Lower Bounds for Symbolic Computation on Graphs

Highlights of Algorithms 2018

Speaker: Wolfgang Dvořák,
Institute of Logic and Computation, TU Wien

Based on:
Motivation

- Graph algorithms are central in the analysis of reactive systems:
  - States of the System → Vertices of the graph
  - State Transitions → Edges of the graph
- The resulting graphs are huge
  - The number of vertices is exponential in the number of variables
  - Explicit representation of graphs is infeasible
  - Graphs are implicitly represented using e.g. binary-decision diagrams (BDDs) = symbolic computation
- Set-based symbolic model of computation
  - Same operations as standard RAM algorithms, except
    - for access to the edges and nodes of the input graph
    - for manipulation of sets of vertices
Model: Set-based Symbolic Computation

- Access to edges: Only through **One-step operations Pre and Post**:
  - Predecessor Operation $Pre(X)$:
    \[
    Pre(X) = \{ v \in V \mid \exists x \in X: (v, x) \in E \} \]
  - Successor Operation $Post(X)$:
    \[
    Post(X) = \{ v \in V \mid \exists x \in X: (x, v) \in E \} \]

- **Manipulation of sets of vertices: Basic set operations**
  - Given one or two sets of vertices, we can perform basic set operations like union, intersection or complement

- **Symbolic Space requirement** = number of sets simultaneously stored by an algorithm
  - We deal with compact representation of huge graphs
  - The number of stored sets should be small w.r.t. size of the graph, ideally constant.
Fundamental problems in graphs

- **Problems on graphs**: Starting from a vertex we have to decide whether there exists an infinite path satisfying a certain objective.
  - Objectives arising in the analysis of reactive systems:
    - Reachability
    - Safety
    - Liveness (Büchi)
    - co-liveness (coBüchi)
- **Computing SCCs** is at the heart of the fastest algorithms for the above problems (and thus of interest)
- Many graphs, e.g., in hardware verification, have small diameter $D$ which (once detected) can be exploited for more efficient algorithms
  - We consider computing the (approximate) diameter of a graph
Results

• First lower bounds for the Set-based Symbolic Model of Computation
  • Demonstrate Communication Complexity to be an appropriate tool to show lower bounds for symbolic computation

• Matching upper and lower bounds for fundamental problems

<table>
<thead>
<tr>
<th></th>
<th>Reach</th>
<th>SCC</th>
<th>Safety</th>
<th>Büchi</th>
<th>coBüchi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reach</td>
<td>$\Theta(D)$</td>
<td>$\Theta(n)$</td>
<td>$\Theta(n)$</td>
<td>$\Theta(n)$</td>
<td>$\Theta(n)$</td>
</tr>
</tbody>
</table>

• Interesting gap between Reach $\Theta(D)$ and the other problems $\Theta(n)$ even for constant diameter graphs
Results

- Refined Analysis of the SCC algorithm by Gentilini et al.
  - and matching lower bounds
    - $\Theta(\min(n, (D \cdot |SCCs(G)|), (\sum_{C \in SCCs(G)}(D_C + 1))))$
    - $D$ ... Diameter of the Graph
    - $D_C$ ... Diameter of the SCC $C$

- Upper and lower bounds for (approximate) diameter

<table>
<thead>
<tr>
<th></th>
<th>exact</th>
<th>$(1 + \epsilon)$ approx</th>
<th>$(3/2 - \epsilon)$ approx</th>
<th>2 approx</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Upper bound</strong></td>
<td>$O(n \cdot D)$</td>
<td>$\tilde{O}(n \cdot \sqrt{D})$</td>
<td>$\tilde{O}(n \cdot \sqrt{D})$</td>
<td>$O(D)$</td>
</tr>
<tr>
<td><strong>Lower bound</strong></td>
<td>$\Omega(n)$</td>
<td>$\Omega(n)$</td>
<td>$\Omega(n)$</td>
<td></td>
</tr>
</tbody>
</table>
Summary and Conclusion

- Different model of computation: **Set-based symbolic computation**
  - First lower bounds and matching upper bounds for
    - fundamental objectives in graphs
    - SCC computation
    - (approximate) diameter
  - Communication Complexity is the right tool for (sub-) linear lower bounds for symbolic algorithms
Summary and Conclusion

- Different model of computation: **Set-based symbolic computation**
  - First lower bounds and matching upper bounds for
    - fundamental objectives in graphs
    - SCC computation
    - (approximate) diameter
  - **Communication Complexity is the right tool** for (sub-)linear lower bounds for symbolic algorithms

Thank you for your attention!
(and see you at the poster)