What can be certified compactly?

Nicolas Bousquet, Laurent Feuilloley, Théo Pierron (CNRS and University of Lyon, France). Appeared at PODC 2022.

Introduction to local certification

- **Where does this take place?** Theory of distributed computing / study of locality and fault-tolerance.
- **General question:** How to convince the node of a network that a global property holds, when they only have a local view? What kind/amount of additional global information do they need?
- **Original motivation:** Self-stabilizing algorithms are distributed algorithms that converge, even if the local memory of nodes is arbitrarily corrupted at the start. They are described by local update rules, based only on the neighbors local memories. Local certification is the main way to ensure that the nodes stop updating, if and only if, they have reached a correct configuration.
- **Alternative motivation:** The amount of information we need to give to the nodes to certify a property is a measure of the globality of this property.

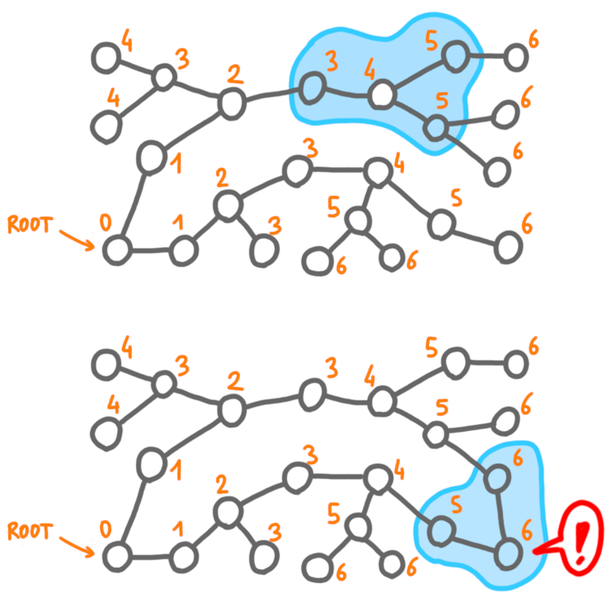
Definition by example: acyclity

Example of a property to check: The network is acyclic (= is a tree, because we assume connectivity).

Model: Every node looks at its neighbors in the graph, and takes a decision: stay silent (accept) or raise an alarm (reject).

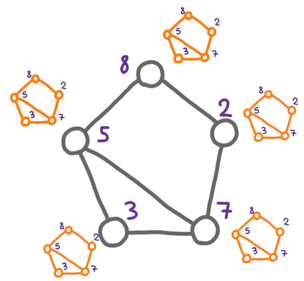
Goal: *If the network has the property, all nodes should accept, otherwise, at least node should reject.*

→ This task is impossible, because we cannot distinguish locally between a cycle and a path (even if we look at large $o(n)$ distance)
→ Now suppose that we give to each vertex its distance to a chosen root, and each vertex checks local consistency. Then we can solve the task: there exist an accepted distance labeling iff the graph is a tree.



Universal scheme

If the network is equipped with node identifiers, any property can be certified, by giving the full map to every node.



Certification size landscape

Certification size of a property = minimum size of the certificates in a correct scheme, function of n.

- Any property has a certification of size in $O(n^2)$.
- Some properties need $\Omega(n^2)$, e.g. symmetric graphs. Also $\Omega(n)$ for "diameter = 3".
- For many property of interest: $\Theta(\log n)$:
  - tree-style problems: acyclicity, spanning tree...
  - classic graph classes: planar, chordal, interval...
- Since for many property going below $\log n$ is impossible, this size is the gold standard of certification (like polynomial-time for standard algorithms).
- A typical exception: k-colorability, in $O(\log k)$.

Key question

What can be certified in $O(\log n)$ bits?

This paper

Our approach: Establish a *meta-theorem*, "all the properties of type X can be certified compactly on graphs of the form Y."

In standard algorithmics, a classic is Courcelles' theorem: Any MSO property can be verified in polynomial time in bounded treewidth graphs.

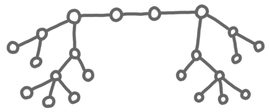
What is MSO? A property is in MSO if it can be expressed in a given graph logic (monadic second-order), which is an extension of first-order logic, with quantification on sets of vertices.
Example: path of even length $\exists S, \forall v, \forall u, u - v \Rightarrow (v \in S \wedge u \notin S) \vee (v \notin S \wedge u \in S)$ and "exactly one endpoint is in S".

Why restricting both formula and shape is needed

- Triangle-freeness requires $\Omega(n/e^{O(\sqrt{n})})$ bits even though it has a simple FO formula:

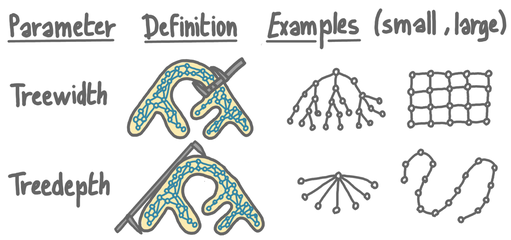
$$\forall x, y, z, \neg(x - y \wedge y - z \wedge z - x)$$

- Symmetric trees (even of bounded depth) require $\Omega(n)$ bits.



Our parameter: treedepth

Treedepth is basically about measuring how large can induced paths be.



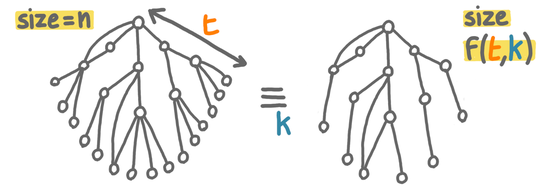
Precisely, a graph has treedepth t is it can be be embedded in an "ancestor-descendant tree" of depth t.

It is basically the most general parameter for a Courcelle-style theorem with reasonable dependency in the parameter.

Technique

Step 1: Certifying treedepth structure. Basically giving the list of ancestors + spanning trees pointing to them.

Step 2: Certified kernelization. The depth k of a formula,is the number quantifiers $\varphi = \exists x_1, \forall x_2, ...\exists x_k, XXX$
We prove that for every k we can compute a compressed version of the graph (the kernel) that satisfies exactly the same MSO formulas of depth k.



Done by pruning branches that have the exact same role in the tree-structure. Key part: *certify the pruning locally.*

Since the kernel is small we can give it to every node, that checks that it satisfies the formula, and that it is locally consistent with the graph and pruning.

More

- Another theorem of the paper: MSO properties can be certified in $O(1)$ bits on trees. Proofs by automata theory.

- Follow-up (Fraigniaud, Montealegre, Rapaport, Todinca): Same with treewidth, but needs $O(\log^2 n)$. Optimal?
- Related important open question: can all minor-closed classes be certified compactly?