Causal Limits of Distributed Computation

Francesco d'Amore

Based on joint works with A. Akbari, X. Coiteux-Roy, R. Gajjala, F. Kuhn, F. Le Gall, H. Lievonen, D. Melnyk, A. Modanese, S. Pai, M. Renou, V. Rozhoň, G. Schmid, and J. Suomela.

ADGA - DISC
28 October 2024

MUR FARE 2020 - Project PAReCoDi

Table of content

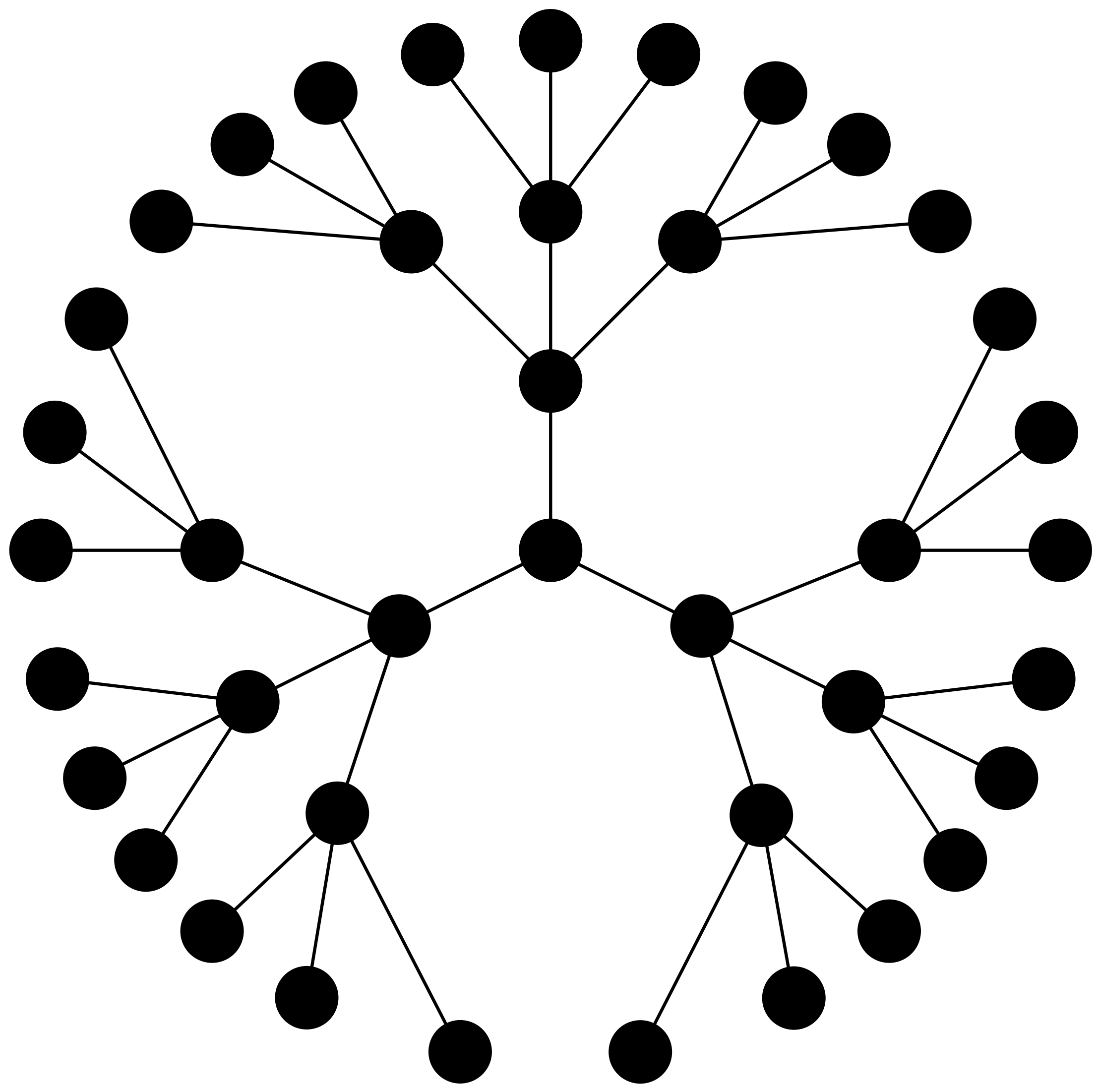
Table of content

The LOCAL model

[Linial, FOCS '87 & SICOMP '92]

- **Distributed network** of n processors/nodes

  - graph $G = (V, E)$ with $|V| = n$
  - E: communication links
  - each node in V runs the same algorithm



The LOCAL model

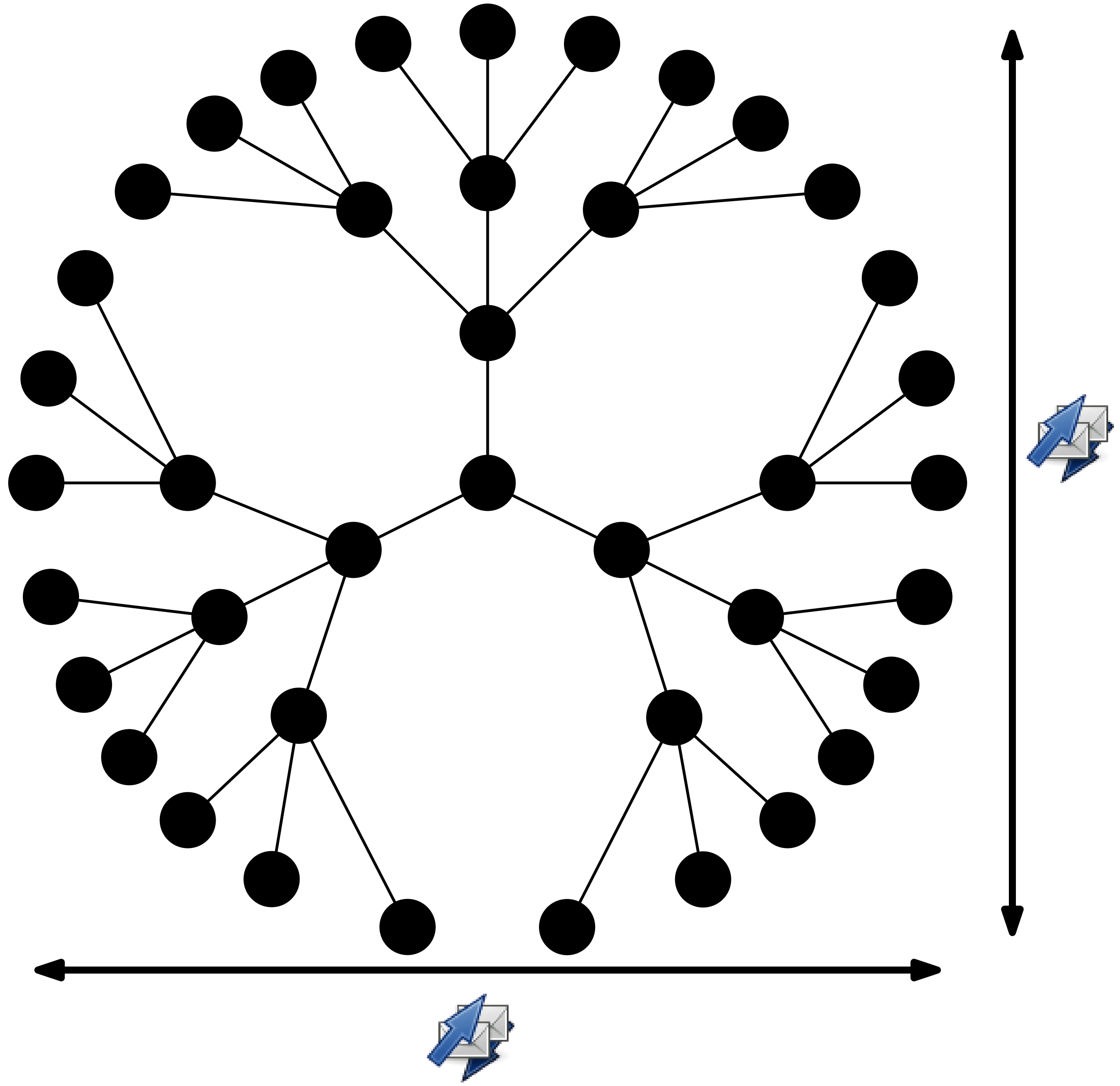
[Linial, FOCS ’87 & SICOMP ’92]

- **Distributed network** of n processors/nodes

  - graph $G = (V, E)$ with $|V| = n$
  - E: communication links
  - each node in V runs the same algorithm

- **Time is synchronous**: nodes alternate

  - arbitrary local computation & update of state variables
  - sending of messages to all neighbors
    * no bandwidth constraints



The LOCAL model

[Linial, FOCS '87 & SICOMP '92]

- **Distributed network** of n processors/nodes

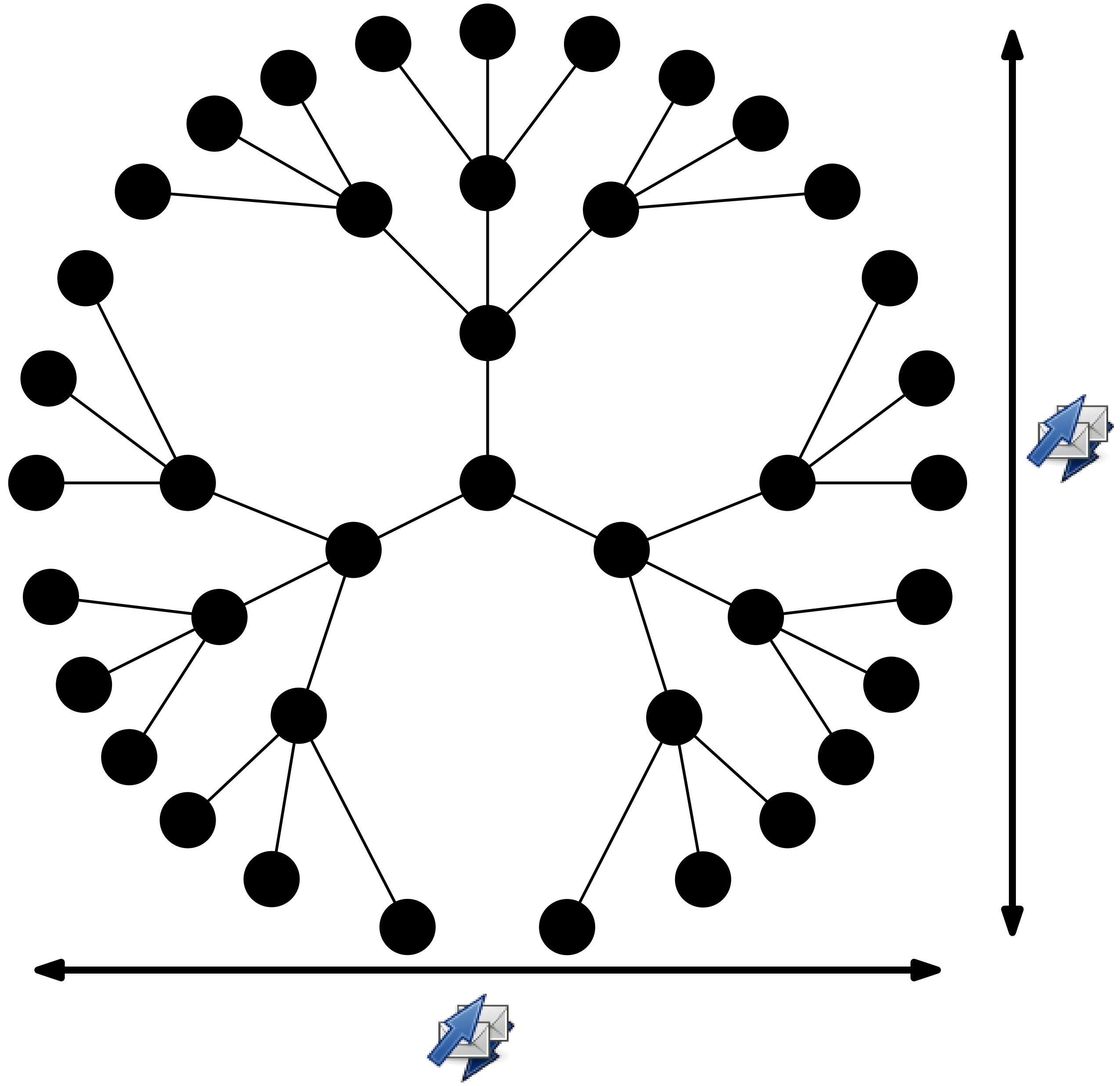
  - graph $G = (V, E)$ with $|V| = n$
  - E: communication links
  - each node in V runs the same algorithm

- **Time is synchronous**: nodes alternate

  - arbitrary local computation & update of state variables
  - sending of messages to all neighbors
    * no bandwidth constraints

- **Unique identifier** to nodes in the set $1, \ldots, \text{poly}(n)$
  * adversarially chosen

  - needed to solve even basic problems (2-coloring a 2-path)



The LOCAL model

[Linial, FOCS '87 & SICOMP '92]

- **Distributed network** of n processors/nodes

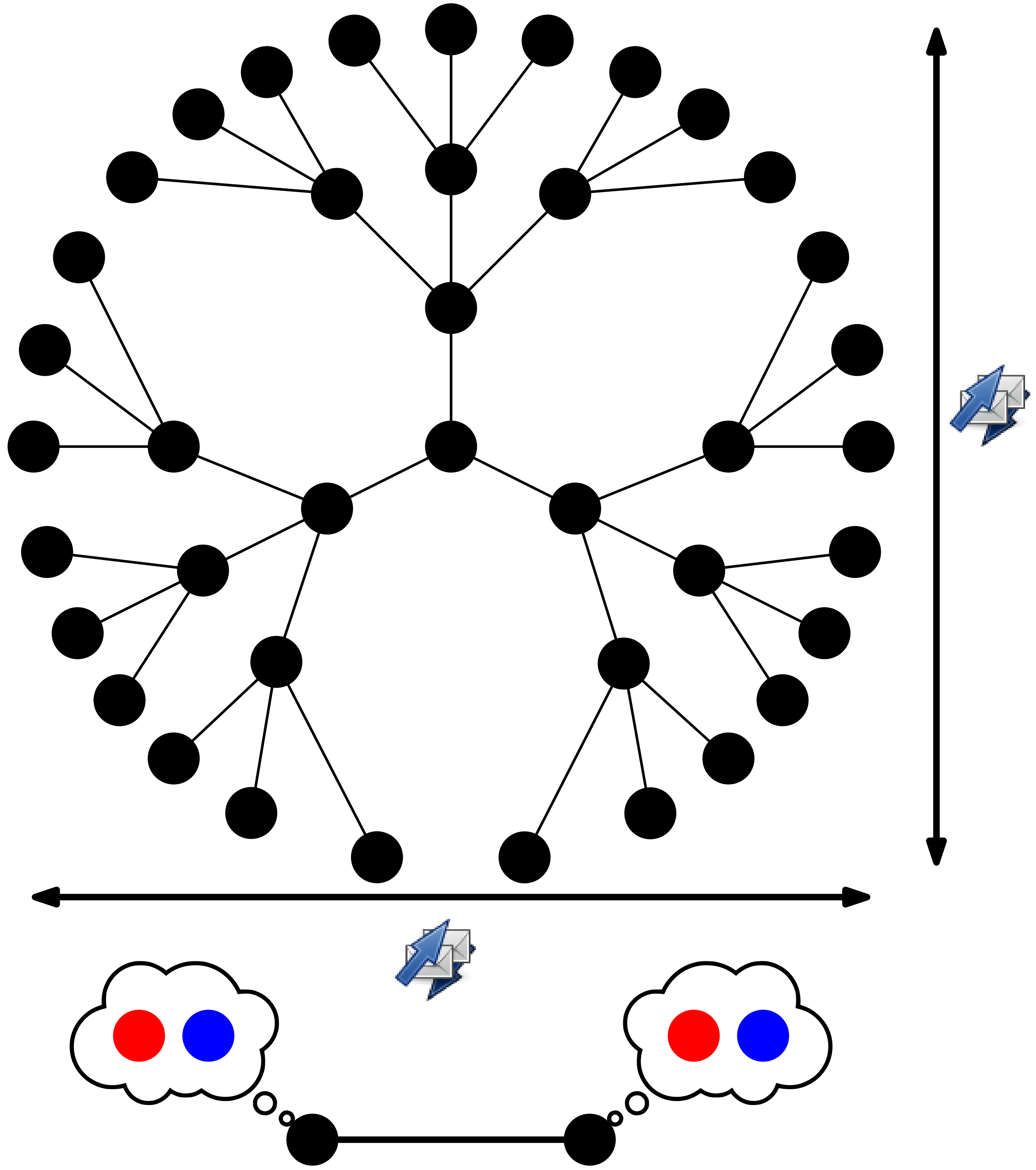
  - graph $G = (V, E)$ with $|V| = n$
  - E: communication links
  - each node in V runs the same algorithm

- **Time is synchronous**: nodes alternate

  - arbitrary local computation & update of state variables
  - sending of messages to all neighbors
    * no bandwidth constraints

- **Unique identifier** to nodes in the set $1, \ldots, \text{poly}(n)$
  * adversarially chosen

  - needed to solve even basic problems (2-coloring a 2-path)



The LOCAL model

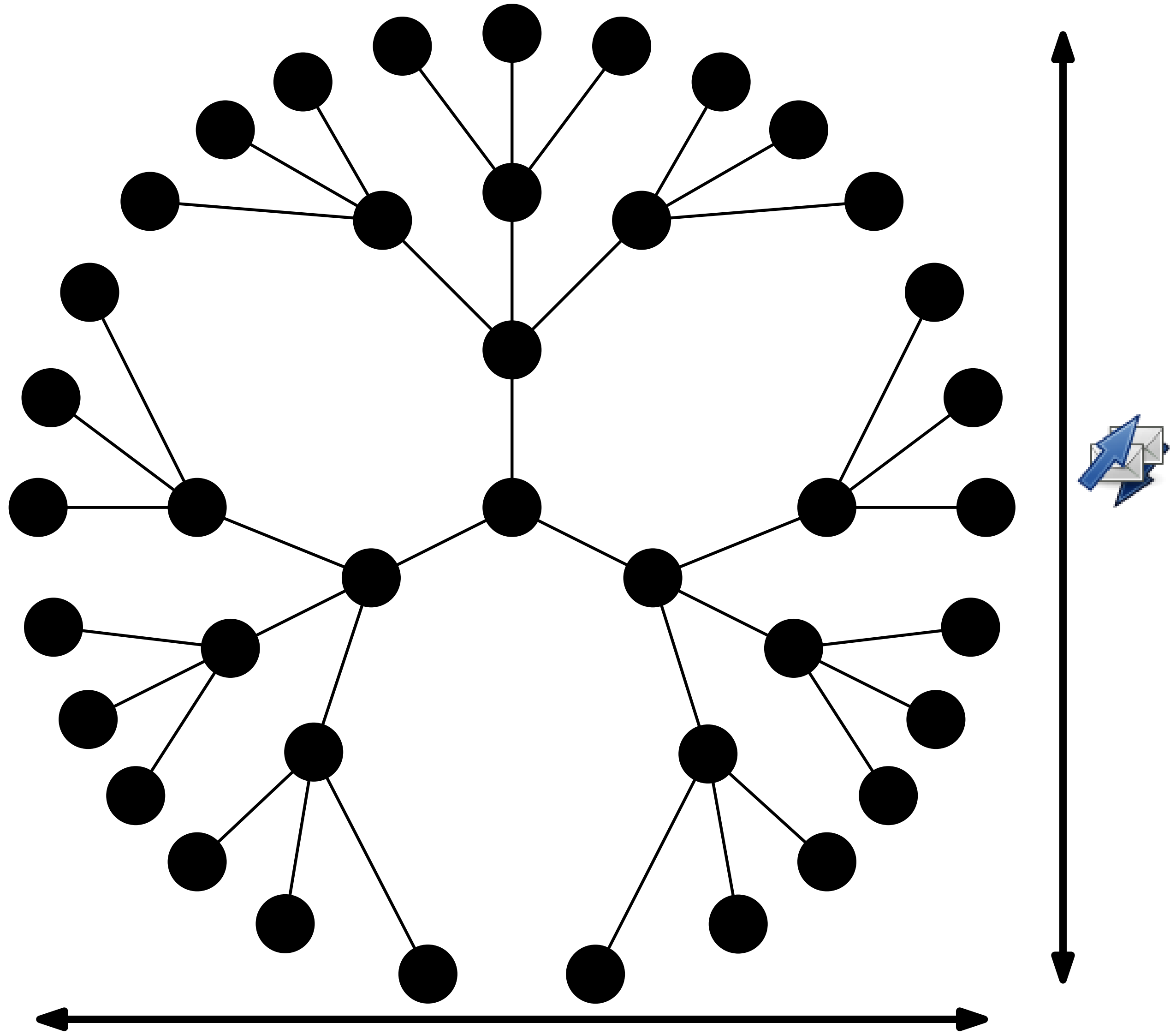
[Linial, FOCS '87 & SICOMP '92]

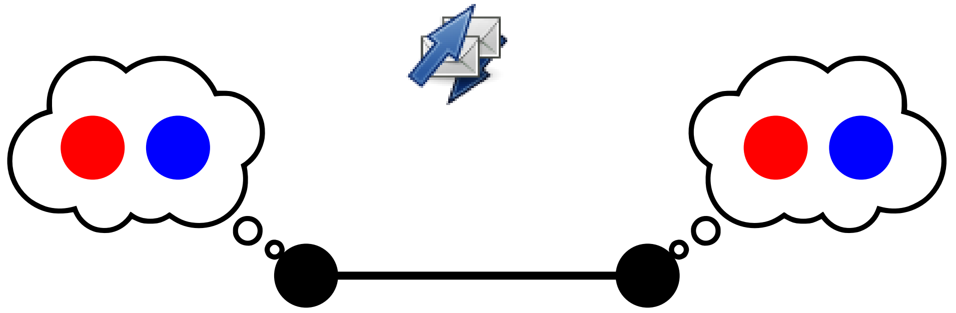
- **Distributed network** of n processors/nodes
  - graph $G = (V, E)$ with $|V| = n$
  - E: communication links
  - each node in V runs the same algorithm

- **Time is synchronous**: nodes alternate
  - arbitrary local computation & update of state variables
  - sending of messages to all neighbors
    * no bandwidth constraints

- **Unique identifier** to nodes in the set $1, \ldots, \text{poly}(n)$
  * adversarially chosen
  - needed to solve even basic problems (2-coloring a 2-path)

- **Complexity measure**: number of communication rounds

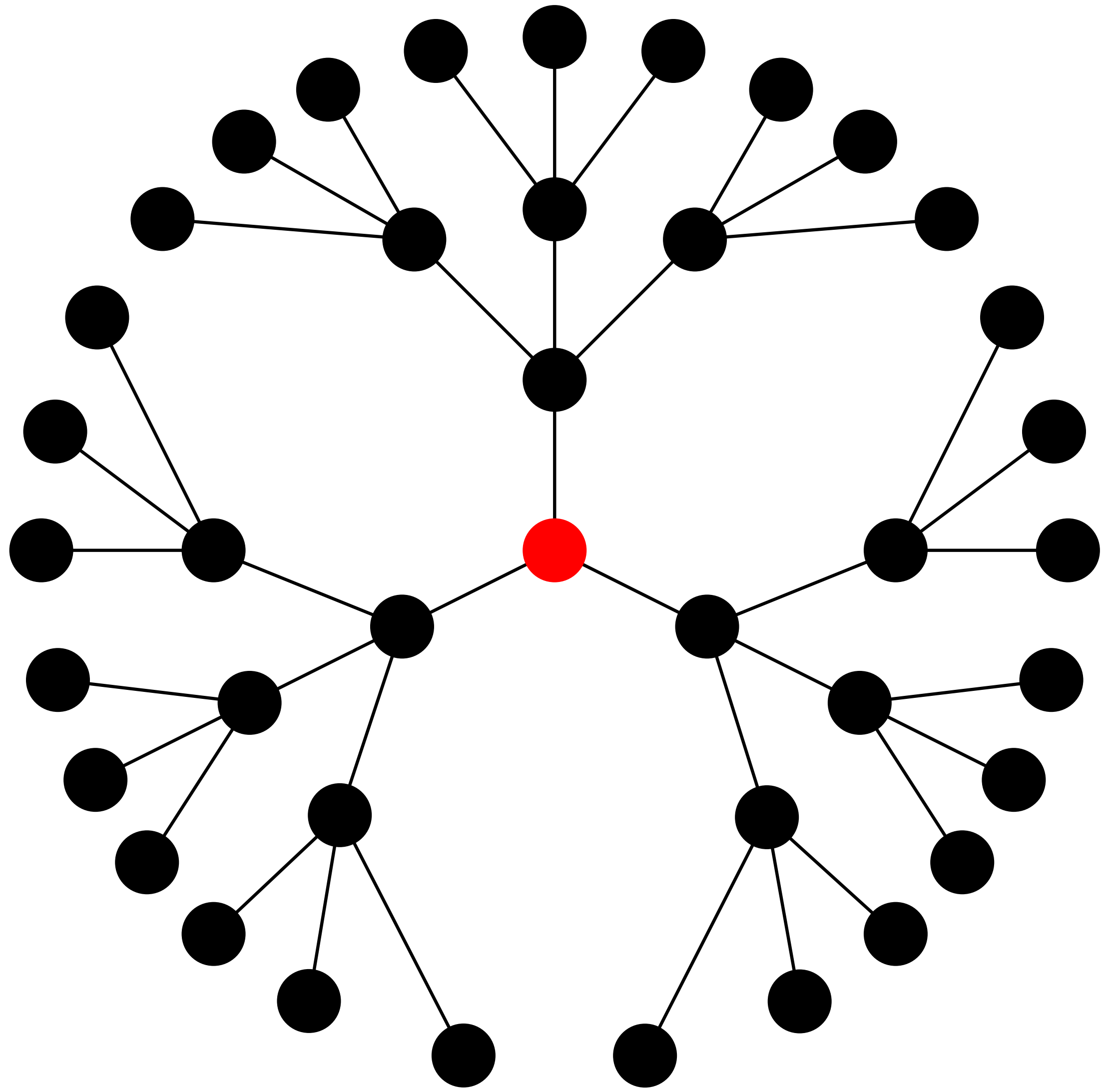




- **Complexity measure**: number of communication rounds

Equivalence:

* n is known to the nodes

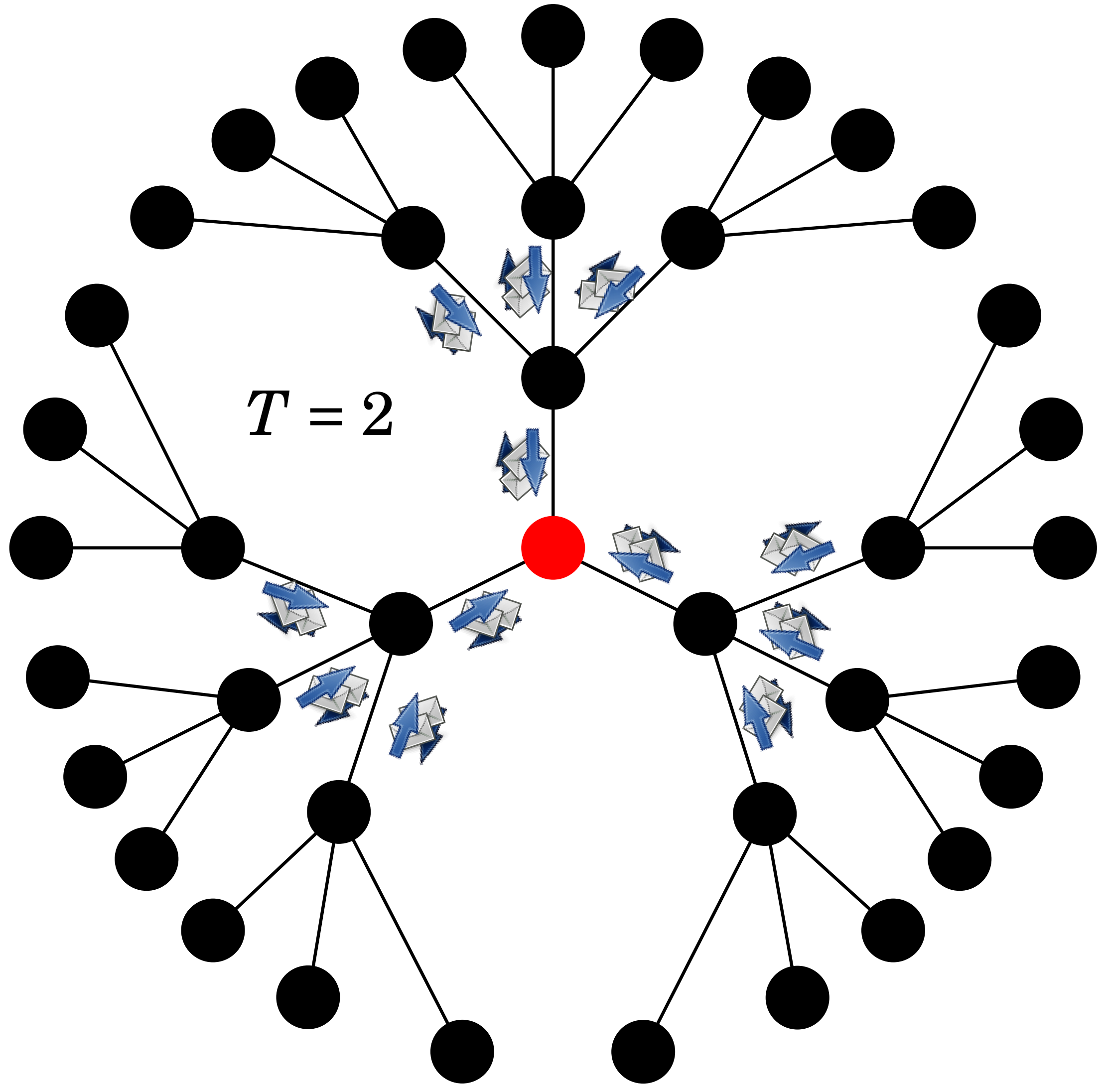


- **Complexity measure**: number of communication rounds

Equivalence:

* n is known to the nodes

- let A be a $T(n)$-round LOCAL algorithm



Locality

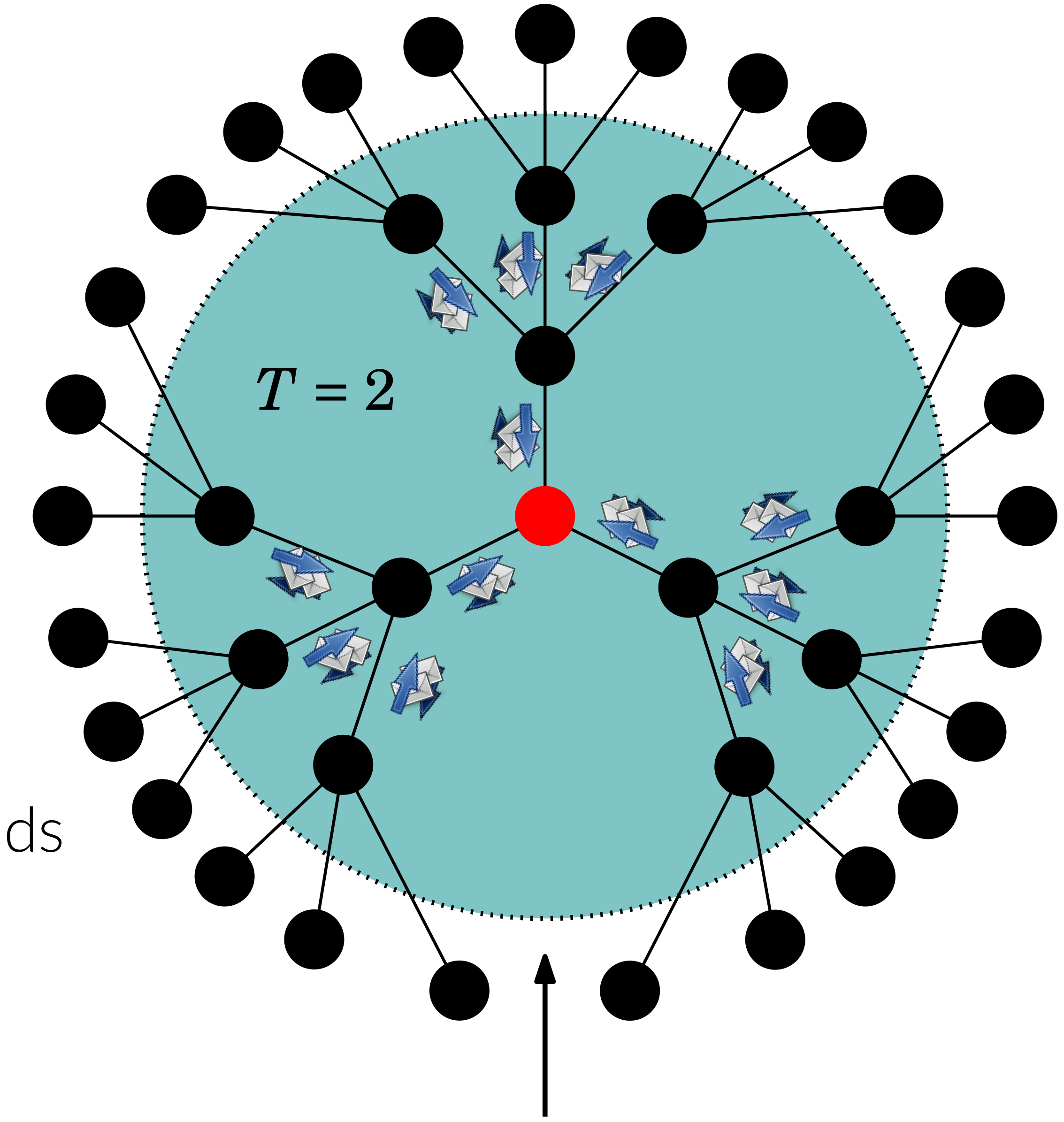
- **Complexity measure**: number of communication rounds

Equivalence:

* n is known to the nodes

- let A be a $T(n)$-round LOCAL algorithm

  - design algo B where each node
    - performs no communication at all
    - gathers the full input in its radius-$T(n)$ neighborhood
    - simulates A by locally simulating the $T(n)$ communications rounds



knowledge after $\mathbf{2}$ rounds of communication

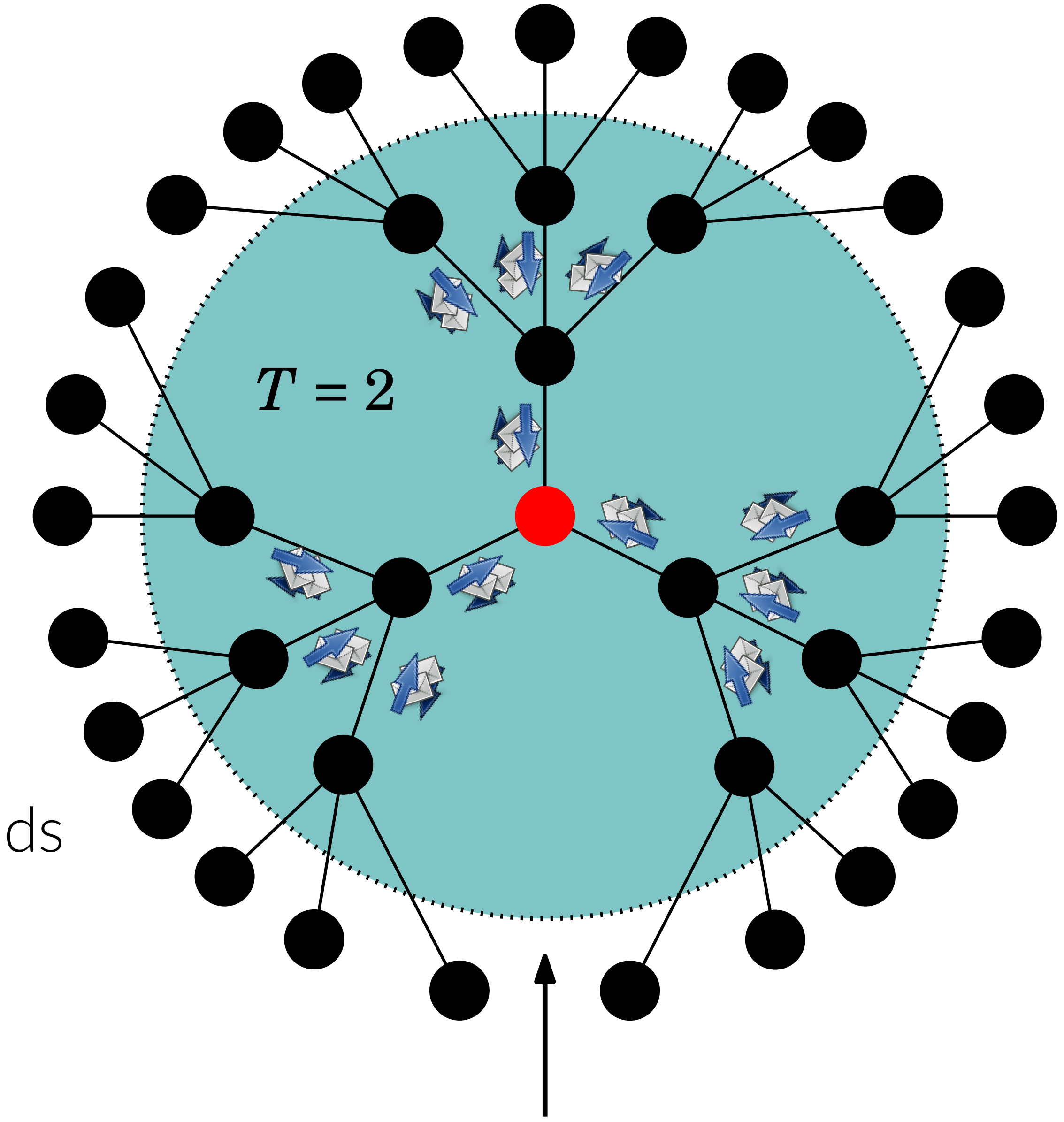
- **Complexity measure**: number of communication rounds

Equivalence:

* n is known to the nodes

- let A be a $T(n)$-round LOCAL algorithm
  - design algo B where each node
    - performs no communication at all
    - gathers the full input in its radius-$T(n)$ neighborhood
    - simulates A by locally simulating the $T(n)$ communications rounds

- let B be a **"gathering"** algorithm with locality $T(n)$
  - design algo A where each node
    - performs $T(n)$ communication rounds
    - outputs simulating B



knowledge after 2 rounds of communication

Locality

- **Complexity measure**: number of communication rounds

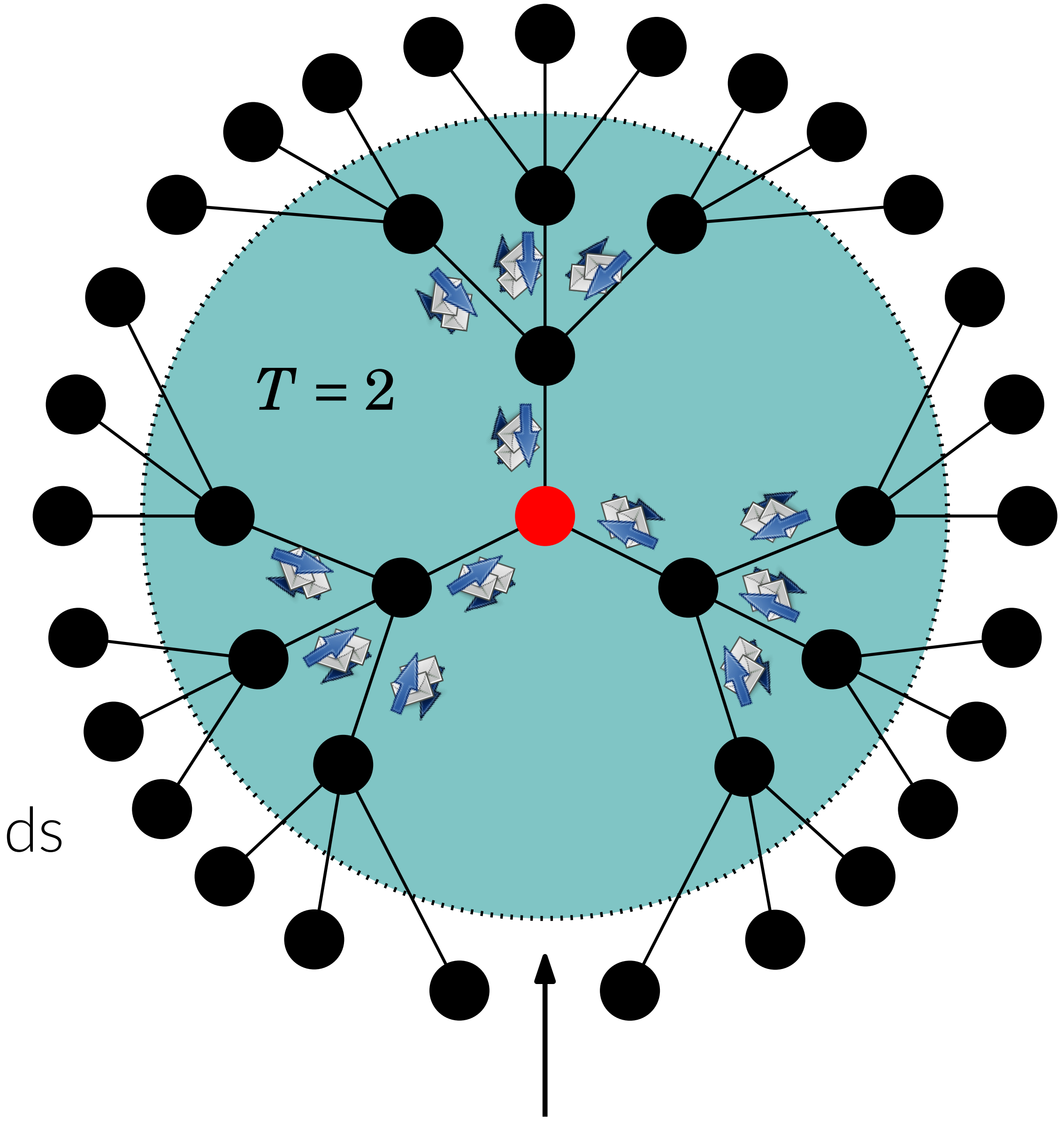
Equivalence:

* n is known to the nodes

- let A be a $T(n)$-round LOCAL algorithm
  - design algo B where each node
    - performs no communication at all
    - gathers the full input in its radius-$T(n)$ neighborhood
    - simulates A by locally simulating the $T(n)$ communications rounds

- let B be a **"gathering"** algorithm with locality $T(n)$
  - design algo A where each node
    - performs $T(n)$ communication rounds
    - outputs simulating B

- Locality $T = \text{diam}(G) + 1$ is always sufficient to solve any problem

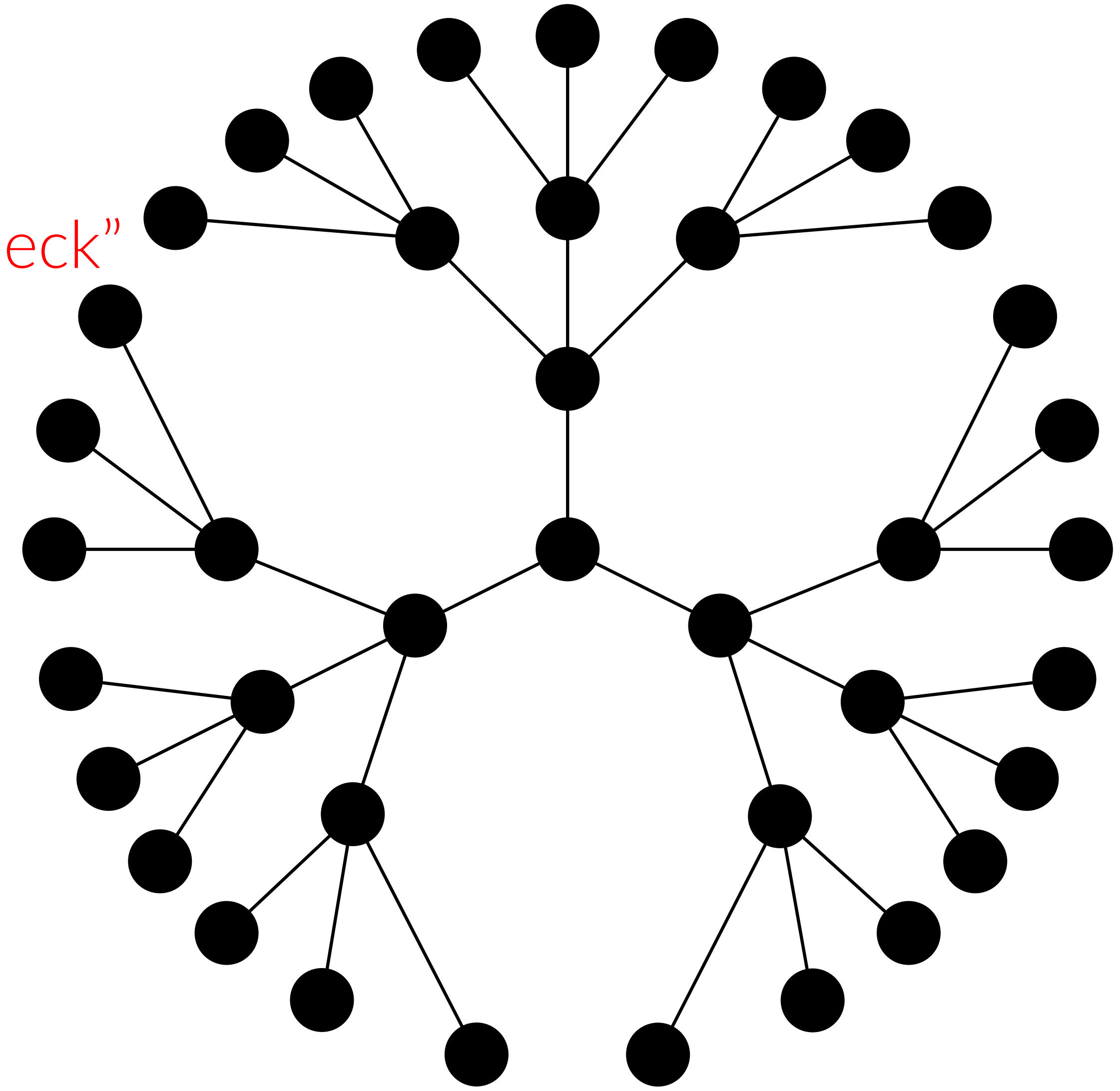


knowledge after 2 rounds of communication

Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC '93 & SICOMP '95]

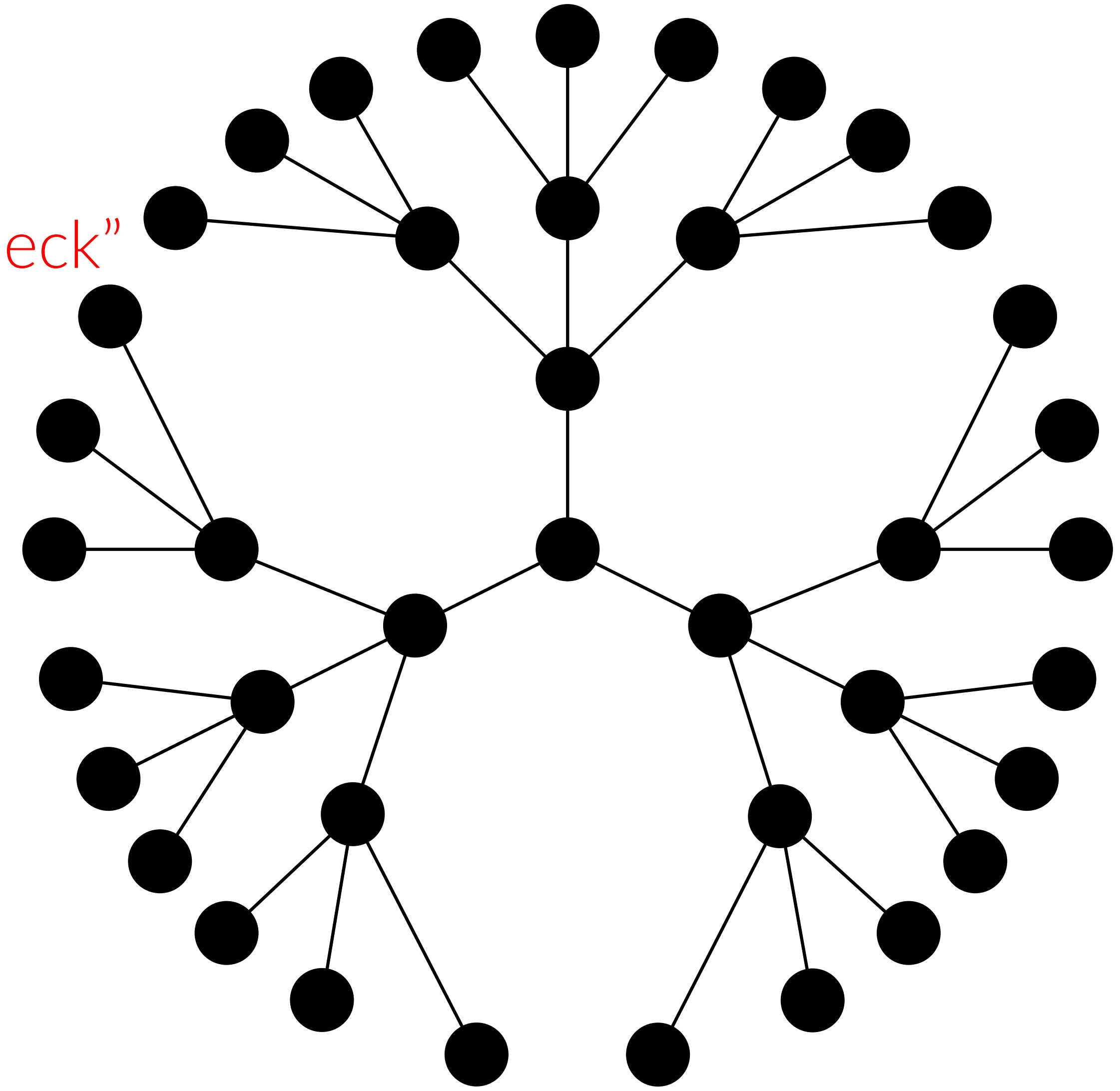
- **Problems** whose solutions might be "hard to find" but are "easy to check"
  - "analogue" of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.



Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC '93 & SICOMP '95]

- **Problems** whose solutions might be "hard to find" but are "easy to check"
  - "analogue" of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.



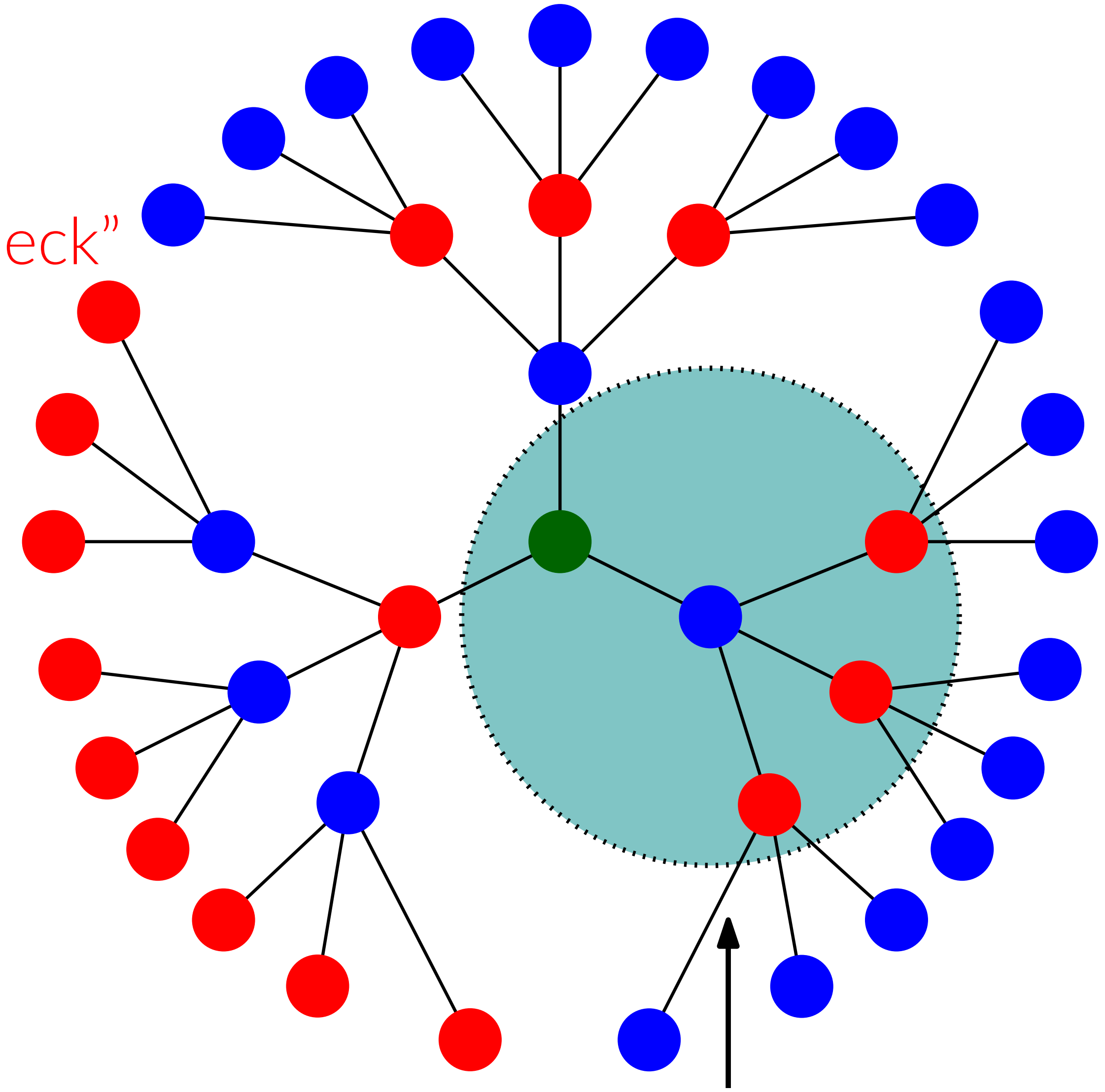
- **"Easy to check"**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy

Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC ’93 & SICOMP ’95]

- **Problems** whose solutions might be “hard to find” but are “easy to check”
  - “analogue” of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.

- **“Easy to check”**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy



3-**coloring**: the blue node checks if its color is different from those of its neighbors

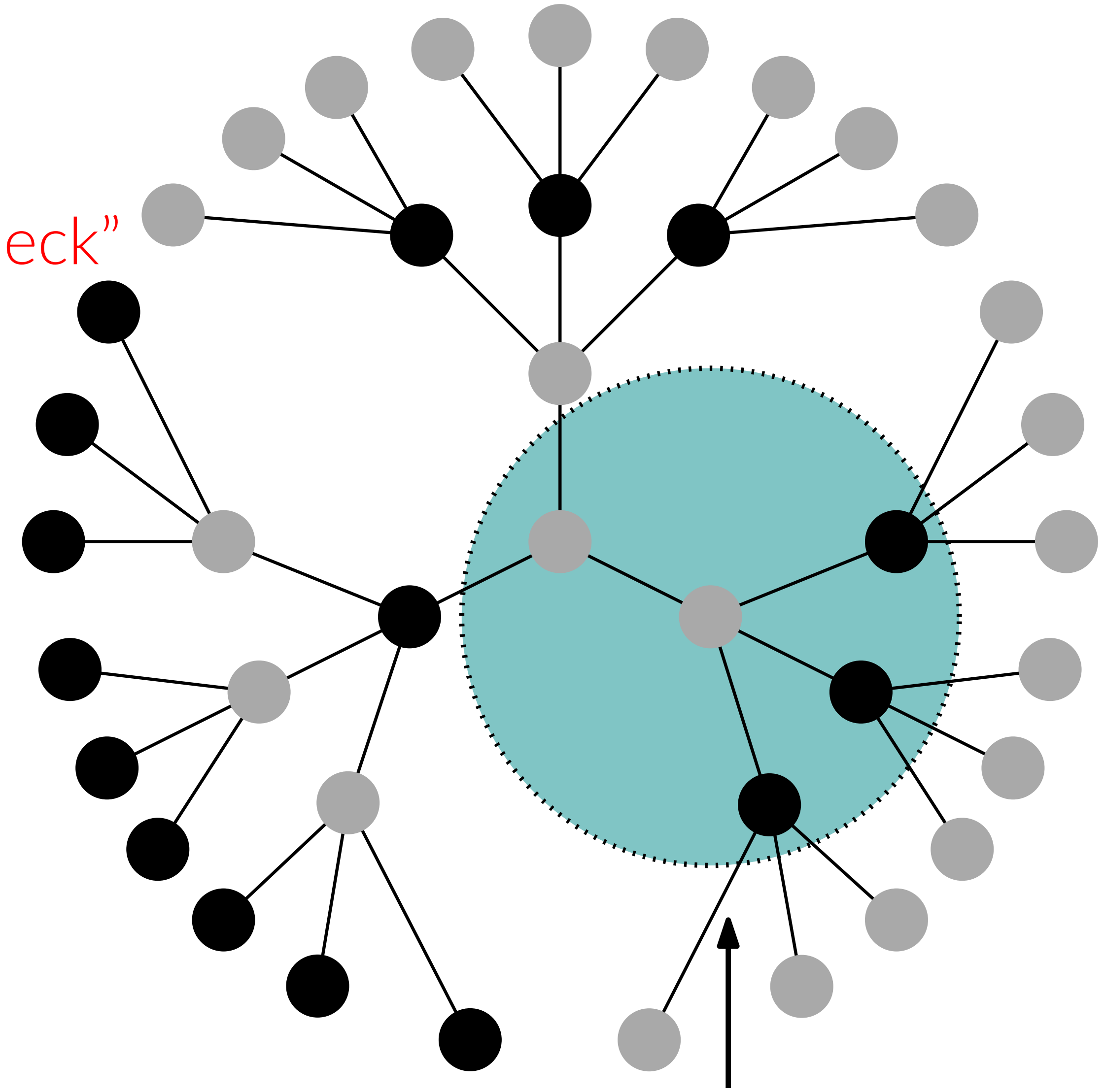
valid LCL

Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC '93 & SICOMP '95]

- **Problems** whose solutions might be "hard to find" but are "easy to check"
  - "analogue" of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.

- **"Easy to check"**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy



MIS: each node checks if it is in the IS or if it has a neighbor in the IS

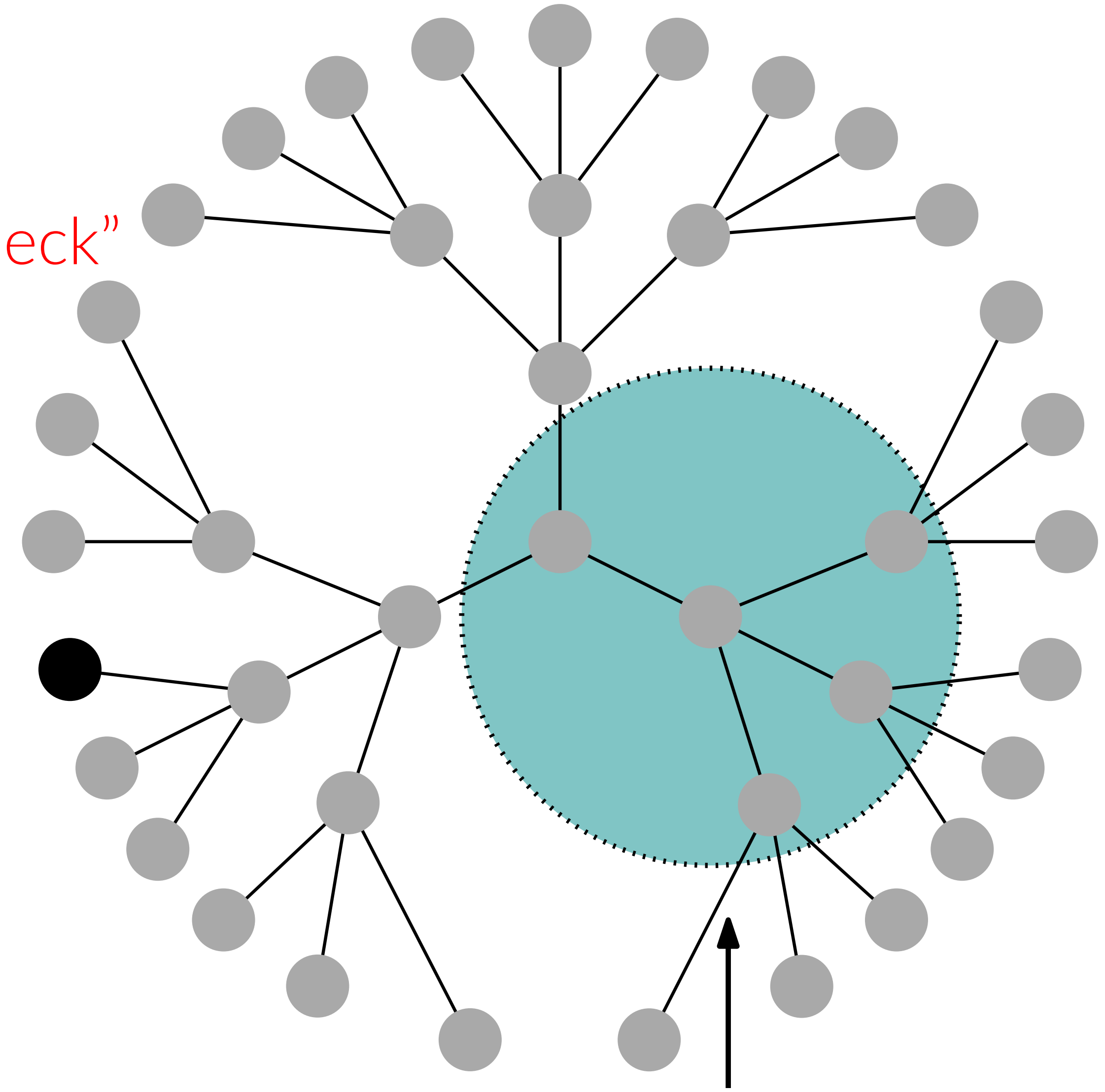
valid LCL

Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC '93 & SICOMP '95]

- **Problems** whose solutions might be "hard to find" but are "easy to check"
  - "analogue" of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.

- **"Easy to check"**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy



Leader election: the checking radius should be $r = \text{diam}(G)$

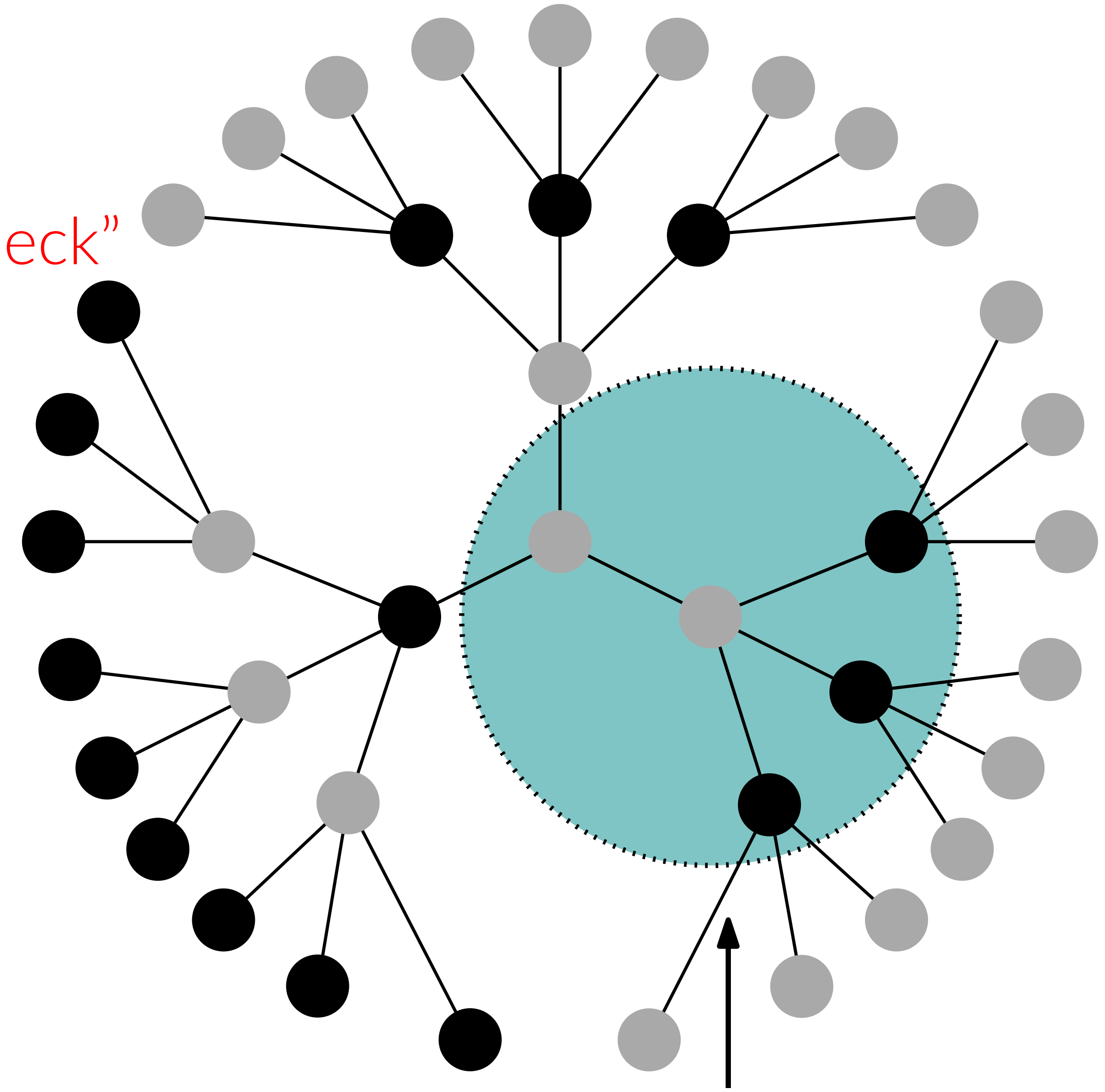
not an LCL

Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC '93 & SICOMP '95]

- **Problems** whose solutions might be "hard to find" but are "easy to check"
  - "analogue" of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.

- **"Easy to check"**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy



MIS: each node checks if it is in the IS or if it has a neighbor in the IS

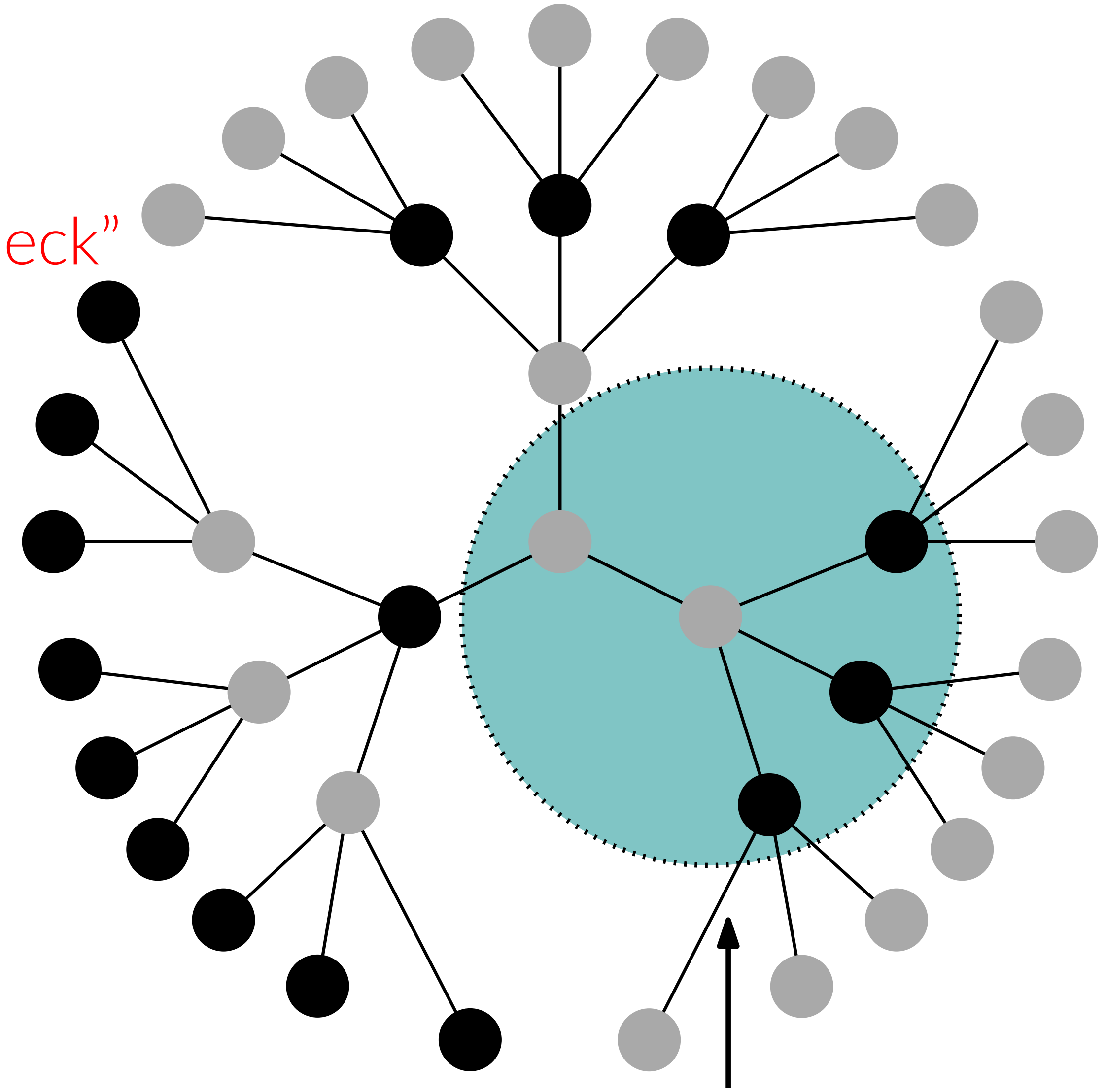
Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC '93 & SICOMP '95]

- **Problems** whose solutions might be "hard to find" but are "easy to check"
  - "analogue" of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.

- **"Easy to check"**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy

- A **lot of literature** studying LCLs:
  - classification of LCLs based on complexity (locality)
  - e.g.: complexity $T(n)$ in randomized-LOCAL $\implies O(T(2^{n^2}))$ in deterministic-LOCAL [Chang et al., SICOMP '19]



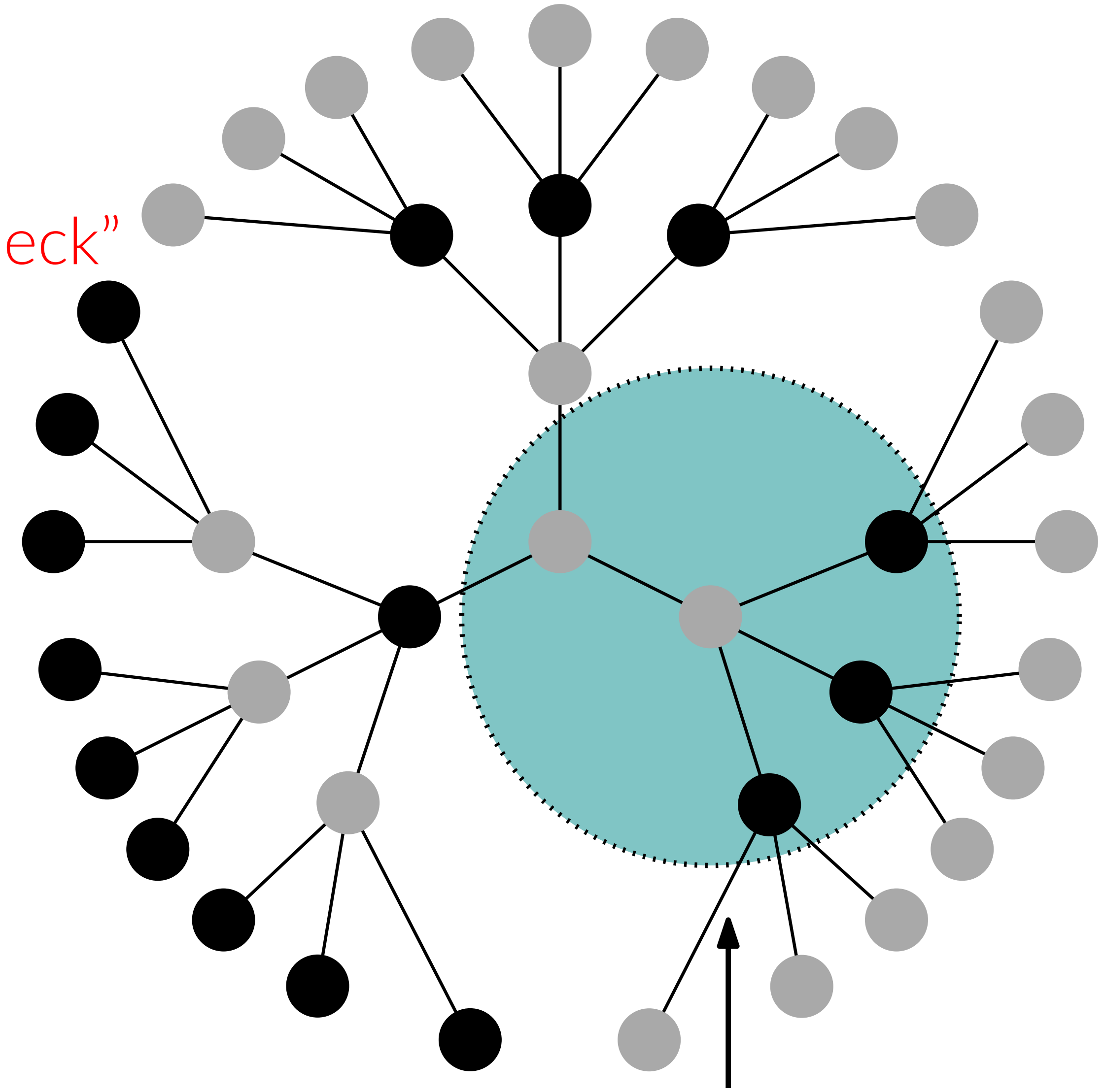
MIS: each node checks if it is in the IS or if it has a neighbor in the IS

Locally checkable labeling (LCL) problems

[Naor and Stockmeyer, STOC ’93 & SICOMP ’95]

- **Problems** whose solutions might be “hard to find” but are “easy to check”
  - “analogue” of NP in the distributed setting
  - coloring, maximal independent set, maximal matching, etc.

- **“Easy to check”**
  - radius $r = \Theta(1)$
  - each node can check its solution within its radius-r neighborhood
  - a globally valid iff each node is locally happy



MIS: each node checks if it is in the IS or if it has a neighbor in the IS

- A **lot of literature** studying LCLs:
  - classification of LCLs based on complexity (locality)
  - e.g.: complexity $T(n)$ in randomized-LOCAL $\implies O(T(2^{n^2}))$ in deterministic-LOCAL [Chang et al., SICOMP ’19]
  - [BFHKLRSU STOC ’16; BHKLOPRSU PODC’17; GKM STOC ’17; GHK FOCS ’18; CP SICOMP ’19; BHKLOS STOC ’18; BBCORS PODC ’19; BBOS PODC ’20; BBHORS JACM ’21; BBCOSS DISC ’22; AELMSS ICALP ’23; etc.]

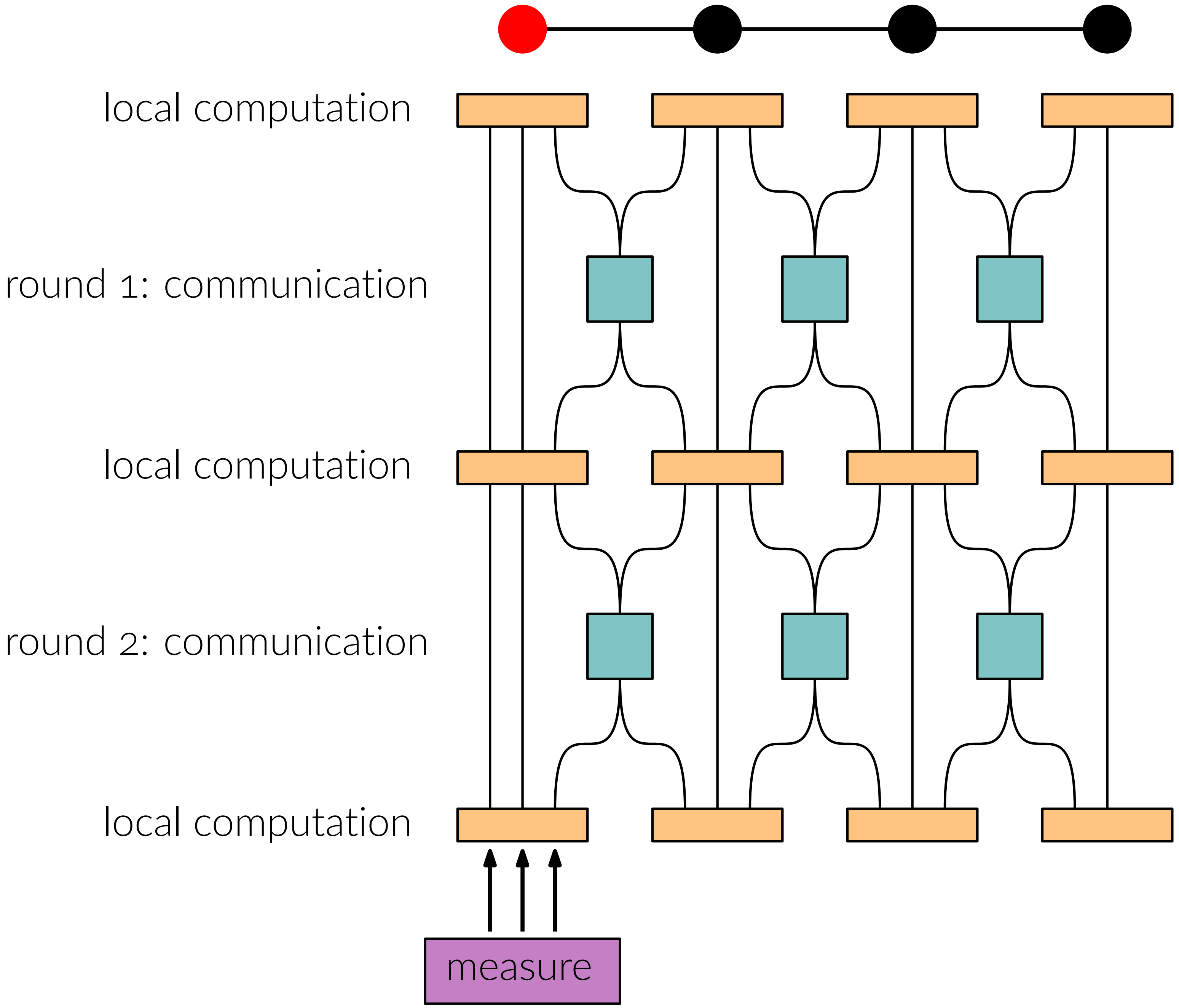
Table of content

1. **Intro**
   - The LOCAL model of computation
   - Locally checkable labeling (LCL) problems
2. **Quantum and causality-based models**
   - The non-signaling model & bounded-dependence model
   - State-of-the-art lower bounds & upper bounds
3. **Locality-based models**
   - The online-LOCAL model
   - Relation with causality-based models
   - Simulation in weaker models
4. **Conclusions and open problems**

Quantum-LOCAL

[Gavoille et al., DISC '09]

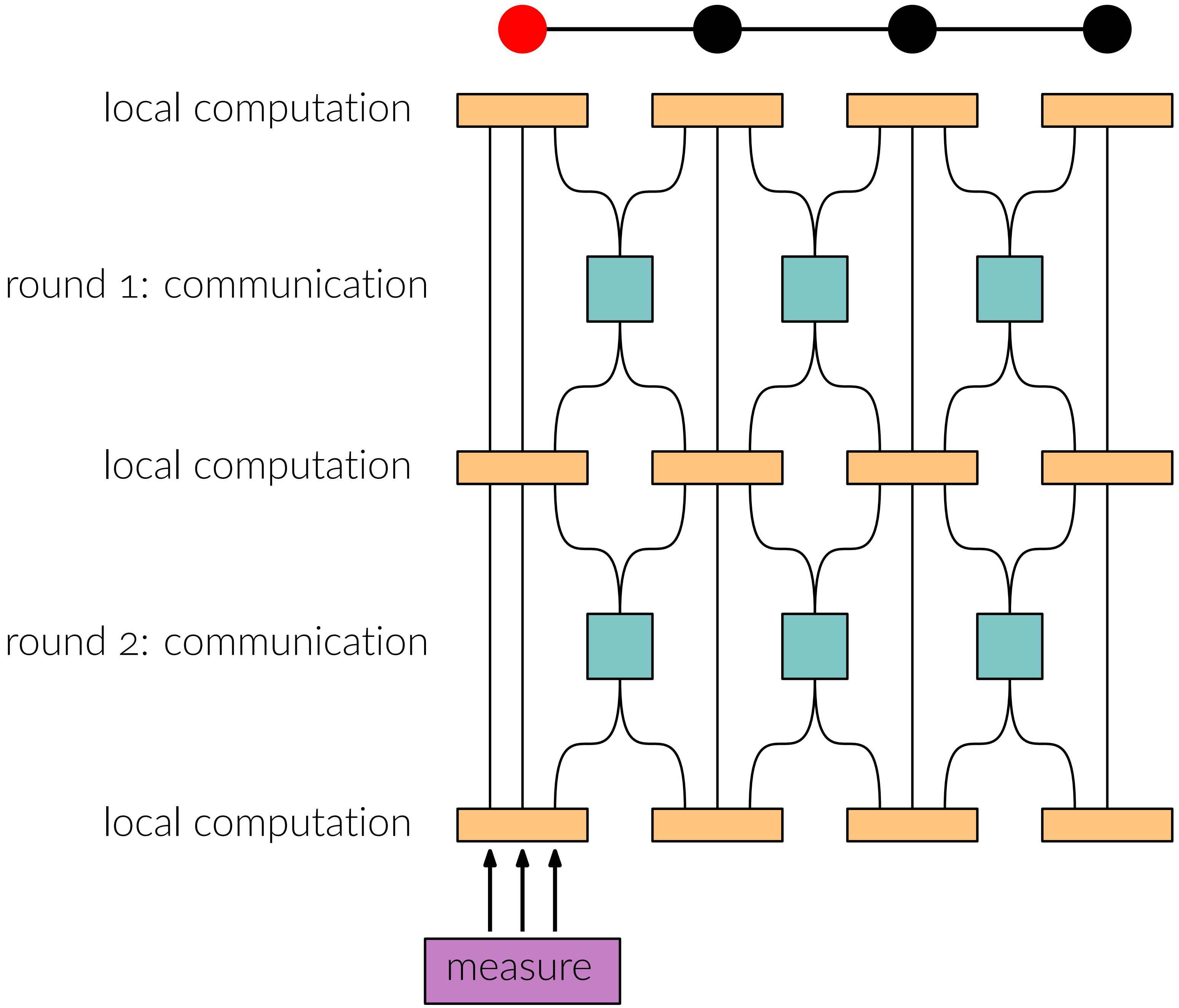
- **Distributed system** of n quantum processors/nodes
  - quantum computation
  - quantum communication (qubits)
  - output: measurement of qubits



Quantum-LOCAL

[Gavoille et al., DISC '09]

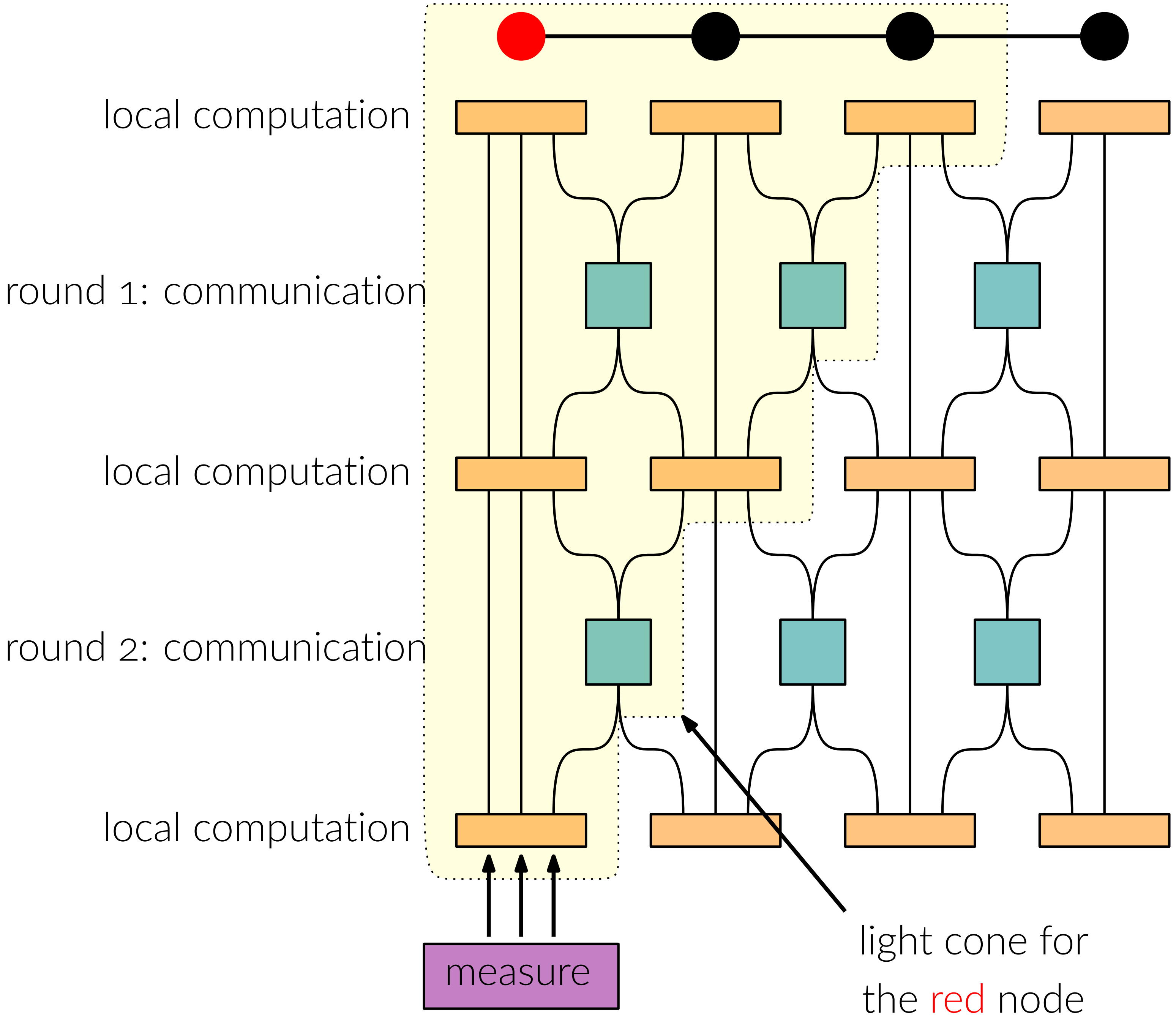
- **Distributed system** of n quantum processors/nodes
  - quantum computation
  - quantum communication (qubits)
  - output: measurement of qubits
- **Complexity measure**: number of communication rounds



Quantum-LOCAL

[Gavoille et al., DISC '09]

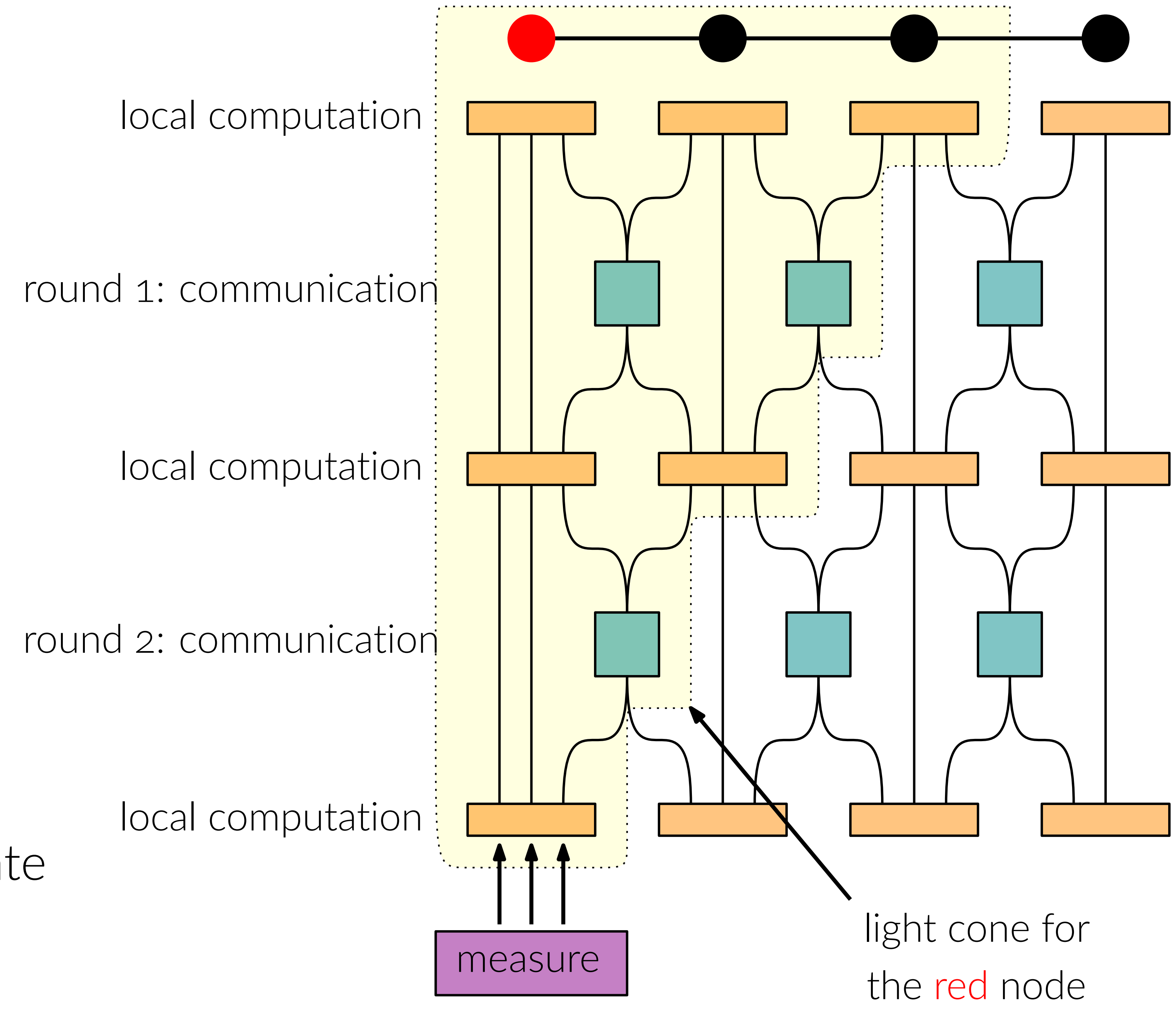
- **Distributed system** of n quantum processors/nodes
  - quantum computation
  - quantum communication (qubits)
  - output: measurement of qubits
- **Complexity measure**: number of communication rounds
- **Gathering algorithms are *weaker***!
  - measuring to clone "corrupts" the quantum state
  - quantum states cannot be *cloned* (no-cloning theorem)



Quantum-LOCAL

[Gavoille et al., DISC '09]

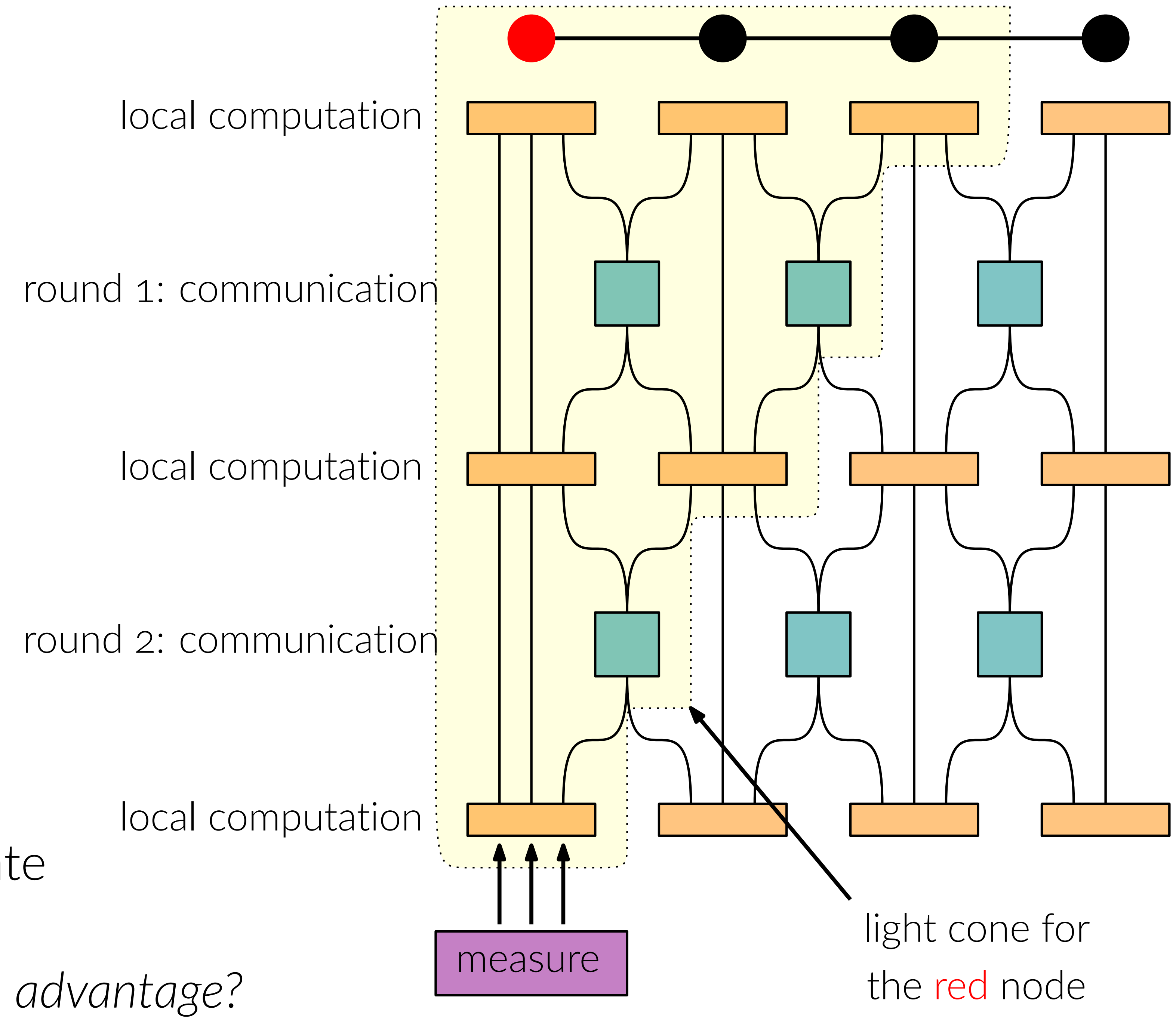
- **Distributed system** of n quantum processors/nodes
  - quantum computation
  - quantum communication (qubits)
  - output: measurement of qubits
- **Complexity measure**: number of communication rounds
- **Gathering algorithms are *weaker*!**
  - measuring to clone "corrupts" the quantum state
  - quantum states cannot be *cloned* (no-cloning theorem)
- Still, locality identifies how *far* nodes need to communicate



Quantum-LOCAL

[Gavoille et al., DISC '09]

- **Distributed system** of n quantum processors/nodes
  - quantum computation
  - quantum communication (qubits)
  - output: measurement of qubits
- **Complexity measure**: number of communication rounds
- **Gathering algorithms are *weaker*!**
  - measuring to clone "corrupts" the quantum state
  - quantum states cannot be *cloned* (no-cloning theorem)
- Still, locality identifies how *far* nodes need to communicate
- **Question**: *is there any graph problem that admits quantum advantage?*



Quantum-LOCAL

- **Question**: *is there any graph problem that admits quantum advantage?*

  - yes! [Le Gall et al., STACS '19]
  - problem with complexity $\Omega(n)$ in randomized-LOCAL and $O(1)$ in quantum-LOCAL

Quantum-LOCAL

- **Question**: *is there any graph problem that admits quantum advantage?*

  - yes! [Le Gall et al., STACS '19]
  - problem with complexity $\Omega(n)$ in randomized-LOCAL and $O(1)$ in quantum-LOCAL

- **Question**: *what about problems that actually interest the distributed computing community?*

Quantum-LOCAL

- **Question**: *is there any graph problem that admits quantum advantage?*

  - yes! [Le Gall et al., STACS '19]
  - problem with complexity $\Omega(n)$ in randomized-LOCAL and $O(1)$ in quantum-LOCAL

- **Question**: *what about problems that actually interest the distributed computing community?*

  - we do not know!

Quantum-LOCAL

- **Question**: *is there any graph problem that admits quantum advantage?*

  - yes! [Le Gall et al., STACS '19]
  - problem with complexity $\Omega(n)$ in randomized-LOCAL and $O(1)$ in quantum-LOCAL

- **Question**: *what about problems that actually interest the distributed computing community?*

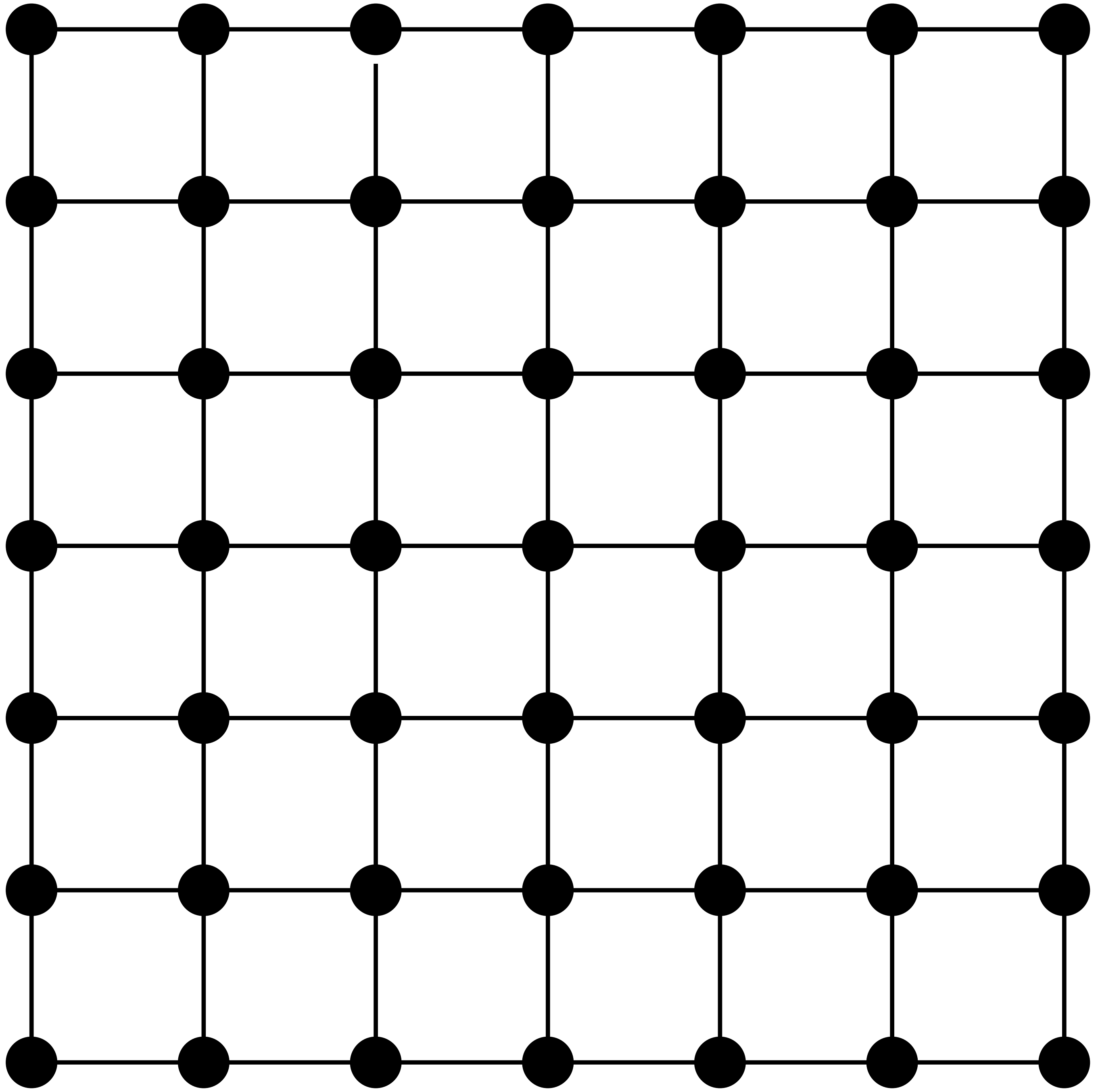
  - we do not know!

- *What do we know?*

  - focus on LCLs
  - input graph degree is bounded by a constant Δ [Naor and Stockmeyer, SICOMP '95]

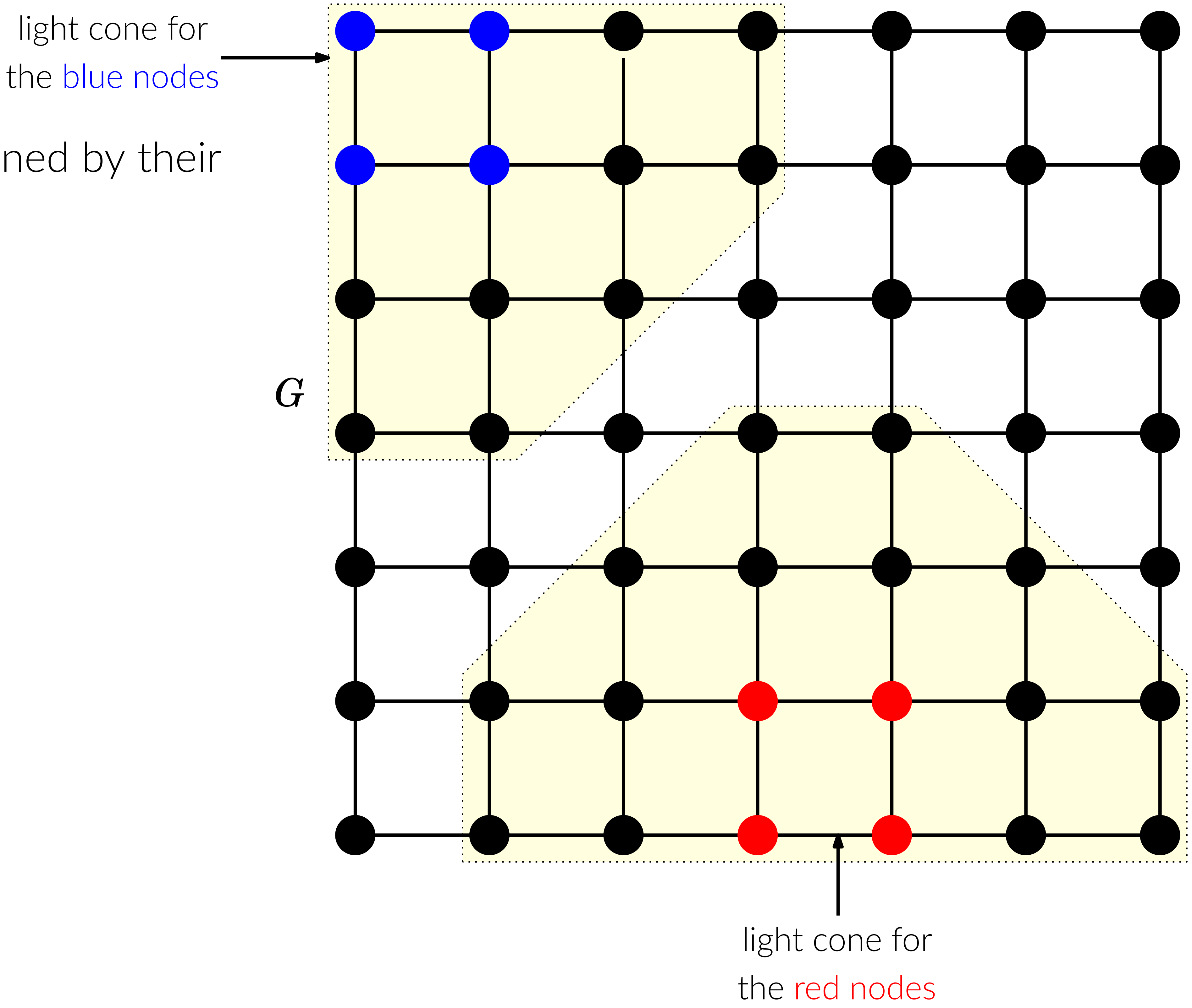
Properties of distributed algorithms

- **Run** a 2-round algorithm A in G



Properties of distributed algorithms

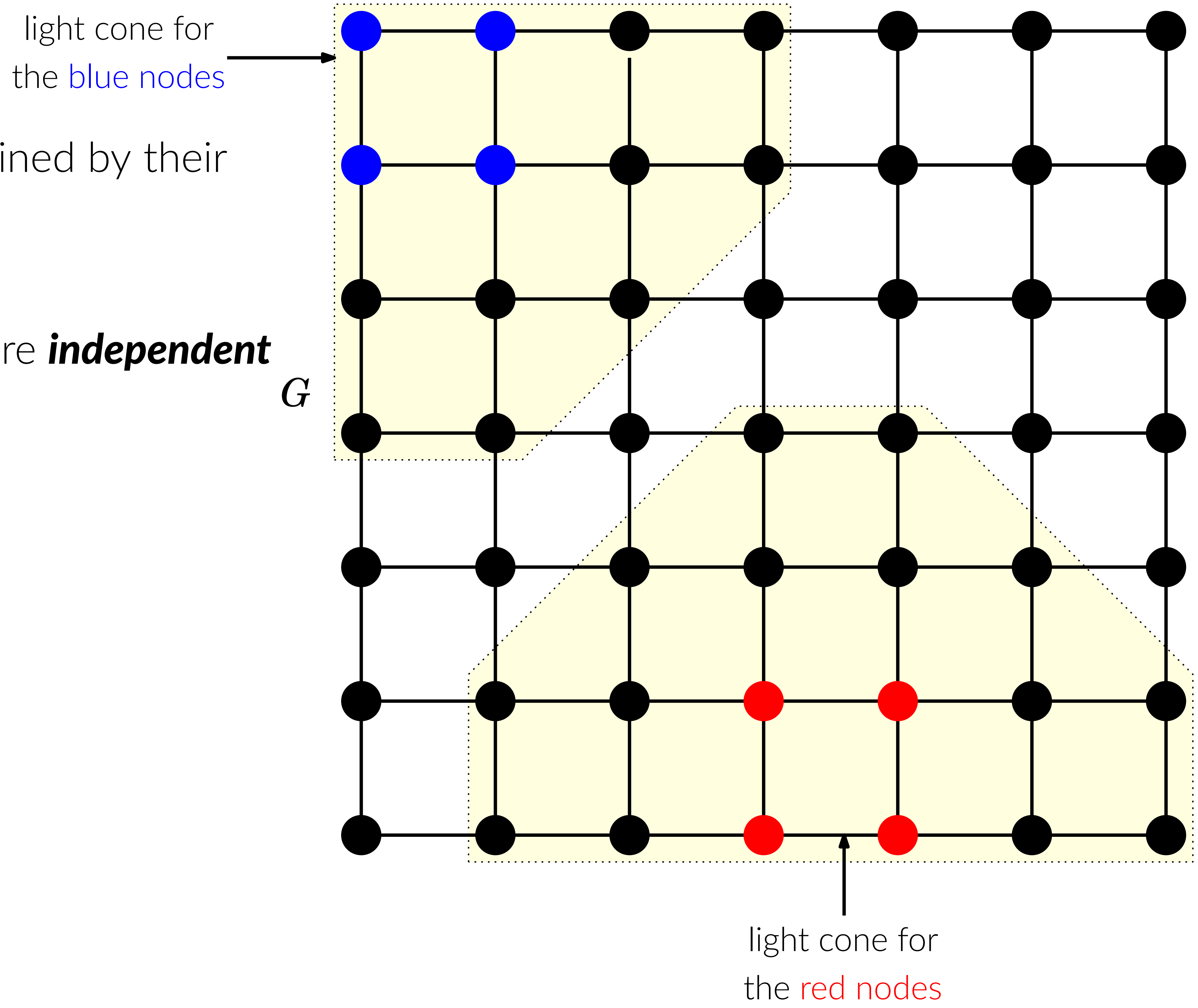
- **Run** a $\boldsymbol{2}$-round algorithm $\boldsymbol{A}$ in $\boldsymbol{G}$
  - output for the red and blue nodes is determined by their respective light cones



Properties of distributed algorithms

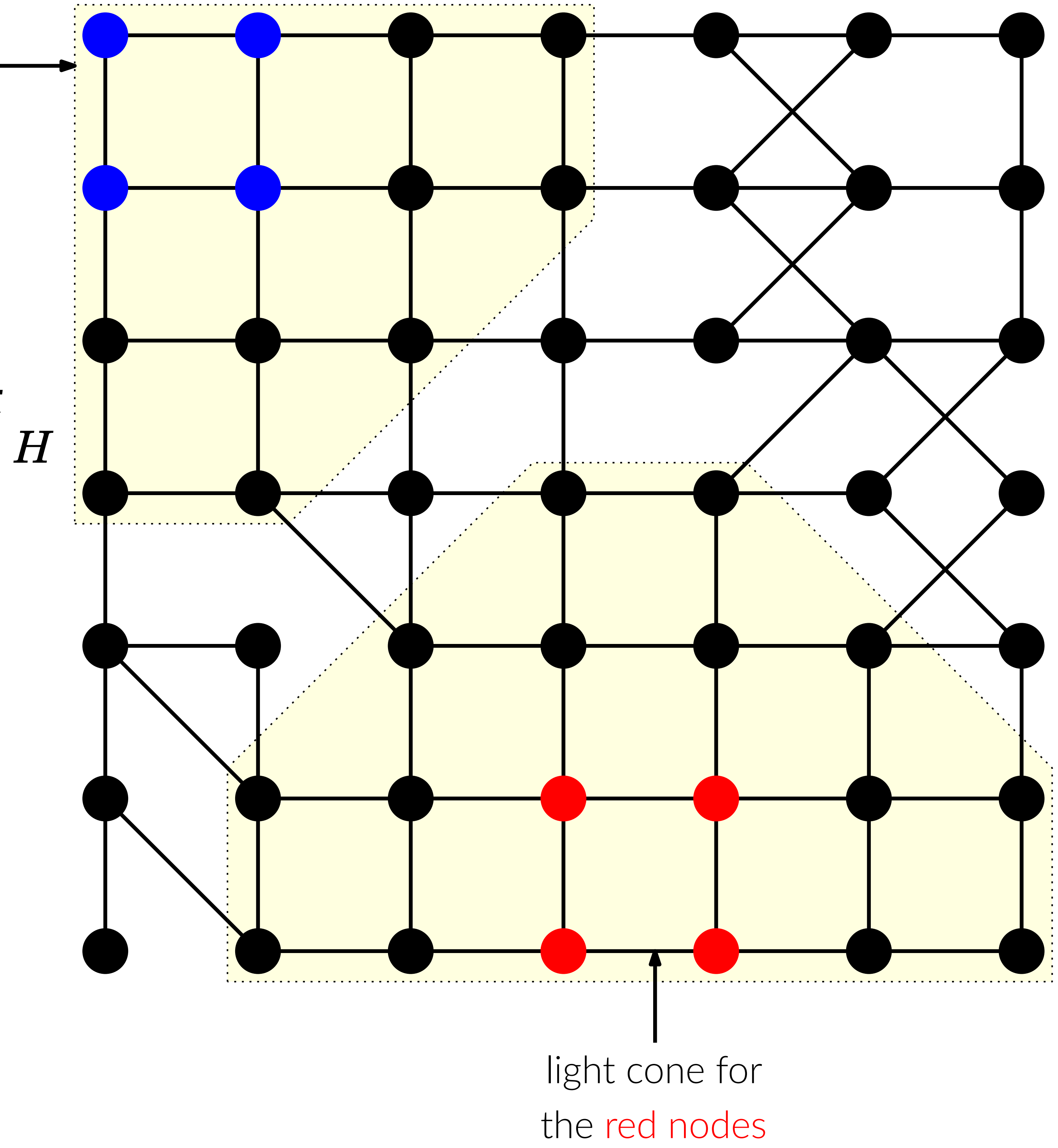
- **Run** a 2-round algorithm A in G
  - output for the red and blue nodes is determined by their respective light cones

- **Output distributions** for red and blue nodes are ***independent***
  - as long as their distance is at least 5



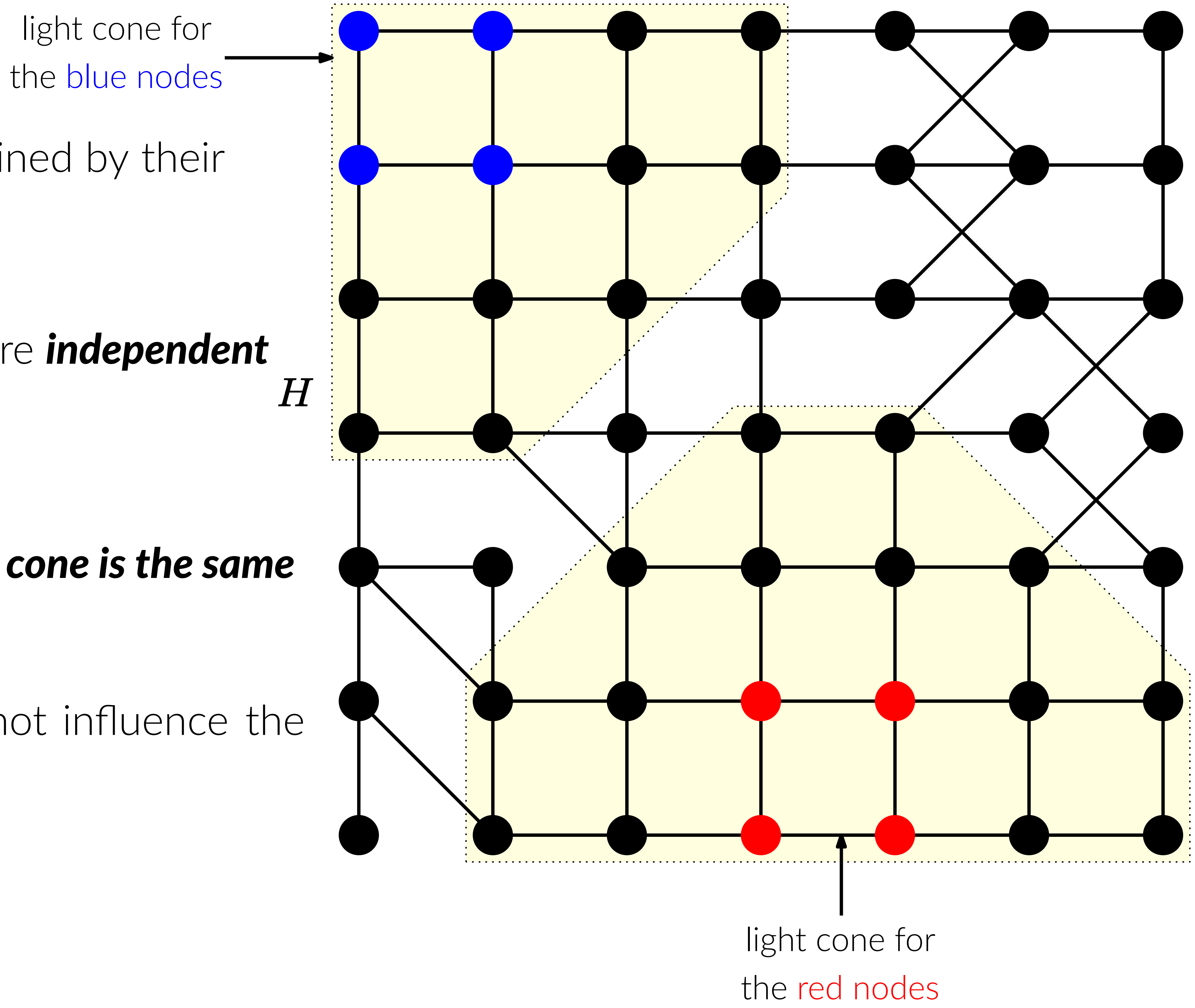
Properties of distributed algorithms

- **Run** a $\mathbf{2}$-round algorithm $\boldsymbol{A}$ in $\boldsymbol{G}$
  - output for the red and blue nodes is determined by their respective light cones
- **Output distributions** for red and blue nodes are ***independent***
  - as long as their distance is at least $\mathbf{5}$



Properties of distributed algorithms

- **Run** a 2-round algorithm A in G
  - output for the red and blue nodes is determined by their respective light cones
- **Output distributions** for red and blue nodes are ***independent***
  - as long as their distance is at least 5
- **Output distributions** remains ***the same*** if ***light cone is the same***
  - non-signaling property
  - changes that are beyond 2-hops away do not influence the output distribution
  - also known as causality



Abstracting output distributions

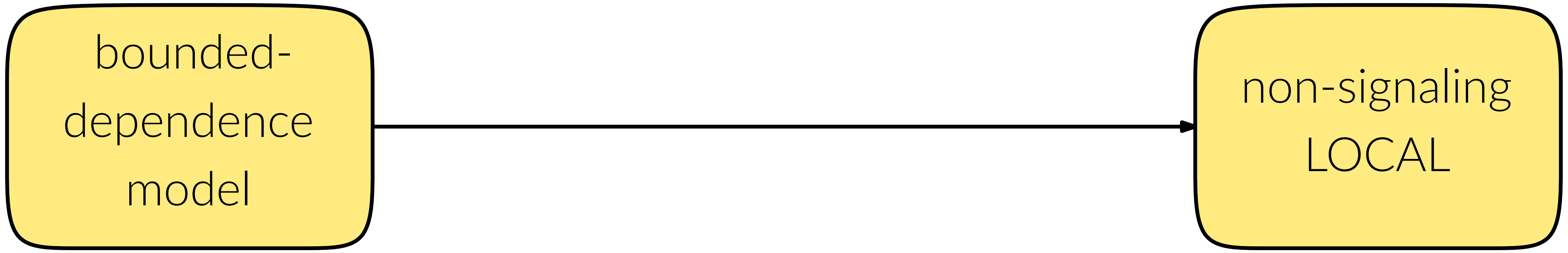
- A T-round distributed algorithm yields an **output distribution** with the following **properties**:
  - outputs of subsets of nodes at distance more than $2T$ are independent
  - non-signaling beyond distance T

Abstracting output distributions

- A T-round distributed algorithm yields an **output distribution** with the following **properties**:
  - outputs of subsets of nodes at distance more than $2T$ are independent
  - non-signaling beyond distance T
- Then we can ***just think about output distributions***!
  - computational models that produce directly distributions with the aforementioned properties

Abstracting output distributions

- A T-round distributed algorithm yields an **output distribution** with the following **properties**:
  - outputs of subsets of nodes at distance more than $2T$ are independent
  - non-signaling beyond distance T
- Then we can ***just think about output distributions***!
  - computational models that produce directly distributions with the aforementioned properties



locality T = independence at distance $2T + 1$ plus non-signaling beyond distance T

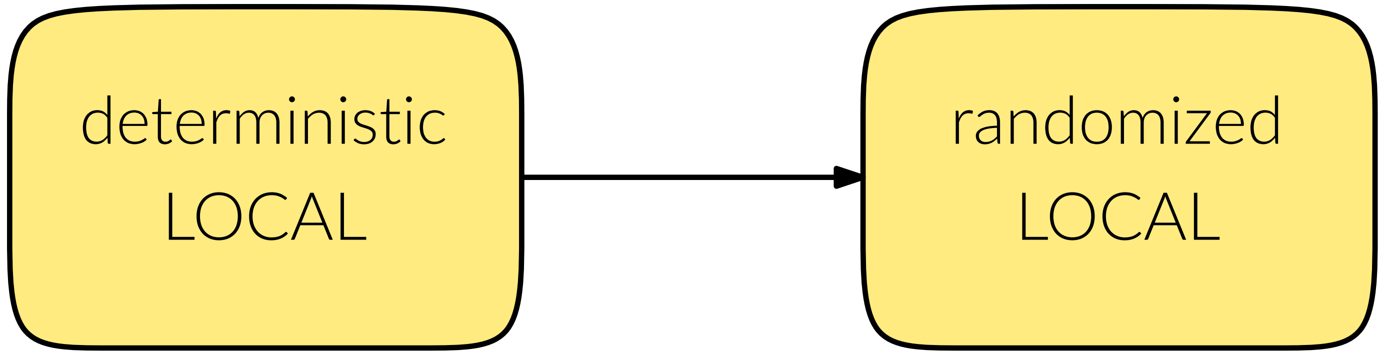
locality T = non-signaling beyond distance T

[Holroyd and Liggett, Forum of Mathematics, Pi '14]
[Akbari et al., 2024] * finitely-dependent distributions if $T = O(1)$

[Gavoille et al., DISC '09]
[Arfaoui and Fraigniaud, PODC '12 & SIGACT News '14]

Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y

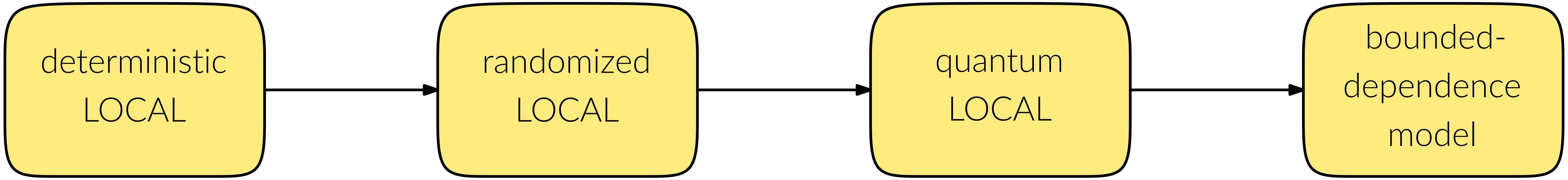


Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y

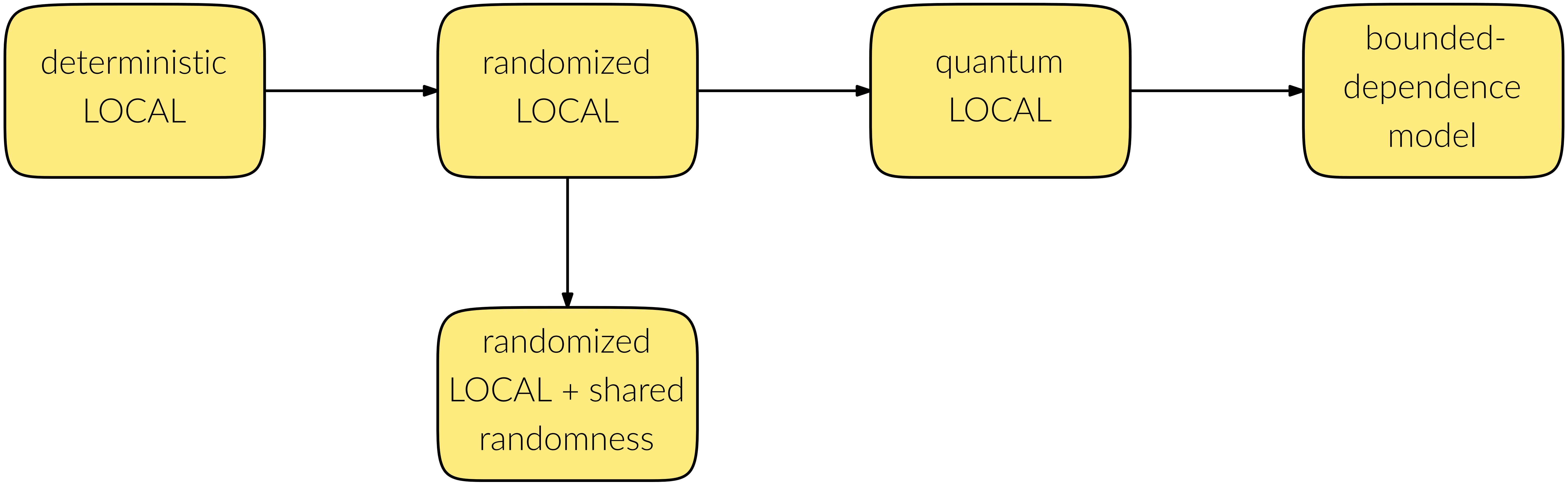
Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y



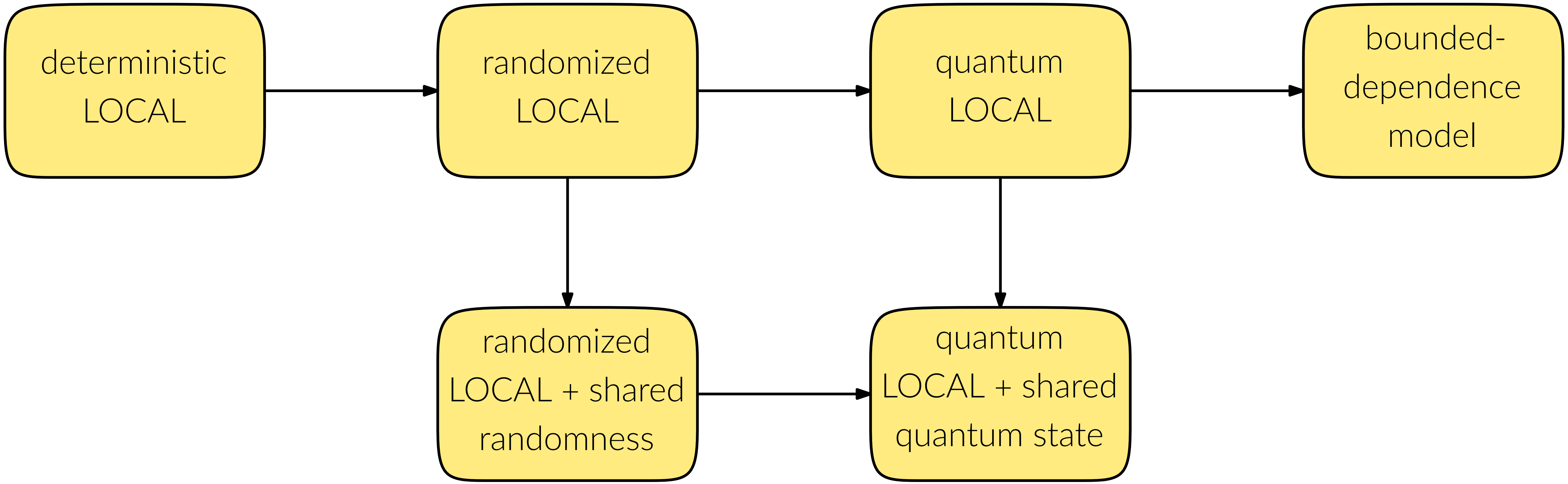
Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y



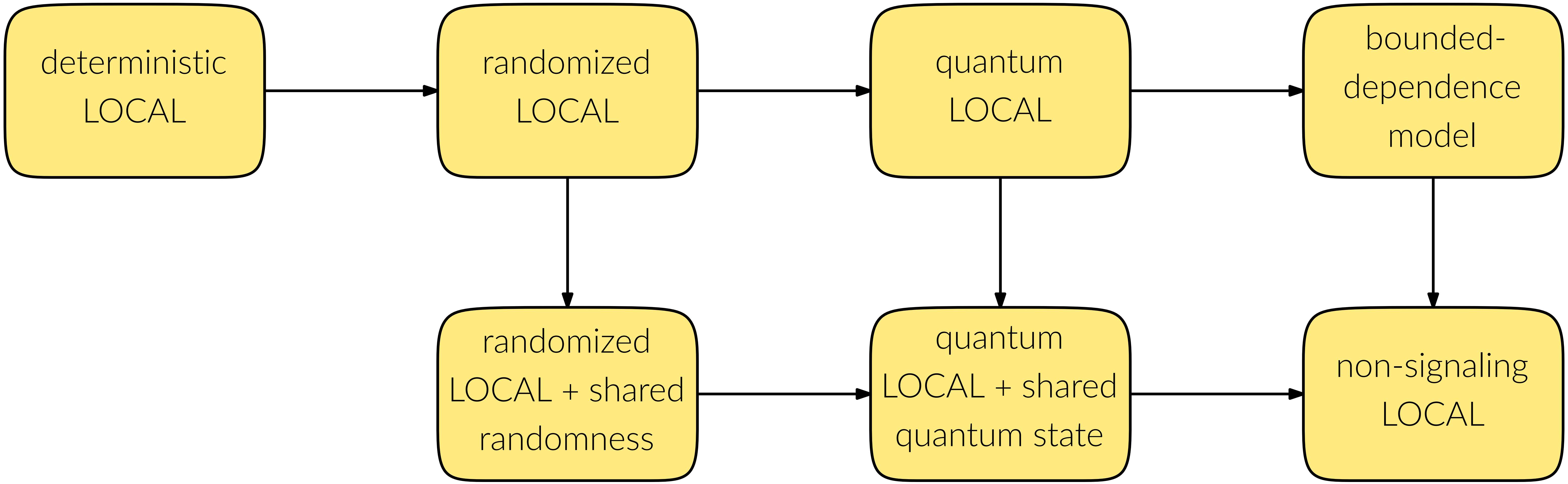
Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y



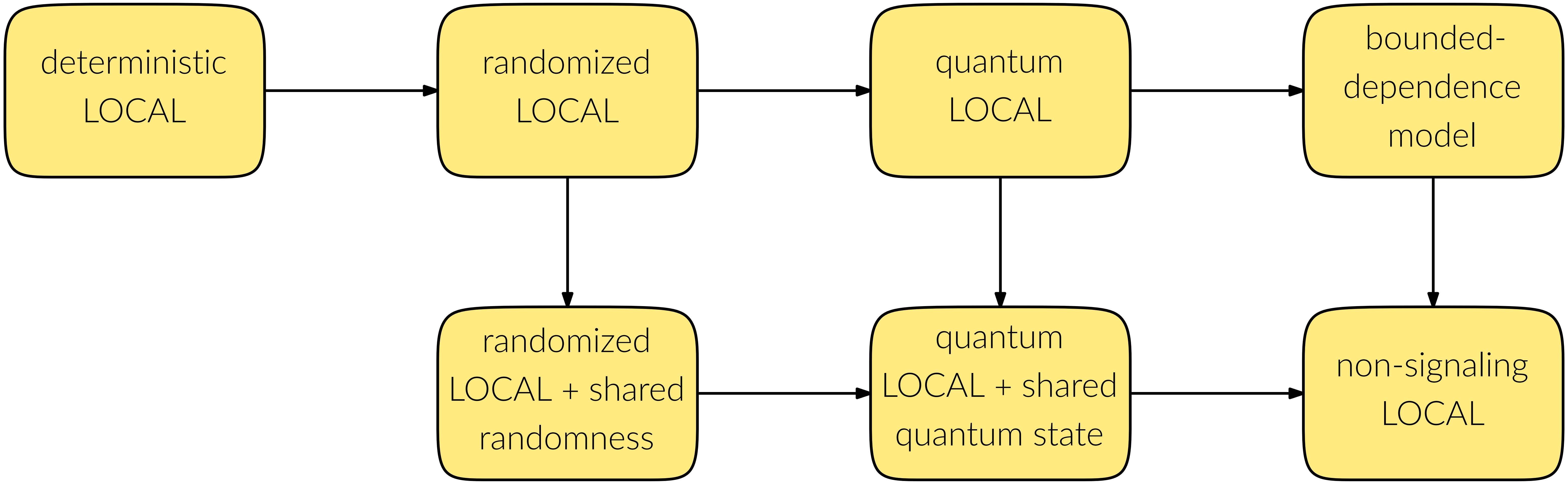
Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y



Relations among models

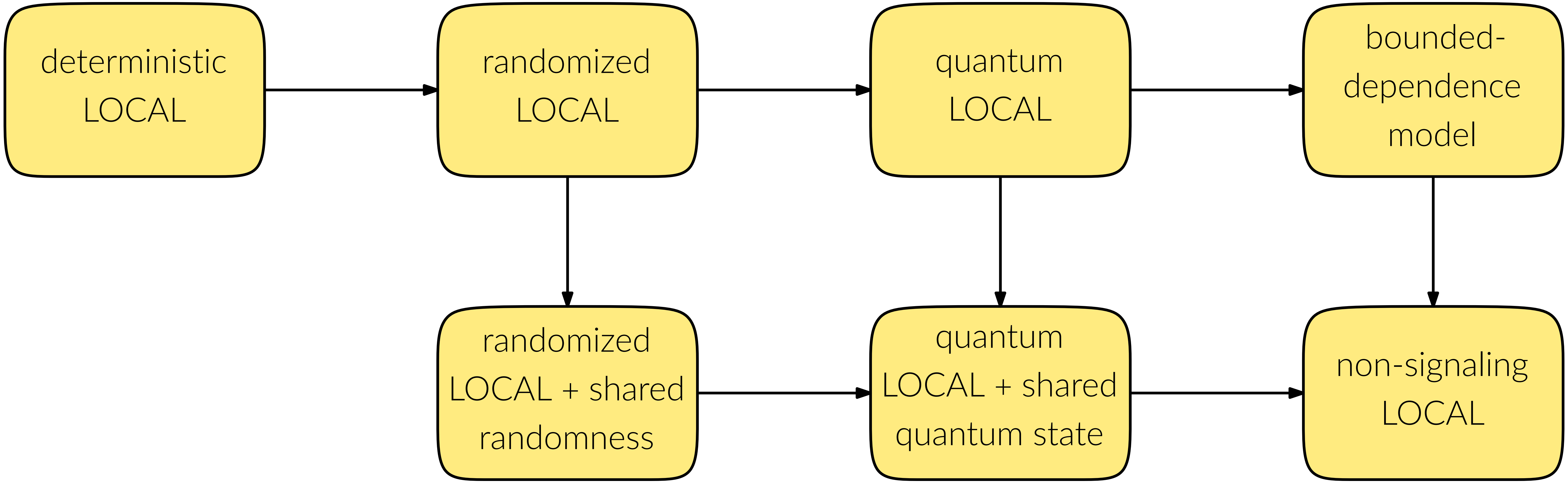
- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y



- Is it possible to **“sandwich” quantum-LOCAL** between weaker and stronger models?

Relations among models

- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y

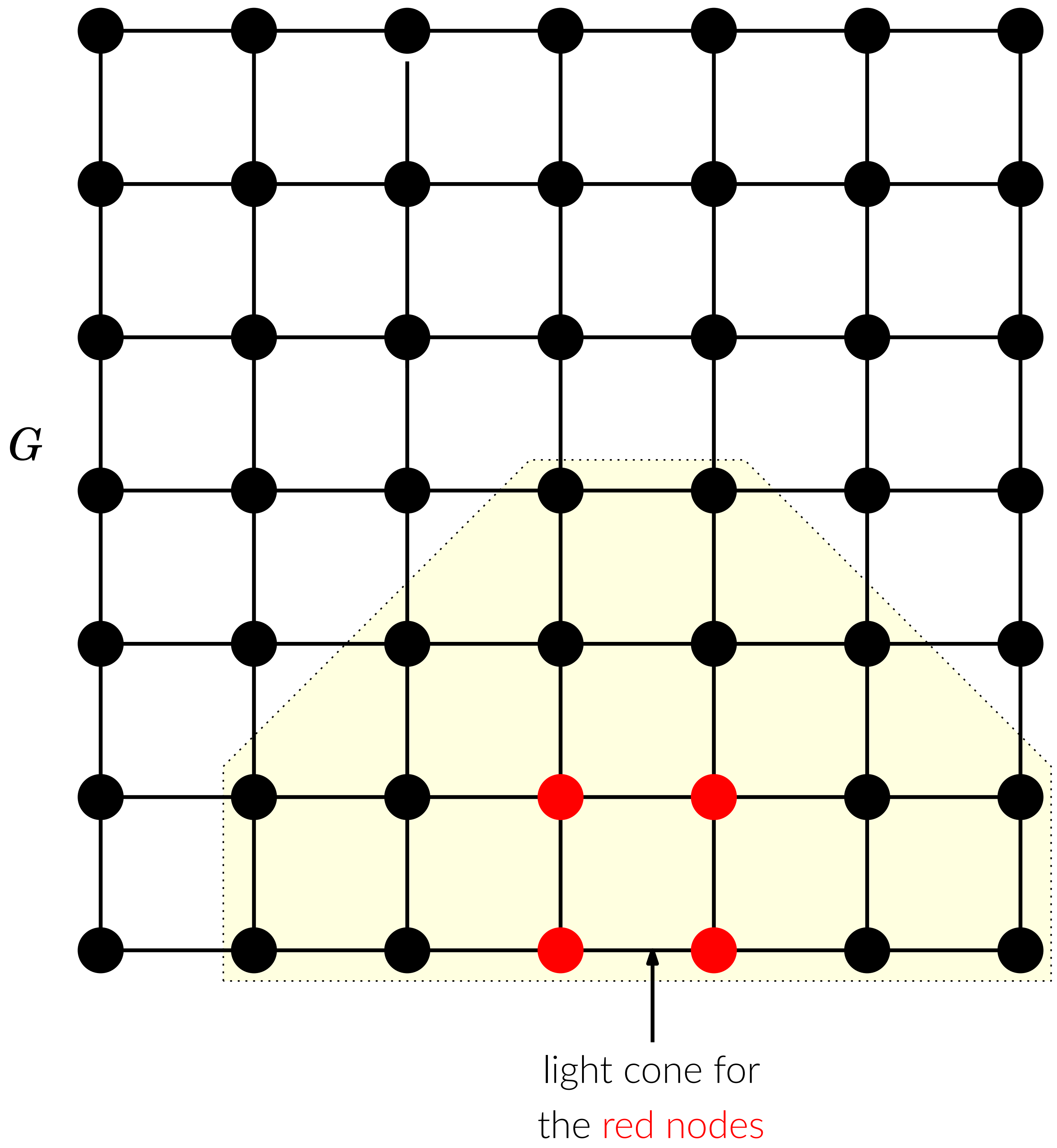


- Is it possible to **“sandwich” quantum-LOCAL** between weaker and stronger models?

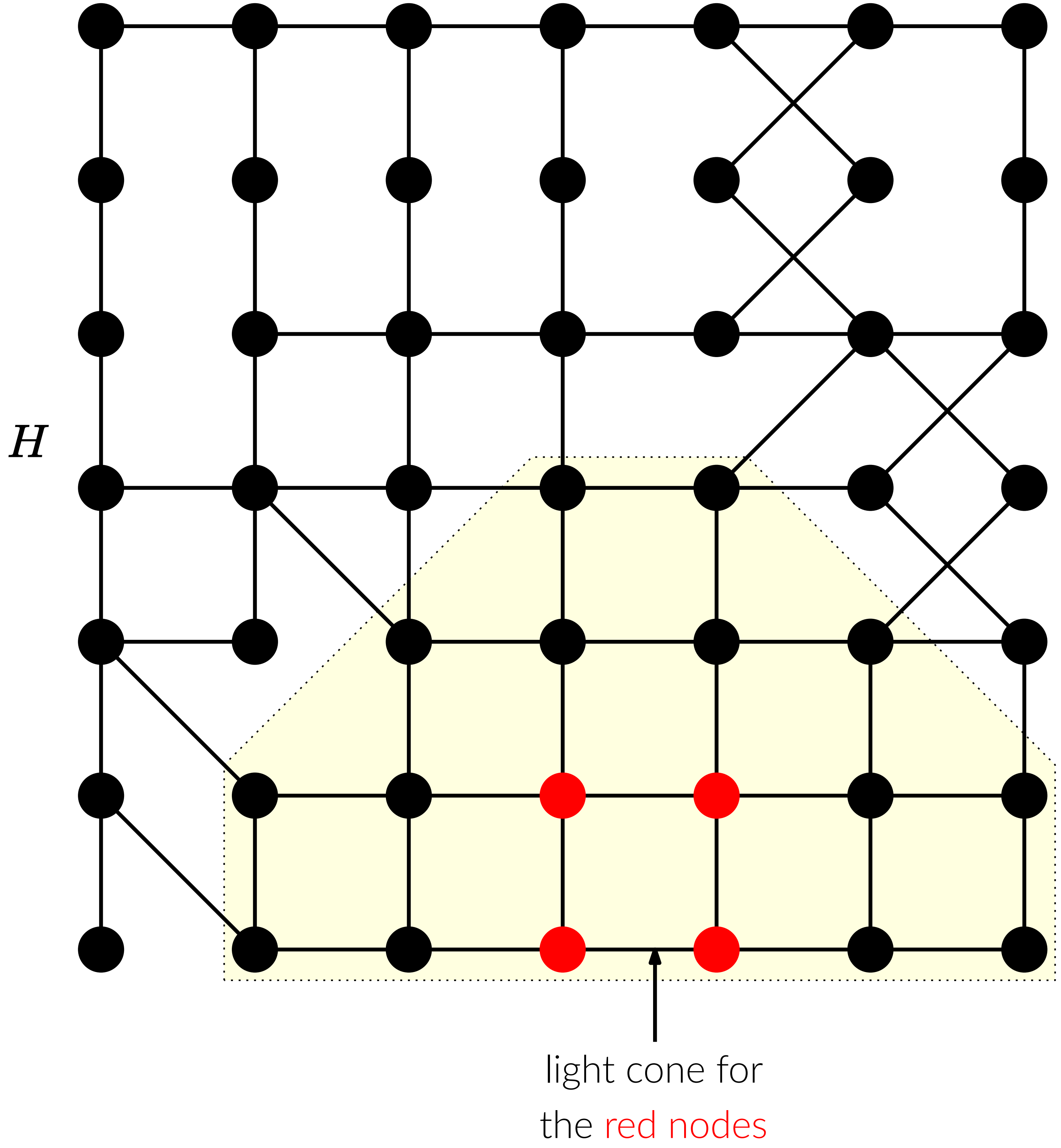
  - yes!

Some results: non-signaling LOCAL model

- T-round **algorithm** in **non-signaling LOCAL**:
  - assigns to each input a distribution over output labelings
  - non-signaling beyond distance T



- T-round **algorithm** in **non-signaling LOCAL**:
  - assigns to each input a distribution over output labelings
  - non-signaling beyond distance T



Some results: non-signaling LOCAL model

- T-round **algorithm** in **non-signaling LOCAL**:

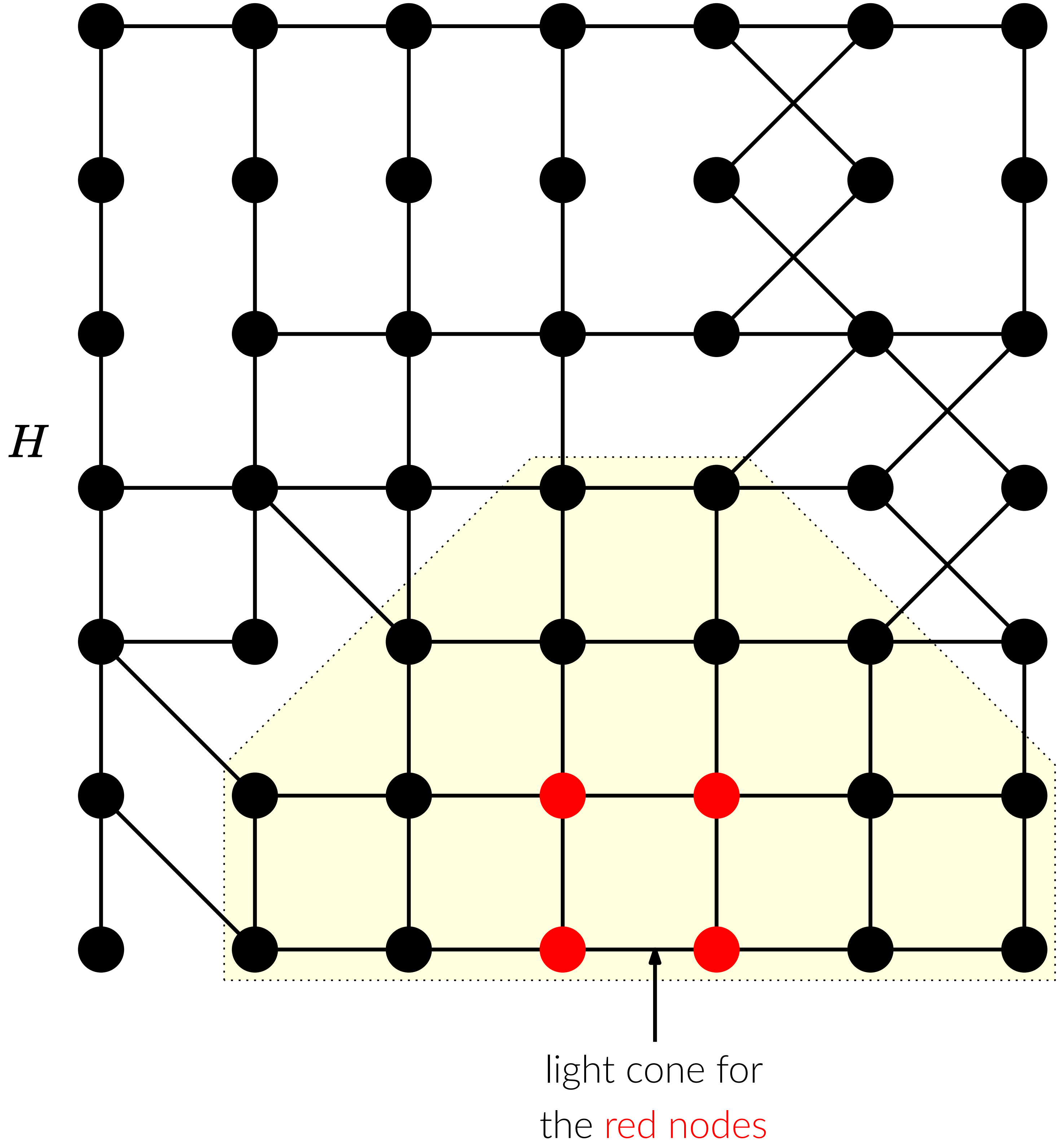
  - assigns to each input a distribution over output labelings

  - non-signaling beyond distance T

[Gavoille et al., DISC '09]

- Propagation arguments based on indistinguishability hold!

  * some care is needed



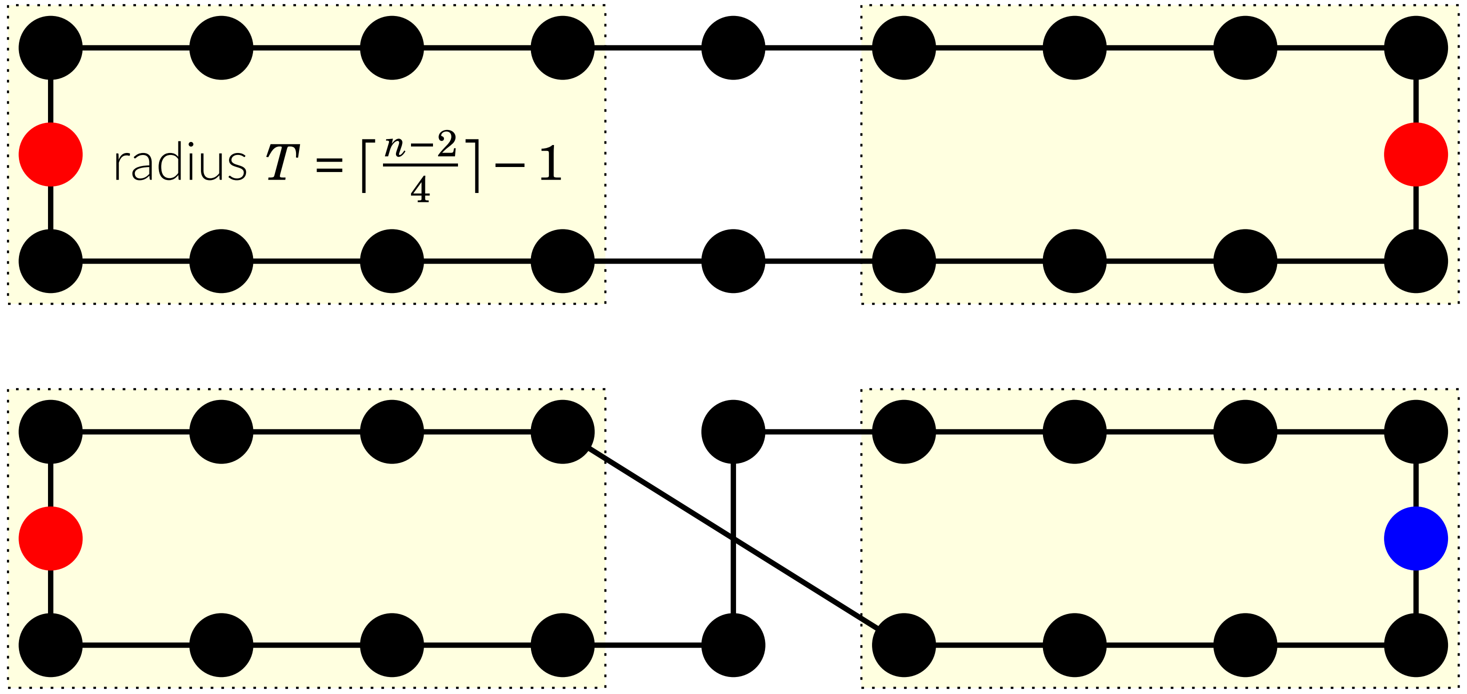
Some results: non-signaling LOCAL model

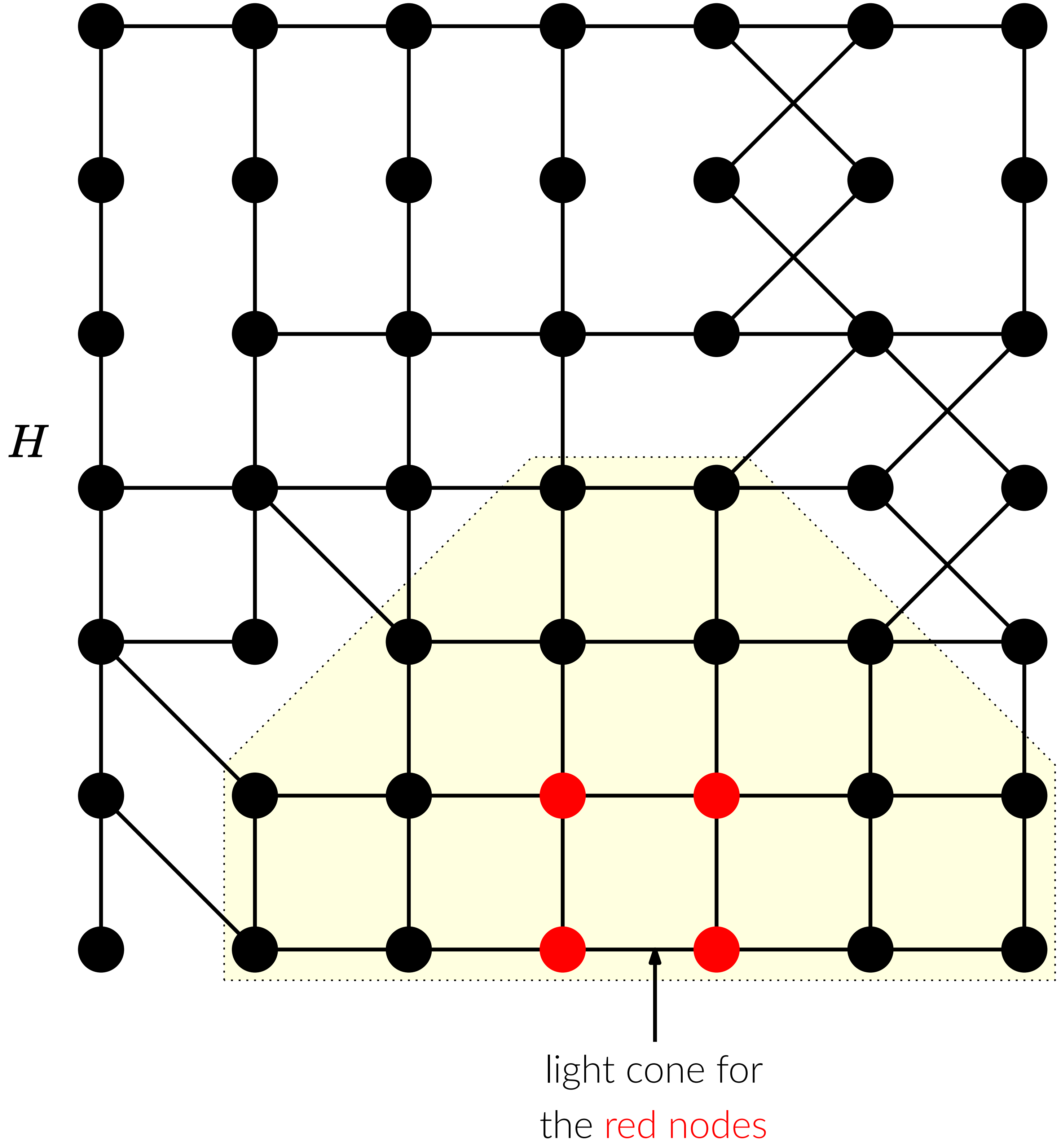
- T-round **algorithm** in **non-signaling LOCAL**:
  - assigns to each input a distribution over output labelings
  - non-signaling beyond distance T

[Gavoille et al., DISC '09]

- Propagation arguments based on indistinguishability hold!

  * some care is needed
  - example: **2-coloring cycles** is hard ($T = \Theta(n)$)





Some results: non-signaling LOCAL model

- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]

Some results: non-signaling LOCAL model

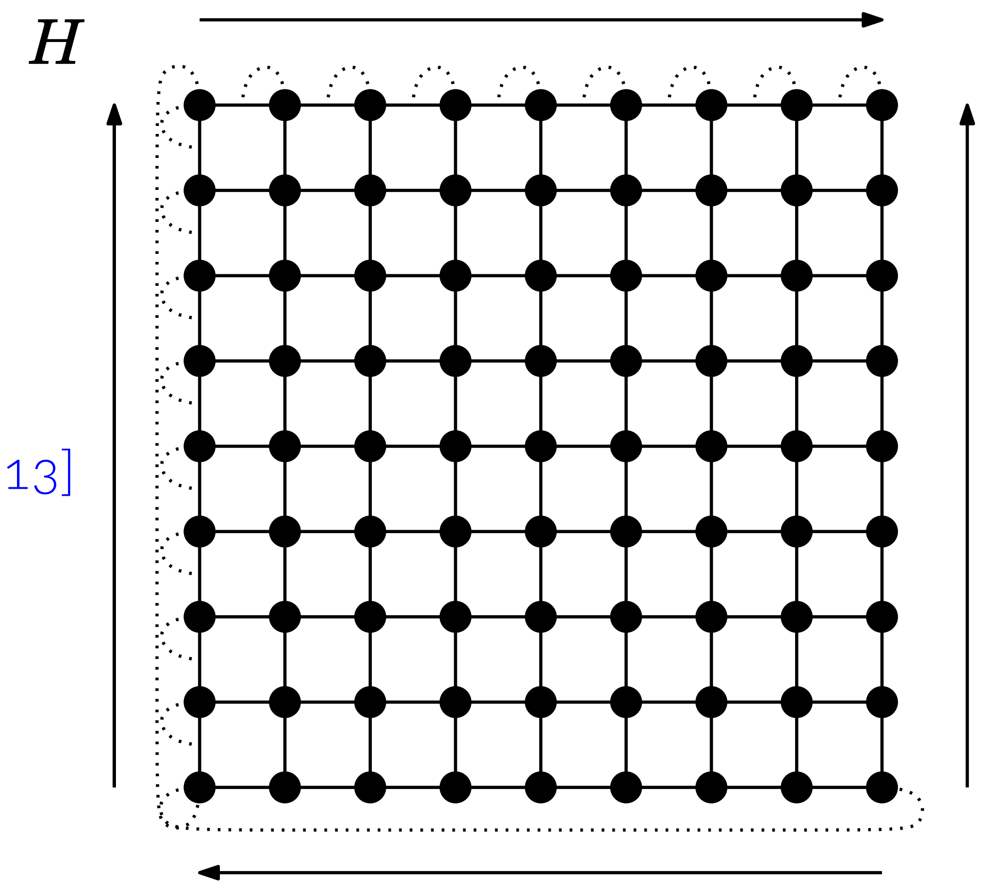
- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]
- Example: **3-coloring** $\sqrt{n} \times \sqrt{n}$ **grids** is hard ($T = \Theta(\sqrt{n})$)

Some results: non-signaling LOCAL model

- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]
- Example: **3-coloring $\sqrt{n} \times \sqrt{n}$ grids** is hard ($T = \Theta(\sqrt{n})$)
  - find graph H that locally is like a grid but chromatic number $\mathcal{X}(H) \geq 4$

H: odd quadrangulation of Klein–bottle

- locally grid-like, $\mathcal{X}(H) = 4$ [Mohar et al., Combinatorica '13]

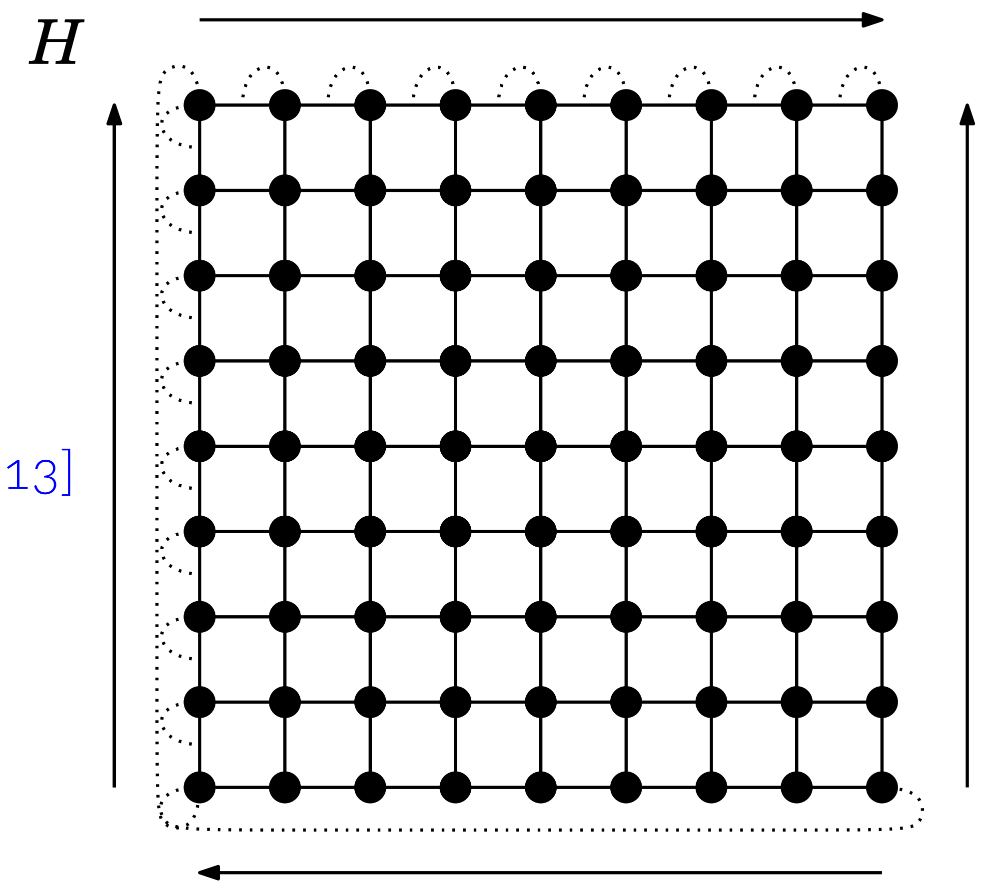


Some results: non-signaling LOCAL model

- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]
- Example: **3-coloring** $\sqrt{n} \times \sqrt{n}$ **grids** is hard ($T = \Theta(\sqrt{n})$)
  - find graph H that locally is like a grid but chromatic number $\mathcal{X}(H) \geq 4$
  - by contradiction: algorithm $\mathcal{A}$ with locality $T \leq \lfloor \frac{\sqrt{n}-2}{4} \rfloor$ that 3-colors grids

H: odd quadrangulation of Klein–bottle

- locally grid-like, $\mathcal{X}(H) = 4$ [Mohar et al., Combinatorica '13]

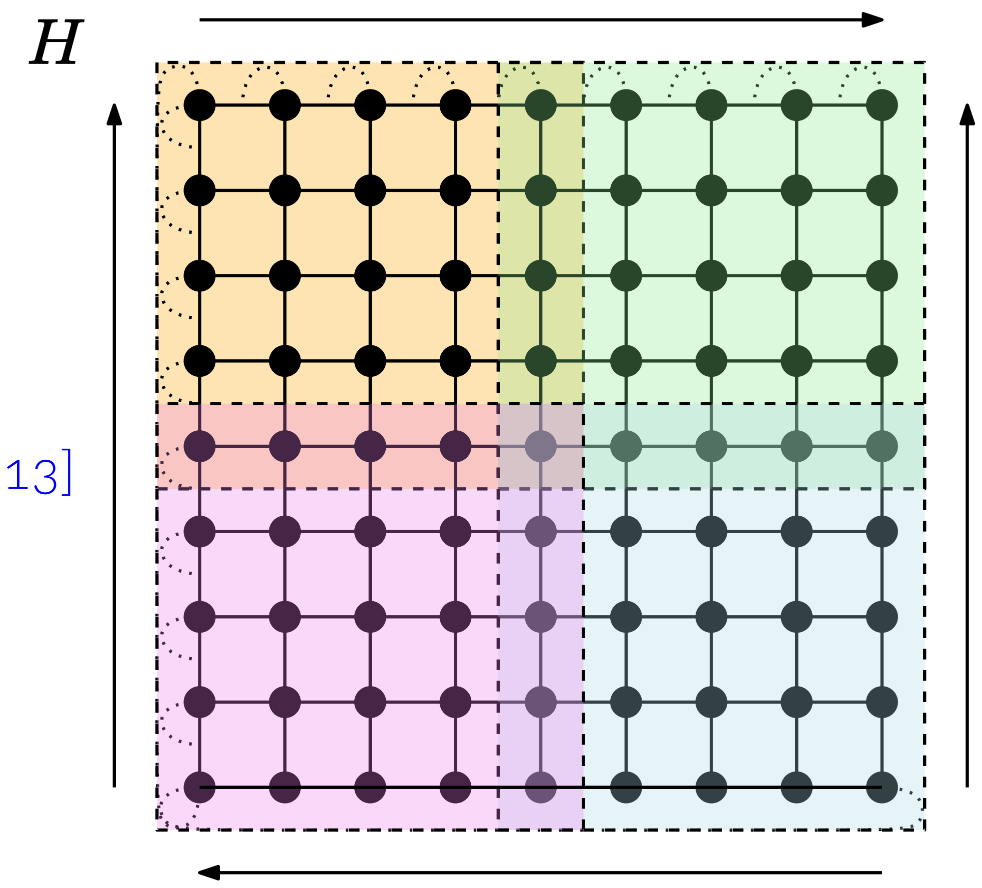


Some results: non-signaling LOCAL model

- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]
- Example: **3-coloring** $\sqrt{n} \times \sqrt{n}$ **grids** is hard ($T = \Theta(\sqrt{n})$)
  - find graph H that locally is like a grid but chromatic number $\mathcal{X}(H) \geq 4$
  - by contradiction: algorithm $\mathcal{A}$ with locality $T \leq \lfloor \frac{\sqrt{n}-2}{4} \rfloor$ that 3-colors grids
  - failure with prob. 1 overall $\Longrightarrow$ failure with prob. $\frac{1}{4}$ in at least one of the regions

H: odd quadrangulation of Klein–bottle

- locally grid-like, $\mathcal{X}(H) = 4$ [Mohar et al., Combinatorica '13]

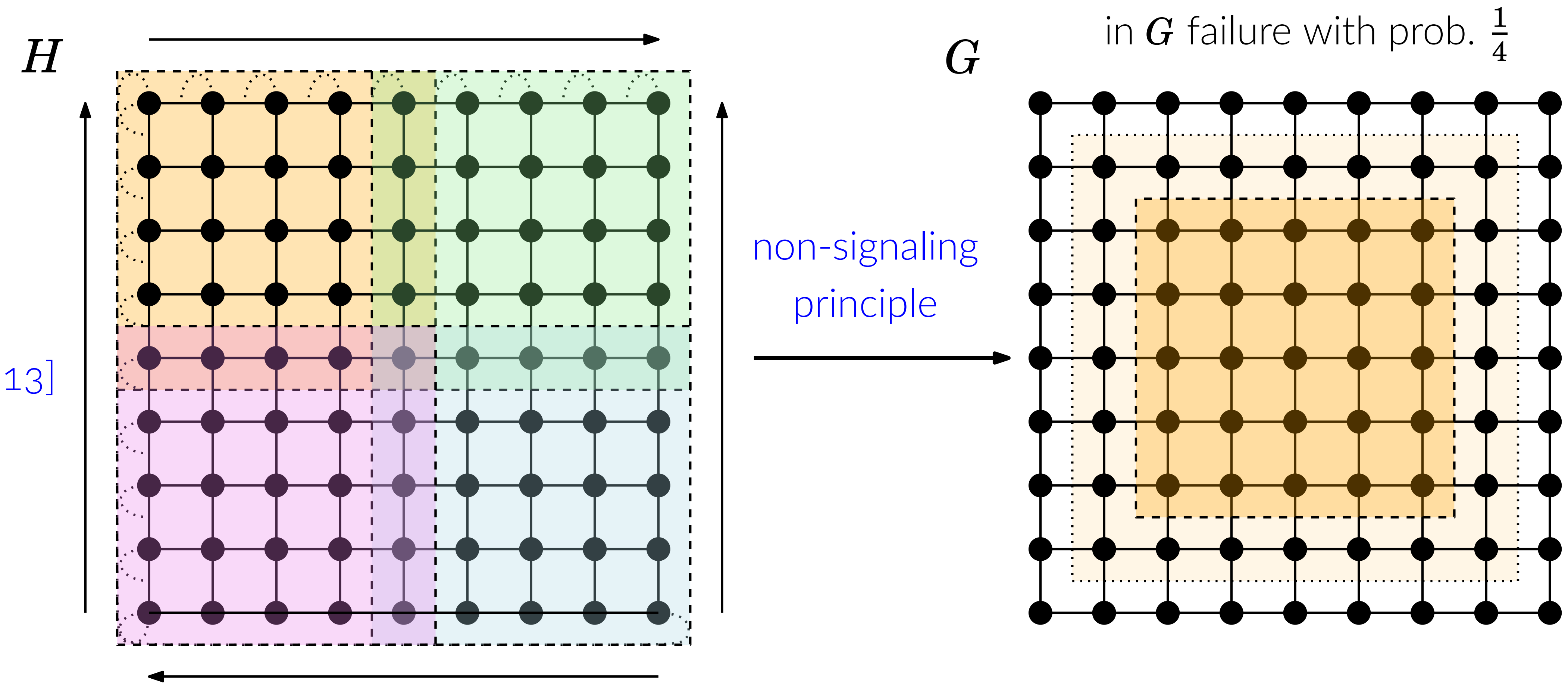


Some results: non-signaling LOCAL model

- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]
- Example: **3-coloring** $\sqrt{n} \times \sqrt{n}$ **grids** is hard ($T = \Theta(\sqrt{n})$)
  - find graph H that locally is like a grid but chromatic number $\mathcal{X}(H) \geq 4$
  - by contradiction: algorithm $\mathcal{A}$ with locality $T \leq \lfloor \frac{\sqrt{n}-2}{4} \rfloor$ that 3-colors grids
  - failure with prob. 1 overall $\implies$ failure with prob. $\frac{1}{4}$ in at least one of the regions
  - exploit non-signaling principle

H: odd quadrangulation of Klein–bottle

- locally grid-like, $\mathcal{X}(H) = 4$ [Mohar et al., Combinatorica '13]

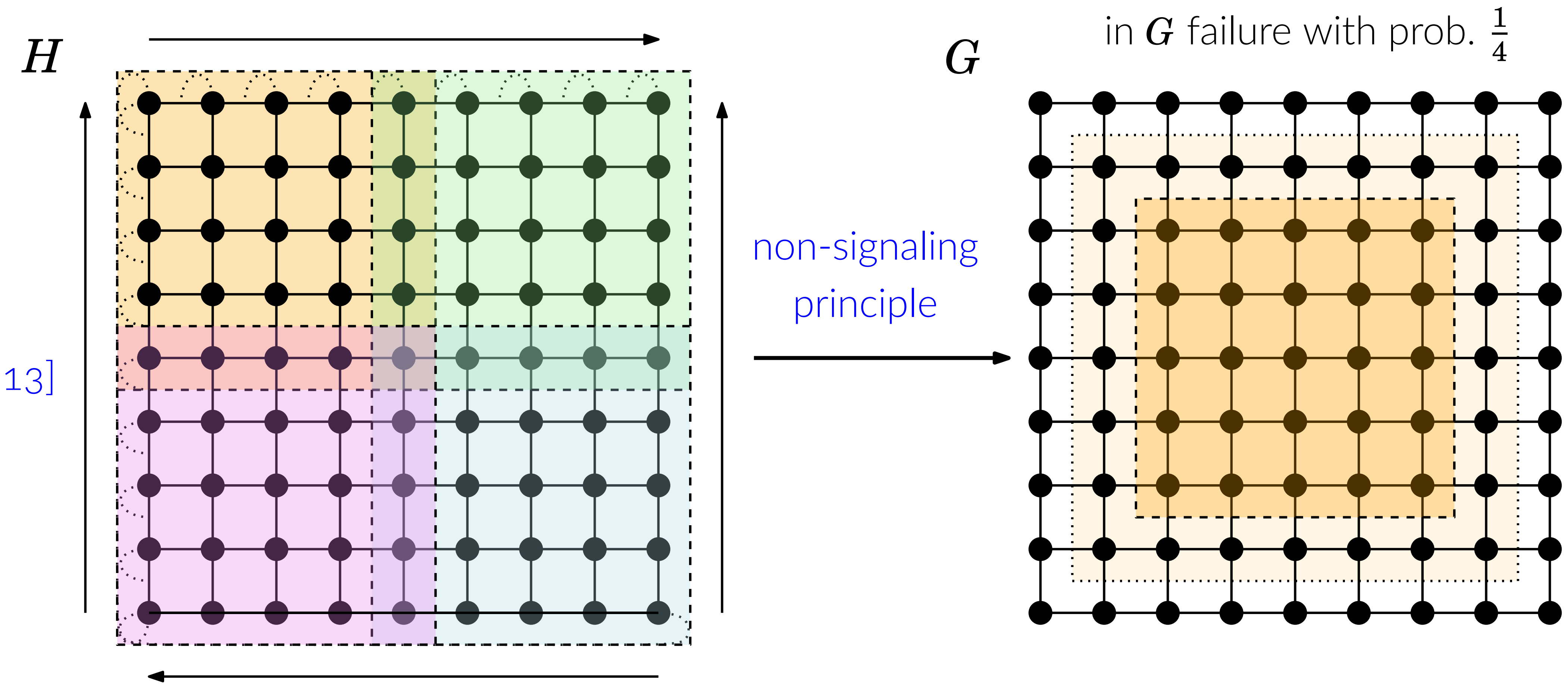


Some results: non-signaling LOCAL model

- **Graph-existential lower bound** arguments based on indistinguishability ***hold***! [Coiteux-Roy et al., STOC '24]
- Example: **3-coloring $\sqrt{n} \times \sqrt{n}$ grids** is hard ($T = \Theta(\sqrt{n})$)
  - find graph H that locally is like a grid but chromatic number $\mathcal{X}(H) \geq 4$
  - by contradiction: algorithm $\mathcal{A}$ with locality $T \leq \lfloor \frac{\sqrt{n}-2}{4} \rfloor$ that 3-colors grids
  - failure with prob. 1 overall $\implies$ failure with prob. $\frac{1}{4}$ in at least one of the regions
  - exploit non-signaling principle

H: odd quadrangulation of Klein–bottle

- locally grid-like, $\mathcal{X}(H) = 4$ [Mohar et al., Combinatorica '13]



- **Boosting failure prob.** is also possible

Some results: non-signaling LOCAL model

Graph-existential lower bound arguments based on indistinguishability

- **Graph coloring**: c-coloring χ-chromatic graphs has complexity $\tilde{\Theta}(n^{1/\lfloor \frac{c-1}{\chi-1} \rfloor})$ [Coiteux-Roy et al., STOC '24]
  - makes use of a "cheating graph" from [Bogdanov, '13]
  - upper bound in deterministic LOCAL, lower bound in non-signaling LOCAL
  - no quantum advantage

Some results: non-signaling LOCAL model

Graph-existential lower bound arguments based on indistinguishability

- **Graph coloring**: c-coloring χ-chromatic graphs has complexity $\tilde{\Theta}(n^{1/\lfloor \frac{c-1}{\chi-1} \rfloor})$ [Coiteux-Roy et al., STOC '24]

  - makes use of a "cheating graph" from [Bogdanov, '13]
  - upper bound in deterministic LOCAL, lower bound in non-signaling LOCAL
  - no quantum advantage

- **Tree coloring**: c-coloring trees has complexity $\Omega(\log_c n)$ [Coiteux-Roy et al., STOC '24]

  - revisitation of [Linial, FOCS '87]'s lower bound
  - no quantum advantage if high degree

Some results: non-signaling LOCAL model

Graph-existential lower bound arguments based on indistinguishability

- **Graph coloring**: c-coloring χ-chromatic graphs has complexity $\tilde{\Theta}(n^{1/\lfloor \frac{c-1}{\chi-1} \rfloor})$ [Coiteux-Roy et al., STOC '24]

  - makes use of a "cheating graph" from [Bogdanov, '13]
  - upper bound in deterministic LOCAL, lower bound in non-signaling LOCAL
  - no quantum advantage

- **Tree coloring**: c-coloring trees has complexity $\Omega(\log_c n)$ [Coiteux-Roy et al., STOC '24]

  - revisitation of [Linial, FOCS '87]'s lower bound
  - no quantum advantage if high degree

- **Grid coloring**: 3-coloring grids of size $n_1 \times n_2$ has complexity $\Omega(\min\{n_1, n_2\})$ [Coiteux-Roy et al., STOC '24]

  - makes use of odd quadrangulations of Klein-bottles [Mohar et al., Combinatorica '13]
  - no quantum advantage

Some results: non-signaling LOCAL model

Graph-existential lower bound arguments based on indistinguishability

- **Graph coloring**: c-coloring χ-chromatic graphs has complexity $\tilde{\Theta}(n^{1/\lfloor \frac{c-1}{\chi-1} \rfloor})$ [Coiteux-Roy et al., STOC '24]
  - makes use of a "cheating graph" from [Bogdanov, '13]
  - upper bound in deterministic LOCAL, lower bound in non-signaling LOCAL
  - no quantum advantage
- **Tree coloring**: c-coloring trees has complexity $\Omega(\log_c n)$ [Coiteux-Roy et al., STOC '24]
  - revisitation of [Linial, FOCS '87]'s lower bound
  - no quantum advantage if high degree
- **Grid coloring**: 3-coloring grids of size $n_1 \times n_2$ has complexity $\Omega(\min\{n_1, n_2\})$ [Coiteux-Roy et al., STOC '24]
  - makes use of odd quadrangulations of Klein-bottles [Mohar et al., Combinatorica '13]
  - no quantum advantage

What about other known lower bounds? E.g., 3-coloring cycles has complexity $\Theta(\log^\star n)$ [Linial, FOCS '87]

Some results: bounded-dependence model

- Can we similarly **rule out quantum advantage for 3-coloring cycles**? Classical complexity $T = \Theta(\log^\star n)$

Some results: bounded-dependence model

- Can we similarly **rule out quantum advantage for 3-coloring cycles**? Classical complexity $T = \Theta(\log^\star n)$
  - no! *There is a finitely-dependent distribution that* $\mathbf{3}$*-colors paths and cycles* ($T = O(1)$)
  - [Holroyd and Liggett, Forum of Mathematics, Pi ’14]
  - [Holroyd et al., Electronic Communications in Probability ’18]

Some results: bounded-dependence model

- Can we similarly **rule out quantum advantage for 3-coloring cycles**? Classical complexity $T = \Theta(\log^\star n)$
  - no! *There is a finitely-dependent distribution that 3-colors paths and cycles ($T = O(1)$)*
  - [Holroyd and Liggett, Forum of Mathematics, Pi '14]
  - [Holroyd et al., Electronic Communications in Probability '18]

- Is there **any quantum-LOCAL algorithm that 3-colors paths and cycles with locality** $T = o(\log^\star n)$?
  - major open question

Some results: bounded-dependence model

- Can we similarly **rule out quantum advantage for 3-coloring cycles**? Classical complexity $T = \Theta(\log^\star n)$
  - no! *There is a finitely-dependent distribution that 3-colors paths and cycles ($T = O(1)$)*
  - [Holroyd and Liggett, Forum of Mathematics, Pi '14]
  - [Holroyd et al., Electronic Communications in Probability '18]

- Is there **any quantum-LOCAL algorithm that 3-colors paths and cycles with locality** $T = o(\log^\star n)$?
  - major open question

- Is there any hope to **rule out quantum advantage for LCLs of complexity $\Theta(\log^\star n)$ in classical LOCAL**?
  * using stronger models

Some results: bounded-dependence model

- Can we similarly **rule out quantum advantage for 3-coloring cycles**? Classical complexity $T = \Theta(\log^\star n)$

  - no! *There is a finitely-dependent distribution that* 3*-colors paths and cycles* ($T = O(1)$)
  - [Holroyd and Liggett, Forum of Mathematics, Pi '14]
  - [Holroyd et al., Electronic Communications in Probability '18]

- Is there **any quantum-LOCAL algorithm that 3-colors paths and cycles with locality** $T = o(\log^\star n)$?

  - major open question

- Is there any hope to **rule out quantum advantage for LCLs of complexity $\Theta(\log^\star n)$ in classical LOCAL**?
  * using stronger models

  - no!
  - For any LCL Π on bounded degree graphs, *there is a finitely-dependent distribution* ($T = O(1)$) *solving* Π
  - [Akbari et al., 2024]

Finitely-dependent distributions for $O(\log^\star n)$-LCLs

- We **build on the 3-coloring distributions for paths and cycles**

  - [Holroyd and Liggett, Forum of Mathematics, Pi ’14]
  - [Holroyd et al., Electronic Communications in Probability ’18]

- Inspired by $(\Delta + 1)$-coloring [Goldberg et al., SICOMP ’88; Panconesi and Rizzi, Dist. Comp. ’01]

Finitely-dependent distributions for $O(\log^\star n)$-LCLs

- We **build on the 3-coloring distributions for paths and cycles**
  - [Holroyd and Liggett, Forum of Mathematics, Pi '14]
  - [Holroyd et al., Electronic Communications in Probability '18]

- Inspired by $(\Delta+1)$-coloring [Goldberg et al., SICOMP '88; Panconesi and Rizzi, Dist. Comp. '01]

- **Fact**: $O(\log^\star n)$-round LOCAL algorithm = find distance-k coloring ($O(\log^\star n)$) + $O(1)$-round LOCAL algorithm
  - [Folklore]

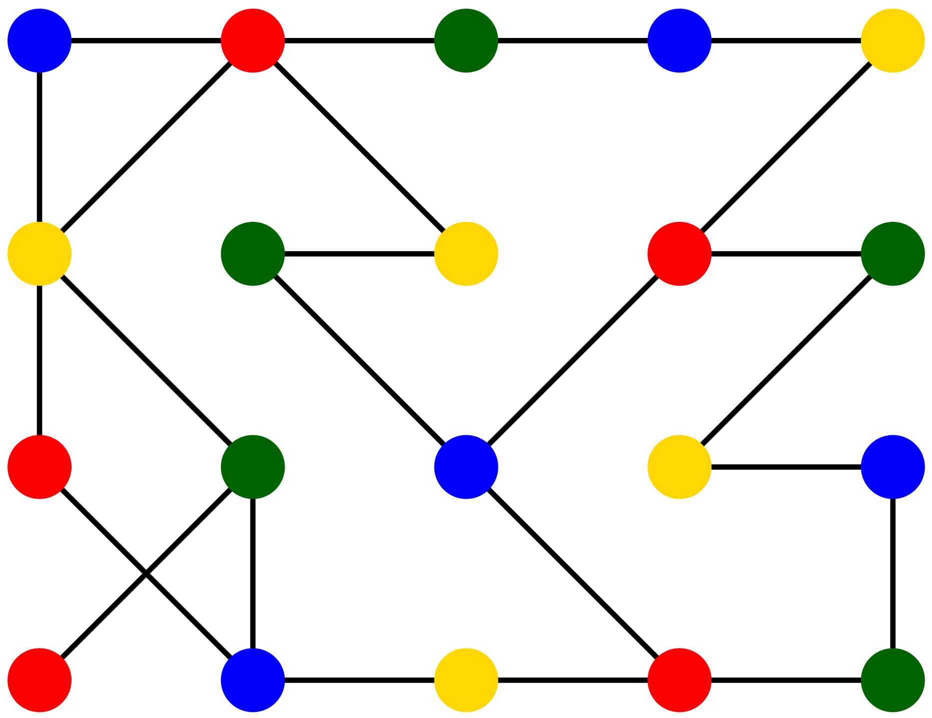
Finitely-dependent distributions for $O(\log^\star n)$-LCLs

- We **build on the 3-coloring distributions for paths and cycles**
  - [Holroyd and Liggett, Forum of Mathematics, Pi '14]
  - [Holroyd et al., Electronic Communications in Probability '18]

- Inspired by $(\Delta + 1)$-coloring [Goldberg et al., SICOMP '88; Panconesi and Rizzi, Dist. Comp. '01]

- **Fact**: $O(\log^\star n)$-round LOCAL algorithm = find distance-k coloring ($O(\log^\star n)$) + $O(1)$-round LOCAL algorithm
  - [Folklore]

distance-2 3-coloring



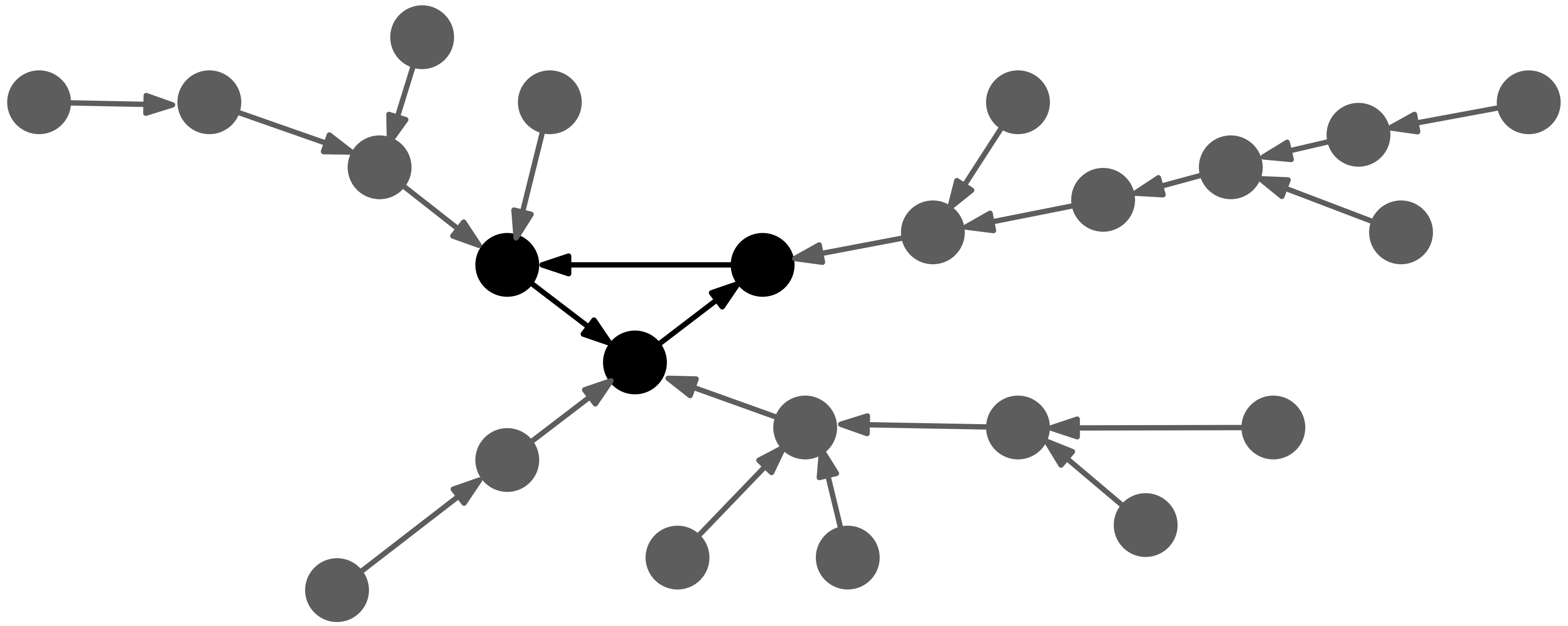
+ constant-round LOCAL algorithm

Finitely-dependent distance-k coloring

- **Step 1**: finitely-dependent 3-coloring of rooted pseudoforests

  - a rooted pseduforoest can be decomposed in node-disjoint paths and cycles

rooted pseudotree



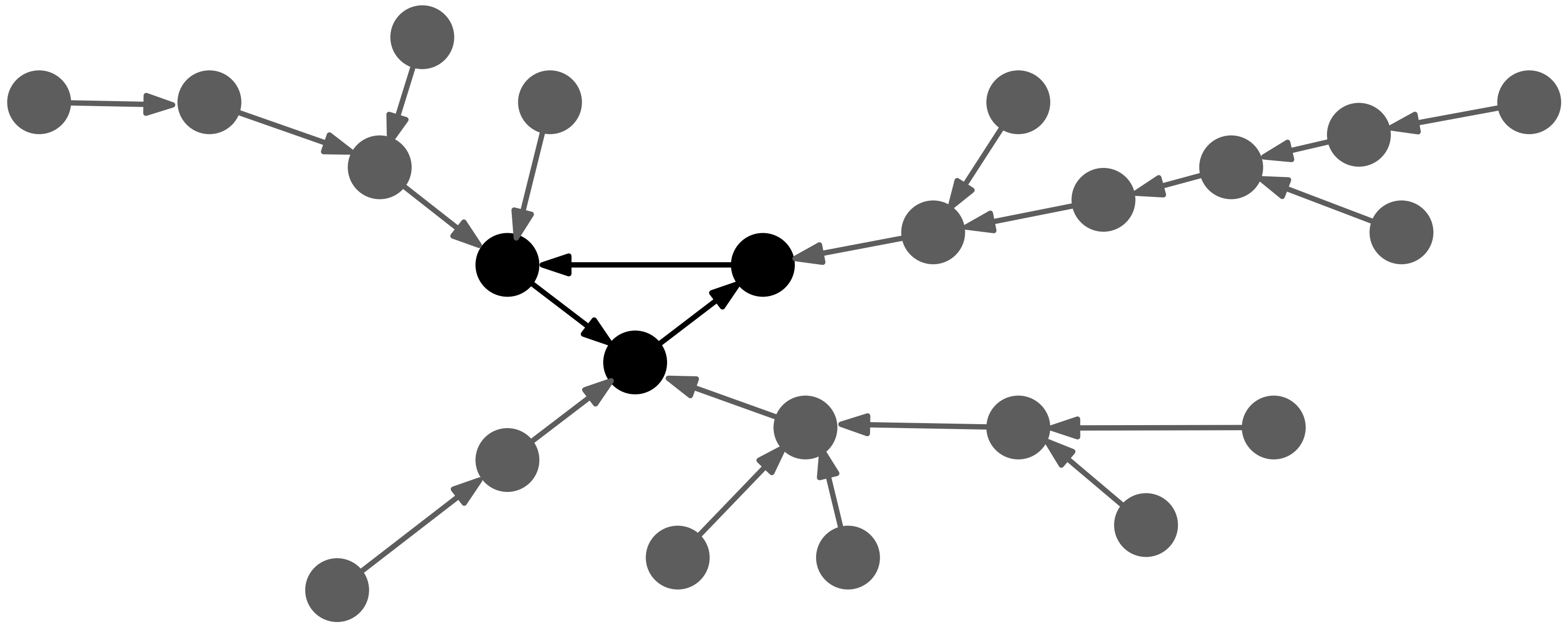
Finitely-dependent distance-k coloring

- **Step 1**: finitely-dependent 3-coloring of rooted pseudoforests

  - a rooted pseduforoest can be decomposed in node-disjoint paths and cycles

- **Step 2**: finitely-dependent $(\Delta+1)$-coloring of bounded-degree graphs

  - a graph can be decomposed in edge-disjoint rooted pseudoforests



rooted pseudotree

Finitely-dependent distance-k coloring

- **Step 1**: finitely-dependent 3-coloring of rooted pseudoforests

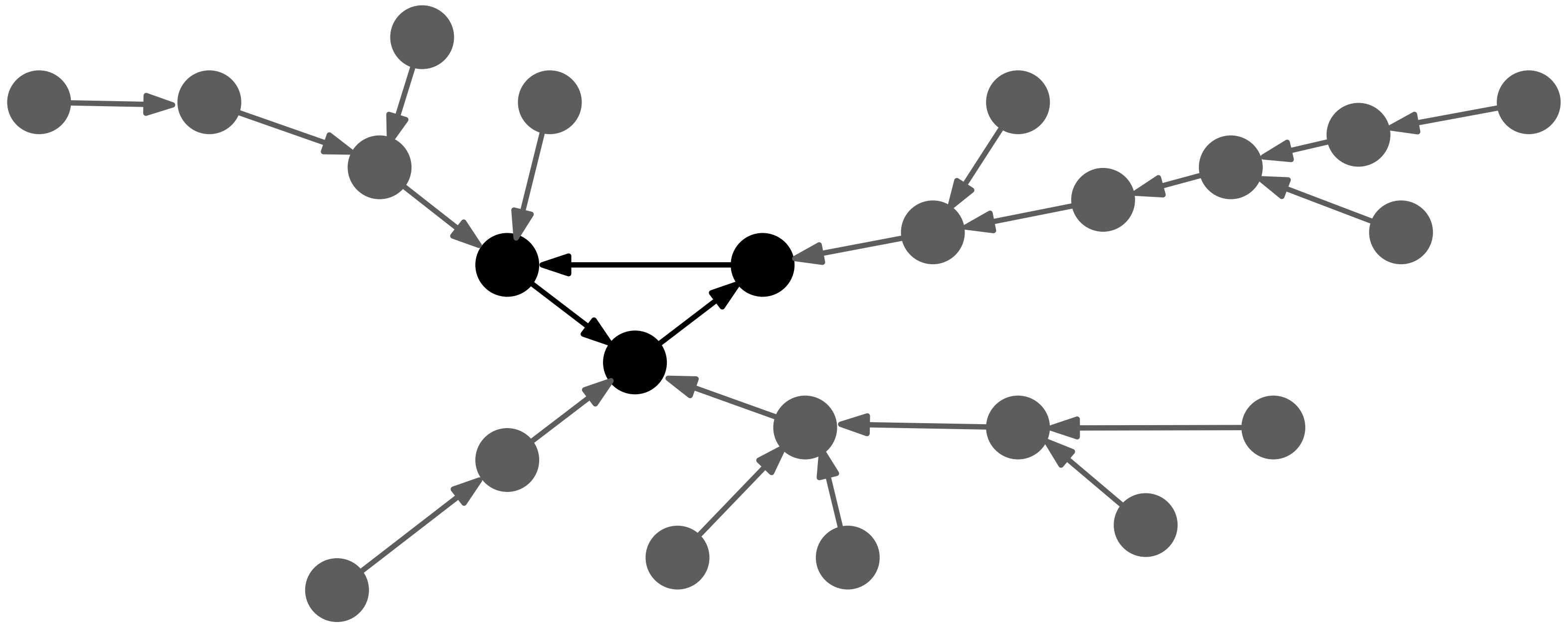
  - a rooted pseduforoest can be decomposed in node-disjoint paths and cycles

- **Step 2**: finitely-dependent $(\Delta + 1)$-coloring of bounded-degree graphs

  - a graph can be decomposed in edge-disjoint rooted pseudoforests

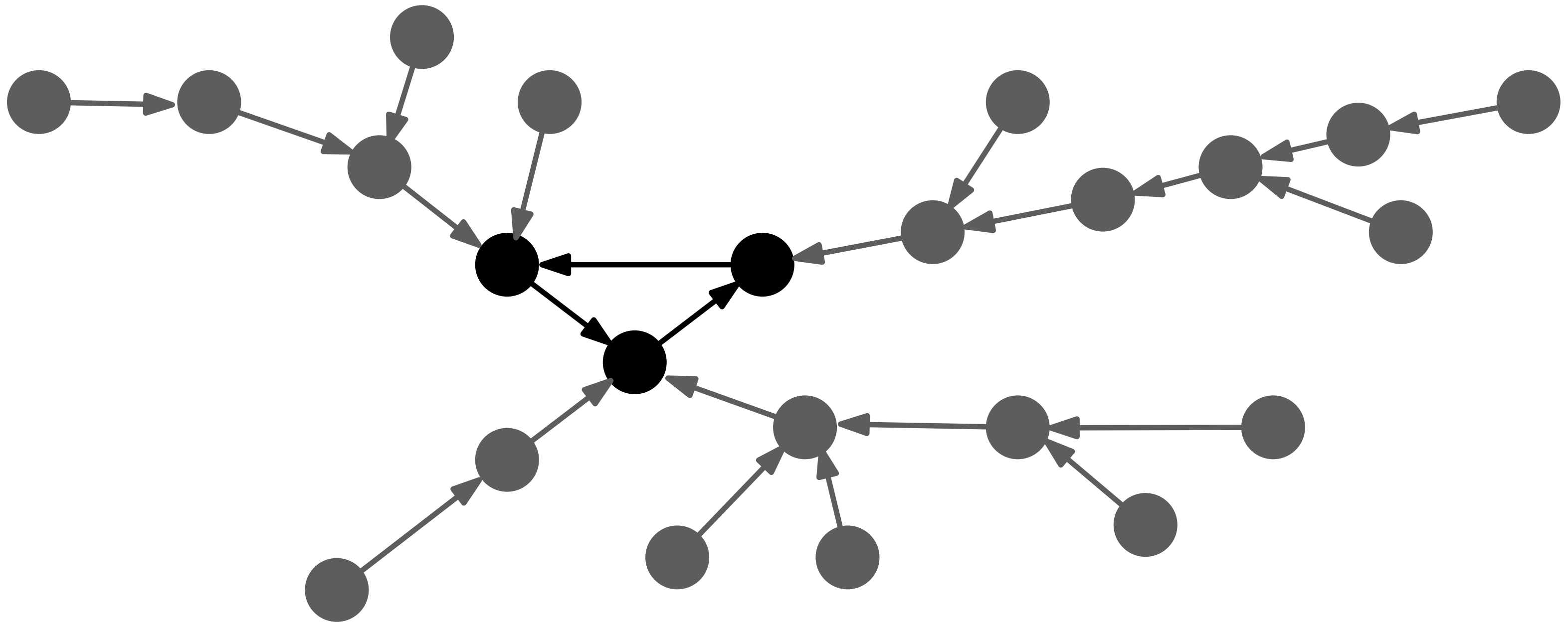
- **Step 3**: apply the $(\Delta + 1)$-coloring in G^k to get distance-k coloring

rooted pseudotree



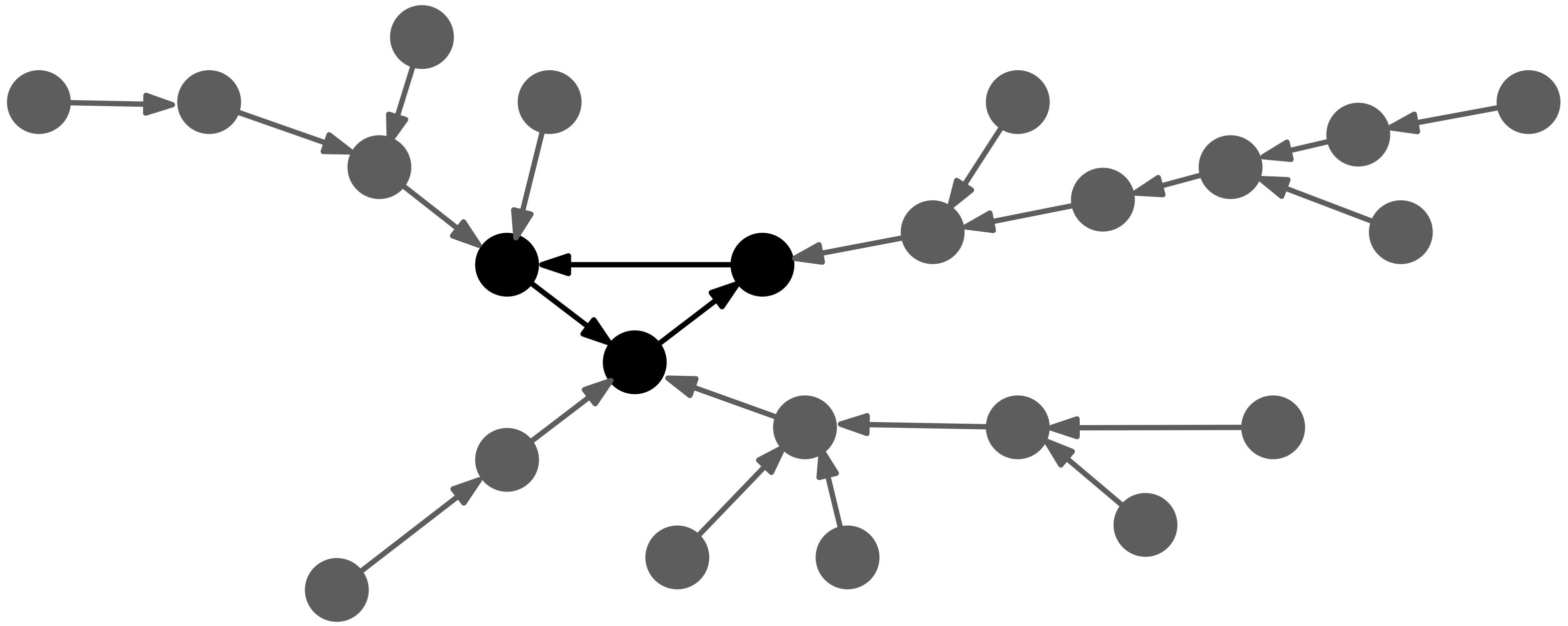
Step 1: finitely-dependent coloring of rooted pseudotree

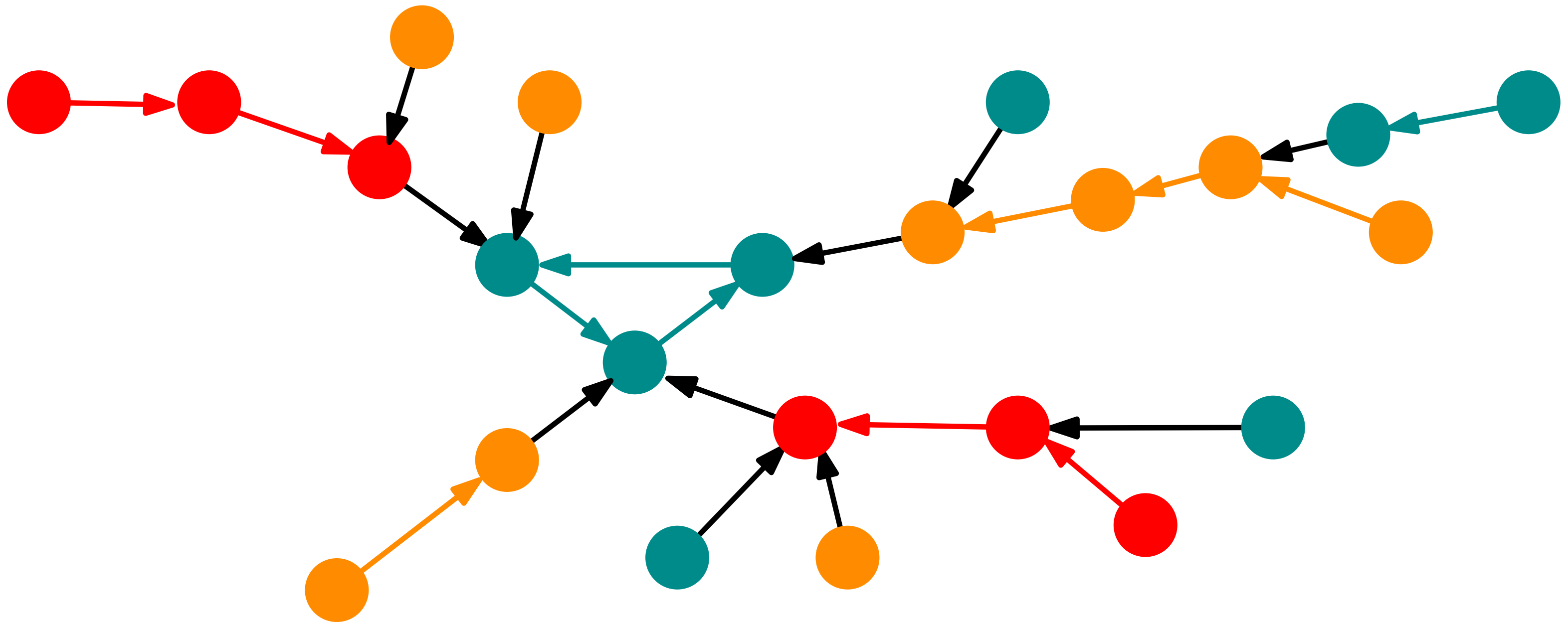
- Each node u colors u.a.r. its in-degree neighbors with colors in $[\text{indeg}(u)]$



Step 1: finitely-dependent coloring of rooted pseudotree

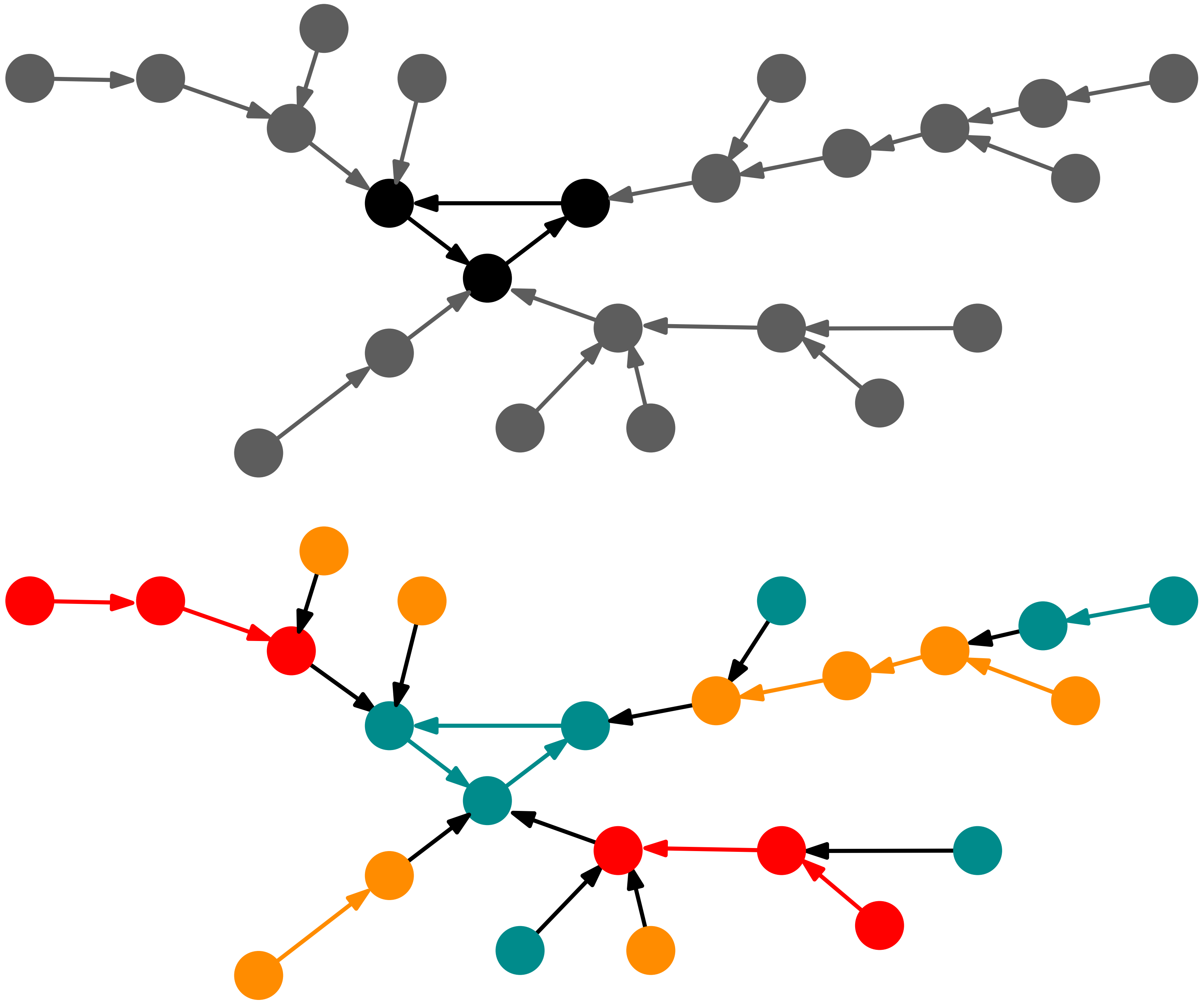
- Each node u colors u.a.r. its in-degree neighbors with colors in $[\text{indeg}(u)]$

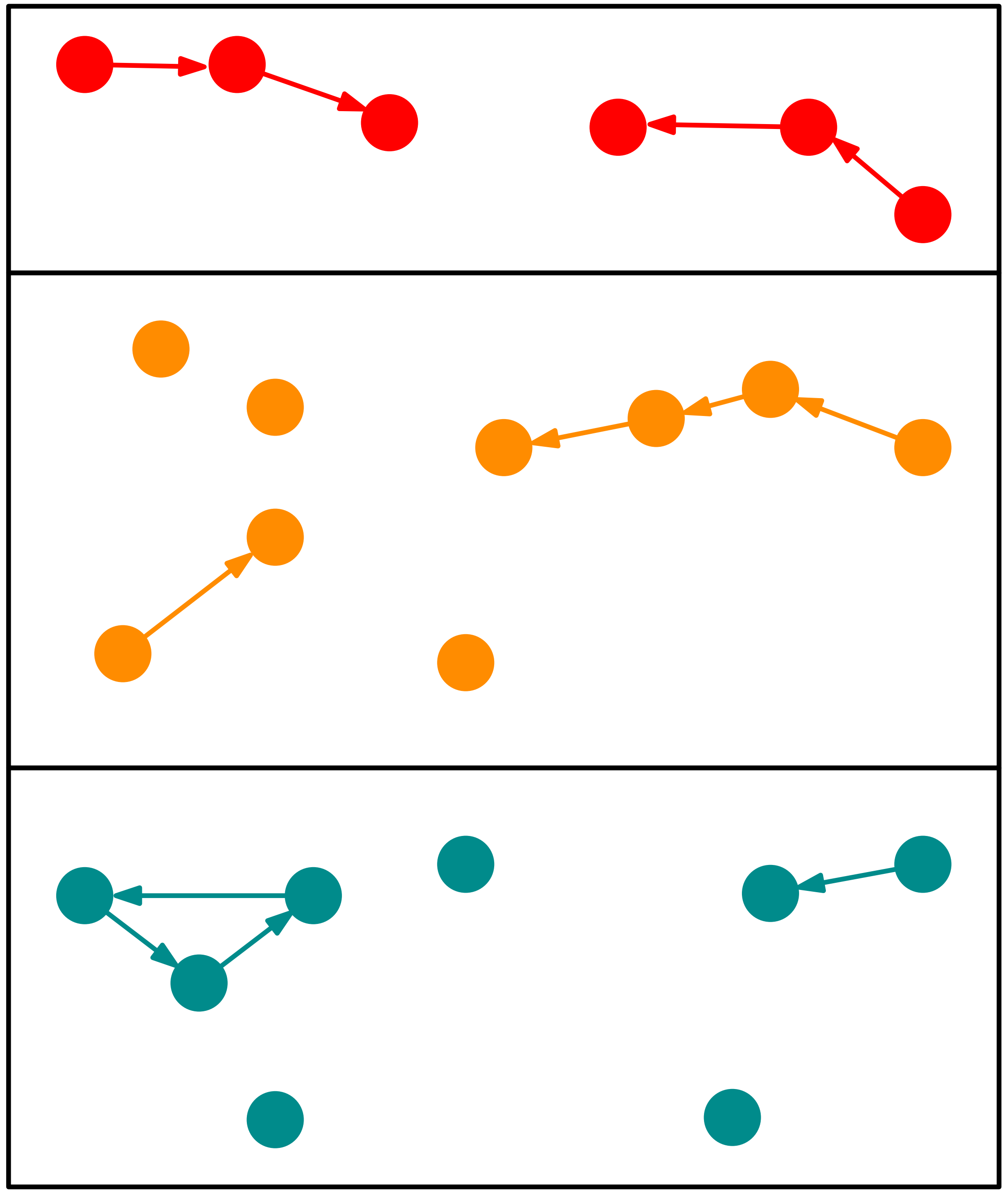




Step 1: finitely-dependent coloring of rooted pseudotree

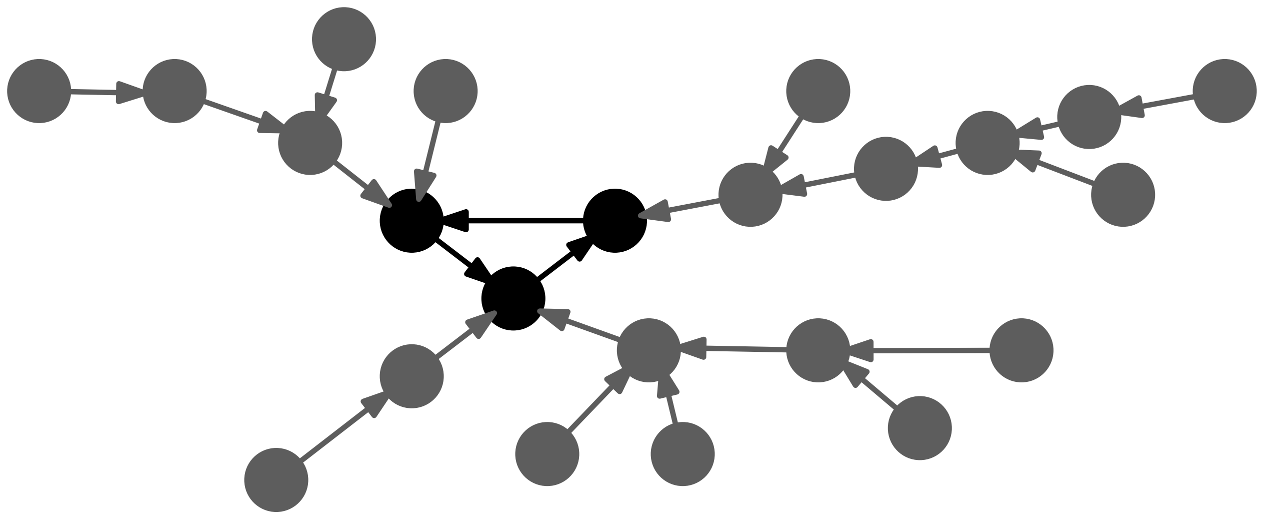
- Each node u colors u.a.r. its in-degree neighbors with colors in $[\mathrm{indeg}(u)]$



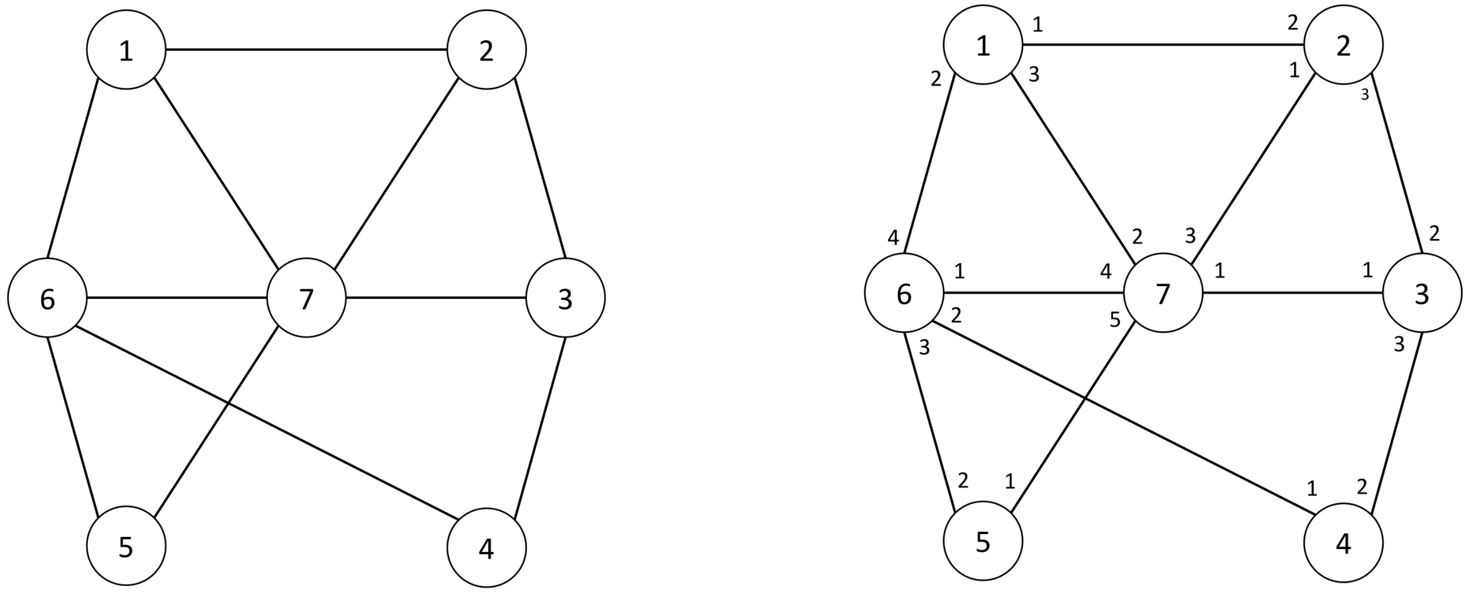


Step 2: finitely-dependent coloring of bounded-degree graph

rooted pseudotree

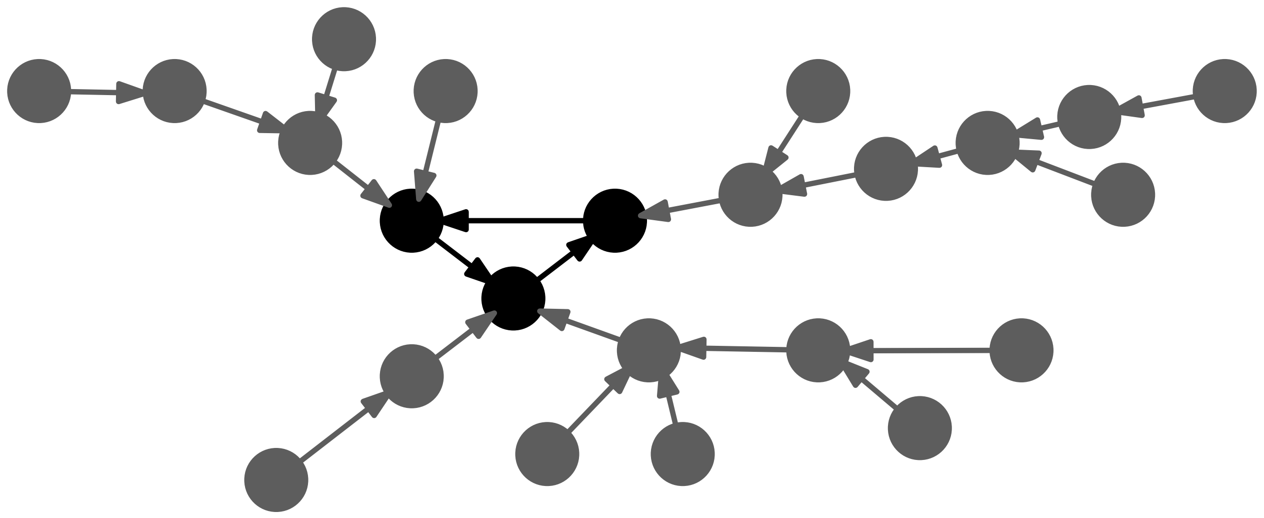


bounded-degree graph and random port-numbering

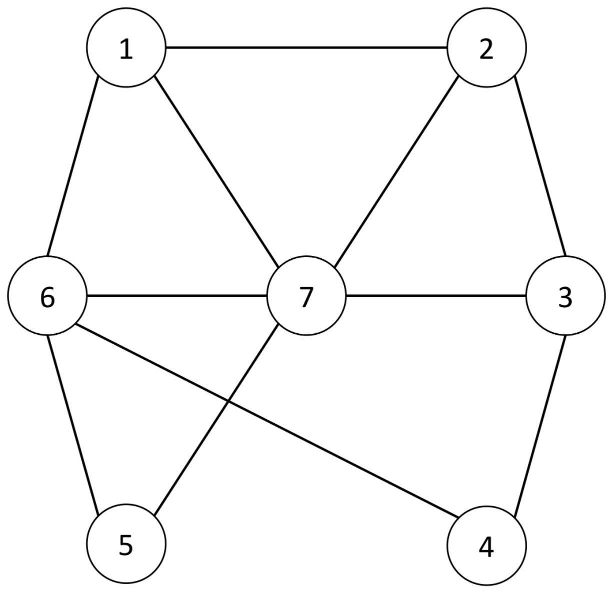


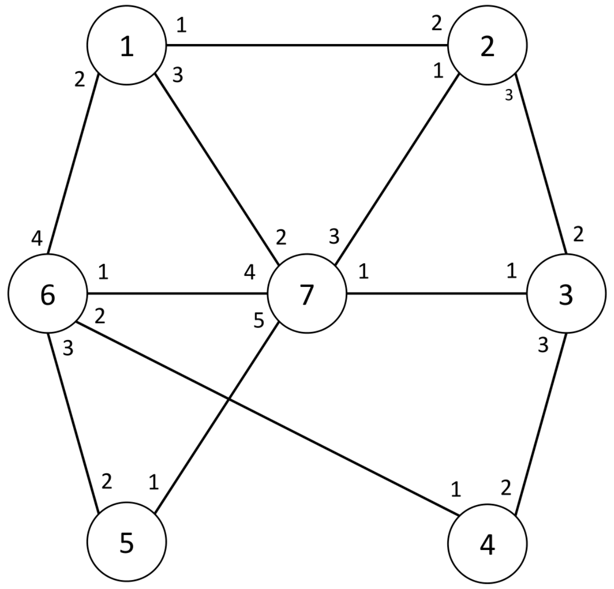
Step 2: finitely-dependent coloring of bounded-degree graph

rooted pseudotree



bounded-degree graph and random port-numbering





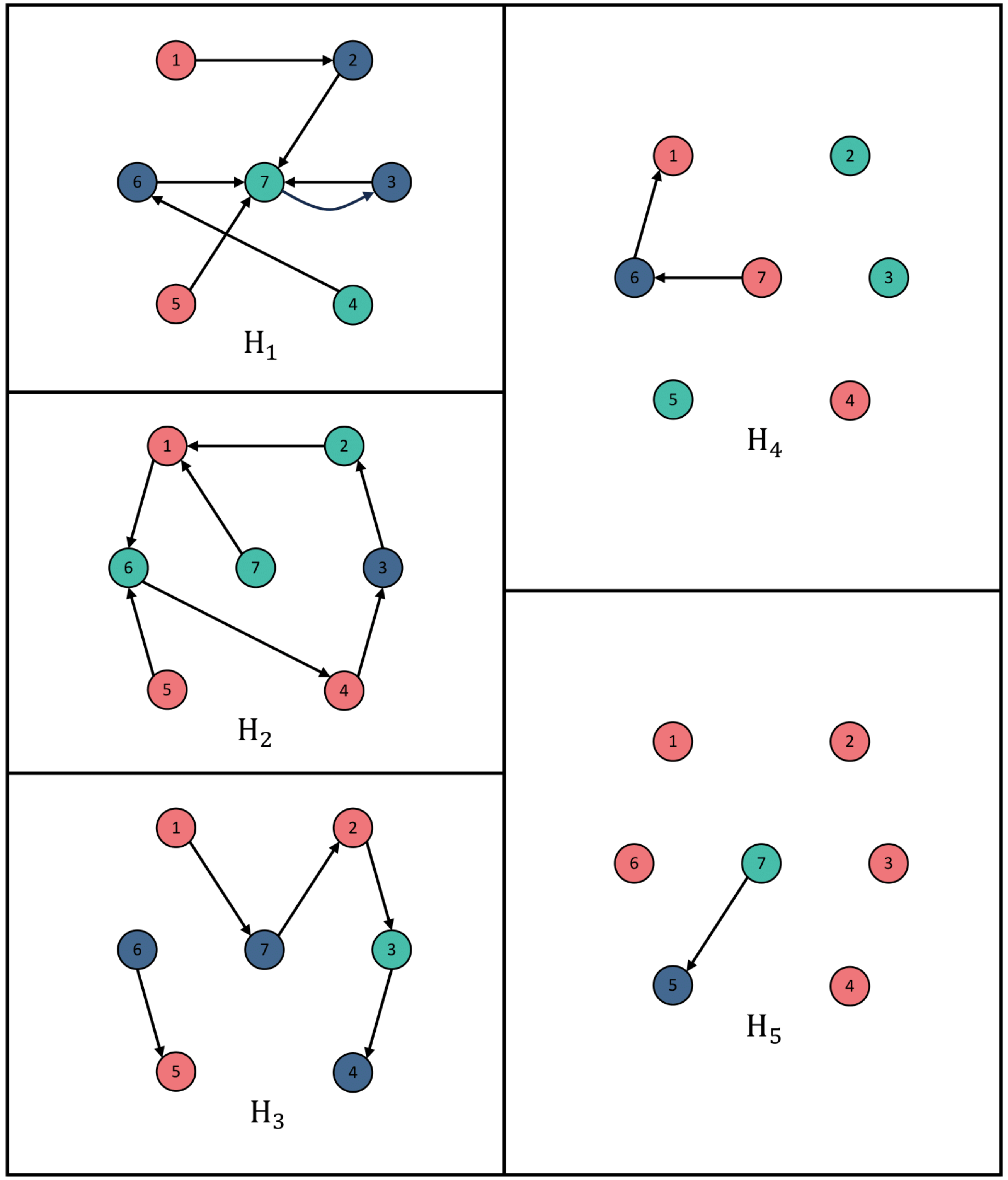


Table of content

1. **Intro**

- The LOCAL model of computation
- Locally checkable labeling (LCL) problems

2. **Quantum and causality-based models**

- The non-signaling model & bounded-dependence model
- State-of-the-art lower bounds & upper bounds

3. **Locality-based models**

- The online-LOCAL model
- Relation with causality-based models
- Simulation in weaker models

4. **Conclusions and open problems**

Other locality-based models

sequential-
LOCAL
(SLOCAL)

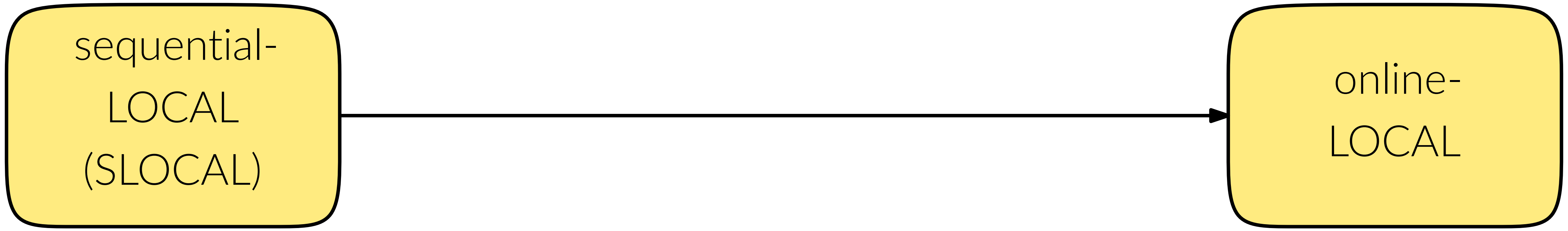
- Similar to LOCAL, but **sequential**

Locality T

- adversary picks a node (each node only once)
- the algorithm gets access to radius-T neighborhood

[Ghaffari et al., STOC '17]

Other locality-based models



- Similar to LOCAL, but **sequential**

Locality T

- adversary picks a node (each node only once)
- the algorithm gets access to radius-T neighborhood

[Ghaffari et al., STOC '17]

- Similar to sequential-LOCAL, but **centralized**

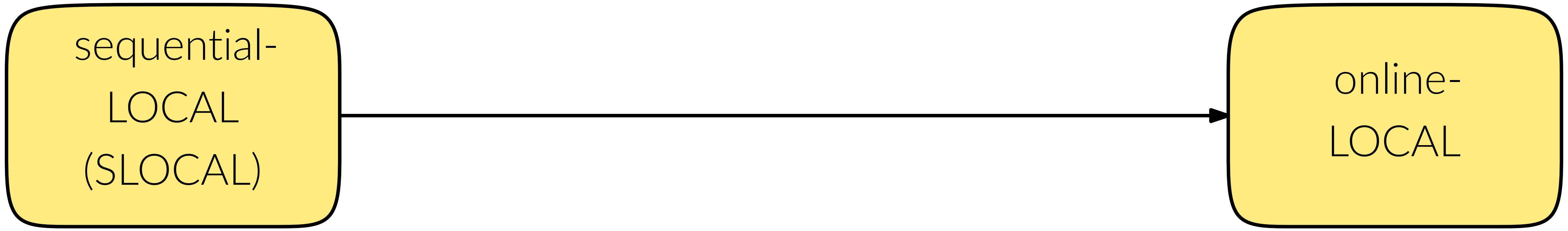
Locality T

- adversary picks a node (each node only once)
- the algorithm gets access to radius-T neighborhood

* plus keeps memory of everything seen so far

[Akbari et al., ICALP '23]

Other locality-based models



- Similar to LOCAL, but **sequential**

Locality T

- adversary picks a node (each node only once)
- the algorithm gets access to radius-T neighborhood

[Ghaffari et al., STOC '17]

- Similar to sequential-LOCAL, but **centralized**

Locality T

- adversary picks a node (each node only once)
- the algorithm gets access to radius-T neighborhood

* plus keeps memory of everything seen so far

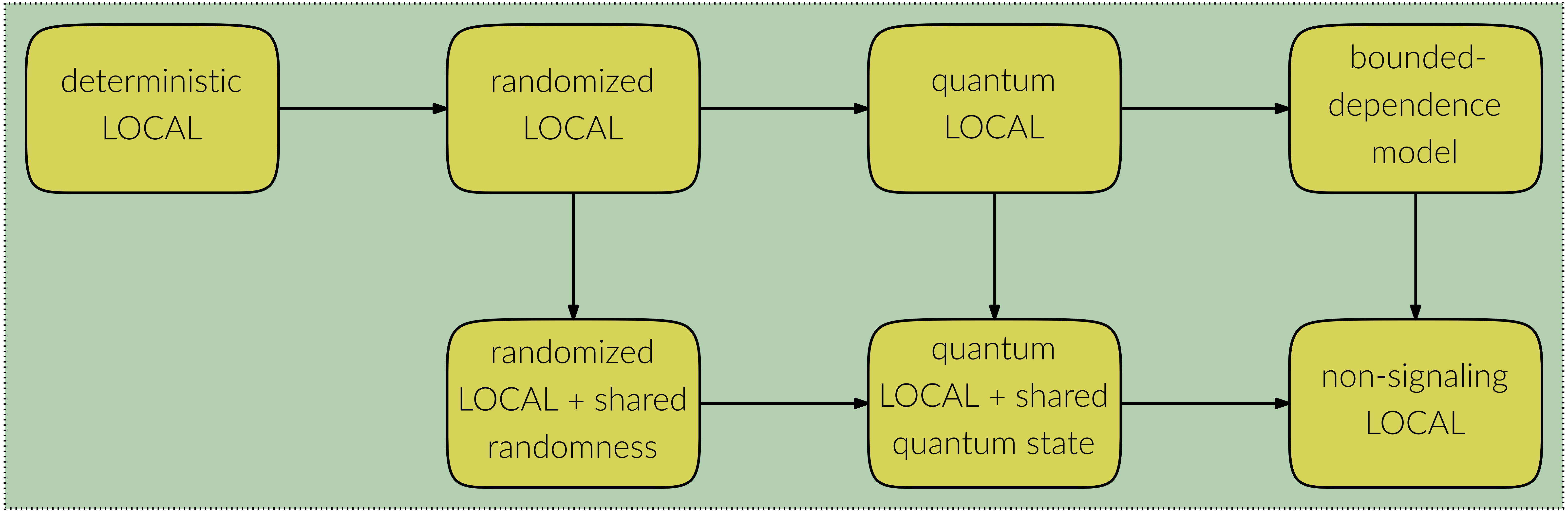
[Akbari et al., ICALP '23]

- There are also the natural **extension to randomness**:

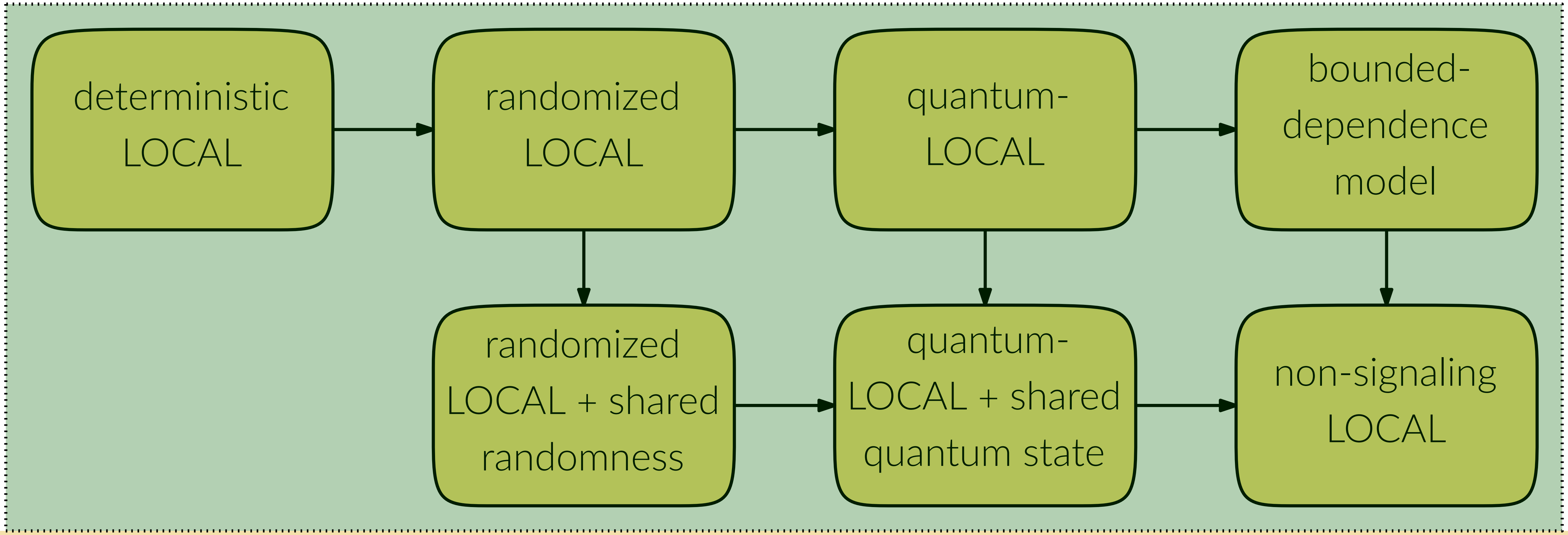
- adversary is oblivious and source of randomess is "infinite" for each node

Full landscape of models

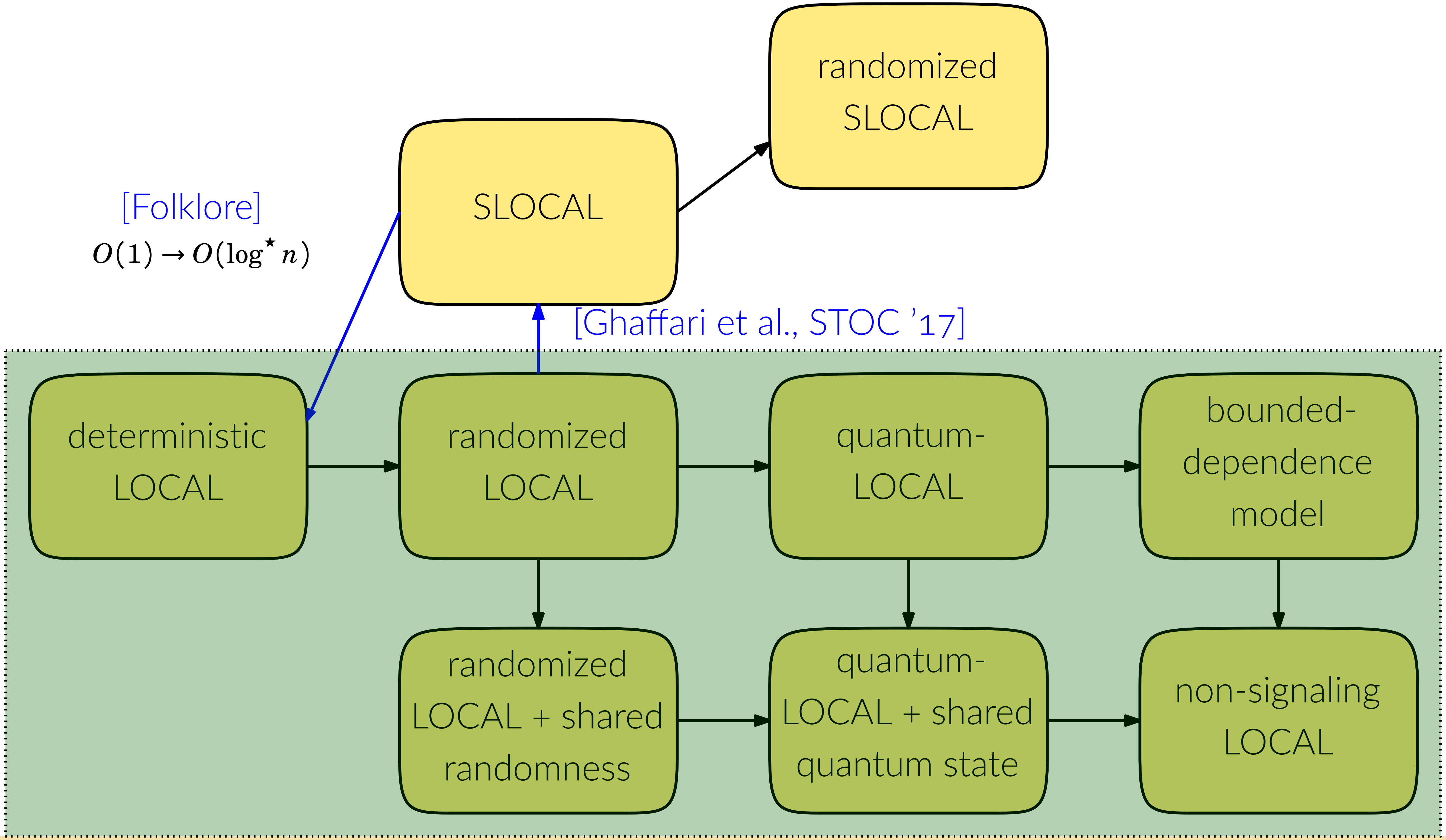
- $X \to Y$ means that locality T in X becomes locality $O(T)$ in Y



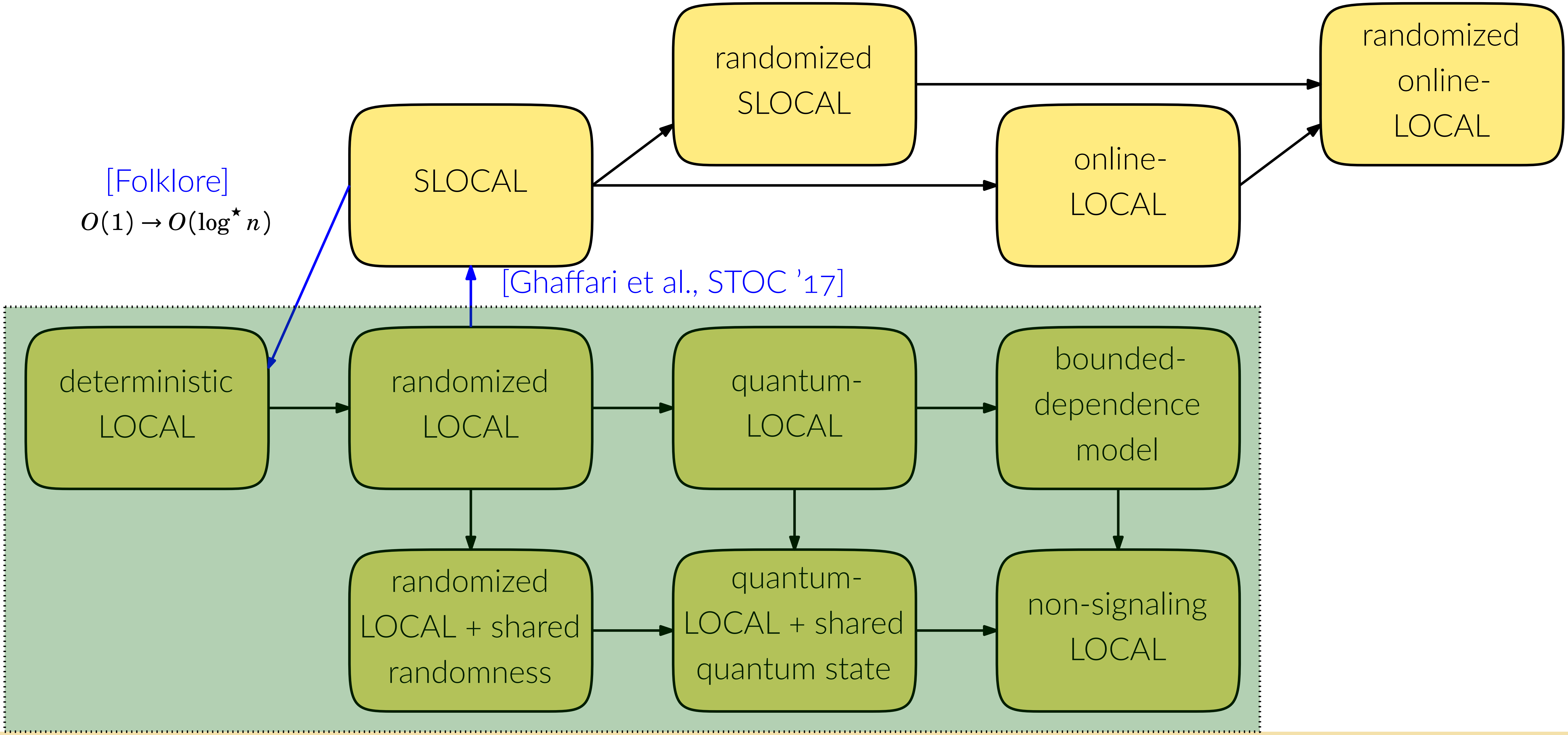
Full landscape of models



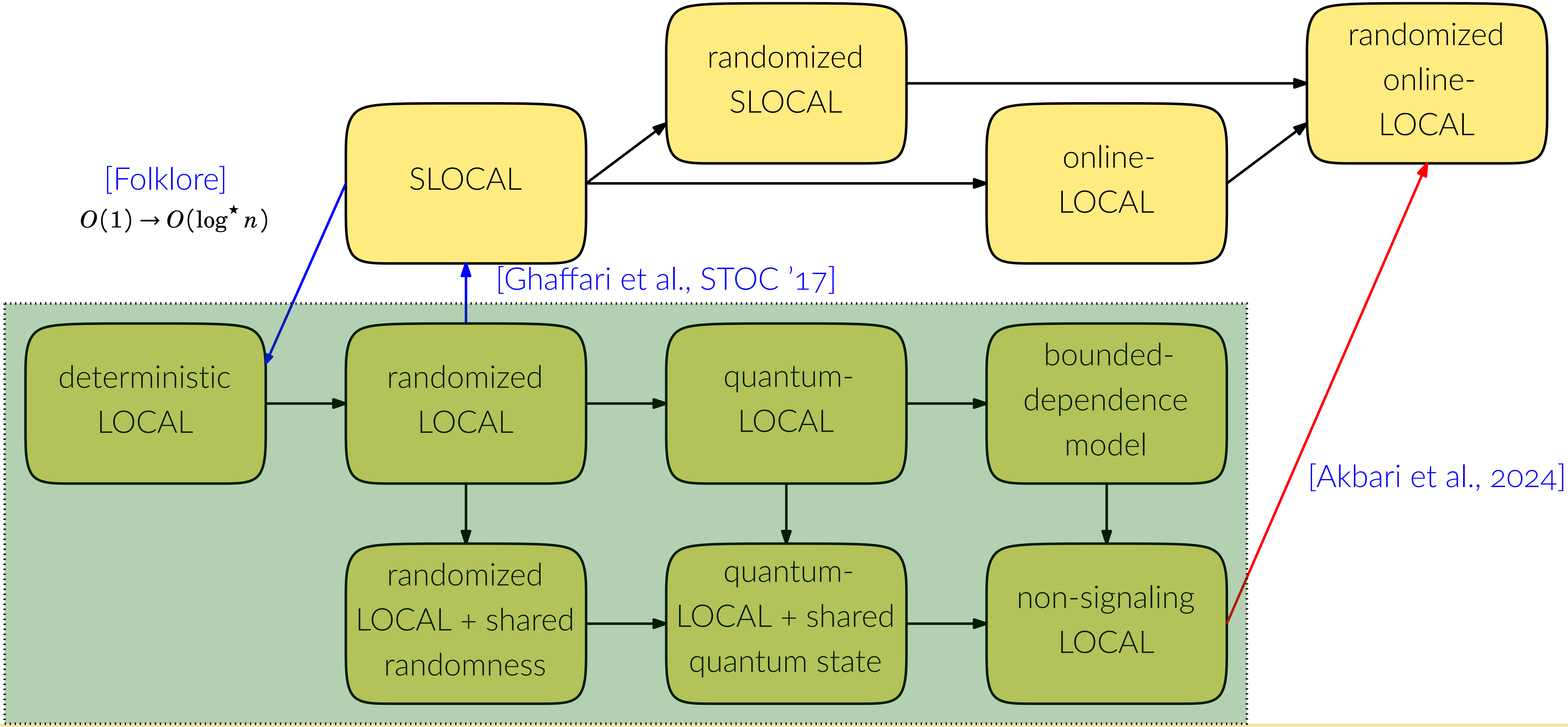
Full landscape of models



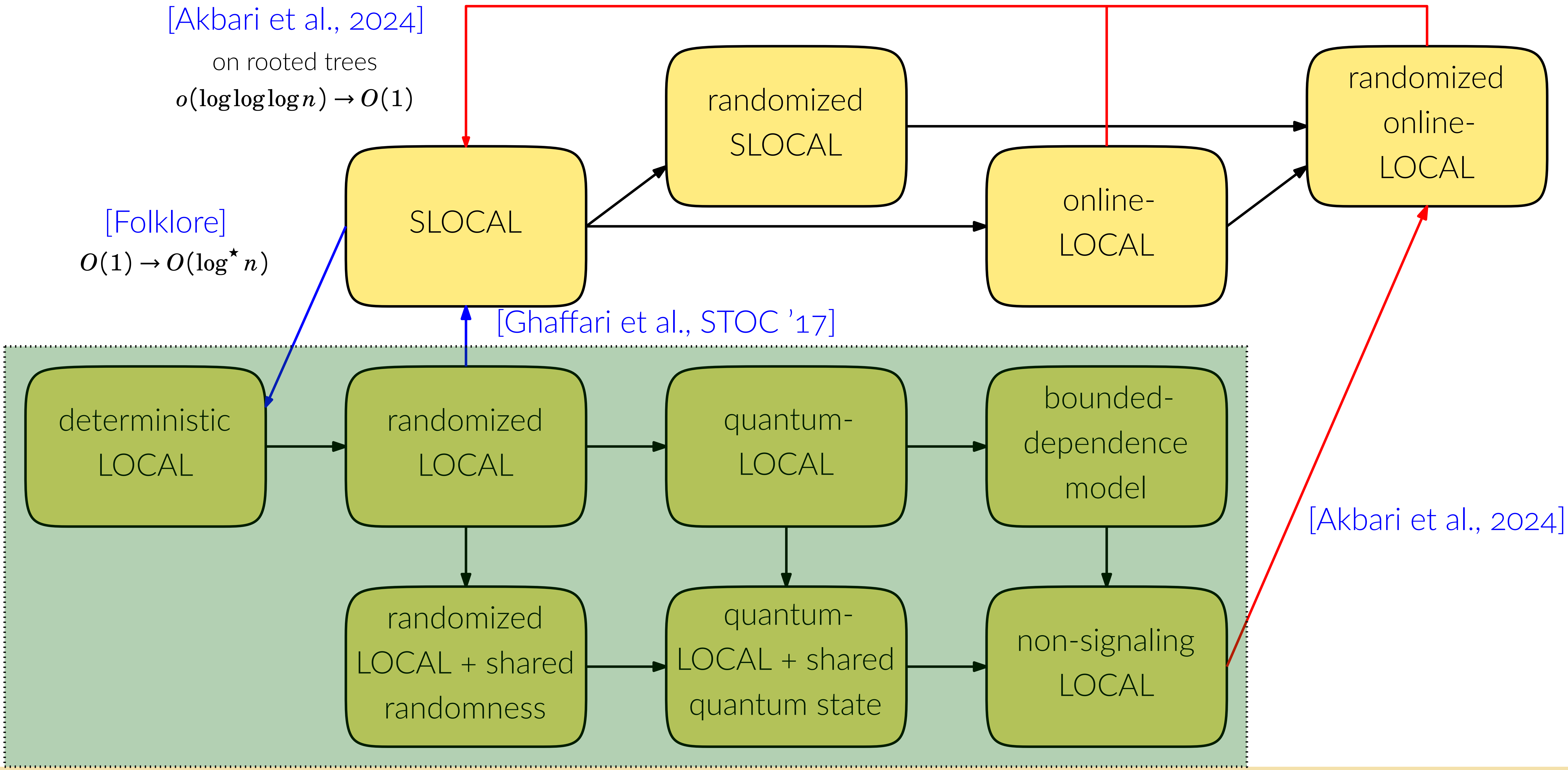
Full landscape of models



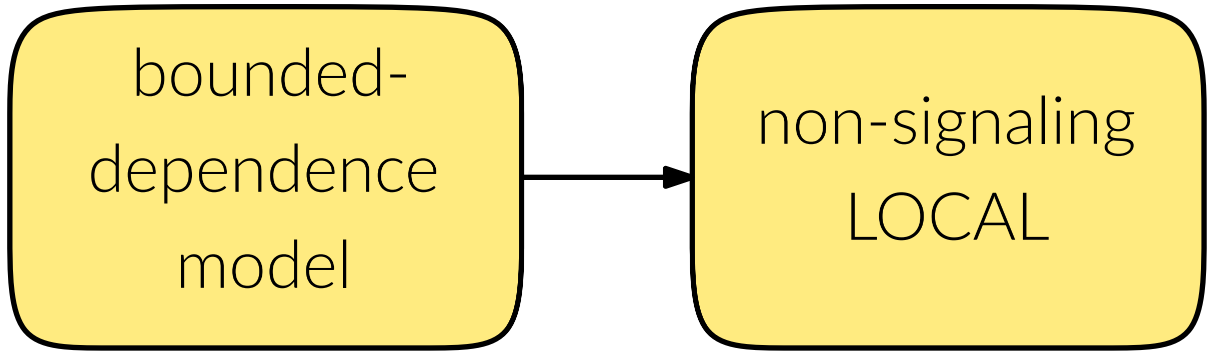
Full landscape of models



Full landscape of models



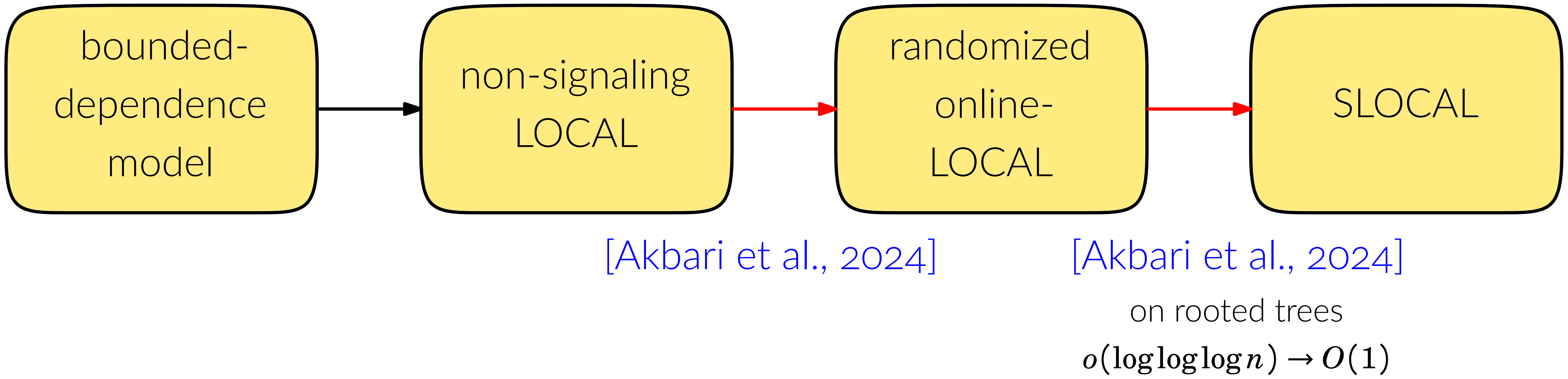
Focus on the implications



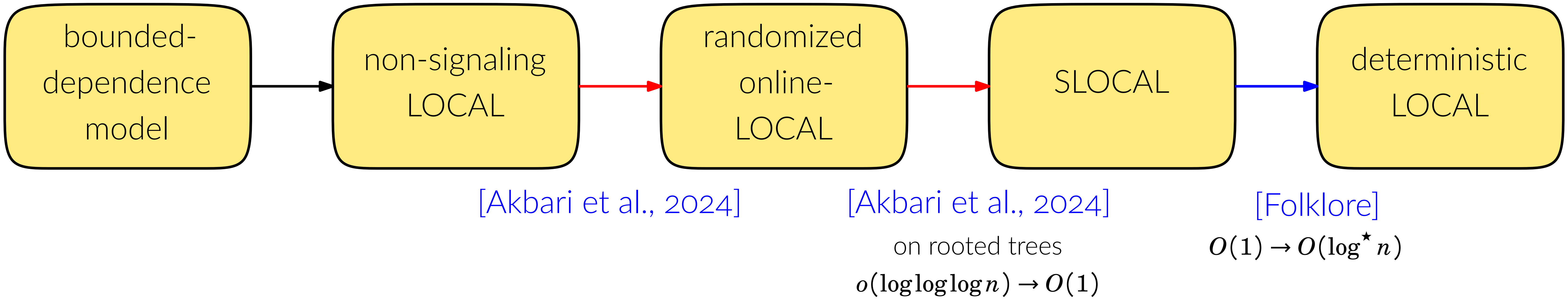
Focus on the implications

[Akbari et al., 2024]

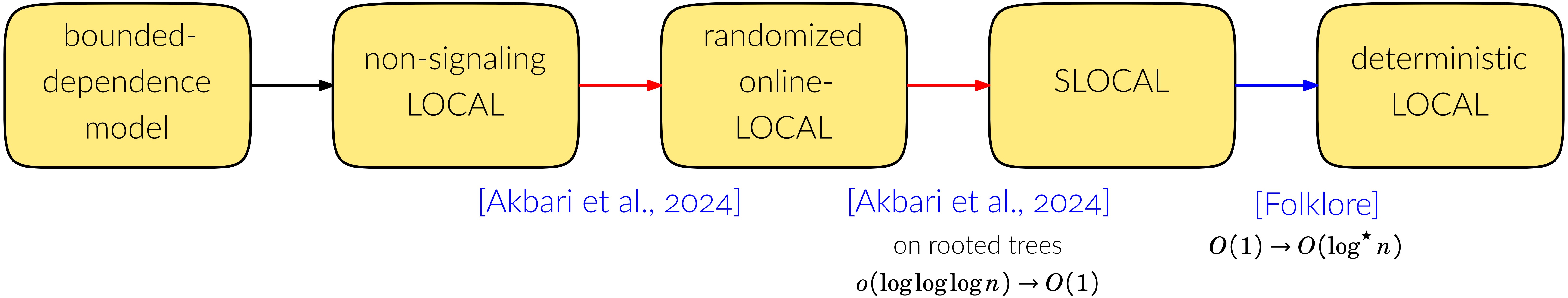
Focus on the implications

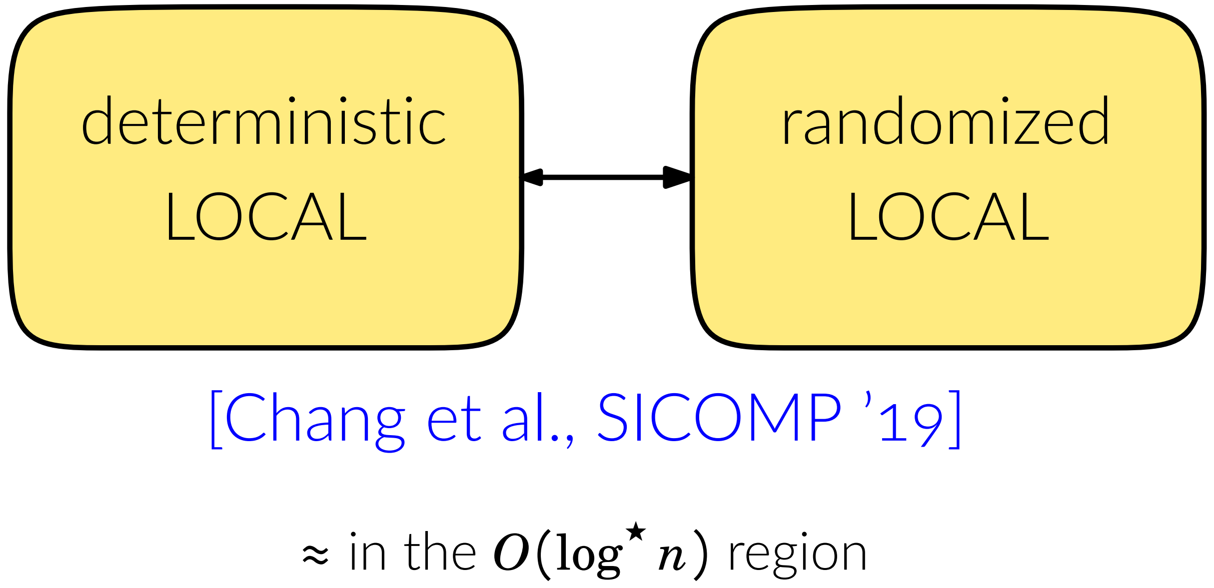


Focus on the implications

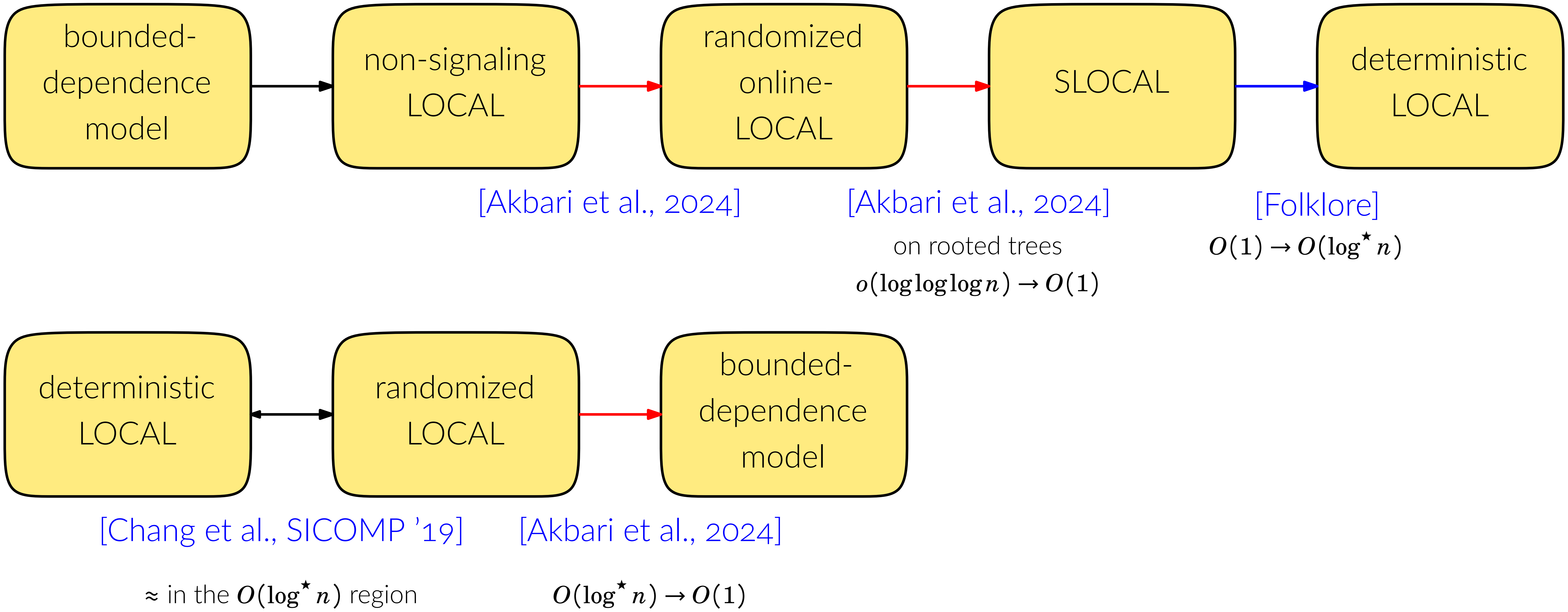


Focus on the implications

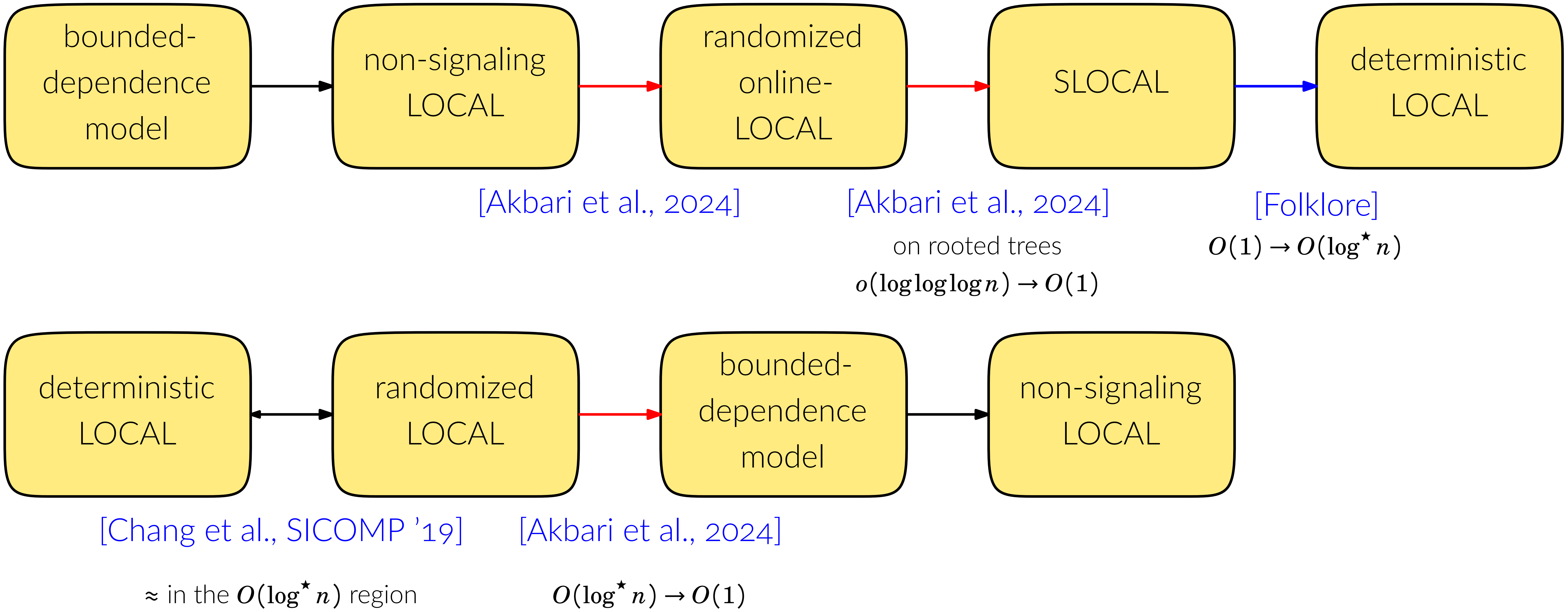




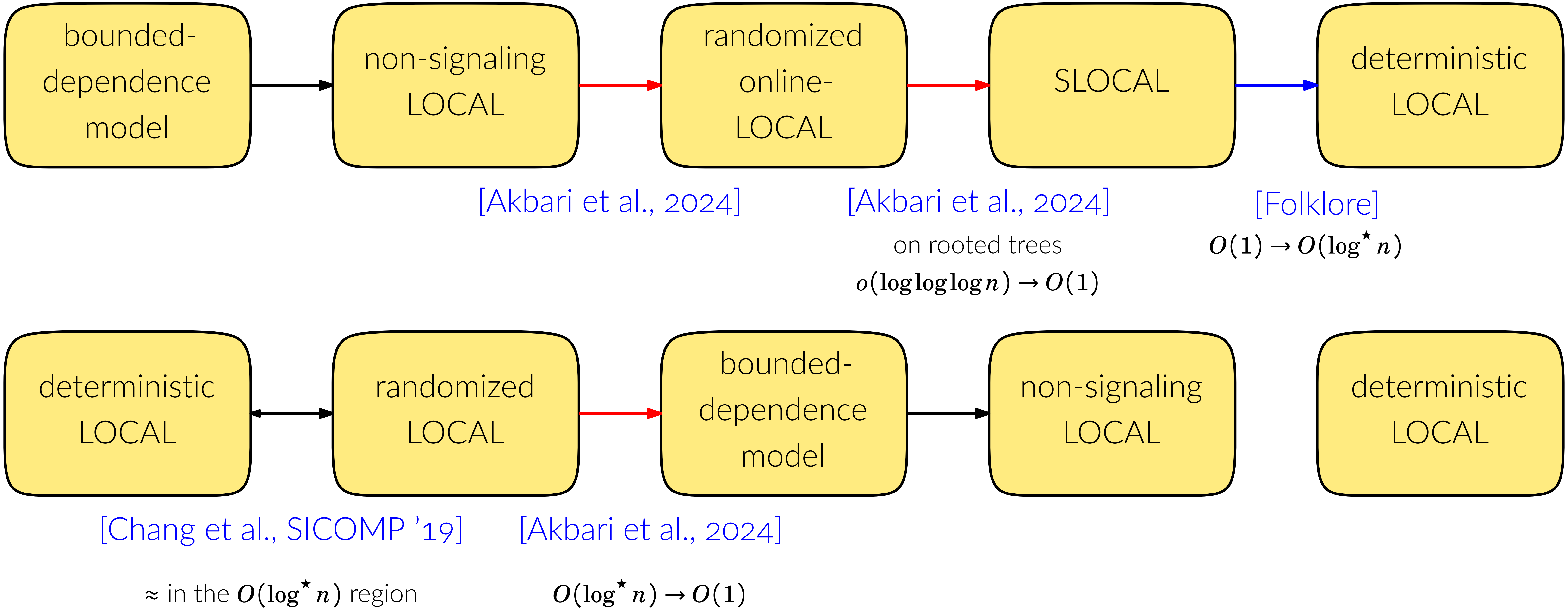
Focus on the implications



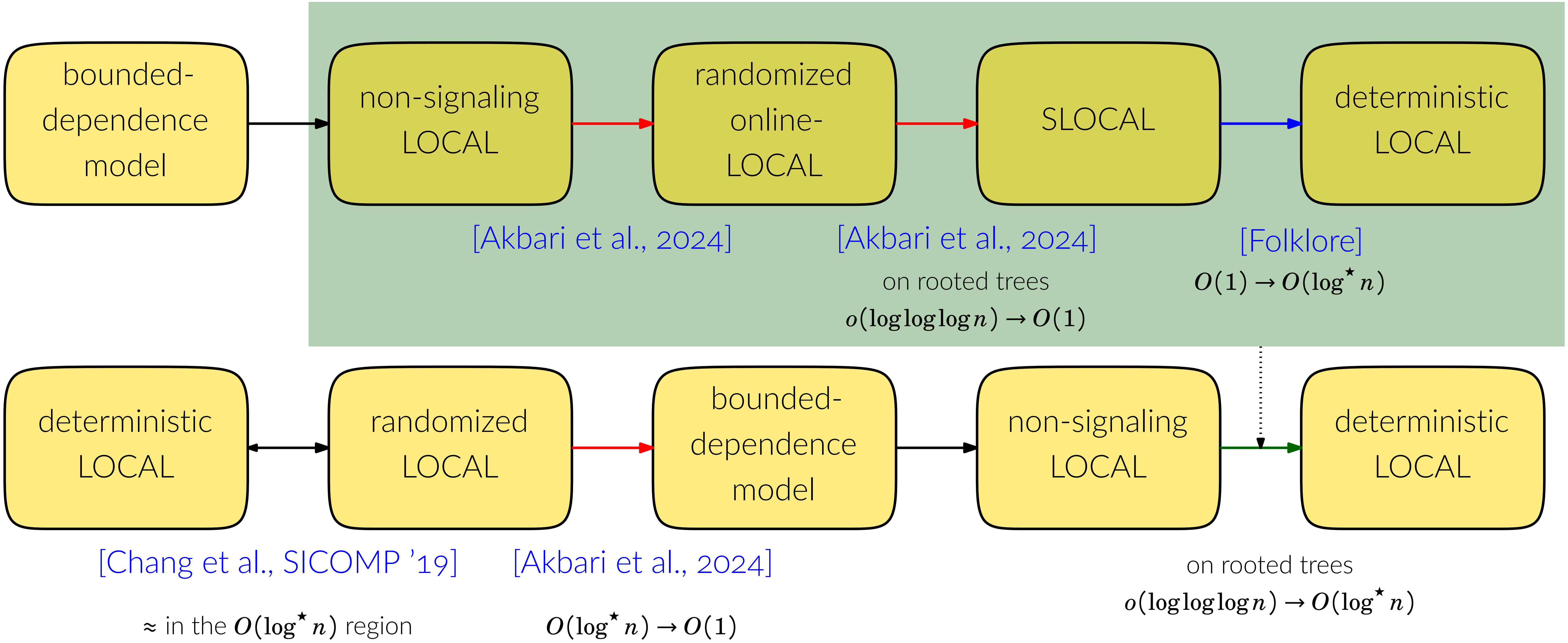
Focus on the implications



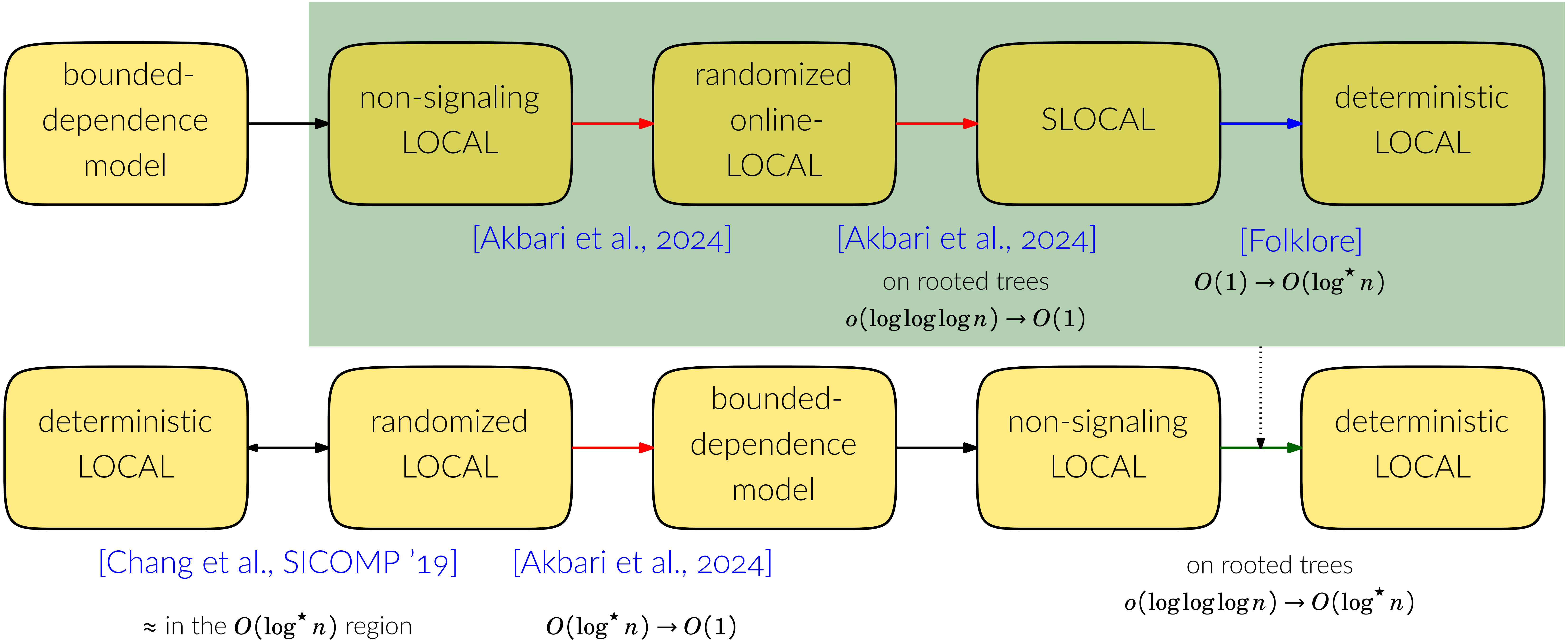
Focus on the implications



Focus on the implications



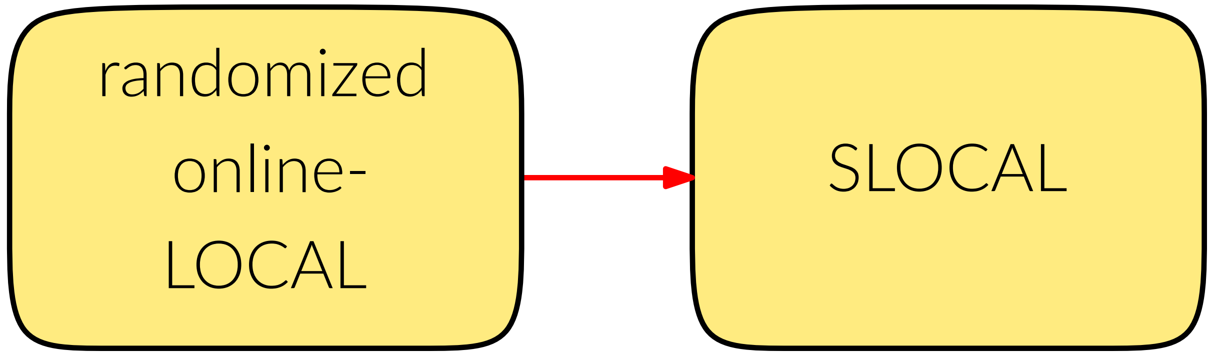
Focus on the implications



- Quantum-LOCAL $O(log^\star n) \to O(log^\star n)$ in LOCAL in **rooted trees**

From randomized online-LOCAL to sequential-LOCAL

- **Main differences** to overcome:
  - global memory vs local memory
  - randomized vs deterministic



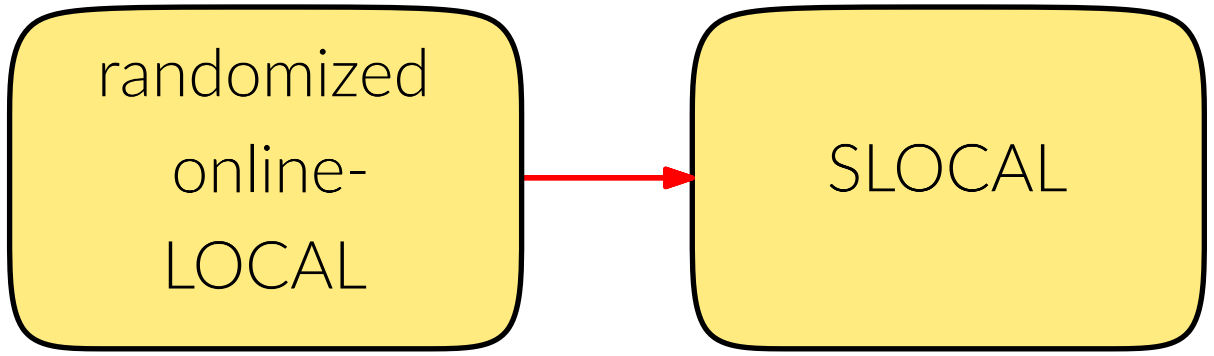
[Akbari et al., 2024]

on rooted trees

$o(\log\log\log n) \to O(1)$

From randomized online-LOCAL to sequential-LOCAL

- **Main differences** to overcome:
  - global memory vs local memory
  - randomized vs deterministic



[Akbari et al., 2024]

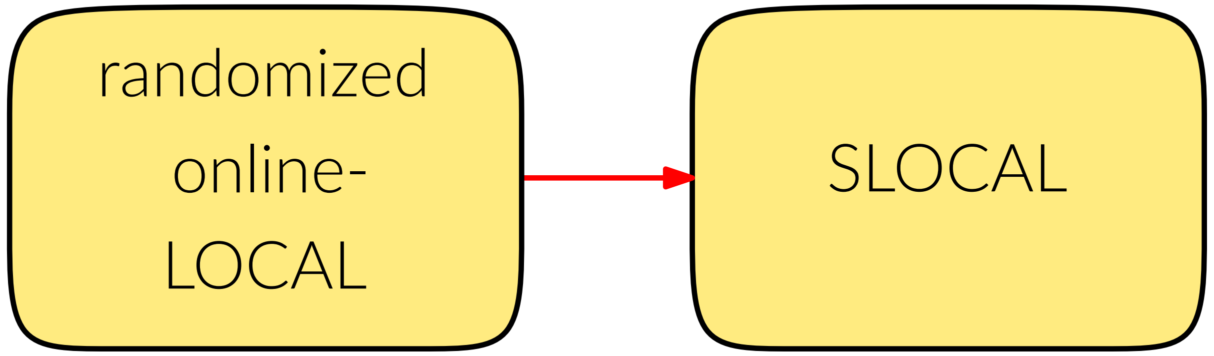
on rooted trees

$o(\log\log\log n) \to O(1)$

- **Step 1**: derandomize
  - $T(n)$-round randomized online-LOCAL $\implies T(2^{O(2^{n^2})})$-round deterministic online-LOCAL
  - while so, we obtain "component-wise" algorithms (memory only in the current connected component)

From randomized online-LOCAL to sequential-LOCAL

- **Main differences** to overcome:
  - global memory vs local memory
  - randomized vs deterministic



[Akbari et al., 2024]

on rooted trees

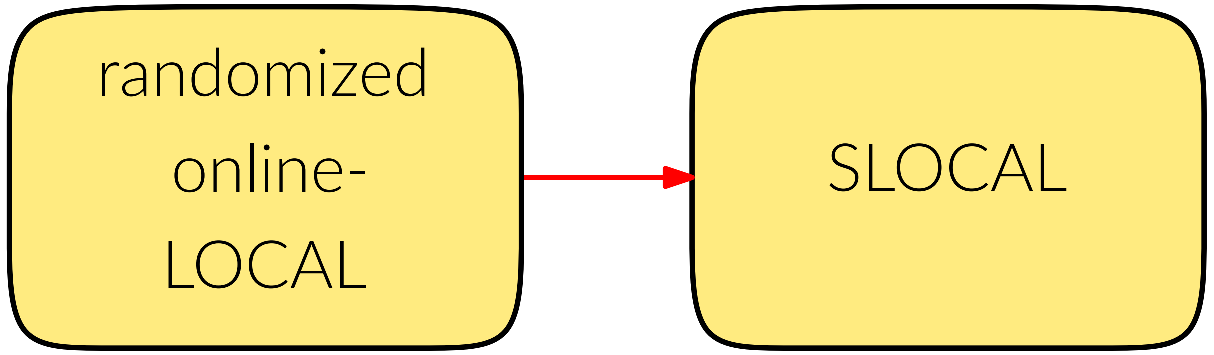
$o(\log\log\log n) \to O(1)$

- **Step 1**: derandomize
  - $T(n)$-round randomized online-LOCAL $\implies T(2^{O(2^{n^2})})$-round deterministic online-LOCAL
  - while so, we obtain "component-wise" algorithms (memory only in the current connected component)

- **Step 2**: from component-wise online-LOCAL algorithms to SLOCAL in rooted trees
  - possible without loosing locality by "splitting" the tree in "subtrees" of small size

From randomized online-LOCAL to sequential-LOCAL

- **Main differences** to overcome:
  - global memory vs local memory
  - randomized vs deterministic



[Akbari et al., 2024]

on rooted trees

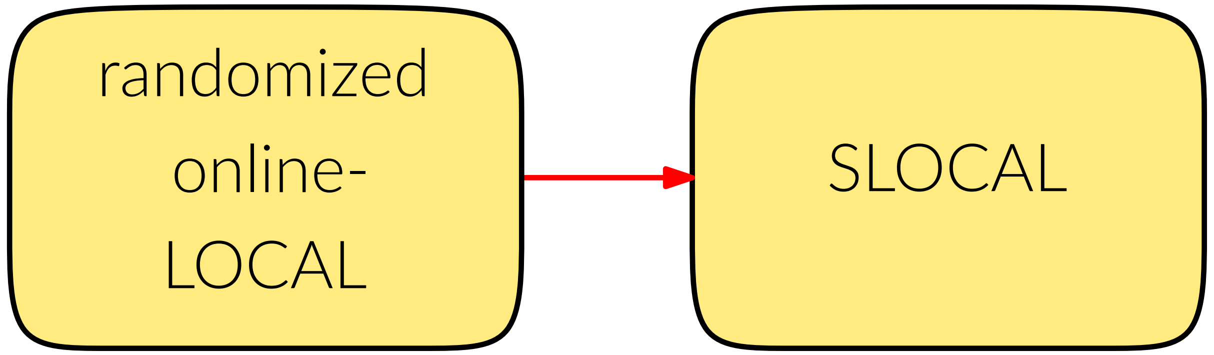
$o(\log\log\log n) \to O(1)$

- **Step 1**: derandomize
  - $T(n)$-round randomized online-LOCAL $\implies T(2^{O(2^{n^2})})$-round deterministic online-LOCAL
  - while so, we obtain "component-wise" algorithms (memory only in the current connected component)

- **Step 2**: from component-wise online-LOCAL algorithms to SLOCAL in rooted trees
  - possible without loosing locality by "splitting" the tree in "subtrees" of small size

- **Step 3**: from $o(\log n) \approx O(1)$ in rooted trees in SLOCAL

From randomized online-LOCAL to sequential-LOCAL



[Akbari et al., 2024]

on rooted trees

$o(\log\log\log n) \to O(1)$

- **Main differences** to overcome:
  - global memory vs local memory
  - randomized vs deterministic

- **Step 1**: derandomize
  - $T(n)$-round randomized online-LOCAL $\Longrightarrow$ $T(2^{O(2^{n^2})})$-round deterministic online-LOCAL
  - while so, we obtain "component-wise" algorithms (memory only in the current connected component)

- **Step 2**: from component-wise online-LOCAL algorithms to SLOCAL in rooted trees
  - possible without loosing locality by "splitting" the tree in "subtrees" of small size

- **Step 3**: from $o(\log n) \approx O(1)$ in rooted trees in SLOCAL

From component-wise to SLOCAL

Idea: compose many SLOCAL algorithms

From component-wise to SLOCAL

Idea: compose many SLOCAL algorithms

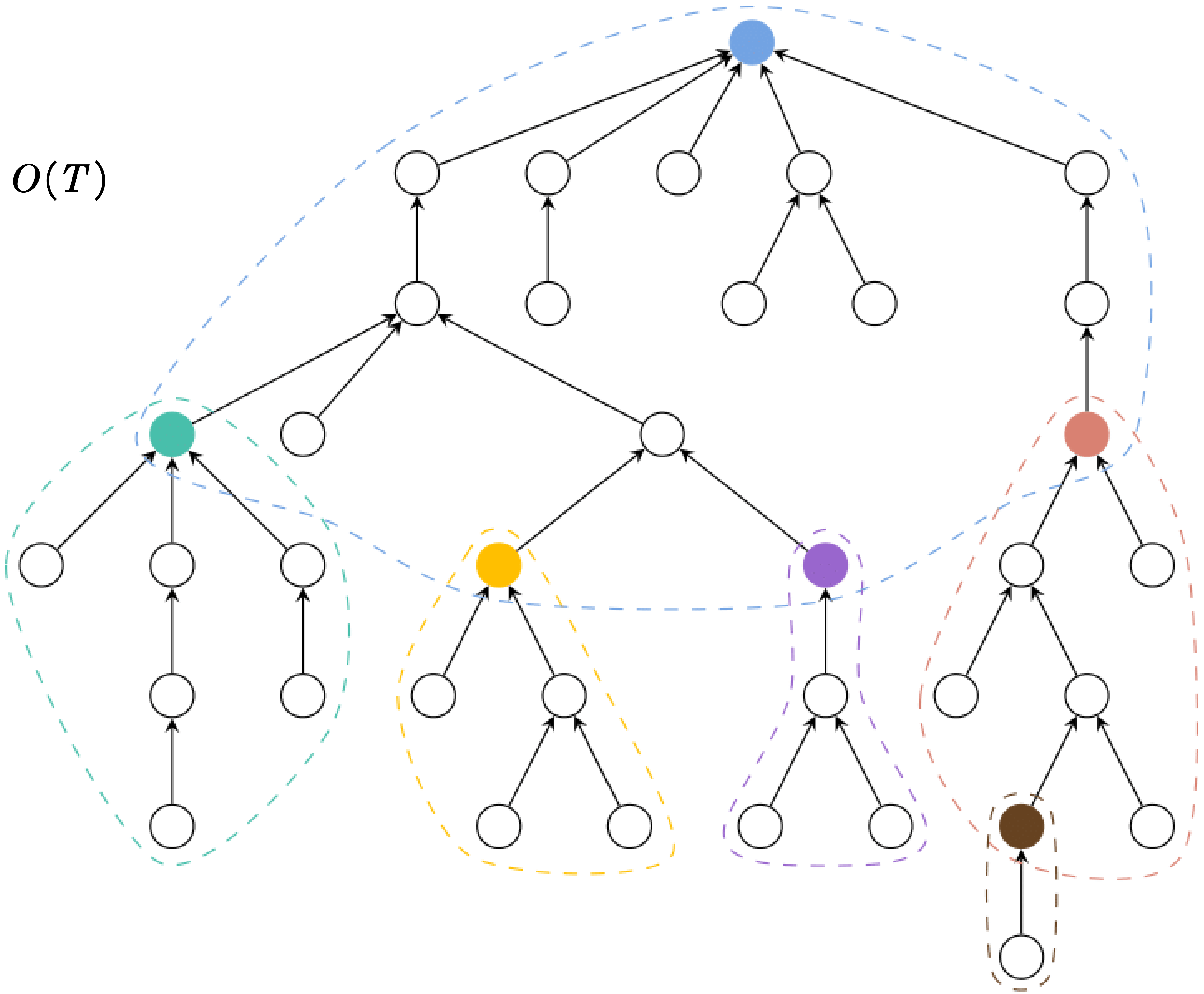
- **First**, create decomposition with trees of size $O(T)$

  - time $O(T)$

From component-wise to SLOCAL

Idea: compose many SLOCAL algorithms

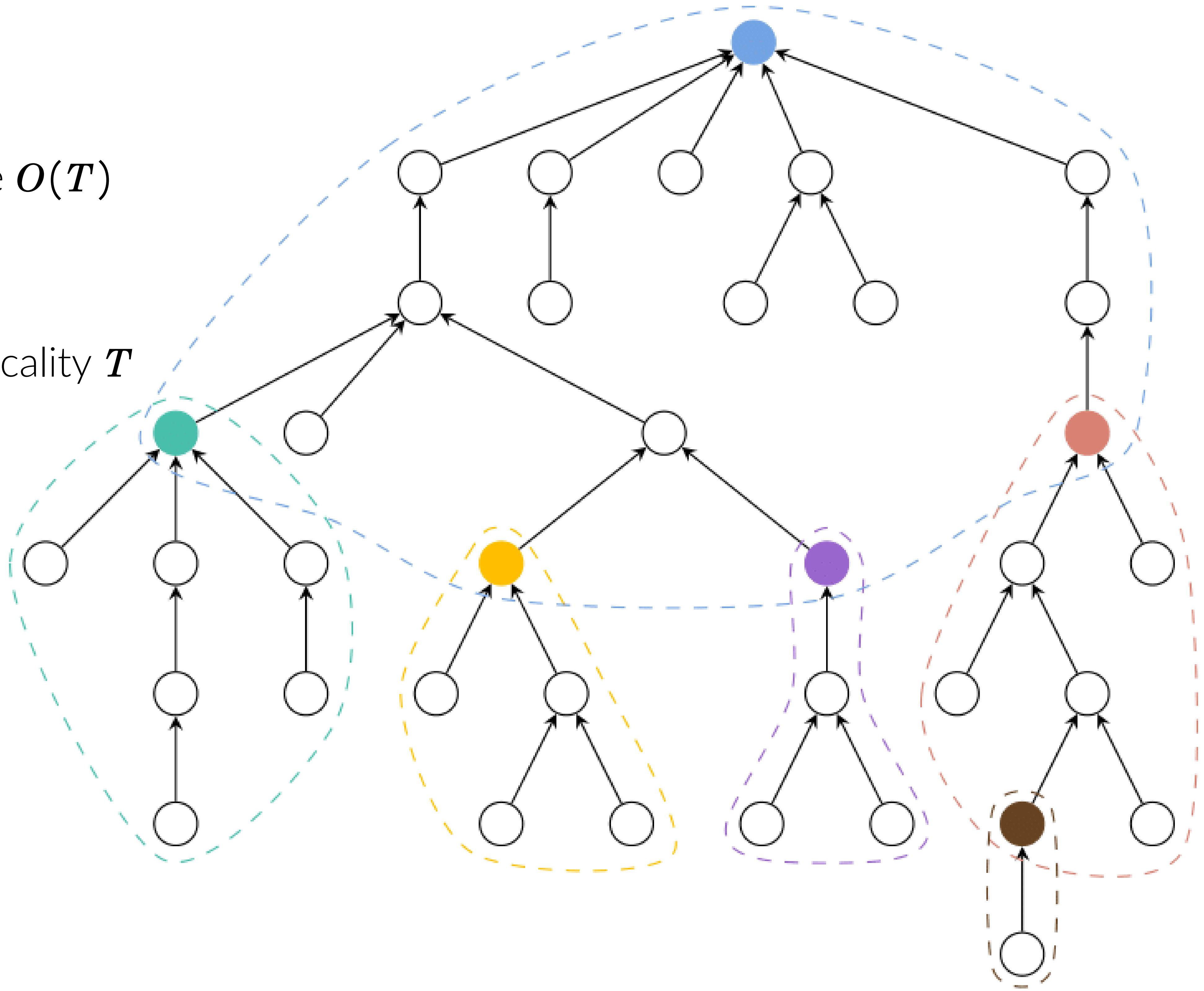
- **First**, create decomposition with trees of size $O(T)$
  - time $O(T)$



From component-wise to SLOCAL

Idea: compose many SLOCAL algorithms

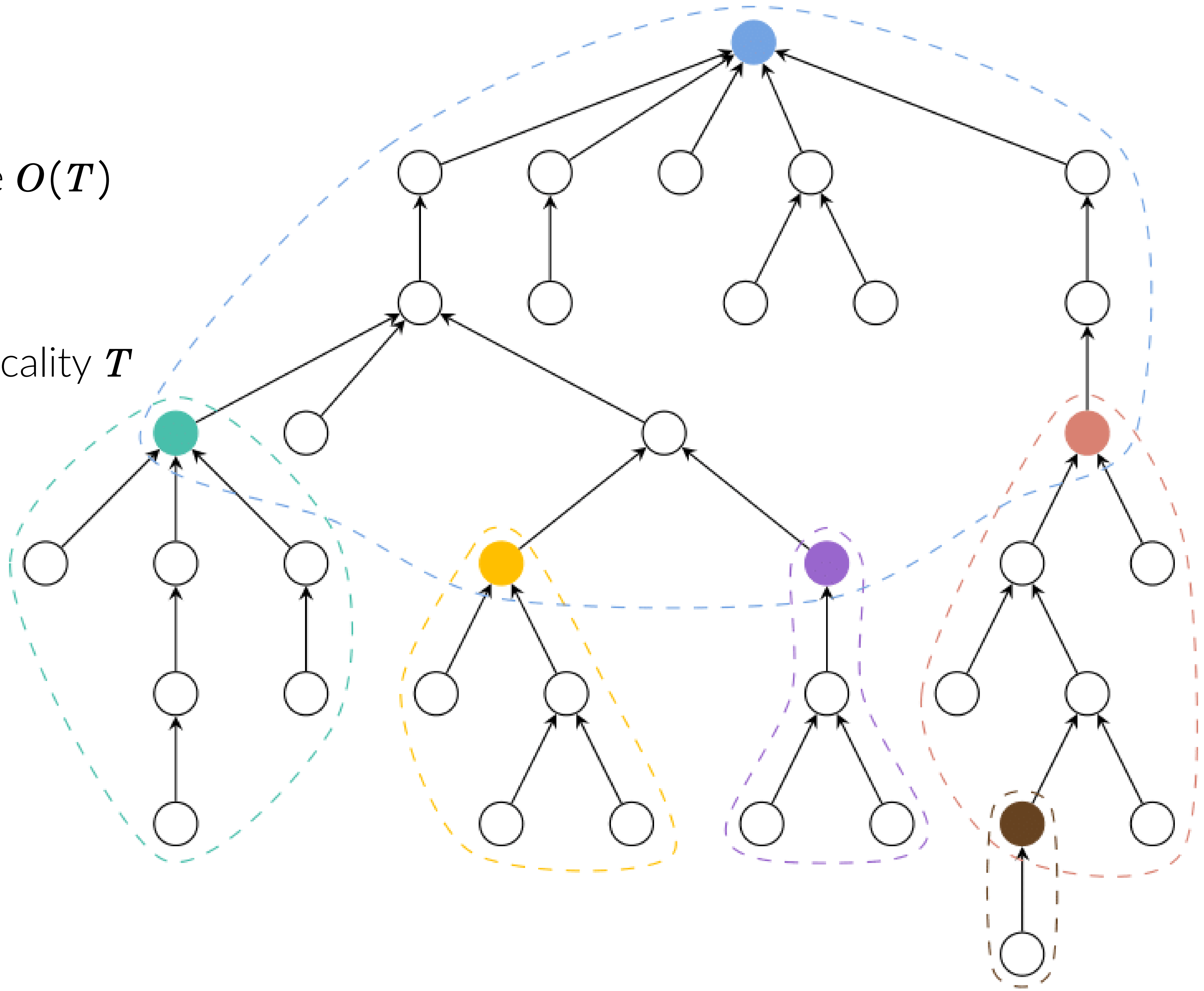
- **First**, create decomposition with trees of size $O(T)$
  - time $O(T)$
- **Second**, use component-wise algorithm of locality T
  - commit at the leader nodes
  - each cluster doable with locality $O(T)$



From component-wise to SLOCAL

Idea: compose many SLOCAL algorithms

- **First**, create decomposition with trees of size $O(T)$
  - time $O(T)$
- **Second**, use component-wise algorithm of locality T
  - commit at the leader nodes
  - each cluster doable with locality $O(T)$
- **Third**, fill the gaps
  - guarantees from "component-wise"
  - each cluster doable with locality $O(T)$



From component-wise to SLOCAL

Idea: compose many SLOCAL algorithms

- **First**, create decomposition with trees of size $O(T)$
  - time $O(T)$
- **Second**, use component-wise algorithm of locality T
  - commit at the leader nodes
  - each cluster doable with locality $O(T)$
- **Third**, fill the gaps
  - guarantees from "component-wise"
  - each cluster doable with locality $O(T)$
- Overall time $O(T)$

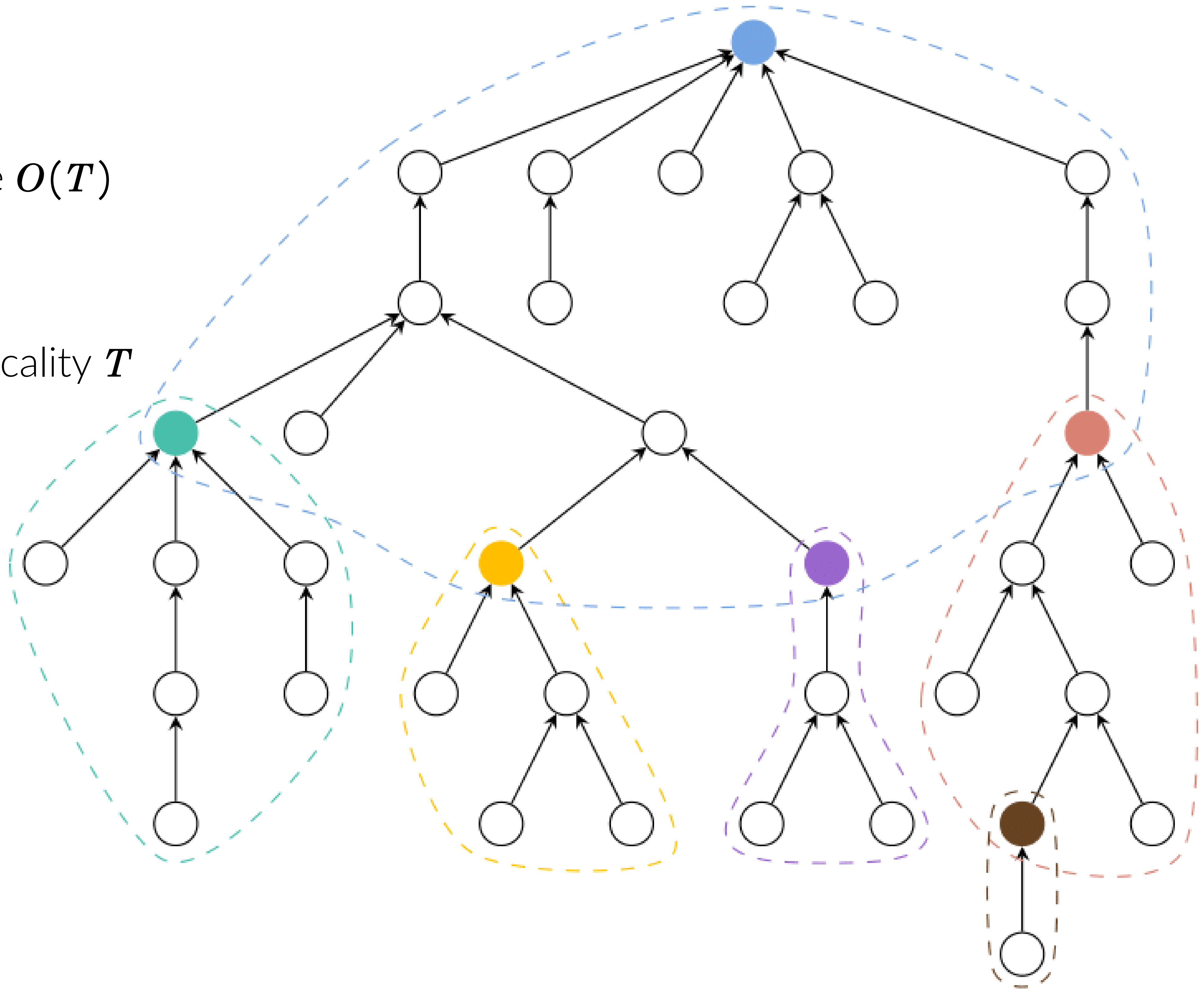
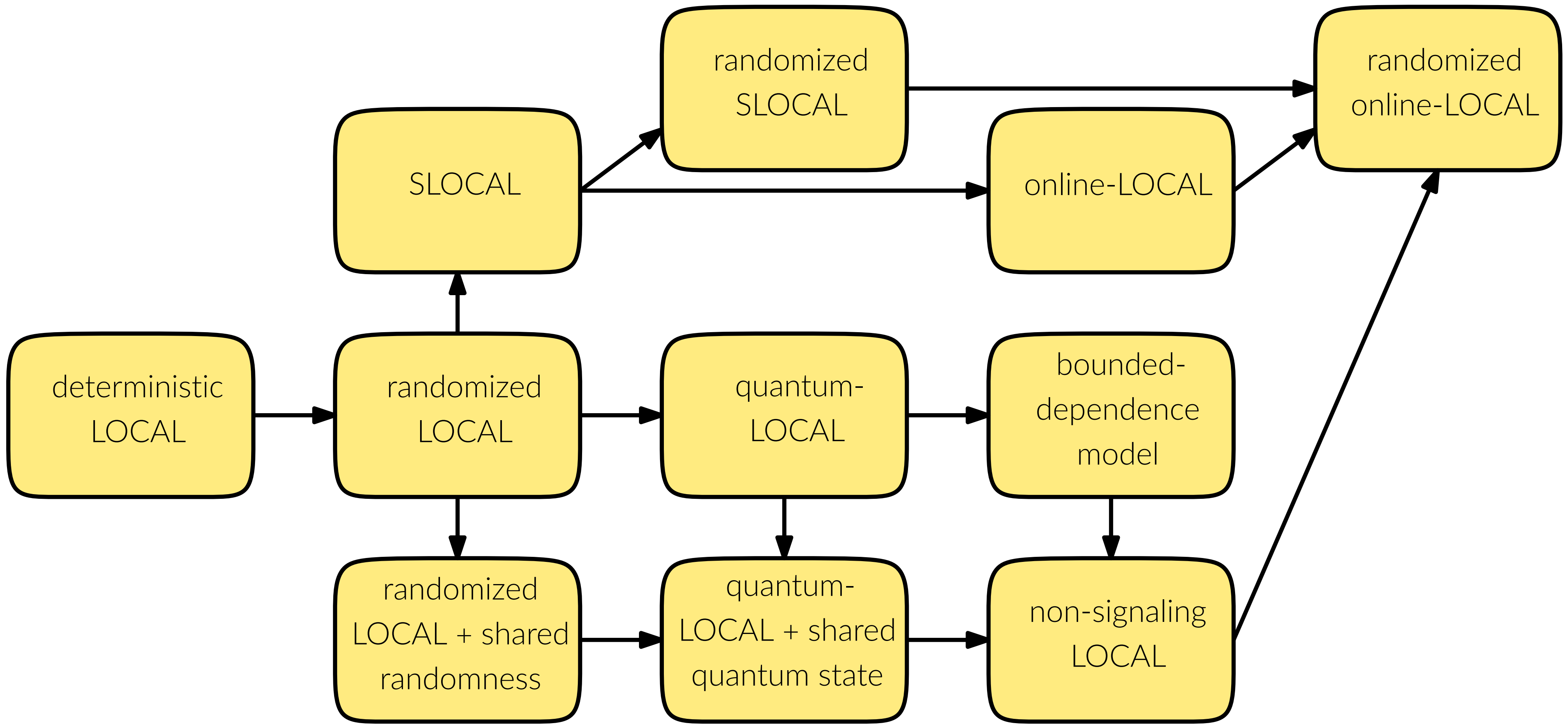


Table of content

Conclusions

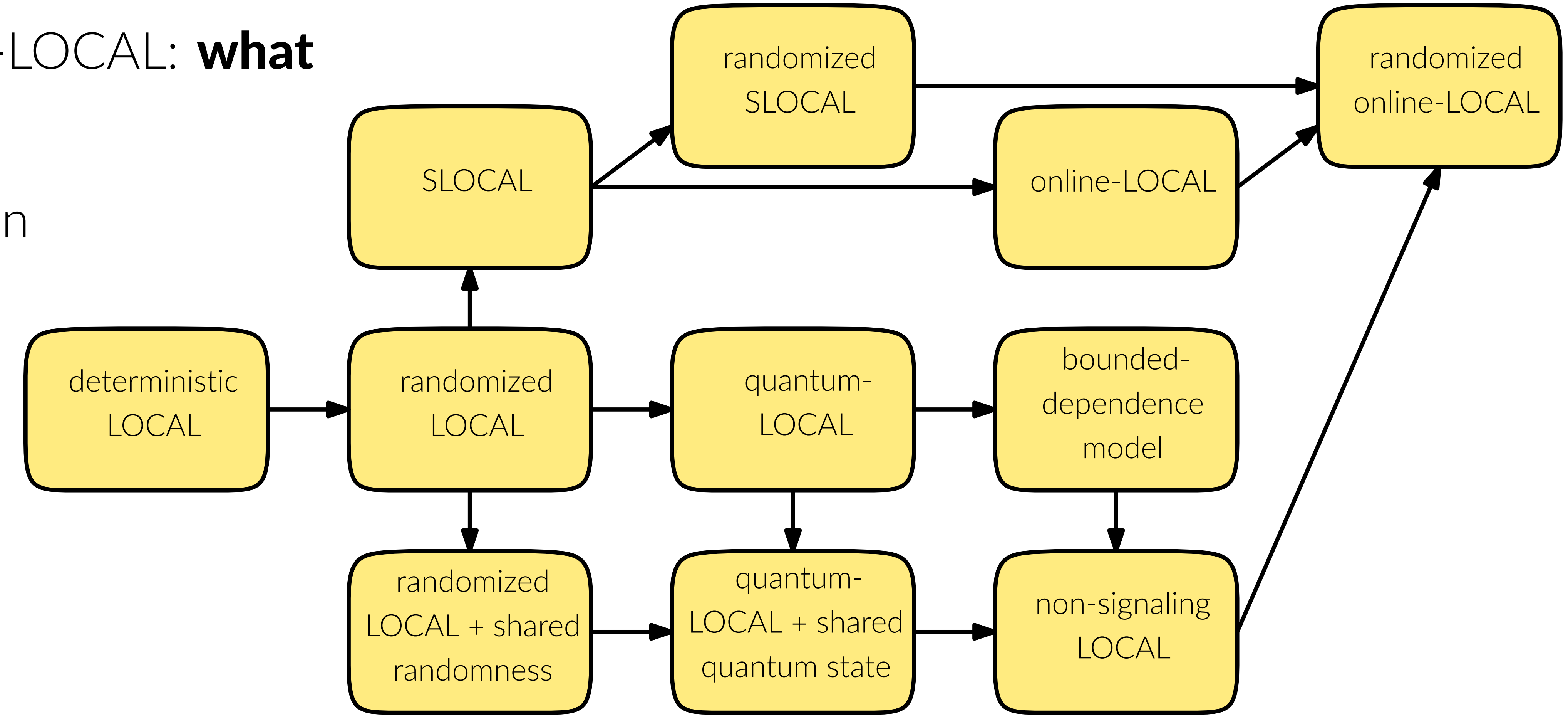
- Possibility to derive **results** for quantum-LOCAL **by “sandwiching”** it between weaker and stronger models



Conclusions

- Possibility to derive **results** for quantum-LOCAL **by "sandwiching"** it between weaker and stronger models

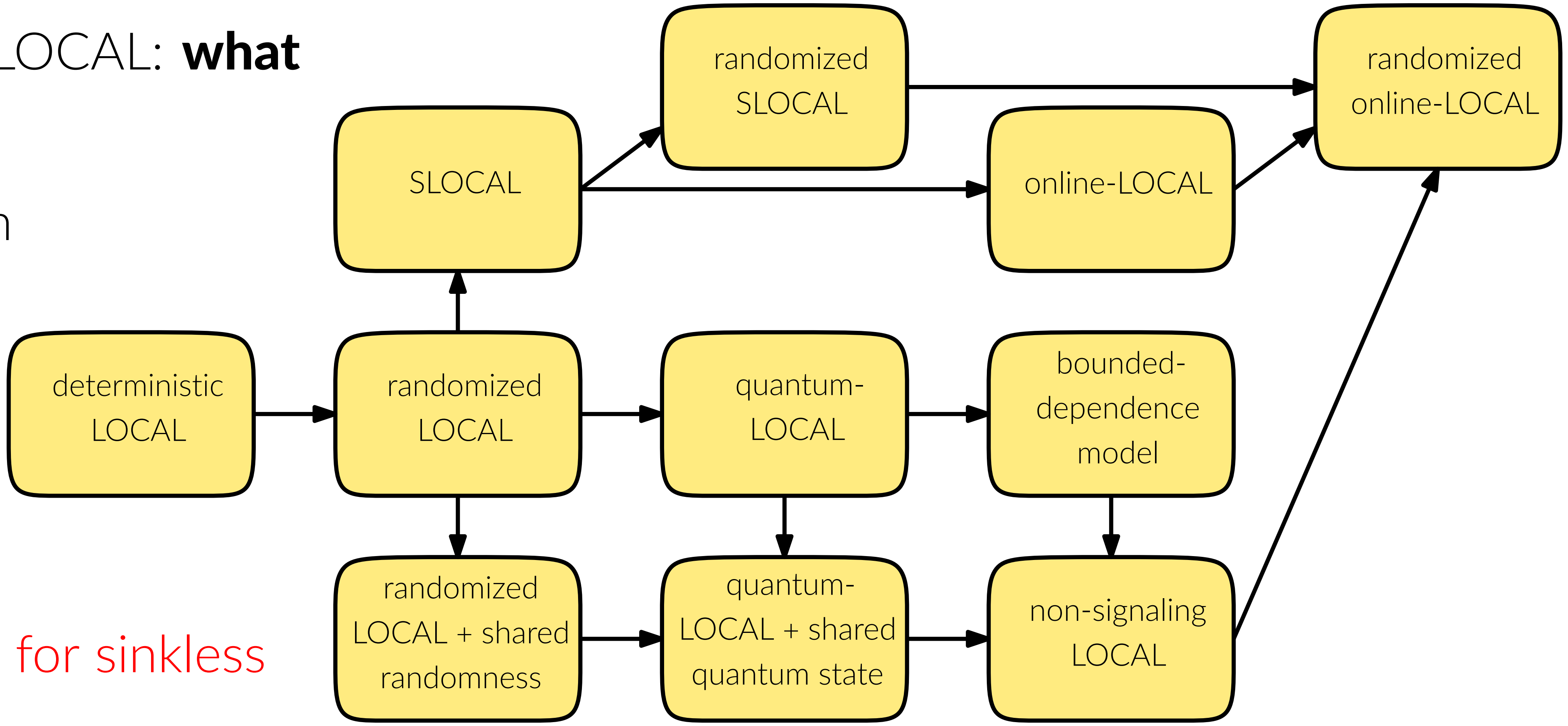
- Randomized online-LOCAL to sequential-LOCAL: **what happens in unrooted trees?**
  - [Dhar et al., '24]: super-logarithmic region
  - other topologies?



- Possibility to derive **results** for quantum-LOCAL **by “sandwiching”** it between weaker and stronger models

- Randomized online-LOCAL to sequential-LOCAL: **what happens in unrooted trees?**
  - [Dhar et al., ’24]: super-logarithmic region
  - other topologies?

- $O(\log^\star n)$ LOCAL $\to$ $O(1)$ bounded-dependence. **Other complexities?**
  - $O(\log n)$? $O(\log\log n)$?
  - is there a finitely-dependent distribution for sinkless orientation?

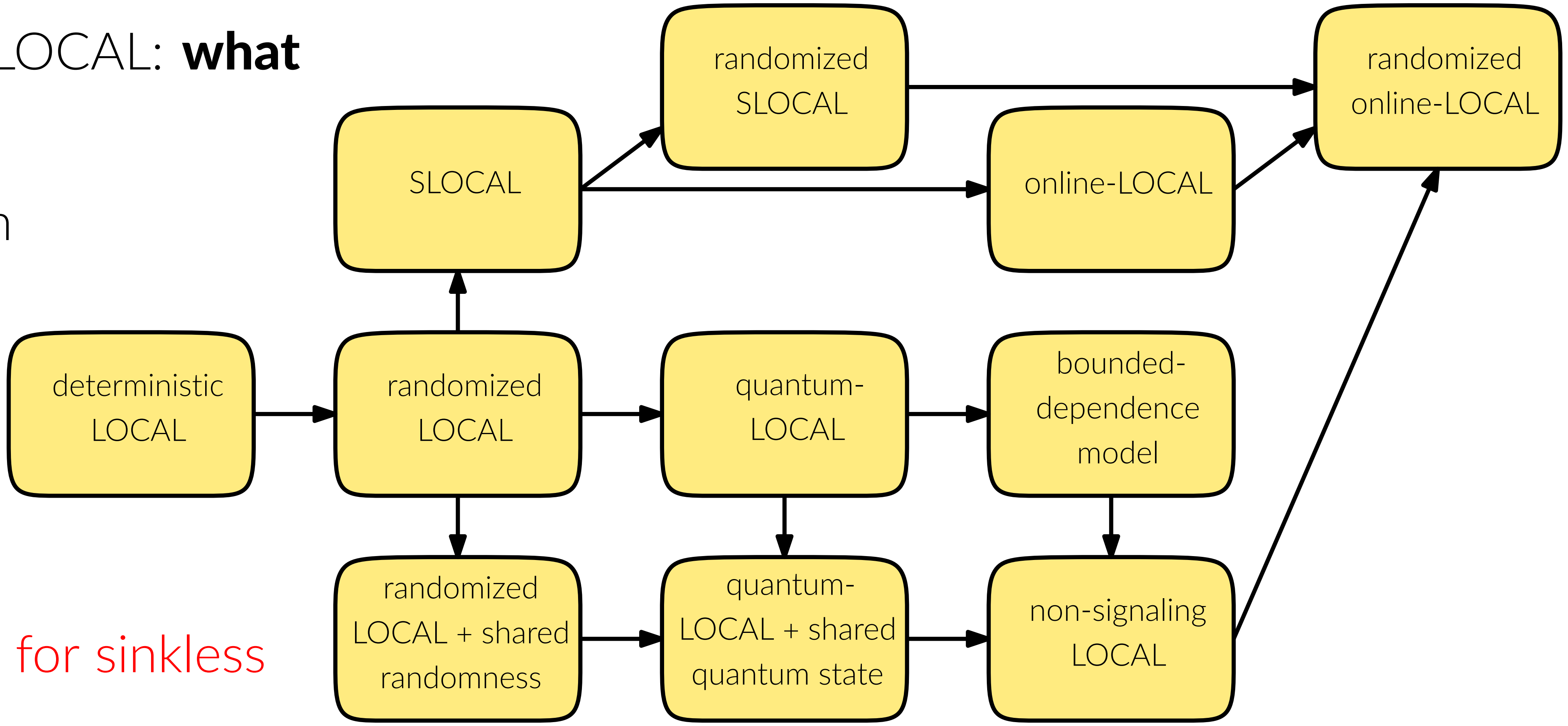


Conclusions

- Possibility to derive **results** for quantum-LOCAL **by "sandwiching"** it between weaker and stronger models

- Randomized online-LOCAL to sequential-LOCAL: **what happens in unrooted trees?**
  - [Dhar et al., '24]: super-logarithmic region
  - other topologies?

- $O(\log^\star n)$ LOCAL $\to$ $O(1)$ bounded-dependence. **Other complexities?**
  - $O(\log n)$? $O(\log\log n)$?
  - is there a finitely-dependent distribution for sinkless orientation?



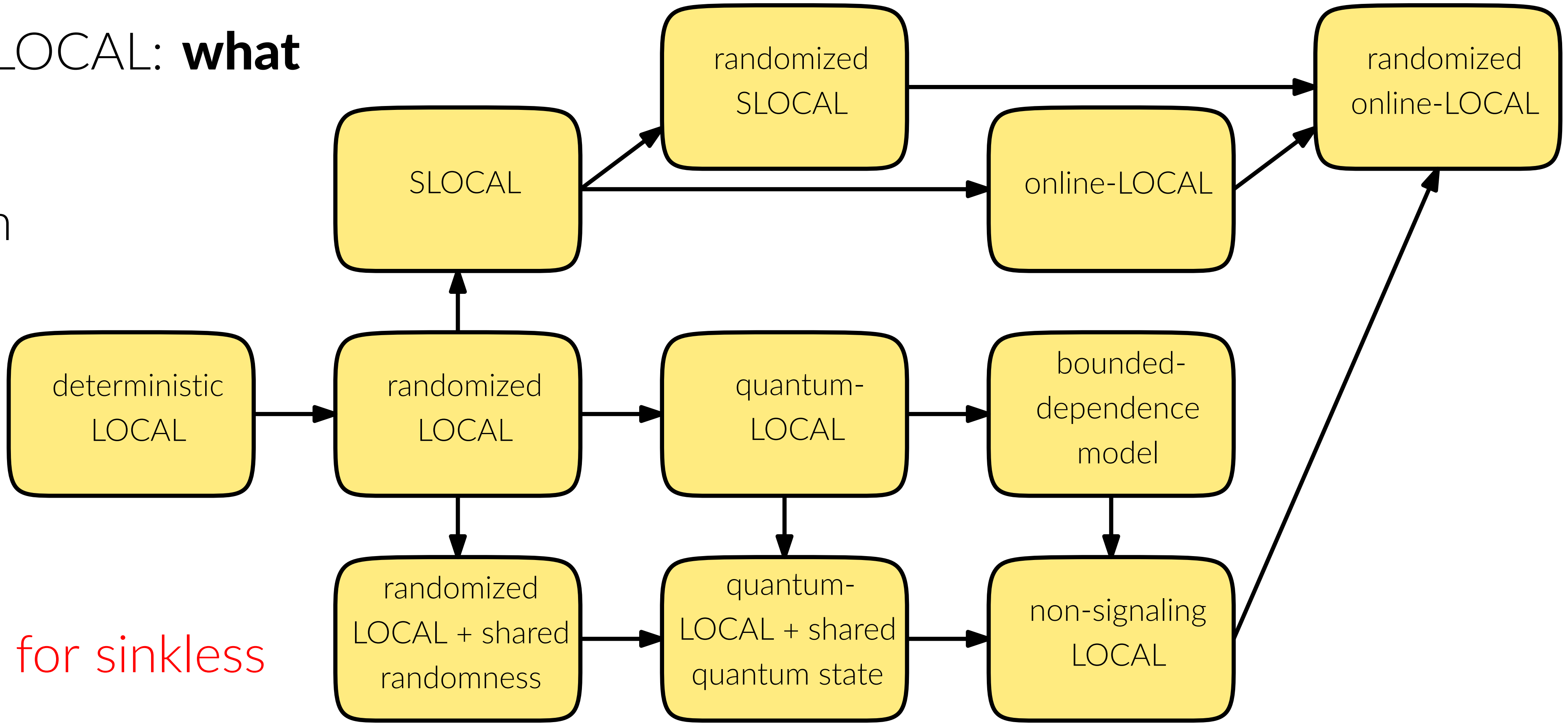
- How to **create bounded-dependence or non-signaling distributions** that do not rely on [Holroyd et al., Forum of Mathematics, Pi '14]?

Conclusions

- Possibility to derive **results** for quantum-LOCAL **by “sandwiching”** it between weaker and stronger models

- Randomized online-LOCAL to sequential-LOCAL: **what happens in unrooted trees?**
  - [Dhar et al., ’24]: super-logarithmic region
  - other topologies?

- $O(\log^\star n)$ LOCAL $\to$ $O(1)$ bounded-dependence. **Other complexities?**
  - $O(\log n)$? $O(\log\log n)$?
  - is there a finitely-dependent distribution for sinkless orientation?



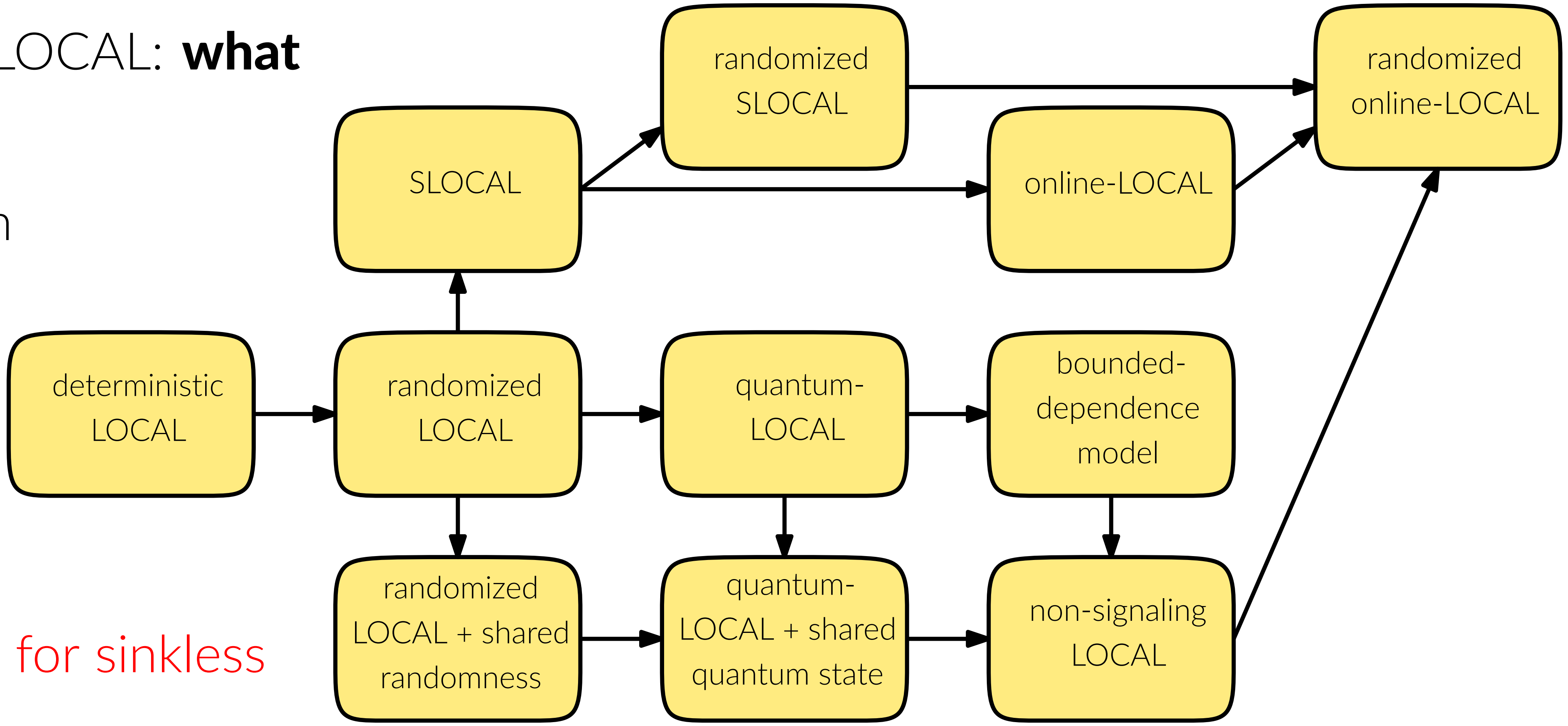
- How to **create bounded-dependence or non-signaling distributions** that do not rely on [Holroyd et al., Forum of Mathematics, Pi ’14]?

- How to get **lower bounds directly in quantum-LOCAL**?

Conclusions

- Possibility to derive **results** for quantum-LOCAL **by "sandwiching"** it between weaker and stronger models

- Randomized online-LOCAL to sequential-LOCAL: **what happens in unrooted trees?**
  - [Dhar et al., '24]: super-logarithmic region
  - other topologies?



- $O(\log^\star n)$ LOCAL → $O(1)$ bounded-dependence. **Other complexities?**
  - $O(\log n)$? $O(\log \log n)$?
  - is there a finitely-dependent distribution for sinkless orientation?

- How to **create bounded-dependence or non-signaling distributions** that do not rely on [Holroyd et al., Forum of Mathematics, Pi '14]?

- How to get **lower bounds directly in quantum-LOCAL**?

THANKS! QUESTIONS?