Adding a referee to an interconnection network: What can be computed with little local information

Florent Becker, Martin Matamala, Nicolas Nisse, Ivan Rapaport, Karol Suchan, Ioan Todinca

DISCO, November 24, 2011

Frugal computation

- Distributed system (arbitrary graph G), synchronous, each node has an identifier
- Frugal computation: during the algorithm, only $O(\log n)$ bits pass through each edge.

Our model: add a referee (universal vertex) u to graph G. What can/cannot be computed frugally?

- Each node knows its neighbors. One more round of communication

Frugal computation

- Distributed system (arbitrary graph G), synchronous, each node has an identifier
- Frugal computation: during the algorithm, only $O(\log n)$ bits pass through each edge.

Our model: add a referee (universal vertex) u to graph G. What can/cannot be computed frugally?

- Each node knows its neighbors. One more round of communication
- u can decide if G is a tree, a planar graph...

Frugal computation

- Distributed system (arbitrary graph G), synchronous, each node has an identifier
- Frugal computation: during the algorithm, only $O(\log n)$ bits pass through each edge.

Our model: add a referee (universal vertex) u to graph G. What can/cannot be computed frugally?

- Each node knows its neighbors. One more round of communication
- u can decide if G is a tree, a planar graph...
- u cannon decide if G has a triangle or a square, if G has diameter ≤ 3

Plan of the talk

1. A model for frugal computation based on a spanning tree [Grumbach, Wu, WG '09]
2. Our (stronger) model: $G + u$
   - Positive results: recognizing trees, planar graphs or any graphs of bounded degeneracy
   - Negative results (in one round): triangle detection
   - Negative results (arbitrary number of rounds): a teaser for communication complexity
3. Several open questions

The model of Grumbach and Wu

- Graph G has a BFS spanning tree T, each node knows its father in the tree.
- If G is of bounded degree any FOL formula ϕ can be evaluated frugally
  - Gaifman normal form: $\exists x_1, \ldots, x_s$, pairwise "far away", and $\phi^{(r)}(x_1) \wedge \cdots \wedge \phi^{(r)}(x_s)$
  - Each node collects the topology information in its r-neighborhood (bounded number of topologies)
  - It is enough to count the isomorphism types up to some constant

The model of Grumbach and Wu

- Graph G has a BFS spanning tree T, each node knows its father in the tree.
- If G is of bounded degree any FOL formula ϕ can be evaluated frugally
  - Gaifman normal form: $\exists x_1, \ldots, x_s$, pairwise "far away", and $\phi^{(r)}(x_1) \wedge \cdots \wedge \phi^{(r)}(x_s)$
  - Each node collects the topology information in its r-neighborhood (bounded number of topologies)
  - It is enough to count the isomorphism types up to some constant
- Similar results for planar G, using tree-decompositions of planar graphs of bounded radius.

Frugally decide if G is a forest

Actually the referee (universal vertex) u will compute the graph G.

- each vertex x sends to the referee vertex u
  - its identifier x
  - its degree $d_G(x)$
  - the sum of its neighbors $\sum_{y \in N_G(x)} y$
- u can recognize the vertices of degree one, then "remove" them; iterate the process

Bounded degeneracy graphs

G is of degeneracy at most k if, by repeatedly removing vertices of degree $\leq k$, we end up with an empty graph.

- Forests are exactly graphs of degeneracy 1
- Planar graphs have degeneracy ≤ 5
- Graphs of treewidth k have degeneracy $\leq k$
- H-minor free graphs have bounded degeneracy

Frugally decide if G is of degeneracy at most k

Actually the universal vertex u will compute graph G.

- each vertex x sends to the special vertex u
  - its identifier x
  - its degree $d_G(x)$
  - k other messages: $m_i(x) = \sum_{y \in N_G(x)} y^i$, for each $1 \leq i \leq k$
- u can recognize the vertices x of degree at most k

Frugally decide if G is of degeneracy at most k

Actually the universal vertex u will compute graph G.

- each vertex x sends to the special vertex u
  - its identifier x
  - its degree $d_G(x)$
  - k other messages: $m_i(x) = \sum_{y \in N_G(x)} y^i$, for each $1 \leq i \leq k$
- u can recognize the vertices x of degree at most k
- and their neighborhoods by solving the system of k equations $X_1^i + X_2^i + \cdots + X_{d(x)}^i = m_i(x)$

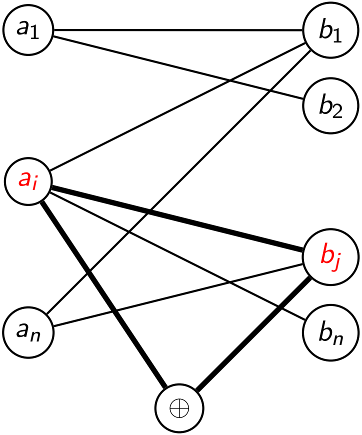
Frugally decide if G is of degeneracy at most k

Actually the universal vertex u will compute graph G.

- each vertex x sends to the special vertex u
  - its identifier x
  - its degree $d_G(x)$
  - k other messages: $m_i(x) = \sum_{y \in N_G(x)} y^i$, for each $1 \leq i \leq k$
- u can recognize the vertices x of degree at most k
- and their neighborhoods by solving the system of k equations $X_1^i + X_2^i + \cdots + X_{d(x)}^i = m_i(x)$
- then u "removes" the vertices of degree $\leq k$ and iterates the process.

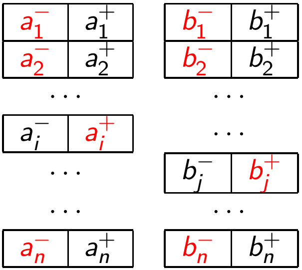
In one round, one cannot decide if G has a triangle

Bipartite graph H plus a "probe node"



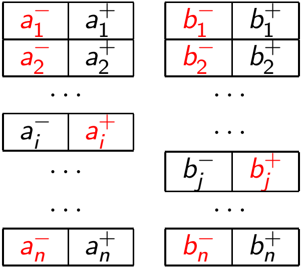
Triangles - part II

- Collect all messages (+ and -) for all vertices
- The red part tells whether there is an edge $a_i b_j$
- For $H \neq H'$, these collections must be different



Triangles - part II

- Collect all messages (+ and -) for all vertices
- The red part tells whether there is an edge $a_i b_j$
- For $H \neq H'$, these collections must be different



$O(n \log n)$ bits do not allow to distinguish $2^{\Theta(n^2)}$ bipartite graphs.

"Reduction" techniques for "hardness"?

We have proven: if there exists a $f(n)$-bits protocol for triangle detection in $2n+1$-vertex graphs, then there also exists a $2f(n)$-bits protocol reconstructing bipartite graphs with n vertices of each color.

- There is no frugal protocol detecting cycles with 4 vertices (easy reduction from RECONSTRUCTION of C_4-free graphs)
- There is no frugal protocol deciding if the diameter is at most 3 (very similar to triangle detection)
- BIPARTITNESS is at least as hard as CONNECTIVITYBIP (so what? see open questions)

A straightforward consequence of comunication complexity results

- Let $G_1 = G[1, 2, \ldots, n/2]$, $G_2 = G[n/2 + 1, n/2 + 2, \ldots, n]$
- Suppose the edges from G_1 to G_2 form a matching $\{i, i + n/2\}$
- One cannot frugally decide if G_2 is a copy of G_1.

A straightforward consequence of comunication complexity results

- Let $G_1 = G[1, 2, \ldots, n/2]$, $G_2 = G[n/2+1, n/2+2, \ldots, n]$
- Suppose the edges from G_1 to G_2 form a matching $\{i, i+n/2\}$
- One cannot frugally decide if G_2 is a copy of G_1.

Why?

A straightforward consequence of comunication complexity results

- Let $G_1 = G[1, 2, \ldots, n/2]$, $G_2 = G[n/2+1, n/2+2, \ldots, n]$
- Suppose the edges from G_1 to G_2 form a matching $\{i, i+n/2\}$
- One cannot frugally decide if G_2 is a copy of G_1.

Why? Communication complexity

A straightforward consequence of comunication complexity results

- Let $G_1 = G[1, 2, \ldots, n/2]$, $G_2 = G[n/2+1, n/2+2, \ldots, n]$
- Suppose the edges from G_1 to G_2 form a matching $\{i, i+n/2\}$
- One cannot frugally decide if G_2 is a copy of G_1.

Why? Communication complexity

- Alice has a boolean vector x_A of size k
- Bob has a boolean vector x_B of size k
- How many bits must Alice and Bob exchange in order to compute some function $f(x_A, x_B)$?

A straightforward consequence of comunication complexity results

- Let $G_1 = G[1, 2, \ldots, n/2]$, $G_2 = G[n/2+1, n/2+2, \ldots, n]$
- Suppose the edges from G_1 to G_2 form a matching $\{i, i+n/2\}$
- One cannot frugally decide if G_2 is a copy of G_1.

Why? Communication complexity

- Alice has a boolean vector x_A of size k
- Bob has a boolean vector x_B of size k
- How many bits must Alice and Bob exchange in order to compute some function $f(x_A, x_B)$?

To compute $EQUAL(x_A, x_B)$, they must exchange k bits [Wikipedia – Communication Complexity].

Summary

A model for frugal computation: $O(\log n)$ bits of communication per edge.

- Positive results for one round of computation: trees bounded degeneracy graphs (planar. . .)
- Unbounded number of rounds: one can do BFS
- Negative results (one round): local properties (triangle, square) and global properties (diameter); reduction techniques
- Negative results if the graph has an $O(n)$ edge cut

Main open question

What abous the CONNECTIVITY of G (in one or more rounds)?

- All our "reductions" may assume that the vertices are initially partitioned in a fixed number of parts (3, for TRIANGLEDETECTION), and the reduction works even if vertices of a same part share their information
- This can not work for CONNECTIVITY, we need new ideas
- (Naive remark) Similar difficulties arise in multiparty communication complexity

More open questions

- Extend the negative results to any constant number of communication rounds
- Find properties which are not decidable in one round, but which are in two or more rounds (candidate: decide if a graph is made of exactly two disjoint cliques)
- Randomized setting? (We did not really think of it)

More open questions

- Extend the negative results to any constant number of communication rounds
- Find properties which are not decidable in one round, but which are in two or more rounds (candidate: decide if a graph is made of exactly two disjoint cliques)
- Randomized setting? (We did not really think of it)

Your questions?