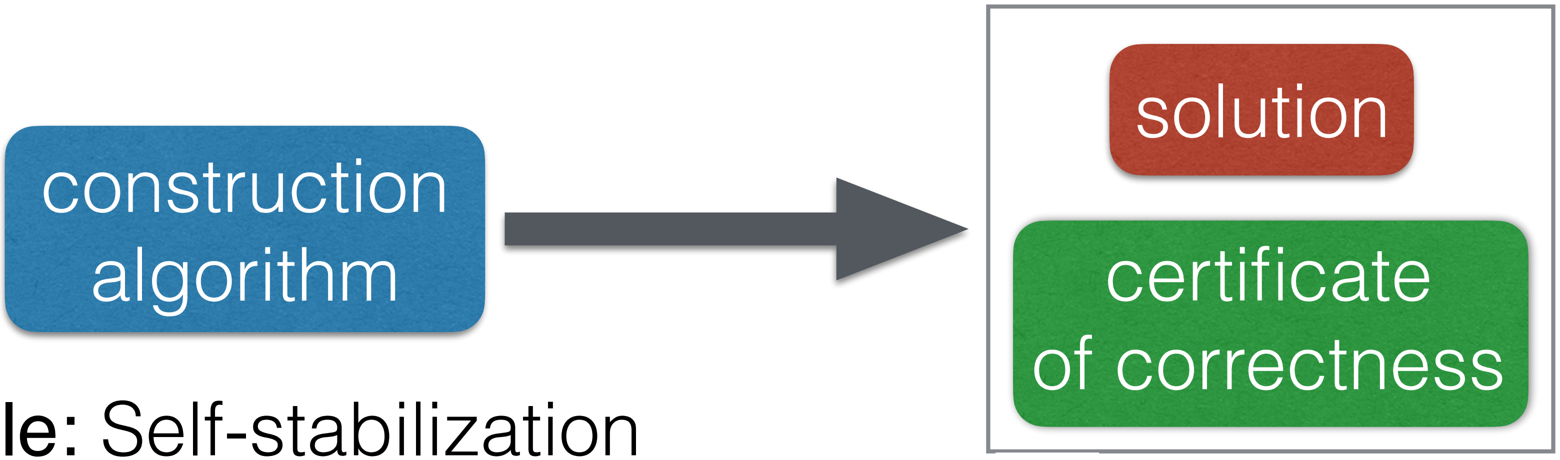
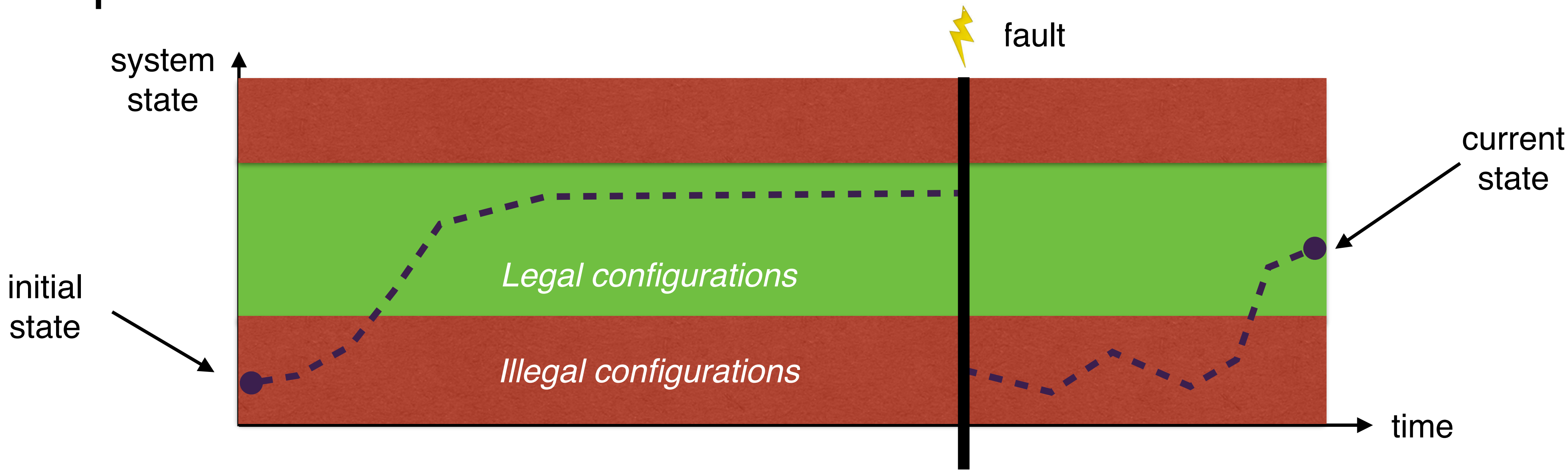
Distributed Certification

- Definition
- Certifying Spanning Tree
- Universal Certification Scheme
- Lower bounds
- Interactive Protocols

Application: Fault-Tolerance

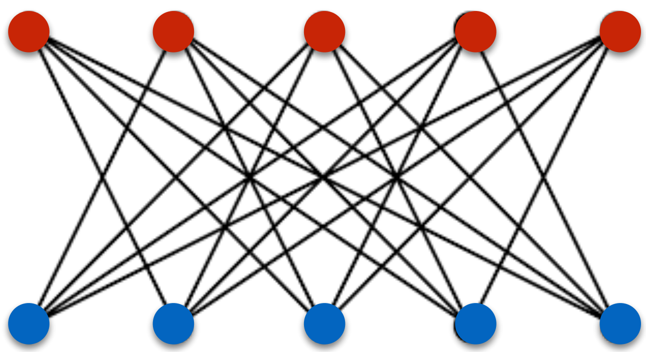


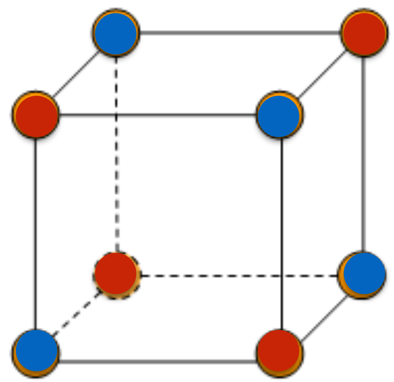
Example: Self-stabilization



Example: Bipartiteness

- **Definition** A graph $G = (V, E)$ is bipartite if V can be partitioned into two sets V_1 and V_2 such that $G[V_1]$ and $G[V_2]$ are stable graphs (i.e., for every edge in E one extremity is in V_1 and the other extremity is in V_2)
- **Remark:** G is bipartite $\iff$ G is 2-colorable.





Verification is local:

- bipartite $\Longrightarrow$ all nodes accept
- non bipartite $\Longrightarrow$ at least one node rejects

Certification Scheme

Given a graph property:

- A non-trustable *prover* assigns *certificates* to the nodes
- A distributed *verifier* checks these certificates at each nodes (in $O(1)$ rounds)

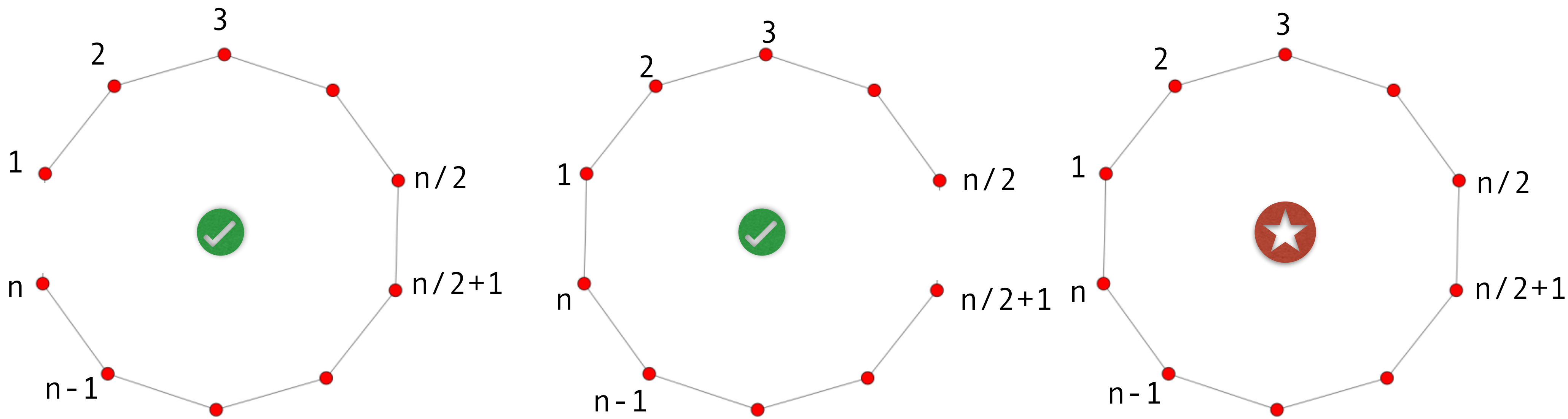
Completeness: If the property is satisfies then there exists certificates such that the verifier accepts at all nodes.

Soundness: If the property is not satisfied, then, for every certificate assignment, the verifier reject in at least one node

Variants of Certification Schemes

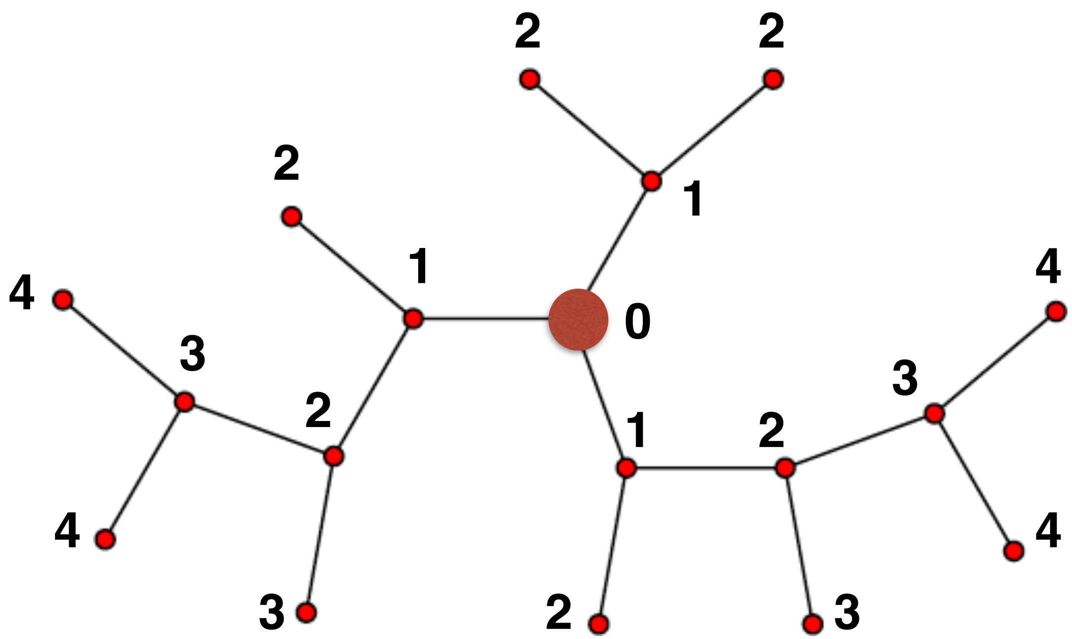
- Locally Checkable Proofs: Verifiers exchange inputs and certificates with neighbors
- Proof-Labeling Scheme: Verifiers exchange only the certificates
- Non-Deterministic Local Decision: Certificates do not depend of the IDs assigned to the node

Cycle-Freeness



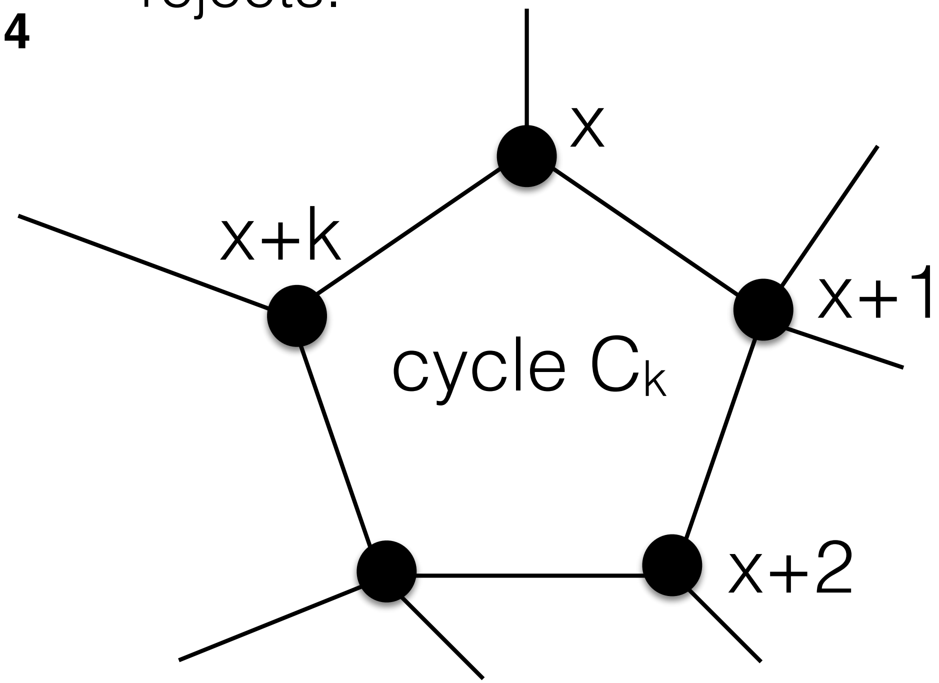
Non locally decidable!

Certifying Cycle-Freeness



if G is acyclic, then there is an assignment of the counter resulting in all nodes accept.

if G is has a cycle, then for every assignment of the counters, at least one node rejects.



Verifier at node u

exchange counters with neighbors
if $\exists!\ v\in N(u)$: $cpt(v)=cpt(u)-1$ and
$\forall\ w\in N(u)\setminus\{v\}$, $cpt(w)=cpt(u)+1$
then accept
else reject

Proof-Labeling Scheme

A distributed algorithm A *verifies* ϕ if and only if:

- $G \models \phi \Rightarrow \exists\, c: V(G) \to \{0,1\}^*$: all nodes accept (G,c)
- $G \not\models \phi \Rightarrow \forall\, c: V(G) \to \{0,1\}^*$ at least one node rejects (G,c)

The bit-string $c(u)$ is called the *certificate* for u (cf. class NP)

Objective: Algorithms in $O(1)$ rounds (ideally, just 1 round in LOCAL)

Examples:

O(log n) bits

- Cycle-freeness: $c(u) = dist_G(u,r)$
- Spanning tree: $c(u) = (dist_G(u,r), ID(r))$

Measure of complexity: $\max_{u \in V(G)} |c(u)|$

Universal PLS

Theorem For any (decidable) graph property ϕ, there exists a PLS for ϕ, with certificates of size $O(n^2)$ bits in n-node graphs.

Proof $c(u) = (M,x)$ where

- M = adjacency matrix of G
- x = table[1..n] with $x(i)$ = ID(node with index i)

Verification algorithm:

1. check local consistency of M using x
2. if no inconsistencies, check whether M satisfies ϕ

G satisfies $\overset{\text{exercice}}{\Longleftrightarrow}$ both tests are passed ❑

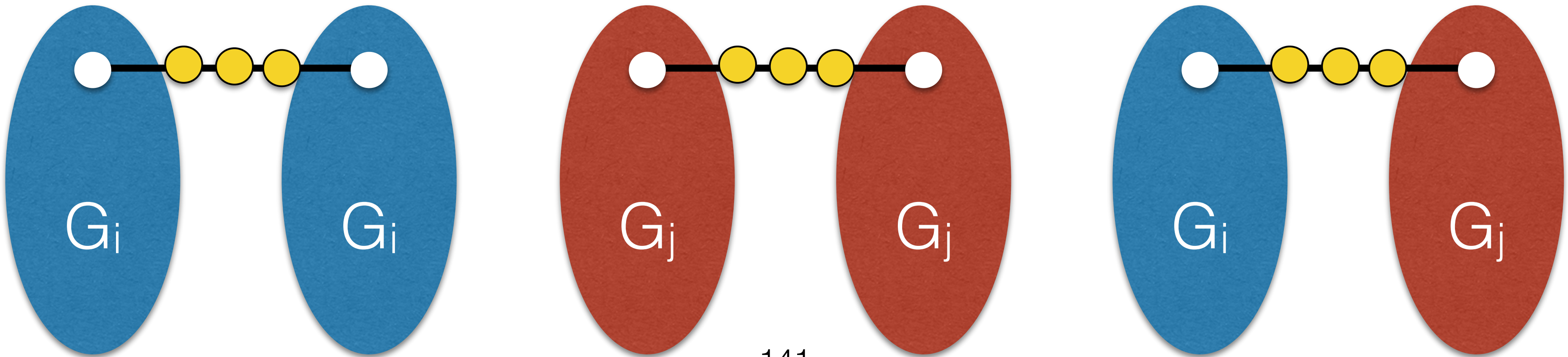
Lower bound

Theorem There exists a graph property for which any PLS has certificates of size $\Omega(n^2)$ bits.

Proof Graph automorphism = bijection $f:V(G) \rightarrow V(G)$ such that $\{u,v\} \in E(G) \iff \{f(u),f(v)\} \in E(G)$

Fact For n large enough, there are $\geq 2^{\varepsilon n^2}$ graphs with no non-trivial automorphism.

If certificates on $< \varepsilon n^2/3$ bits, then $\exists$ $i \neq j$ such that the three nodes have same certificates on G_i-G_i and G_j-G_j.



$\square$

Certifying Diameter

Given $k \geq 1$ certifying $\textsf{Diam}(G) = k$ requires certifying $\textsf{Diam}(G) \leq k$ and $\textsf{Diam}(G) \geq k$

Lemma 1. There exists a PLS for $\textsf{Diam}(G) \geq k$ with certificates on $O(\log n)$ bits.

Lemma 2. There exists a PLS for $\textsf{Diam}(G) \leq k$ with certificates on $\tilde{O}(n)$ bits.

Remark: Certifying $\textsf{Diam}(G) \leq k$ requires certificates on $\tilde{\Omega}(n)$ bits (cf. Réduction to DISJ)

PLS for $\text{Diam}(G) \geq k$

If $\text{Diam}(G) \geq k$ then there are two nodes u, v (identified by their IDs) at distance k

- Prover uses:
  - two trees T_u and T_v rooted at u, v, respectively, to certify the existence of these two nodes
  - a third tree T, which is a shortest path tree rooted at u with nodes labeled with distance to u

- Verifier at each node:
  - Checks consistency of T_u, T_v and T (exercice)

PLS for Diam$(G) \leq k$

- Prover gives to each node u:
  - Table D_u where $D_u[v] = \text{dist}(u, v)$

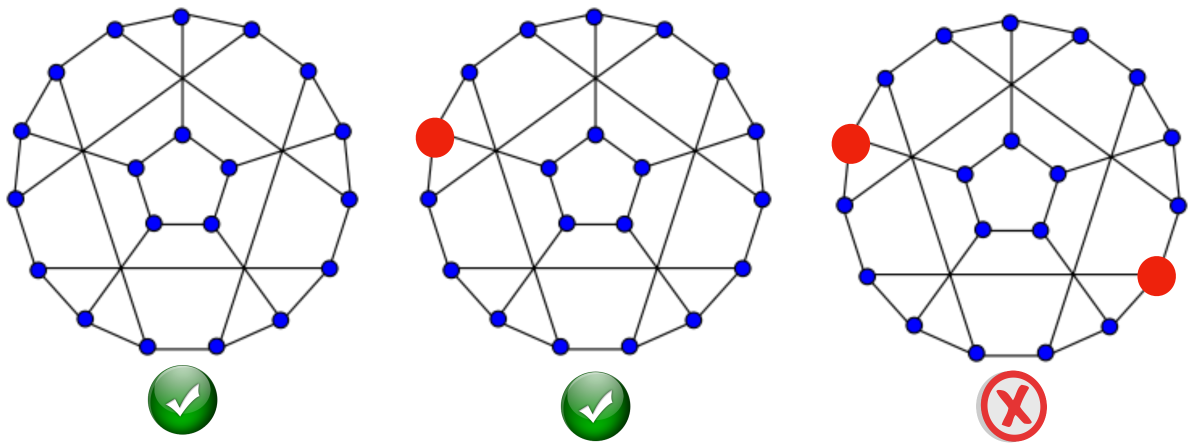
- Verifier at each node u checks:

  $D_u[u] = 0$, and, for every $v \in \{1, \ldots, n\} \setminus \{u\}$,
  - $D_u[v] \leq k$
  - $\exists u' \in N(u) : D_{u'}[v] = D_u[v] - 1$
  - $\forall u' \in N(u) : D_{u'}[v] \geq D_u[v] - 1$

Interactive Proofs

Randomized Protocols

- At most one selected (AMOS)

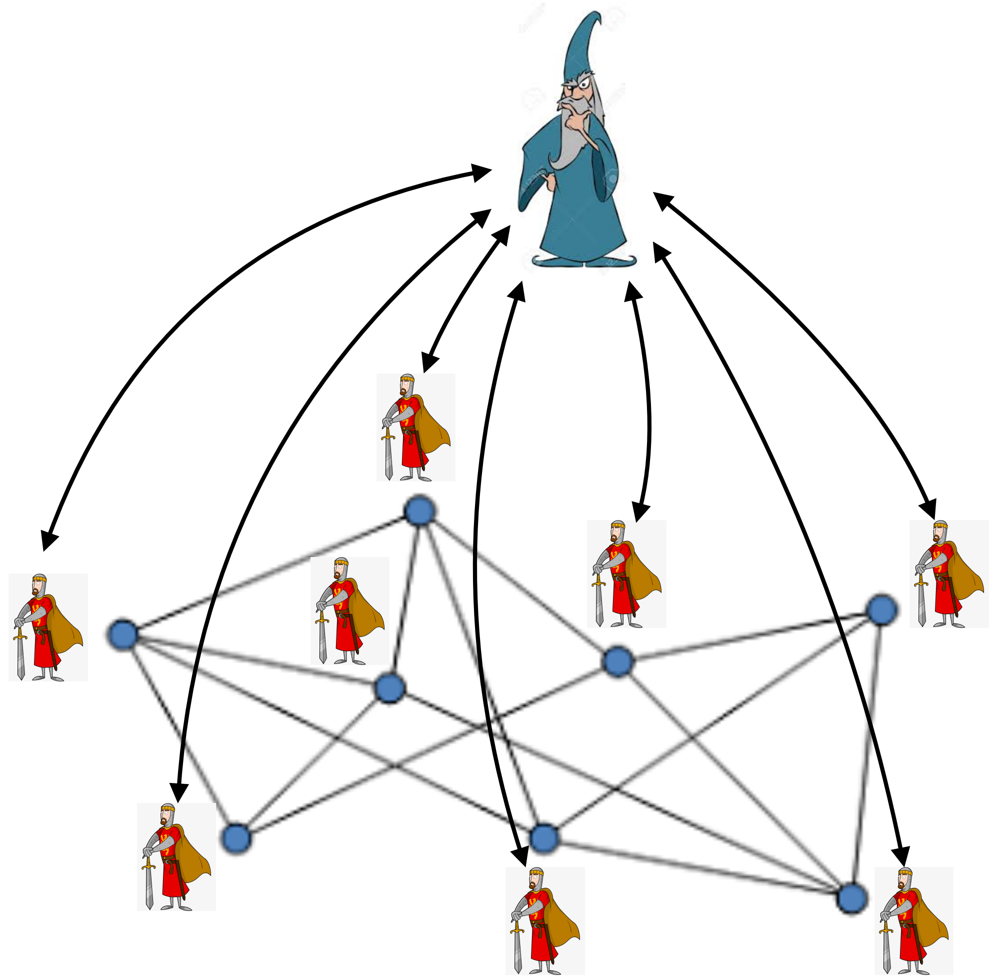


- Decision algorithm (2-sided):
  - let p = $(\sqrt{5}-1)/2$ = 0.61…
  - If not selected then accept
  - If selected then accept w/ prob p, and reject w/ prob 1-p

- Issue with boosting! — But OK for 1-sided error

Distributed Interactive Protocols

[KOS, 2018]

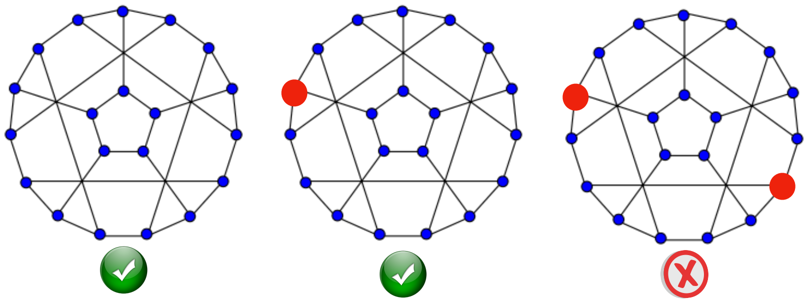


- Arthur-Merlin Phase *(no communication, only interactions)*
- Verification Phase *(only communications)*

- Merlin has infinite computation power
- Arthur is randomized

- k = #interactions
- dAM[k] or dMA[k]
- dAM = dAM[2]
- dAMA = dMA[3]

Example: AMOS

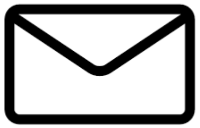


- Locally checkable with success probability $(\sqrt{5}-1)/2$

- In dAM(r) with r random bits, and success prob $1 - 1/2^r$
  - Arthur independently picks a r-bit index x_u at each node u, u.a.r.
  - Merlin answer $y = \perp$ if no nodes selected, or the index $y = x_v$ of the selected node v
  - Verifier checks with neighbors that all nodes get same value y from Merlin, and selected nodes v checks that $y = x_v$.

Parameters

- Number of interactions between  and 
- Size of 

- Size of 
- Number of random 
- Shared vs distributed 

Sequential setting

- For every $k \geq 2$, $\mathsf{AM}[k] = \mathsf{AM}$
- $\mathsf{MA} \subseteq \mathsf{AM}$ because $\mathsf{MA} \subseteq \mathsf{MAM} = \mathsf{AM}[3] = \mathsf{AM}$
- $\mathsf{MA} \in \Sigma_2\mathsf{P} \cap \Pi_2\mathsf{P}$
- $\mathsf{AM} \in \Pi_2\mathsf{P}$
- $\mathsf{AM}[\mathsf{poly}(n)] = \mathsf{IP} = \mathsf{PSPACE}$

Distributed Setting

[KOS 2018, NPY 2018]

- Sym $\in$ dAM($n \log n$)
- Sym $\in$ dMAM($\log n$)
- Any dAM protocol for Sym requires $\Omega(\log\log n)$-bit certificates
- $\neg$Sym $\in$ dAMAM($\log n$)

Example: Set Equality

- Every node u is given $a_u, b_u \in \{1,\ldots,n\}$
- Let $A = \{a_u : u \in V(G)\}$
- Let $B = \{b_u : u \in V(G)\}$
- Legality: $A = B$ as multisets (i.e., with repetitions)

Theorem SET-EQ is in dAM$(O(\log n))$

Proof

Let q be prime, with $3n < q < 6n$

Let us consider two polynomials in $\mathbb{F}_q$:

$P_A(X) = \prod_{u \in V(G)} (X - a_u)$ and $P_B(X) = \prod_{u \in V(G)} (X - b_u)$

Note also that $P(X) = P_A(X) - P_B(X)$ is of degree n, and thus has at most n roots in $\mathbb{F}_q$

In particular: $A = B \iff P_A(X) = P_B(X)$

Proof (continued)

- Every node u picks rand$(u) \in \mathbb{F}_q$ u.a.r. and sends it to Merlin
- Merlin sends to all nodes:
  - node r with smallest ID, with a spanning tree T rooted at r
  - the value $x =$ rand(r)
  - the value $P_A^u(x) = \prod_{v \in V(T_u)} (x - a_v)$
  - the value $P_B^u(x) = \prod_{v \in V(T_u)} (x - b_v)$
- Arthur checks consistency with neighbors at every node
- Root r checks that $P_A^r(x) = P_B^r(x)$

Proof (end)

- Completeness is satisfied with probability 1
- Soundness: if $A \neq B$ then $P_A(X) \neq P_B(X)$

  if all tests are passed, then $P_A(x) = P_B(x)$

  Since $x \in \mathbb{F}_q$ is random, $P_A(x) = P_B(x)$ occurs with probability $\leq n/q < \dfrac{1}{3}$ ❑

End Lecture 7


Final exam
Feb 28, 10h30-12h00