Proving lower bounds in the LOCAL model

Juho Hirvonen, IRIF, CNRS, and Université Paris Diderot

ADGA, 20 October, 2017

Talk outline

- Sketch two lower bound proof techniques for distributed graph algorithms
- In general, simulation is a very powerful tool for lower bounds
- We have the beginnings of a complexity theory: can use heavy hammers in lower bound proofs
- Combining lower bounds for coordination and symmetry breaking beyond our techniques

LOCAL model

- **input =** the communication network
- output = every computer produces local output
- global output = the union of local outputs



LOCAL model

Nodes are running a **synchronous loop:**

- 1. exchange messages
- 2. **update** state

No bounds on computation, messages, no failures: trying to abstract away all challenges **except locality**

LOCAL model

 complexity = the number of synchronous communication rounds until all nodes have stopped and announced output



The model

- each node has a unique name in poly(n)
- graph has bounded maximum degree $\Delta = O(1)$
- graph size n is known

Notation: **DLOCAL(1)** and **RLOCAL(log log n)**, etc.

Information-limited

- In trounds, information can propagate at most t
 hops in the network
- After t rounds, output cannot depend on input that is more than t hops away
- Gathering the t-hop neighborhood is all an algorithm can do!

time ≈ distance

Information-limited

t-round algorithm

function on t-hop neighbourhoods

First lower bound

Based on*

A lower bound for the distributed Lovász local lemma, Brandt, Fischer, <u>Hirvonen</u>, Keller, Lempiäinen, Rybicki, Suomela, and Uitto, STOC 2016

An exponential separation between randomized and deterministic complexity in the LOCAL model Chang, Kopelowitz, and Pettie, FOCS 2016

The Complexity of Distributed Edge Coloring with Small Palettes, Chang, He, Li, Pettie, and Uitto, SODA 2018

Sinkless orientation



All edges are oriented with no sinks

The lower bound

Sinkless orientation requires **Ω(log_Δ log n)** randomized time

Sinkless orientation requires **Ω(log_Δ n)** deterministic time



complexity: f(n)

A (simple) deterministic lower bound

We will start by proving a simpler lower bound for a simpler deterministic model:

Finding a **sinkless orientation** requires **Ω(log_Δ n)** communication rounds in this model

Lower bound: sinkless orientation (simple) model: d-regular graphs, 2-vertex col. c-edge col. (for **c >> d**) graphs have large (logarithmic) girth

(Very) high level proof

- In high-girth graphs a o(log∆ n)-round algorithm for sinkless orientation implies a O-round algorithm for sinkless orientation
- 2. There is no **0-round** algorithm for *sinkless orientation* in high-girth graphs

Lower bound: sinkless orientation

For algorithm A, define running time profile $\mathbf{t} = (t_1, t_2, \dots, t_c)$

Edges of color **i** must halt after **t**_i rounds*

Lower bound: sinkless orientation

Assume algorithm has running time profile $\mathbf{t} = (t, t, ..., t)$

Edges of **all colors** halt in *t* communication rounds

Lower bound: sinkless orientation For example, assume d=3 and c=5

 $\mathbf{t} = (t, t, t, t, t)$ speed up color 5 by simulation $t^{(1)} = (t, t, t, t, t-1)$ speed up color 4 by simulation $t^{(2)} = (t, t, t, t-1, t-1)$

Lower bound: sinkless orientation For example, assume **d=3** and **c=5** t = (t, t, t, t, t)speed up each color t-1 = (t-1, t-1, t-1, t-1)repeat **t** times $\mathbf{0} = (0, 0, 0, 0, 0)$

Lower bound: sinkless orientation For example, assume d=3 and c=5

algorithm with running time profile $\mathbf{0} = (0,0,0,0,0)$

easy to show that this is impossible!

We can apply argument if initial **t** = o(log_Δ n)



Simulation

possible outputs given **2-neighbourhood**?



Simulation

possible outputs given **2-neighbourhood**?



3-neighbourhood of violet edge



3-neighbourhood of red edge



intersection of **3-neighbourhoods = 2-neighbourhood of**



outputs on the two sides are **independent** given **orange**



is it possible for endpoint to be a **sink** for the other edges?



I input s.t. other edges pointed towards node?

is it possible for endpoint to be a **sink** for the other edges?



Other endpoint a sink

now assume the first endpoint is a **potential sink**



I input s.t. other edges pointed towards node?

Other endpoint a sink

now assume both endpoints potential sinks



Other endpoint a sink

now assume both endpoints potential sinks



no feasible output left for middle edge

Lower bound: sinkless orientation For example, assume **d=3** and **c=5** t = (t, t, t, t, t)speed up each color t-1 = (t-1, t-1, t-1, t-1)repeat **t** times $\mathbf{0} = (0, 0, 0, 0, 0)$

Problem with LOCAL model

- **Unique identifiers** induce dependencies between possible inputs of distant nodes
- Argument that we can **force a sink** unless one endpoint is safe is no longer true

Roundabout solution: randomize

- Now consider the randomized setting
- In addition to the colouring, nodes have access to u.a.r. real number
- Can get identifiers w.h.p.

Theorem: sinkless orientation requires Ω(Δ-1log_Δ log n) rounds
Lower bound: updated strategy

A:
$$\mathbf{t} = (t,t,t,t,t)$$
 error with prob. < \mathbf{p}
speed up color 5 by simulation

A': $t^{(1)} = (t,t,t,t,t-1)$ error with prob. $< 3p^{1/3}$

speed up color 4 by simulation

A'' :
$$t^{(2)} = (t,t,t,t-1,t-1)$$

Lower bound: updated strategy

$$A_t$$
 : $\mathbf{t} = (t,t,t,t,t)$ error with prob. $< \mathbf{p}$ \mathbf{A}_t : $\mathbf{t} = (t,t,t,t,t)$ error with color A_{t-1} : $\mathbf{t} = (t-1,t-1,t-1,t-1,t-1)$ error with \mathbf{A}_{t-1} : $\mathbf{t} = (t-1,t-1,t-1,t-1,t-1)$ prob. $< \mathbf{O}(\mathbf{p}^{-3^{(2d-1)}})$ \mathbf{A}_0 : $\mathbf{0} = (0,0,0,0,0)$ error with \mathbf{A}_0 : $\mathbf{0} = (0,0,0,0,0)$ prob. $< \mathbf{O}(\mathbf{p}^{-3^{(t(2d-1))}})$

Lower bound: updated strategy

start with alg. **A**, running time **t**, error prob. **p**₀

algorithm **A'** with running time **0**, error prob. < **O(p^{-3^(t(2d-1))})**

0 rounds: must have error probability $\mathbf{p} > 1/8^d$ $\int_{\mathbf{t}} \mathbf{t} = \Omega(\Delta^{-1} \log \log n)$

Lower bound: updated strategy For example, assume **d=3** and **c=5**

A :
$$\mathbf{t} = (t,t,t,t,t)$$
 error with prob. $< \mathbf{p}$
 \downarrow speed up a color by simulation
A' : $\mathbf{t}^{(1)} = (t,t,t,t,t-1)$ error with prob. $< 3p^{1/3}$

$$t^{(2)} = (t,t,t,t-1,t-1)$$

Outputs of incident edges

black endpoint potential **sink** w.p. **> p**?



Outputs of incident edges

white endpoint potential **sink** w.p. **> p**?



Back to deterministic

Theorem (Chang et al., FOCS 2016): Assume that for LCL L there exists an algorithm with running time t = o(log_A n), then there exists an algorithm with running time t' = O(log* n)

Corollary: sinkless orientation requires **Ω(log_Δ n)** deterministic time

Automatic speed-up

- Another black box simulation
- A given algorithm A is "fooled" to run faster: compute locally unique "identifiers" (a colouring) and run A on those
- Efficient solving of LCLs reduces to coloring + constant time

Back to randomized

Theorem (Chang et al., FOCS 2016): randomized complexity of an LCL on instances of **size n** is at least the deterministic complexity on instances of **size (log n)**^{1/2}

Corollary: sinkless orientation requires **Ω(log_Δ log n)** randomized time

What just happened?

deterministic: **Ω(log_Δ n)** Proof technique doesn't work for identifiers

> IDs → randomness randomized: Ω(Δ⁻¹ log log n)

automatic connection randomized: $\Omega(\log \log n)$ automatic speed-up deterministic: $\Omega(\log n)$

Trouble with IDs



Other applications? *A hammer but no nails?*

- Identify other **intermediate** problems
- Lower bounds as function of Δ (maximum degree)?
 - Best algorithms for maximal matching and maximal independent set linear in Δ
 - Weaker lower bounds exist

Lifting lower bounds Examples and obstacles

A general strategy

- **Lifting** of lower bounds from weaker models into stronger ones a powerful technique
 - Proving lower bounds potentially easier in a weaker model (say LOCAL without unique identifiers)
 - Use simulation to show that a fast algorithm in strong model implies a fast algorithm in weak model

Simulation results

- OI(1) = ID(1) for LCLs [Naor and Stockmeyer, SIAM J. Comput., 1994]
- PO(1) = ID(1) for approximation [Göös, H., and Suomela, J.ACM, 2013]
- Non-local probes do not help for LCLs [Göös et al., DISC 2016]

A case study

Coordination vs. symmetry breaking

e.g. maximal matching: $O(\Delta + \log^* n)$

MM: existing bounds

- MM requires Ω(log Δ / log log Δ) rounds in RLOCAL [Kuhn et al., J.ACM, 2016]
- MM requires Ω(Δ) in EC
 [H. and Suomela, PODC 2012]
- FMM requires Ω(Δ) in RLOCAL(1)
 [Göös, H., and Suomela, PODC 2014]
- O(Δ^{1+η})-COL requires Ω(Δ^{1/3-η/3}) in broadcast
 DLOCAL [Hefetz et al., DISC 2016]

Conjecture*

Coordination necessary if time budget small as **function of n**

No algorithm for maximal matching with running time $O(\log^* n)$ and $o(\Delta)$



Look at a single bound

Fractional maximal matching requires $\Omega(\Delta)$ rounds in **RLOCAL(1)**

(If no dependency on **n**, then must spend $\Omega(\Delta)$ rounds)

Fractional maximal matching



Fractional maximal matching



A lower bound for MM?

- Implies a "if cows could fly" lower bound for maximal matching
 - When solving maximal matching in O(1) time, must use Ω(Δ) rounds
 - Solving maximal matching requires Ω(log* n) rounds

Structure of the proof

- Base case: FMM requires Ω(Δ) rounds in the EC model
- Lift the bound using a series of simulations

$\begin{array}{cccc} \mathsf{EC} & \longrightarrow & \mathsf{PO} & \longrightarrow & \mathsf{OI} & \longrightarrow & \mathsf{DLOCAL}(1) \\ \Omega(\Delta) & & & & & \downarrow \\ & & & & \mathsf{RLOCAL}(1) \end{array}$



Base case



- Start with completely symmetric graph:
 ∆ ≈ degrees of symmetry
- Merge two graphs, forcing the algorithm to look 1
 step further, lose 1 unit of symmetry

Structure of the proof

- Base case: FMM requires Ω(Δ) rounds in the EC model
- Lift the bound using a series of simulations









Structure of the proof

- Base case: FMM requires Ω(Δ) rounds in the EC model
- Lift the bound using a series of simulations



a < b < c < d < e

PO -> OI

- OI-algorithms have to work on locally tree-like graphs
- Use port-numbering and orientation to construct
 a locally consistent ordering on nodes

Order-invariant tree



For any **r**, there exists an infinite **2r**-regular **PO-tree** with *homogeneous* linear order

homogeneous = translation invariant = order of **u** and **v** defined by the labels on the path **p(u,v)**

PO -> OI



Consider the tree-like view of node v

PO -> OI


Structure of the proof

- Base case: FMM requires Ω(Δ) rounds in the EC model
- Lift the bound using a series of simulations



$OI \rightarrow DLOCAL(1)$

- Ramsey-argument: in a large identifier space there exists a subspace s.t. DLOCAL-algorithm has to behave like an OI-algorithm
 - Trick: outputs of non-constant size, control an indicator algorithm A*

$OI \rightarrow DLOCAL(f(n))?$

- Ramsey-type argument cannot go beyond
 o(log* n)
 - It is not clear if we can get even there!
- Real goal was Ω(Δ) when dependency on n is
 O(log* n) or even o(log n)

Going forward

How to proceed?

- One possible strategy: divide the proof into two parts:
 - Show that MM / MIS in vertex-colored graphs requires Ω(Δ) rounds (with moderate dependency on n)
 - 2. Show that **unique identifiers** do not help compared to a **colouring**

P1: $\Omega(\Delta)$ lower bound

- Existing lower bounds no good for various reasons
 - Indistinguishability doesn't work for maximality constraints
 - KMW lower bound for approximation, probably capped around $\Omega(\log \Delta)$
 - **EC** lower bound cannot handle coloring
 - So simulation speed-up?

Simulation for MM

- A concrete research question: solving maximal matching in 2-vertex colored graphs
- Can be solved in time O(Δ) using a simple greedy proposal algorithm

Simulation for MM?

- Simulation proof works for input vertex coloring
- Currently technique works only for sinkless orientation and coloring
- Need a different invariant: current invariants cycle length, success probability, and size of color palette

P2: ID → coloring

Usually control identifiers by **Ramsey's theorem:** create areas of homogeneous output

This proof technique *cannot* go to **O(log* n)** and beyond

"Reason": **OI-algorithms** much weaker in trivial breaking of symmetry

P2: ID → coloring

Chang et al.:

In bounded-degree graphs, any algorithm for an LCL problem with complexity **o(log n)** can be sped up to **O(log* n)** time

In particular, solving LCLs in **DLOCAL(log* n)** decomposes into **coloring + DLOCAL(1)**

LCL speed-up

- Algorithm A with complexity t(n) = o(log n)
- Given instance G of size N, tell algorithm it is running in instance of size n = n(A)
- Colour G^{2t(n)+1+O(1)} with n² colours (e.g. Linial)
- Everything looks locally like an instance of size n, so simulate A in t(n) rounds

t(n) = O(1), colouring in time O(log*n)

LCL speed-up

- Every LCL solvable in time o(log n) can be solved in time O(log*n) by first coloring + a constanttime distributed algorithm
- Compare with the natural algorithm for MIS: colour with Δ+1 colours, apply greedy

LCL speed-up

- The algorithm implied by LCL speed-up still requires a coloring with long-distance dependencies
- Does not keep dependency on Δ
- Two lower bound techniques still **incompatible**



Role of identifiers

- Additional ideas required to truly understand power of unique identifiers and in general input labels with long-range dependencies
 - Very few examples where these actually help!
 - Include "cheating", e.g. **output is unbounded** (*Kuhn, 2009, Hasemann et al., 2012*),
 - Different setting: **local decision** [*Fraigniaud et al., 2015*]

Concluding

Simulation

Simulation can be a powerful and simple technique for proving lower bounds

Big open questions

- Power of unique identifiers / power of colorings
- Coordination / symmetry breaking

 Extending speed-up simulation to new problems