Lower Bounds for Symbolic Computation on Graphs

Krishnendu Chatterjee[1], Wolfgang Dvořák[2,3], Monika Henzinger[3], and Veronika Loitzenbauer[4]

[1] IST Austria [2] TU Wien, Austria [3] University of Vienna, Austria [4] Johannes Kepler University Linz, Austria

Motivation

Graph algorithms are central in the *formal analysis of reactive systems*. A reactive system consists of a set of variables and a state of the system corresponds to a set of valuations, one for each of these variables. This induces a directed graph: Each vertex represents a state of the system and each directed edge represents a possible state transition.

- The resulting *graphs are huge* (exponential in the number of variables)
  ↪ Explicit representation of graphs is infeasible
- *Graphs are implicitly represented* using e.g. binary-decision diagrams (BDDs)

To avoid considering specifics of the implicit representation and their manipulation, an elegant theoretical model for algorithms that work on this implicit representation has been developed, called *symbolic algorithms*.

Set-based Symbolic Model of Computation

Symbolic Algorithms allow the same operations as standard RAM algorithms, except

- for access to the vertices and edges of the input graph, and
- for manipulation of sets of vertices.

Symbolic Operations

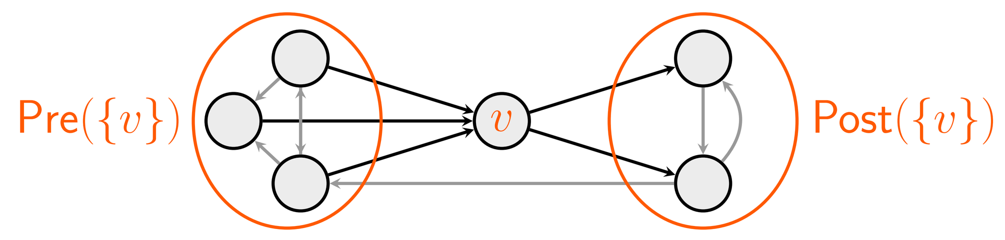
The input graph can be only accesses via the following symbolic operations.

- *Access to edges*: Only through *One-step operations* Pre and Post:
  - Successor Operation

$$\mathsf{Post}(S) = \{v \in V \mid \exists s \in S : (s, v) \in E\}$$

  - Predecessor Operation

$$\mathsf{Pre}(S) = \{v \in V \mid \exists s \in S : (v, s) \in E\}$$



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

Symbolic Space

The *Symbolic Space requirement* of an algorithm is the number of sets simultaneously stored by the algorithm. As we deal with compact representation of huge graphs and the number of stored sets should be small w.r.t. size of the graph, i.e., $O(\log n)$.

Our Results

We provide the first lower bounds for the Set-based Symbolic Model of Computation and provide matching upper and lower bounds for fundamental problems.

Computing the (approximate) diameter of a graph

The *diameter D of a graph* is defined as the largest finite distance in the graph. Many graphs, e.g., in hardware verification, have small diameter D which can be exploited for more efficient algorithms. We provide a lower bound and a symbolic approx. scheme.

	# symbolic operations			
approx.	exact	$1+\varepsilon$	$3/2-\varepsilon$	2
upper bound	$O(n \cdot D)$	$\widetilde{O}(n\sqrt{D})$	$\widetilde{O}(n\sqrt{D})$	$O(D)$
lower bound	$\Omega(n)$	$\Omega(n)$	$\Omega(n)$	

Deciding Fundamental Objectives

Starting from a vertex we want to decide whether there exists an infinite path satisfying certain objectives arising in the analysis of reactive systems, e.g., Reachability, Safety, Liveness (Büchi), and co-liveness (coBüchi).

# symbolic operations			
Reach	Safety	Büchi	coBüchi
$\Theta(D)$	$\Theta(n)$	$\Theta(n)$	$\Theta(n)$

Interesting gap between $\Theta(D)$ for Reach and $\Omega(n)$ for the other objectives even for constant diameter graphs.

Strongly Connected Components

Computing Strongly Connected Components (SCCs) is at the heart of the algorithms for the above objectives. The best *known upper bound* $O(\min(n, D \cdot |SCCs(G)|))$ is by the algorithm of Gentilini et al. [2]. We give *matching lower bounds* and a *refined analysis* in terms of the diameters of the SCCs of the graph. Here $SCCs(G)$ is the set of SCCs of G and D_C is the diameter of the SCC C. We obtain that computing SCC is in

$$\Theta\big(\min\big(n\ ,\ D \cdot |SCCs(G)|\ ,\ \textstyle\sum_{C \in SCCs(G)}(D_C+1)\big)\big)$$

Communication Complexity

Our lower bounds are by *reductions from the Set Disjointness Problem* from communication complexity.

Two-party Communication Complexity Model

- Two parties Alice, Bob
  - Alice and Bob need to compute a function $f(x, y)$, but
  - $x \in X$ is only known to Alice and $y \in Y$ is only known to Bob.
- Aim: send as few bits as possible between Alice and Bob
  ↪ *We do not count computation but only communication*
- Communication protocol ("the algorithm")
  - determines which player sends which bits when,
  - is fixed beforehand, and is known to both Alice and Bob.

Set Disjointness Problem (with parameter k)

- Universe $U = \{0, \ldots, k-1\}$
- Alice's input: bit vector x of length k
- Bob's input: bit vector y of length k
- Function $f(x, y) = 1$ iff for all $0 \le i \le k-1$ either $x_i = 0$ or $y_i = 0$

Thm 1. [3] *Any protocol for Set Disjointness sends $\Omega(k)$ bits in the worst case.*

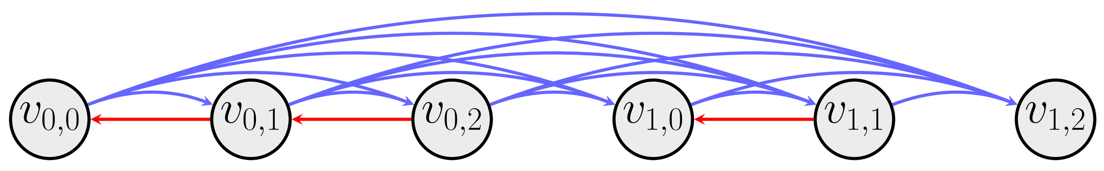
A Lower Bound for Computing SCCs

We exemplify our technique by one result for SCCs. Given an instance (x, y) of Set Disjointness with $k = \ell \cdot \bar{k}$ we define the graph $G = (V, E)$ with:

- $V = \bigcup_{i=0}^{\ell-1} V_i$ with $V_i = \{v_{i,0}, \ldots, v_{i,\bar{k}}\}$
- Forward Edges $\{(v_{i,j}, v_{i',j'}) \mid i < i' \text{ or } i = i' \wedge j < j'\}$
- Backward Edges $\{(v_{i,j+1}, v_{i,j}) \mid x_{i \cdot \bar{k}+j} = 0 \text{ or } y_{i \cdot \bar{k}+j} = 0\}$

Alice and Bob both know the Forward Edges and only need to communicate about the Backward edges.

The graph G for $k = 4, \ell = 2$ and $x = (0, 0, 1, 1)$, $y = (1, 1, 0, 1)$ is given below:



Lem 2. *$f(x, y) = 1$ iff G has exactly ℓ SCCs.*

One can show that each symbolic operation can be performed in a communication protocol with $O(1)$ bits of communication.

Lem 3. *For any algorithm that computes SCCs with N symbolic operations there is a communication protocol for Set Disjointness that requires $O(N)$ Communication.*

Thus by Thm 1 we have a $\Omega(n)$ lower bound for computing SCCs, even in graphs of constant diameter.

Thm 2. *Any symbolic algorithm that computes the SCCs of graphs with n vertices needs $\Omega(n)$ symbolic one-step operations.*

Summary & Conclusion

We consider a set-based symbolic computation as a different model of of computation:

- We give the first lower bounds and matching upper bounds for fundamental objectives in graphs like SCC computation and (approximate) diameter.
- Demonstrate communication complexity to be a suitable tool to show (sub-)linear lower bounds for symbolic computation.

We identify the following *challenges for future research*.

- Super-linear set-based symbolic lower bounds, in particular tight bounds for diameter.
- A notion of (symbolic) reduction between set-based symbolic problems. Notice that all our reductions start from a communication complexity problem.

References

[1] Chatterjee, K., Dvořák, W., Henzinger, M., and Loitzenbauer, V. (2018). Lower bounds for symbolic computation on graphs: Strongly connected components, liveness, safety, and diameter. In Proc. *SODA 2018*, pages 2341–2356.

[2] Gentilini, R., Piazza, C., and Policriti, A. (2008). Symbolic graphs: Linear solutions to connectivity related problems. *Algorithmica*, 50(1):120–158. Announced at SODA'03.

[3] Kushilevitz, E., and Nisan, N. (1997). Communication Complexity. *Cambridge University Press* (cit. on pp. 4, 5)

IST Austria — TU Wien Fakultät für Informatik — universität wien — WWTF Wiener Wissenschafts-, Forschungs- und Technologiefonds — JKU Johannes Kepler Universität Linz — RiSE