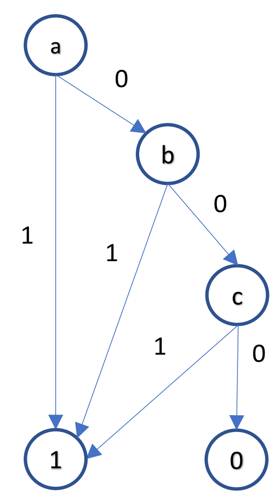
MCSP is Hard for Read-Once Nondeterministic Branching Programs

Ludmila Glinskih and Artur Riazanov

Boston University EPFL

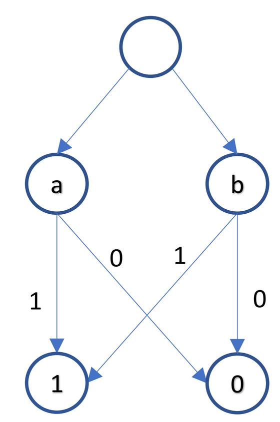
Branching program



- Acyclic directed graph
- Two sinks: 1-sink and 0-sink
- All vertices labeled by variables
- Value: sink label at the end of the path that corr. to the subst.
- Size = number of nodes

Nondeterministic branching program

- Has nodes without labels
- Value equals one if there exists a path from source to 1-sink
- Size = number of labeled nodes



Read-once branching program

- Every path has only *1* occurrence of each variable

Minimum Circuit Size Problem

Input:
- truth table of a Boolean function $f: \{0,1\}^n \to \{0,1\}$
- size parameter s

Output:
yes, if f can be computed by a circuit of size at most s

In a **Partial** MCSP input is a truth-table of a partial f

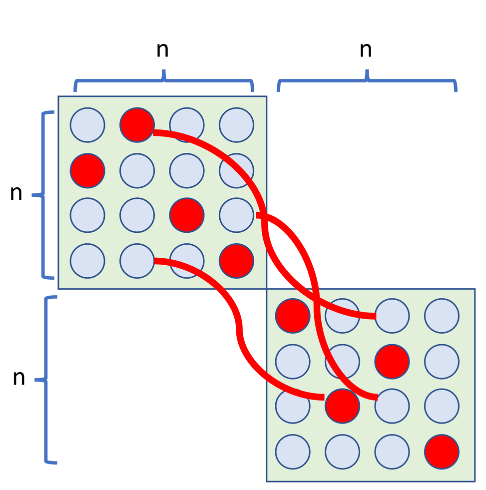
Hardness of MCSP for BPs implications

Theorem: if MCSP cannot be computed by a branching program of size $N^{2.01}$ then NP $\not\subset$ C-SIZE$[n^k]$ for all k [Chen, Jin, Williams, 2019]

The best lower bound: BP(ED)=$\Omega\left(\frac{n^2}{\log^2 n}\right)$ [Nechiporuk, 1966]

To develop new techniques to show lower bounds for BPs we study hardness of restricted versions of BP

(n x n)-Bipartite Permutation Independent Set Problem



- Graph with 2n x 2n vertices,
- Edges exist only between vertices from two quadrants
- Need to find exactly one vertex from every row, and exactly one vertex from every column, such that
  - These vertices are from the two quadrants
  - These vertices form independent set

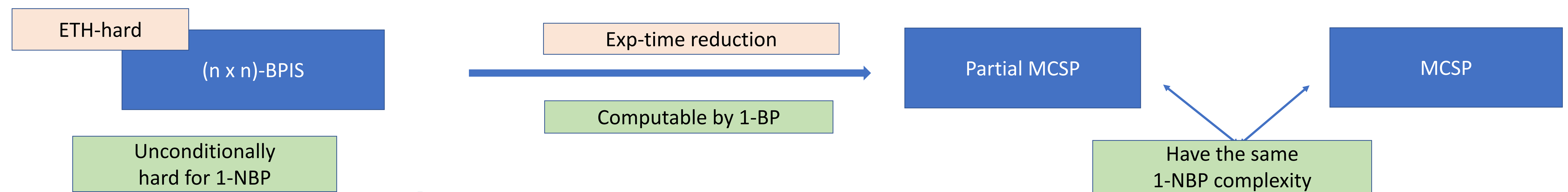
Main result

Theorem: size of 1-NBP computing MCSP is $N^{\Omega(\log\log N)}$

This result is tight for MCSP with linear size parameter.

To prove this lower bound we adapt a framework from the work [Ilango 2020], in which the author showed an ETH-hardness of partial MCSP.

Sketch of the proof of the Theorem:
Assume there is a small 1-NBP computing MCSP. As the sizes of 1-NBP for MCSP and Partial MCSP are polynomially related, there is a small 1-NBP computing Partial MCSP.
Then, from this small 1-NBP for Partial MCSP we can get a small read-once 1-NBP for (n x n)-BPIS. Which is impossible unconditionally. Hence, MCSP cannot be computed by a small 1-NBP.



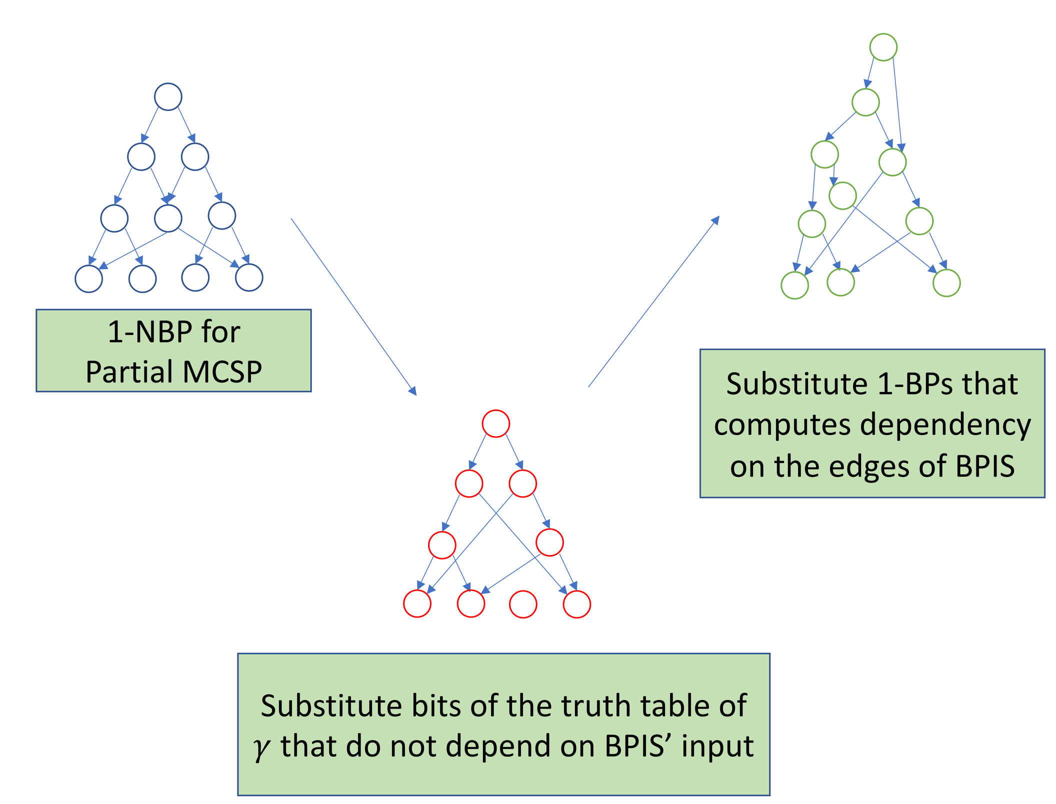
Lemma: size of 1-NBP computing an (n x n)-BPIS is $2^{\Omega(n \log n)}$

Idea of the proof:
- Show that the minimum 1-NBP for BPIS has the same size as the minimum 1-NBP for Bipartite Permutation Clique
- Adapt the proof of the lower bound on 1-NBP for Clique_Only to get a lower bound on BPC
  - Use bottleneck arguments
  - BPC is like "clique" with permutations

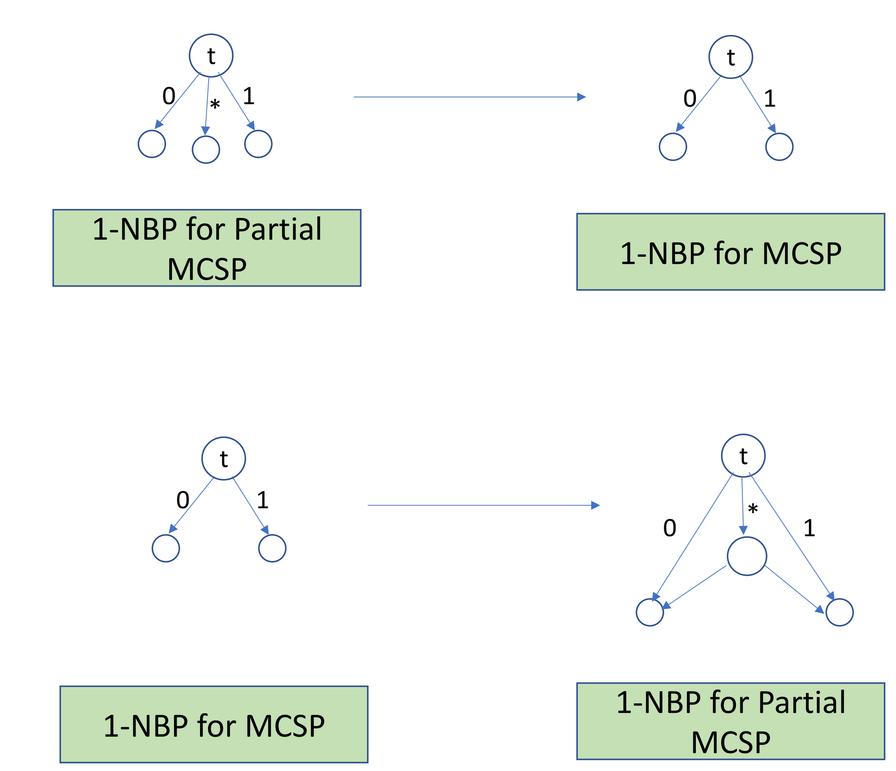
Lemma: (n x n)-BPIS is 1-BP reducible to Partial MCSP

$$\begin{cases} \bigvee_{i\in[2n]}(y_i \wedge z_i) & \text{, if } x = 0^{2n} \\ \bigvee_{i\in[2n]} z_i & \text{, if } x = 1^{2n} \\ \bigvee_{i\in[2n]}(x_i \vee y_i) & \text{, if } z = 1^{2n} \\ 0 & \text{, if } z = 0^{2n} \\ \mathsf{OR}_n(x_1,\dots,x_n) & \text{, if } z = 1^n0^n \text{ and } y = 0^{2n} \\ \mathsf{OR}_n(x_{n+1},\dots,x_{2n}) & \text{, if } z = 0^n1^n \text{ and } y = 0^{2n} \\ 1 & \text{, if } \exists\, ((j,k),(j',k')) \in E \text{ such that } (x,y,z) = (\overline{e_k e_{k'}}, 0^{2n}, e_j e_{j'}) \\ \star & \text{, otherwise} \end{cases}$$

Only these bits of the truth table depend on the input bits of BPIS



Lemma: the size of the minimal 1-NBP computing Partial MCSP equals the size of the minimal 1–NBP computing MCSP



Upper Bound

Lemma: MCSP on an input of length 2^n with a size parameter s can be computed by a 1-NBP of size $2^n 2^{O(s \log s)}$

Corollary: our $2^{\Omega(n \log n)}$ lower bound is tight for inputs with a linear size parameter

Future work

Show tight lower bound for MCSP with higher size parameters
- The same technique cannot work, as we cannot construct a truth table of a function with higher than linear circuit complexity

Extend this result to other models of computations
- For any model in which (n x n)-BPIS is hard and the reduction to the truth table is efficiently computable the same size lower bound will hold