On the diameter of locally constrained trees

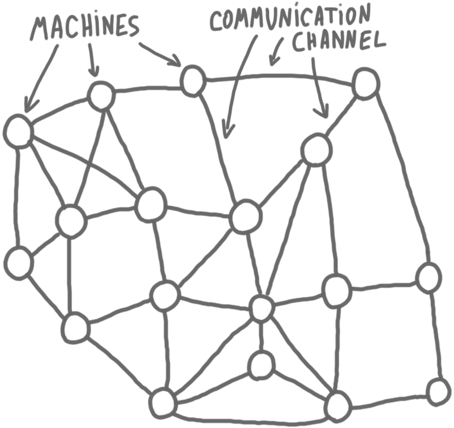
from distributed computing to living beings?

Nicolas Bousquet
Laurent Feuilloley
Antonin Kiladjian
Théo Pierron

Combinatorics and Life Science
Lyon · Fall 2025

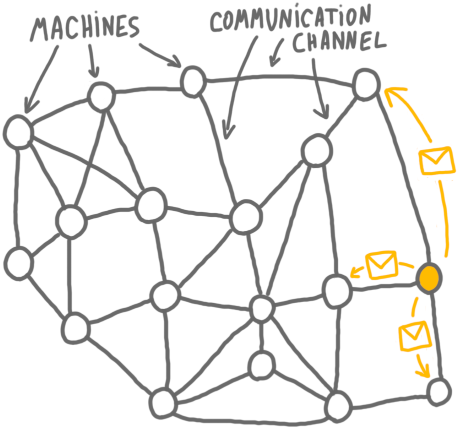
Distributed graph algorithms

- Network of machines modeled as a graph.
- Communication by synchronous rounds.
- Restrict/count communication, not computation.
- Each node must hold its part of the solution.
- Notations: n nodes, diameter D.
- (Unique identifiers.)



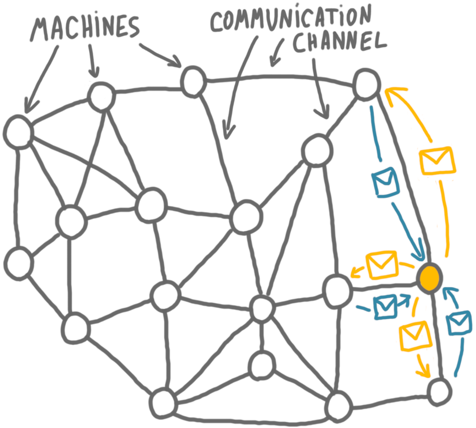
Distributed graph algorithms

- Network of machines modeled as a graph.
- Communication by synchronous rounds.
- Restrict/count communication, not computation.
- Each node must hold its part of the solution.
- Notations: n nodes, diameter D.
- (Unique identifiers.)



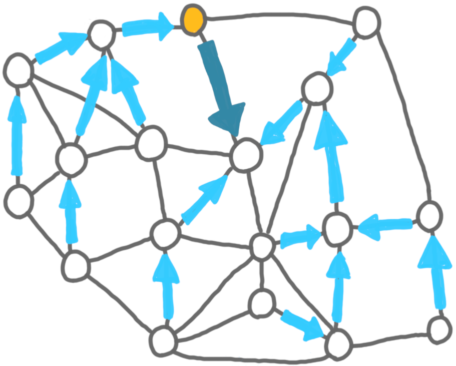
Distributed graph algorithms

- Network of machines modeled as a graph.
- Communication by synchronous rounds.
- Restrict/count communication, not computation.
- Each node must hold its part of the solution.
- Notations: n nodes, diameter D.
- (Unique identifiers.)



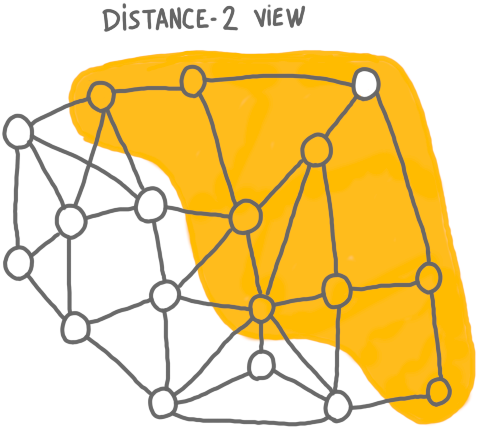
Distributed graph algorithms

- Network of machines modeled as a graph.
- Communication by synchronous rounds.
- Restrict/count communication, not computation.
- Each node must hold its part of the solution.
- Notations: n nodes, diameter D.
- (Unique identifiers.)



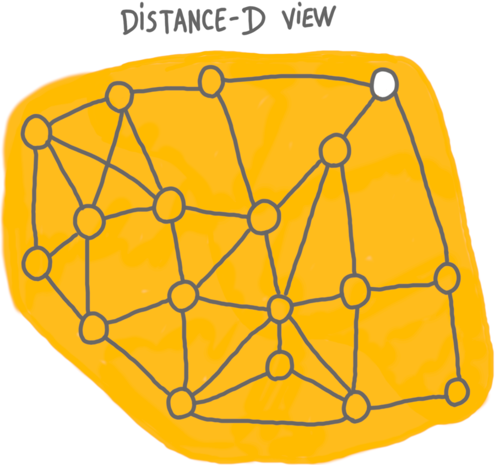
LOCAL model

- No constraint on message size.
- Complexity: number of rounds.
- **Theorem:** k-round algorithm equivalent to a mapping from distance k neighborhood to output.
- Generic upper bound on complexity: $O(n)$ or $O(D)$.
- Example of a global problem: 2-coloring a path.



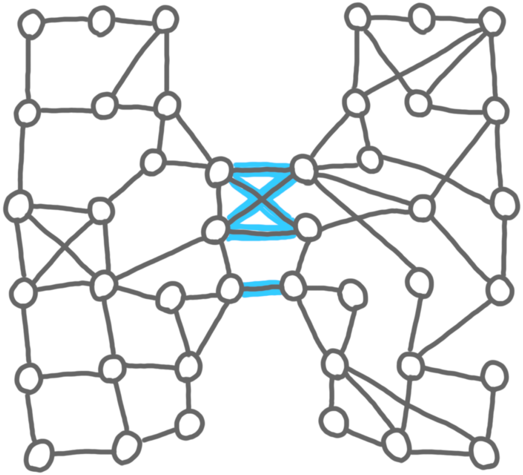
LOCAL model

- No constraint on message size.
- Complexity: number of rounds.
- **Theorem:** k-round algorithm equivalent to a mapping from distance k neighborhood to output.
- Generic upper bound on complexity: $O(n)$ or $O(D)$.
- Example of a global problem: 2-coloring a path.



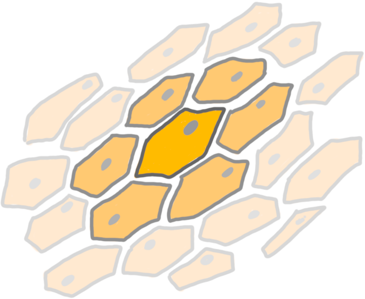
CONGEST model

- **Constraint:** $O(logn)$-bit messages.
- **Complexity:** Number of rounds.
- Generic upper bound: $O(n^2)$.

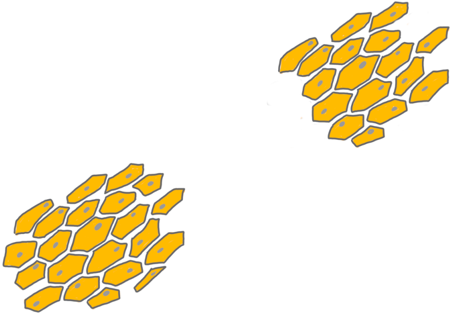


Locality and congestion in living beings

Locality



Congestion

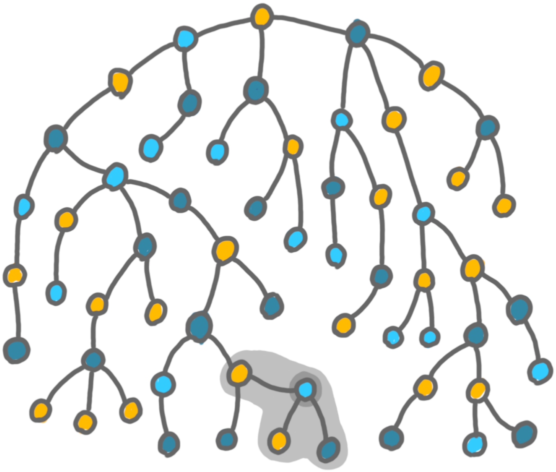


Local problems

Definition: Locally checkable labelings are the class of problems with:

- constant size outputs
- where the output can be checked locally.

Examples: k-coloring, maximal independent set, dominating set.

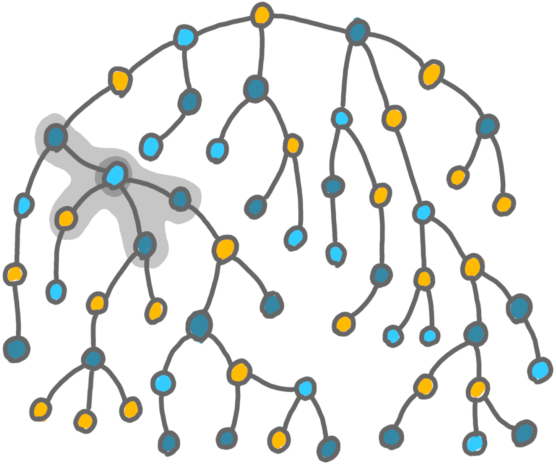


Local problems

Definition: Locally checkable labelings are the class of problems with:

- constant size outputs
- where the output can be checked locally.

Examples: k-coloring, maximal independent set, dominating set.

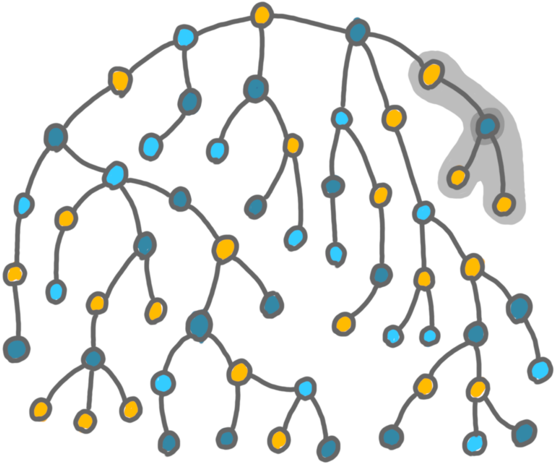


Local problems

Definition: Locally checkable labelings are the class of problems with:

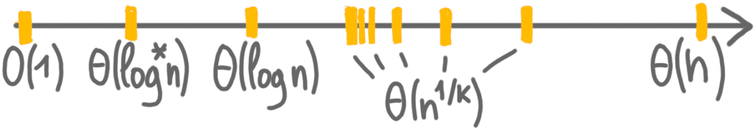
- constant size outputs
- where the output can be checked locally.

Examples: k-coloring, maximal independent set, dominating set.



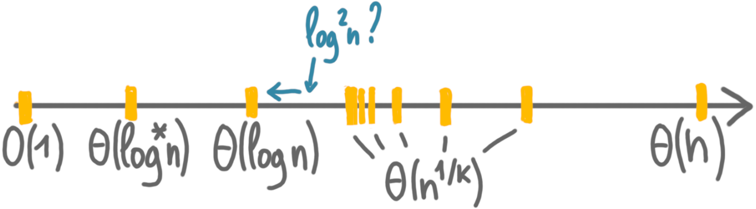
Landscape theorem for LCLS

Theorem [many authors]: In the LOCAL model, in bounded-degree trees, the complexity of solving an LCL can only be of the following form: $O(1)$, $\Theta(\log^* n)$, $\Theta(\log n)$, $\Theta(n^{1/k})$, $\Theta(n)$.



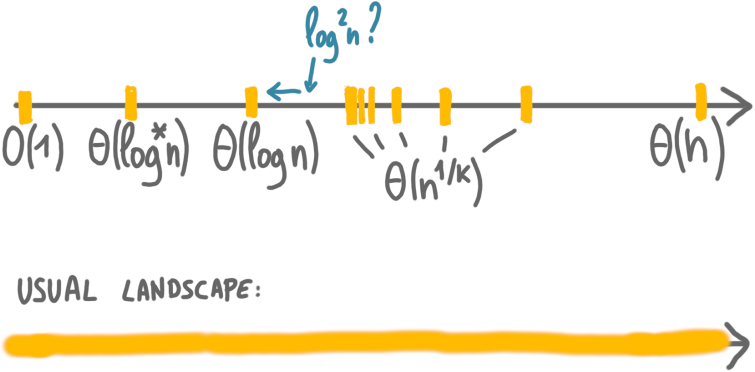
Landscape theorem for LCLS

Theorem [many authors]: In the LOCAL model, in bounded-degree trees, the complexity of solving an LCL can only be of the following form: $O(1)$, $\Theta(\log^* n)$, $\Theta(\log n)$, $\Theta(n^{1/k})$, $\Theta(n)$.



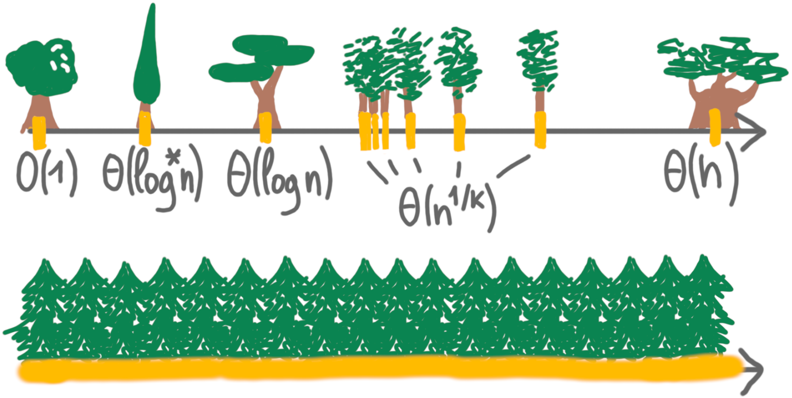
Landscape theorem for LCLS

Theorem [many authors]: In the LOCAL model, in bounded-degree trees, the complexity of solving an LCL can only be of the following form: $O(1)$, $\Theta(\log^* n)$, $\Theta(\log n)$, $\Theta(n^{1/k})$, $\Theta(n)$.



Landscape theorem for LCLS

Theorem [many authors]: In the LOCAL model, in bounded-degree trees, the complexity of solving an LCL can only be of the following form: $O(1)$, $\Theta(\log^* n)$, $\Theta(\log n)$, $\Theta(n^{1/k})$, $\Theta(n)$.



What about unbounded degree?

Question: Does the landscape survives if we remove the bounded degree constraints?

No. :(

Theorem: For 'any' function f, there exists an LCL that has complexity $\Theta(f)$.

Equivalent to the following:

Theorem: For 'any' function f, there exists a local checker such that the maximum diameter of the trees accepted is $\Theta(f)$.

Local checkers

Local checker: A mapping from radius-k neighborhoods to accept/reject.
A tree is accepted by a local checker if all nodes accept.

Local checkers

Local checker: A mapping from radius-k neighborhoods to accept/reject.
A tree is accepted by a local checker if all nodes accept.

Example:

Local checker for caterpillars. Every node checks that it is:

- A leaf or
- A node of degree d with 2 neighbors of degree $d-1$ or $d+1$
- an "endpoint"

Local checkers

Local checker: A mapping from radius-k neighborhoods to accept/reject.
A tree is accepted by a local checker if all nodes accept.

Example:

Local checker for caterpillars. Every node checks that it is:

- A leaf or
- A node of degree d with 2 neighbors of degree $d-1$ or $d+1$
- an "endpoint"

Local checkers

Local checker: A mapping from radius-k neighborhoods to accept/reject.
A tree is accepted by a local checker if all nodes accept.

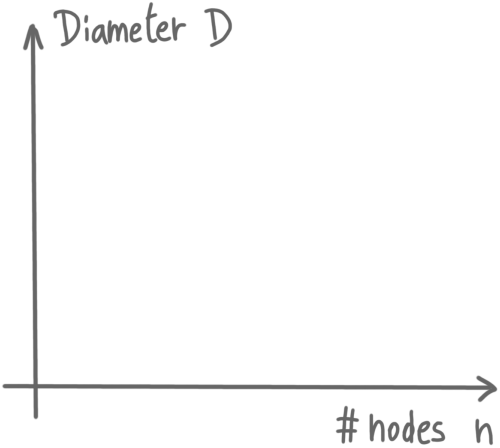
Example:

Local checker for caterpillars. Every node checks that it is:

- A leaf or
- A node of degree d with 2 neighbors of degree $d-1$ or $d+1$
- an "endpoint"

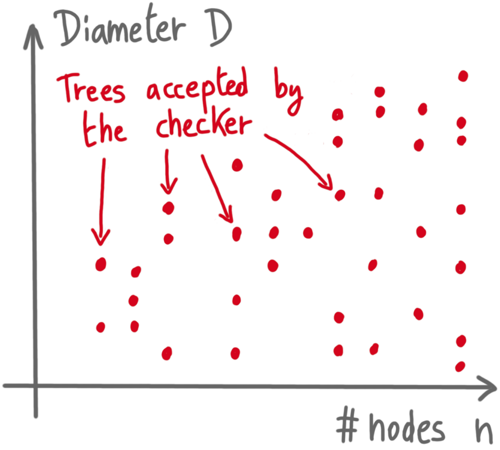
Max diameter of a local checker

- Fix a local checker.
- For each tree accepted, consider its diameter D, as a function of the number of nodes n.
- Focus on the maximum diameter, and smoothed version of it captured by a function.



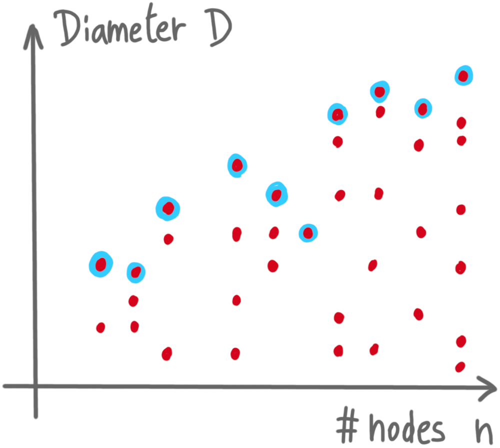
Max diameter of a local checker

- Fix a local checker.
- For each tree accepted, consider its diameter D, as a function of the number of nodes n.
- Focus on the maximum diameter, and smoothed version of it captured by a function.



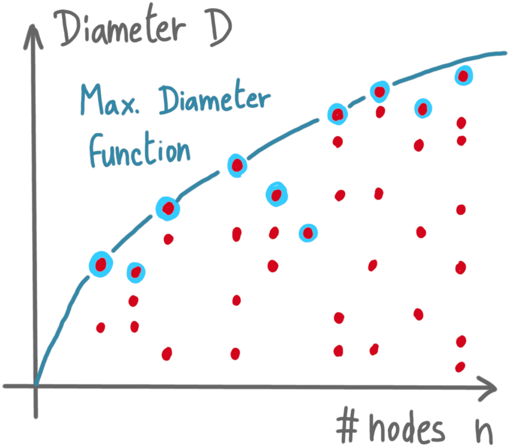
Max diameter of a local checker

- Fix a local checker.
- For each tree accepted, consider its diameter D, as a function of the number of nodes n.
- Focus on the maximum diameter, and smoothed version of it captured by a function.



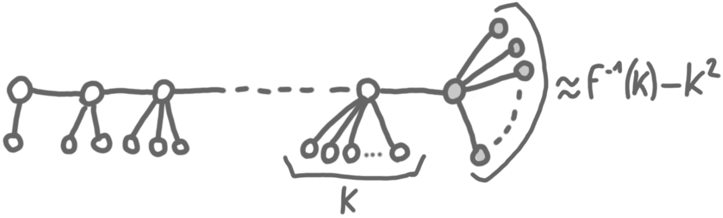
Max diameter of a local checker

- Fix a local checker.
- For each tree accepted, consider its diameter D, as a function of the number of nodes n.
- Focus on the maximum diameter, and smoothed version of it captured by a function.



Any diameter by padding

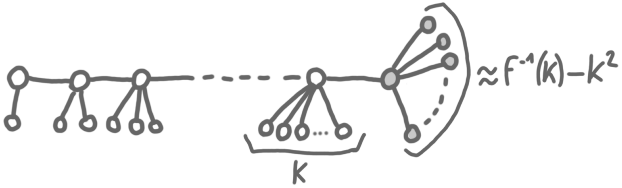
Theorem: For ‘any’ function f, there exists a local checker such that the maximum diameter of the trees accepted is $\Theta(f)$.



To get the LCL theorem, define an LCL such that:

- on the graph accepted by the local checker the complexity is the diameter (global problem)
- otherwise it is easier.

A landscape for nice unbounded degree?



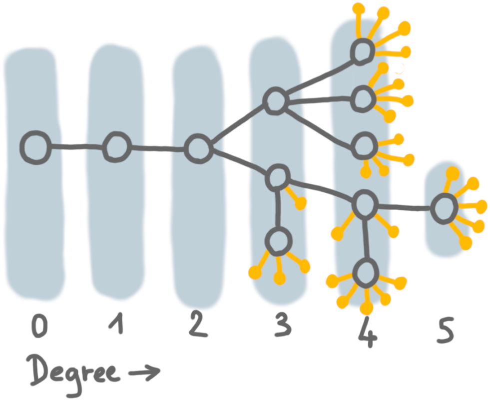
The construction is not very satisfying:

- Arbitrary jump in degree and local computation of f
- Does not feel homogeneous/intrinsic/natural.

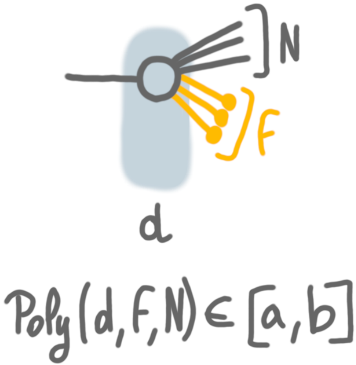
Question: Can we define reasonable constraints on local checkers and get back a nice landscape for maximum diameter of trees?

Two constraints

Laminated trees



Polynomial constraints



Landscape for constrained trees

Theorem: For laminated trees with polynomial constraints, the possible maximum diameters are:
$O(1)$, $\Theta(\log n / \log\log n)$, $\Theta(\log n)$, $\Theta(n^{a/b})$, and $\Theta(n)$
with $a/b \in [1/3, 1/2]$.

Back to life (science)

Take home message: There are landscape theorem in distributed computing, and now in “pure” combinatorics. They are useful and interesting.

Questions:

- Relevance to biology? Same flavor as L-systems.
- Dynamic vs static. Fixing "faults".
- Beyond deterministic maximum diameter.
- Beyond trees.