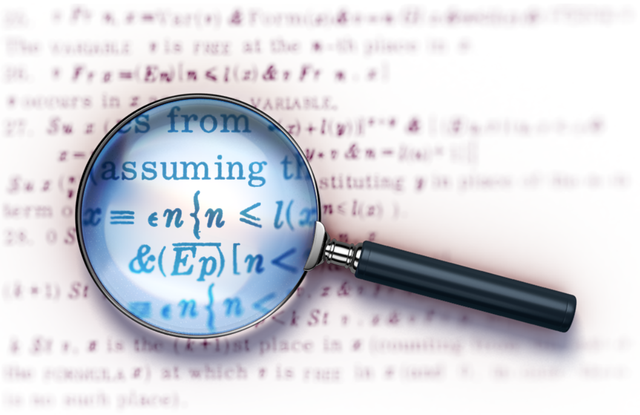


Locally Checkable Proofs

Mika Göös & Jukka Suomela
Helsinki Institute for Information Technology HIIT

Basic Question

1. What **global** information can we infer from **local** structure?

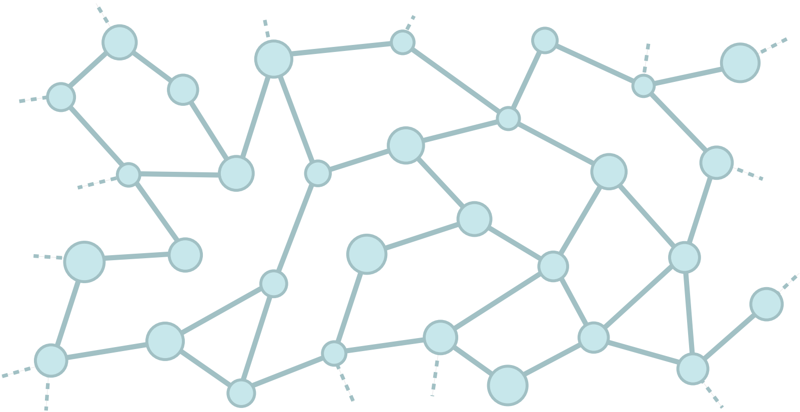
Basic Question

1. What **global** information can we infer from **local** structure?

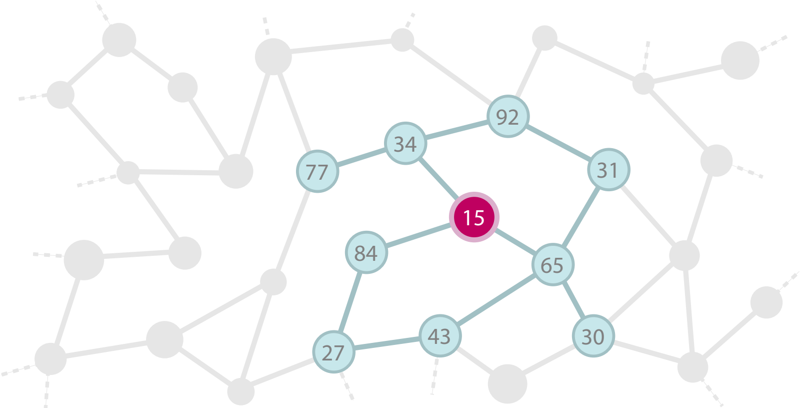
   ⋮

2. *Specifically:* Can we **prove** to a **distributed local verifier** that a graph has a certain **global property**?

Local Algorithms



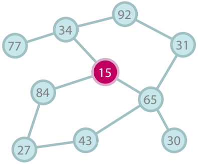
Local Algorithms



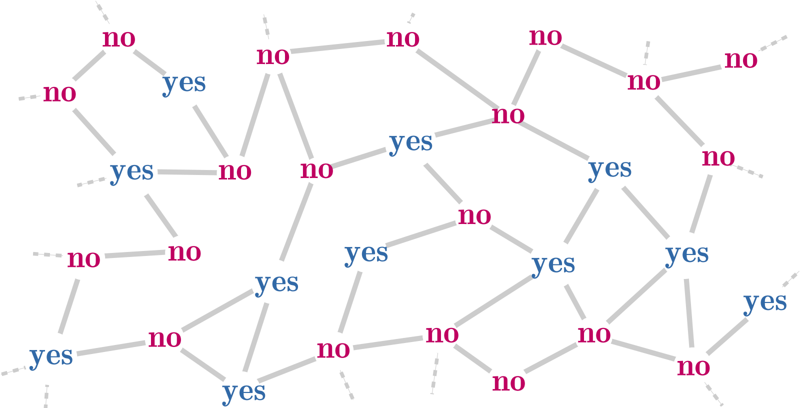
Locality condition: constant running time $t \in \mathbb{N}$

Local Algorithms

Definition:

$\mathcal{A} : \{$  $\} \to \{\mathbf{yes}, \mathbf{no}\}$

Local Algorithms

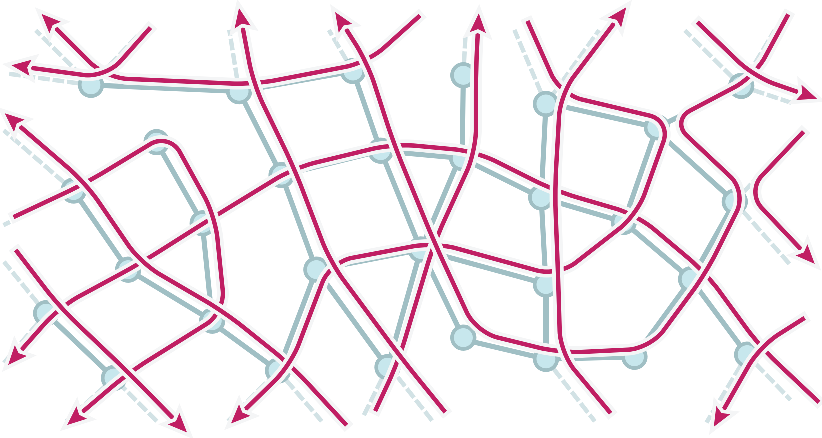


Graph is accepted $\overset{\text{def}}{\iff}$ **all** nodes output **yes**

Locally Checkable Properties

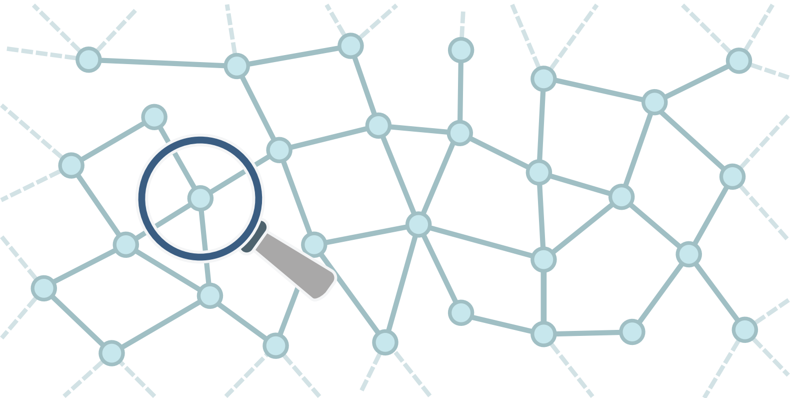
[Naor & Stockmeyer, 1995]

Locally Checkable Properties



e.g. Eulerian graphs

Locally Checkable Properties



Graph Eulerian $\iff$ **all** vertices have even degree

Locally Checkable Proofs

1. Very few properties are locally checkable

Locally Checkable Proofs

1. Very few properties are locally checkable
2. *Extension:* Add information to local neighbourhoods:

Proof labels: $P : V(G) \to \{0,1\}^\star$

Locally Checkable Proofs

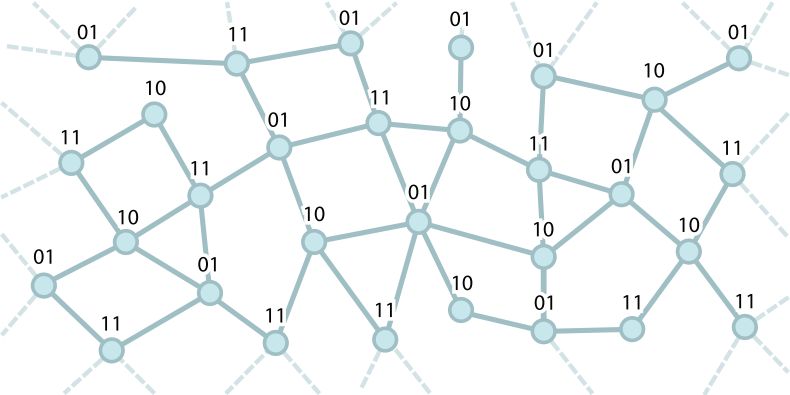
1. Very few properties are locally checkable
2. *Extension:* Add information to local neighbourhoods:

$$\textbf{Proof labels:} \quad P : V(G) \to \{0,1\}^\star$$

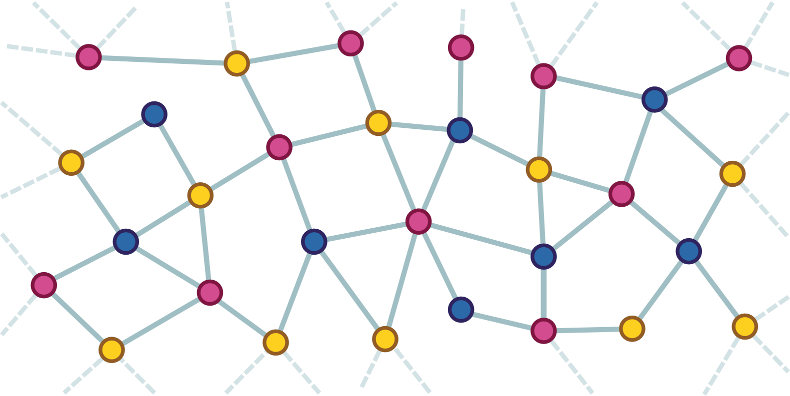
3. "Proof Labelling Schemes"
   [Korman, Kutten & Peleg, PODC 2005]
   [Korman & Kutten, 2007]
   [Fraigniaud, Korman & Peleg, 2010]

Example: 3-Colourability

Example: 3-Colourability



Example: 3-Colourability



$\exists c : V \to \{1, 2, 3\}$ s.t. **all** edges non-monochromatic

Locally Checkable Proofs ($\mathbf{LCP}$) — Definition

A graph property $\mathcal{P}$ admits **locally checkable proofs of size** $f : \mathbb{N} \to \mathbb{N}$ if there exists a local algorithm $\mathcal{A}$ so that

$G \in \mathcal{P}$: There exists a proof

$$P : V(G) \to \{0,1\}^{f(n(G))}$$

so that $\mathcal{A}(G, P, v)$ outputs **yes** on **all** nodes v.

$G \notin \mathcal{P}$: For every proof P, $\mathcal{A}(G, P, v)$ outputs **no** on **some** node v.

Complexity Theory Analogue

Locally checkable properties

$\simeq$

$\mathbf{P}$

$\Rightarrow$

Locally checkable proofs

$\simeq$

$\mathbf{NP}$

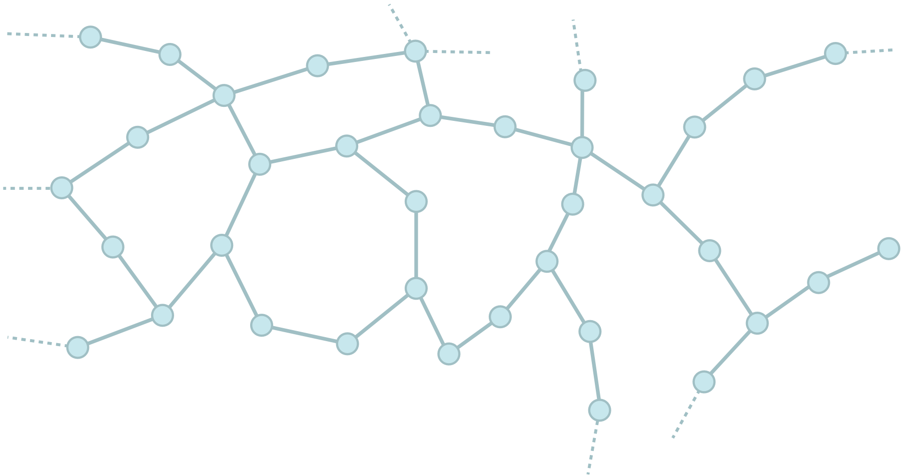
Our Contributions

1. We study the Locally Checkable Proof (**LCP**) hierarchy

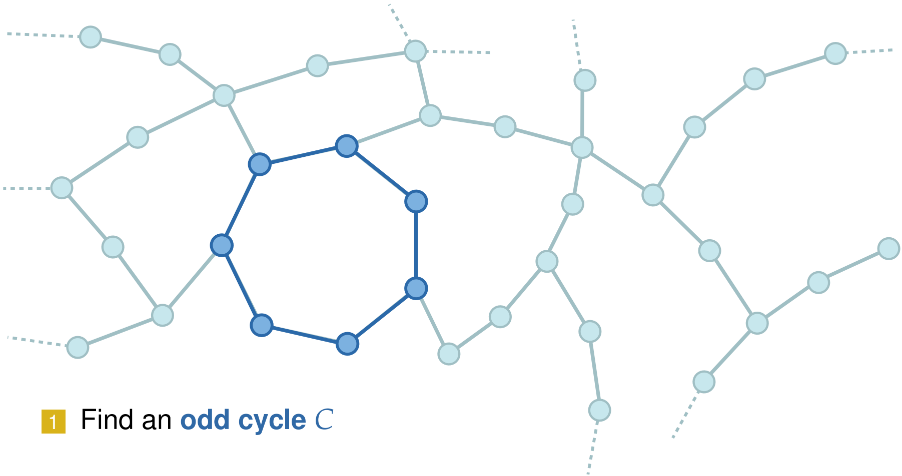
$$\mathbf{LCP}(0) \subset \mathbf{LCP}(O(1)) \subset \mathbf{LCP}(O(\log n)) \subset \mathbf{LCP}(O(n^2))$$

2. Extending the results of [Korman et al., 2005]
   - Our model is strictly stronger
3. Lower-bound constructions—using e.g.
   - Extremal graph theory
   - Gadgets (from **NP**-completeness theory)
   - Communication complexity

Non-bipartiteness in $\mathbf{LCP}(O(\log n))$

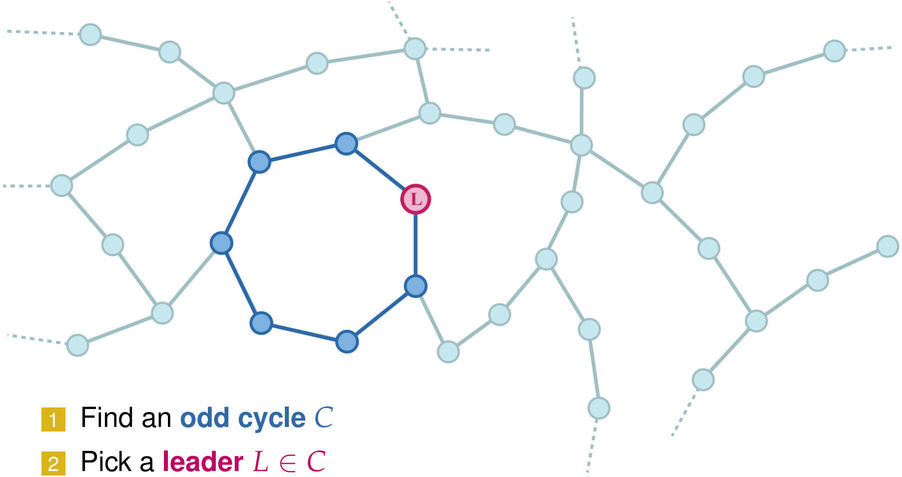


Non-bipartiteness in $\mathbf{LCP}(O(\log n))$



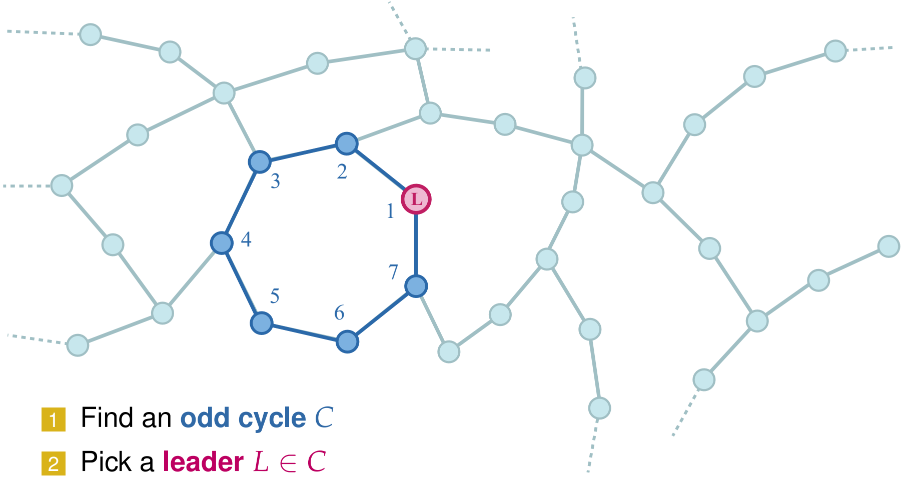
1. Find an **odd cycle** C

Non-bipartiteness in $\mathbf{LCP}(O(\log n))$



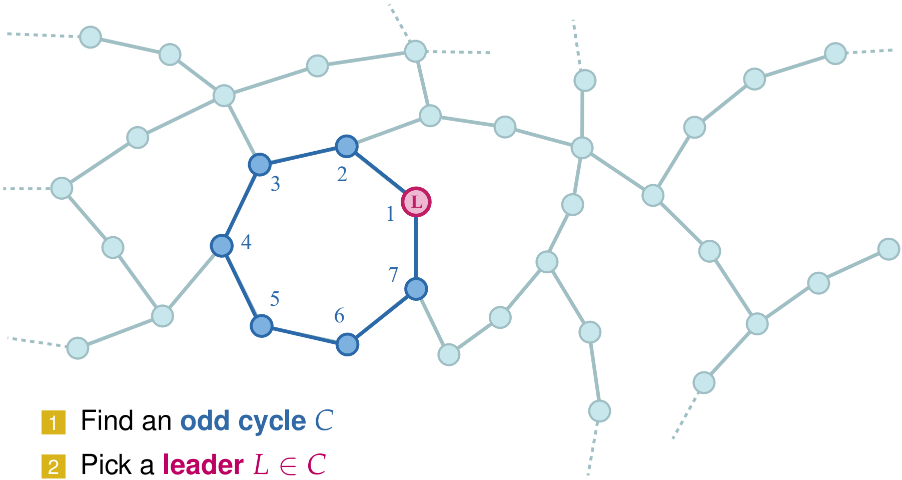
1. Find an **odd cycle** C
2. Pick a **leader** $L \in C$

Non-bipartiteness in $\mathbf{LCP}(O(\log n))$



1. Find an **odd cycle** C
2. Pick a **leader** $L \in C$
3. Equip C with node counters

Non-bipartiteness in $\mathbf{LCP}(O(\log n))$



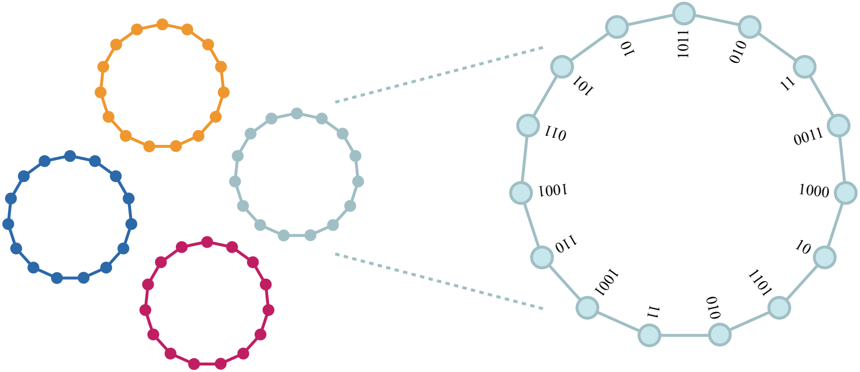
1. Find an **odd cycle** C
2. Pick a **leader** $L \in C$
3. Equip C with node counters
4. Prove the existence of a unique L using spanning tree methods

Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$

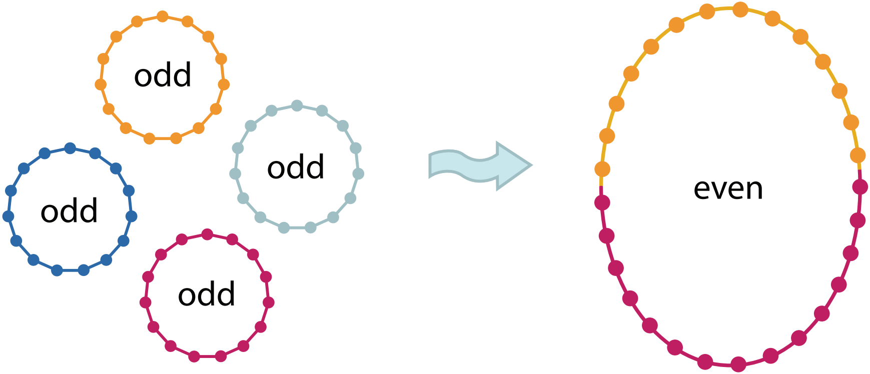
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



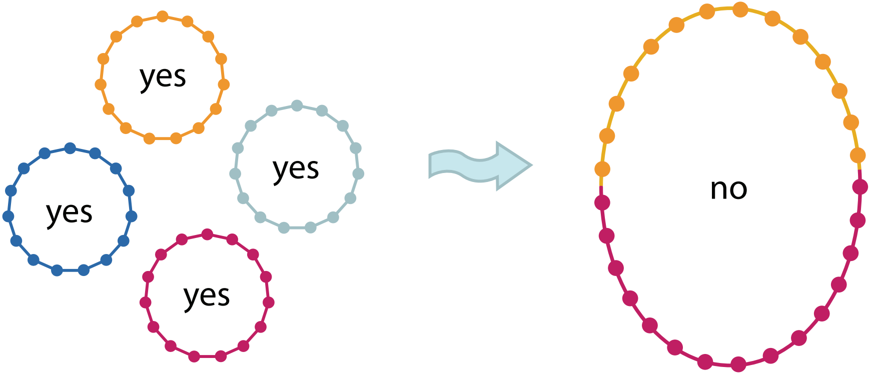
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



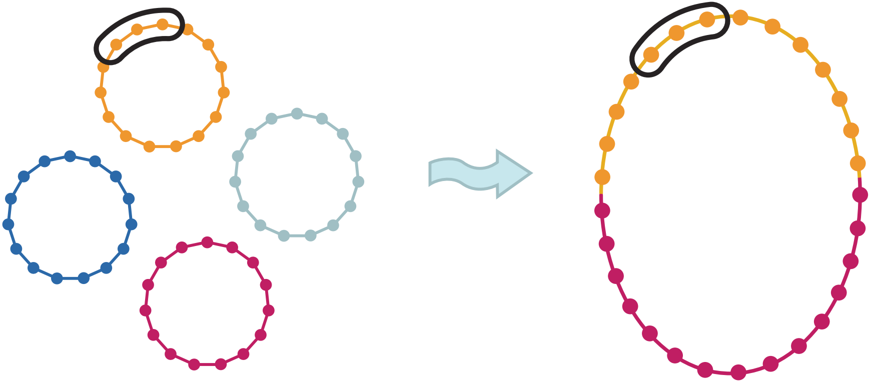
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



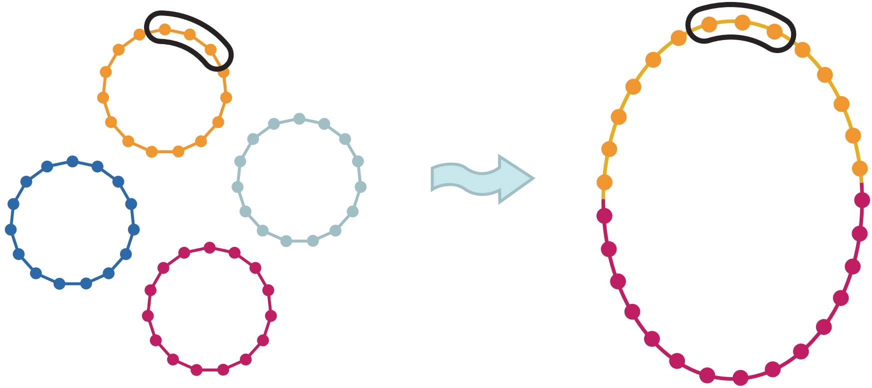
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



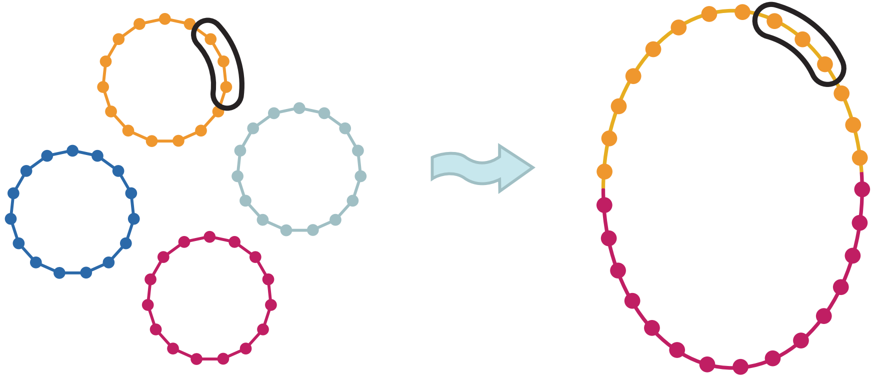
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



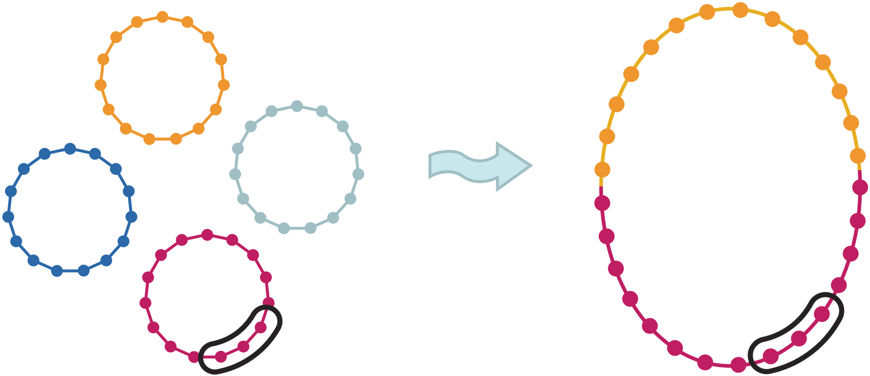
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



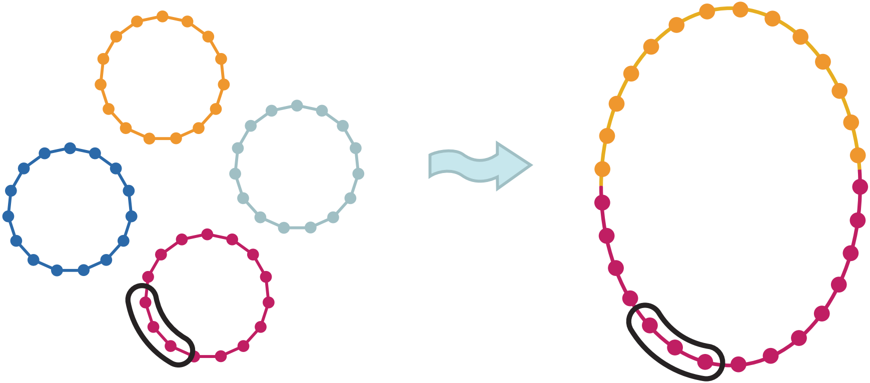
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



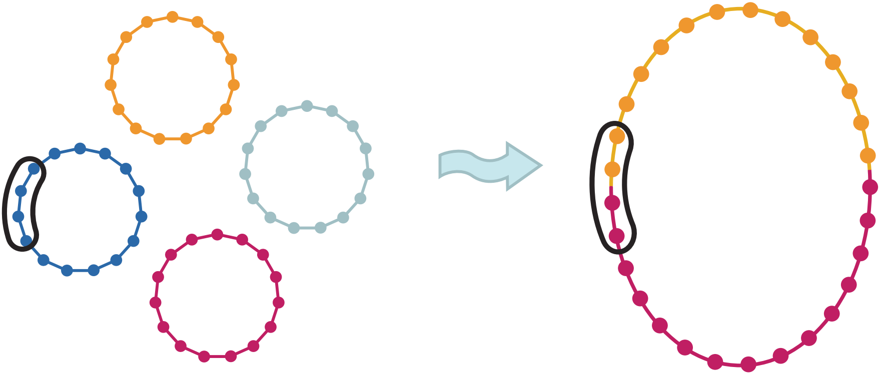
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



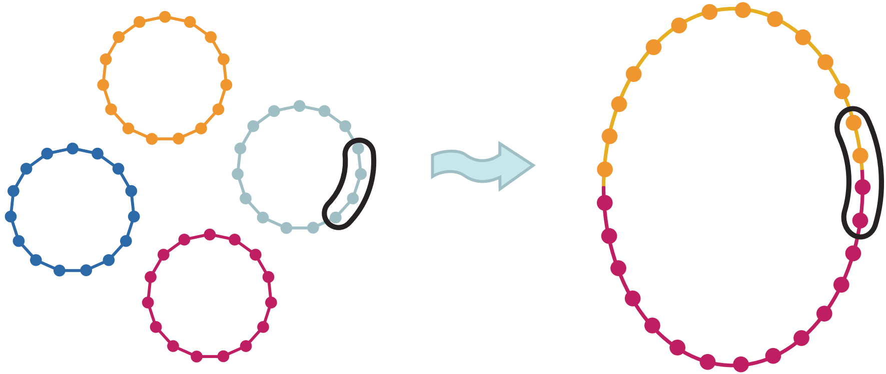
Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



Proving Lower Bounds

1. Suppose *non-bipartiteness* admits proof of size $o(\log n)$ with local algorithm $\mathcal{A}$
2. Then $\mathcal{A}$ accepts **odd cycles** with short proofs:



Local Proof Complexities 1

Class	Proof size	Graph property	Graph family
$\mathbf{LCP}(0)$	0	Eulerian graphs	connected
	0	line graphs	general
$\mathbf{LCP}(O(1))$	$\Theta(1)$	s–t reachability	undirected
	$\Theta(1)$	s–t unreachability	undirected
	$\Theta(1)$	s–t unreachability	directed
	$\Theta(1)$	s–t connectivity $= k$	planar
	$\Theta(1)$	bipartite graphs	general
	$\Theta(1)$	even $n(G)$	cycles
$\mathbf{LCP}(O(\log k))$	$O(\log k)$	s–t connectivity $= k$	general
	$O(\log k)$	chromatic number $\leq k$	general

Local Proof Complexities 2

Class	Proof size	Graph property	Graph family
$\mathbf{LCP}(O(\log n))$	$O(\log n)$	any $\mathbf{coLCP}(0)$ property	connected
	$O(\log n)$	any monadic Σ_1^1 property	connected
	$\Theta(\log n)$	odd $n(G)$	cycles
	$\Theta(\log n)$	chromatic number > 2	connected
$\mathbf{LCP}(\mathbf{poly}(n))$	$\Theta(n)$	fixpoint-free symmetry	trees
	$\Theta(n^2)$	symmetric graphs	connected
	$\Omega(n^2 / \log n)$	chromatic number > 3	connected
	$O(n^2)$	any computable property	connected
—	—	connected	general

Open Problems

1. The exact local proof complexity for many classical problems remains unknown
2. Is it the case that, when $\Delta = O(1)$,

$$\mathbf{LCP}(O(1)) \subseteq \mathbf{NP} \ ?$$

Note: we already know that

$$\mathbf{LCP}(0) \subseteq \mathbf{P} \qquad \& \qquad \mathbf{LCP}(O(\log n)) \begin{cases} \not\subseteq \mathbf{NP} \\ \subseteq \mathbf{NP}_{/\mathbf{poly}} \end{cases}$$

Thank you!