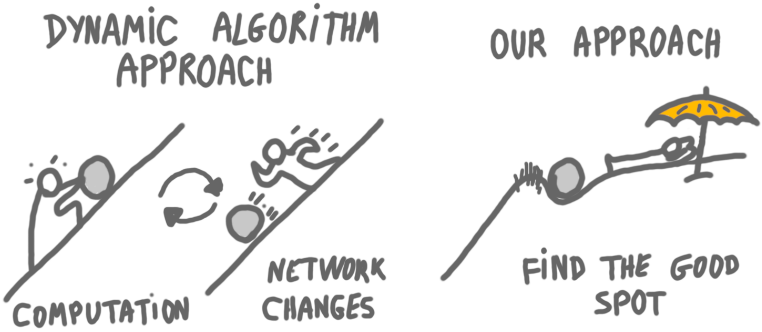
Sorbonne University, University of Lyon, Paris Cité University

SAND 2023 · Pisa

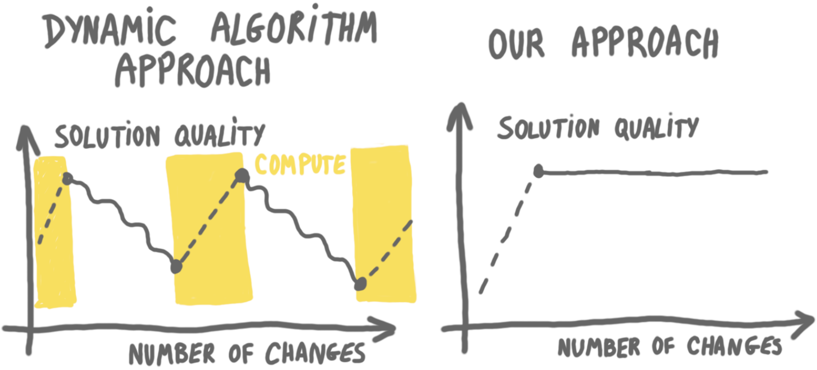
Dynamic algorithms and our approach



Dynamic algorithms and our approach



Dynamic algorithms and our approach



General goal

Identify instances and solutions
that are more robust than others

Necessary restrictions

In some situations there is no hope to get any sort of robustness.

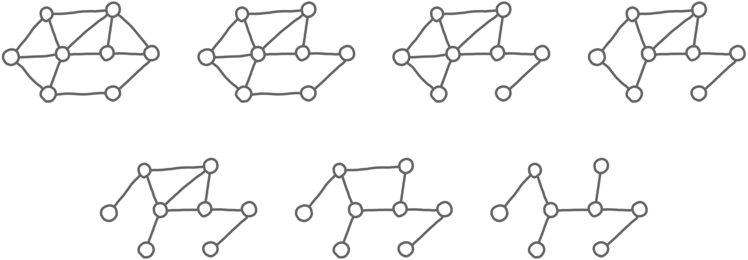
→ Very sensitive problems, eg maintain the number of edges

→ Arbitrary dynamic.

Restriction of the dynamic

Assumption on the dynamic:

- Edge removals only (static vertex set, no edge additions)
- Keeping connectivity.



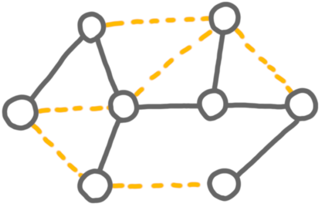
Restriction of the dynamic

Assumption on the dynamic:

- Edge removals only (static vertex set, no edge additions)
- Keeping connectivity.

Motivations:

- Decaying networks
- More promising
- Eventual footprint.

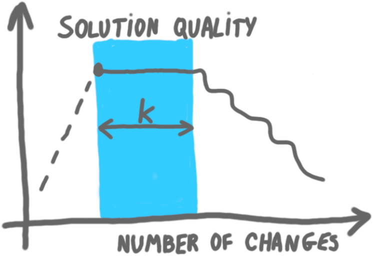


Robustness: A new form of heredity motivated by dynamic networks by Casteigts, Dubois, Petit, Robson. 2020.

Granularity of edge-removals

New parameter: k = number of edges removed.

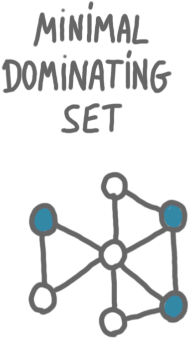
A solution is k-robust if it is still correct after k edge removals.

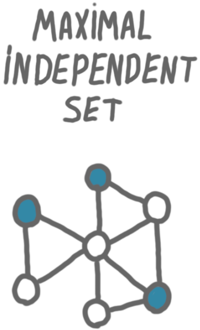


Focus on local problems

Locally checkable problems (LCL): the solution is correct if and only if it is locally correct everywhere.

Our three problems:





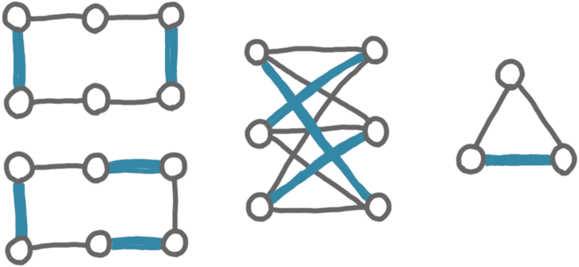
Two basic questions

For a problem P and a parameter k:

Universal question: What is the set of graphs such that all solutions are k-robust, $\mathcal{U}_k^P$?

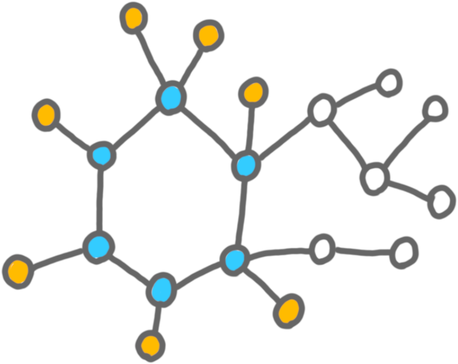
Existential question: What is the set of graphs such that there exists a solution that is k-robust, $\mathcal{E}_k^P$?

$\rightarrow$ Examples with $P =$ maximal matching and $k = 1$.



Basic building blocks

Trees

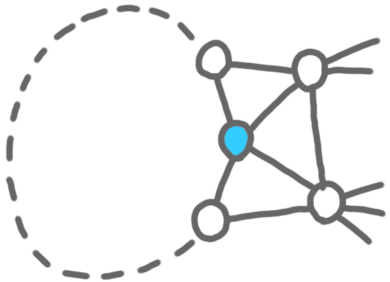


Sputnik graphs

Minimal dominating set results

Theorem: $\forall k,\ \mathcal{U}_k^{MDS} = \{\text{Sputnik graphs}\}$.

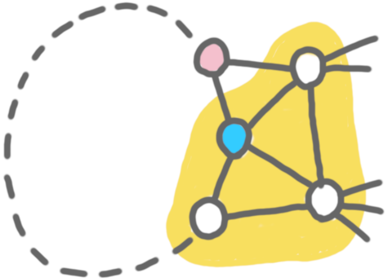
Proof idea:



Minimal dominating set results

Theorem: $\forall k$, $\mathcal{U}_k^{MDS} = \{\text{Sputnik graphs}\}$.

Proof idea:

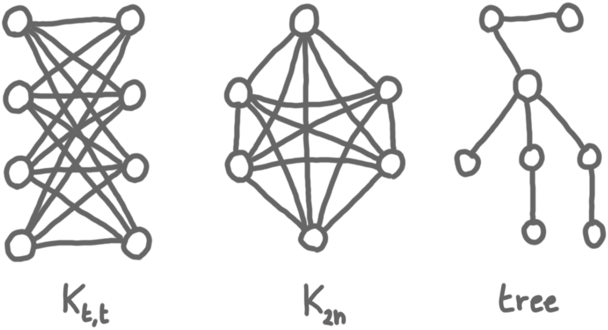


Maximal matchings results

Theorem:

$\mathcal{U}_1^{MM} =$ balanced bicliques ($K_{t,t}$), even cliques (K_{2n}), and trees.

$\mathcal{U}_{k>1}^{MM} =$ cycle C_4, and trees.



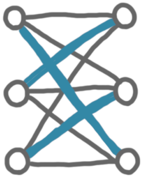
Maximal matchings results

Theorem:

$\mathcal{U}_1^{MM} =$ balanced bicliques ($K_{t,t}$), even cliques (K_{2n}), and trees.

$\mathcal{U}_{k>1}^{MM} =$ cycle C_4, and trees.

Core of the proof: proving that "$G \in \mathcal{U}_1^{MM}$" is equivalent to "all maximal matchings in G are perfect matchings" (except for trees).



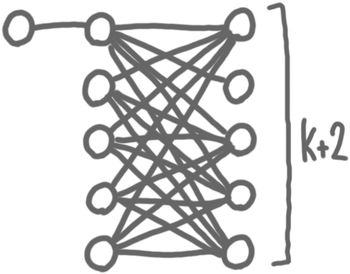
Theorem [Summer79]: Randomly matchable graphs $= \{K_{tt}, K_{2n}\}$.

Maximal independent set results

$\rightarrow$ No exact characterization of $\mathcal{U}_k^{MIS}$. :-(

Theorem: A strict hierarchy: $\mathcal{U}_{k+1}^{MIS} \subsetneq \mathcal{U}_k^{MIS}$.

Proof: $G_k \in \mathcal{U}_k^{MIS} \setminus \mathcal{U}_{k+1}^{MIS}$:

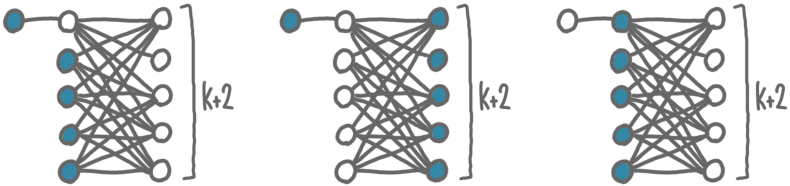


Maximal independent set results

$\rightarrow$ No exact characterization of $\mathcal{U}_k^{MIS}$. :-(

Theorem: A strict hierarchy: $\mathcal{U}_{k+1}^{MIS} \subsetneq \mathcal{U}_k^{MIS}$.

Proof: $G_k \in \mathcal{U}_k^{MIS} \setminus \mathcal{U}_{k+1}^{MIS}$:

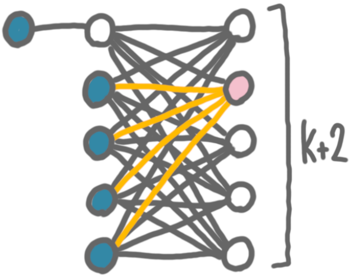


Maximal independent set results

→ No exact characterization of $\mathcal{U}_k^{MIS}$. :-(

Theorem: A strict hierarchy: $\mathcal{U}_{k+1}^{MIS} \subsetneq \mathcal{U}_k^{MIS}$.

Proof: $G_k \in \mathcal{U}_k^{MIS} \setminus \mathcal{U}_{k+1}^{MIS}$:

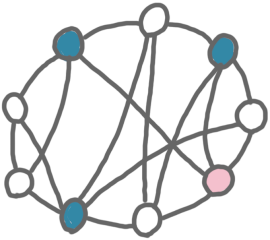


About the existential question

Theorem [CDPR20]: There exists a polynomial time algorithm to decide if a graph belongs to $\mathcal{E}_{\infty}^{MIS}$.

Theorem: Deciding $\mathcal{E}_{1}^{MIS}$ is NP-hard.

Proof: Focus on the 2-connected case where is it equivalent to "perfect stable" (= every node not in the set is dominated at least twice).



Theorem [Croitoru-Suditu 83] Deciding if there exists a perfect stable is NP-hard.

Conclusion

- Universal question: Good understanding, but small classes.
- Existential question: Large classes but difficult to recognize and use.
- Maybe the right question is in between: graphs such that there exist robust solutions that are easy to find/maintain.
- Even better: new criteria to quickly measure the robustness of subgraphs, in order to guide algorithms.

.