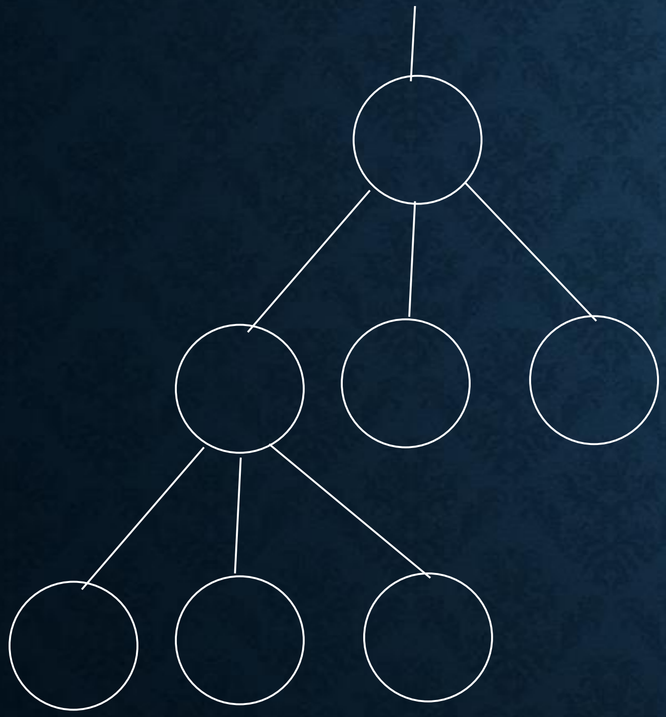
QUANTUM ALGORITHMS FOR TREE SIZE ESTIMATION, WITH APPLICATIONS

Andris Ambainis, Mārtiņš Kokainis

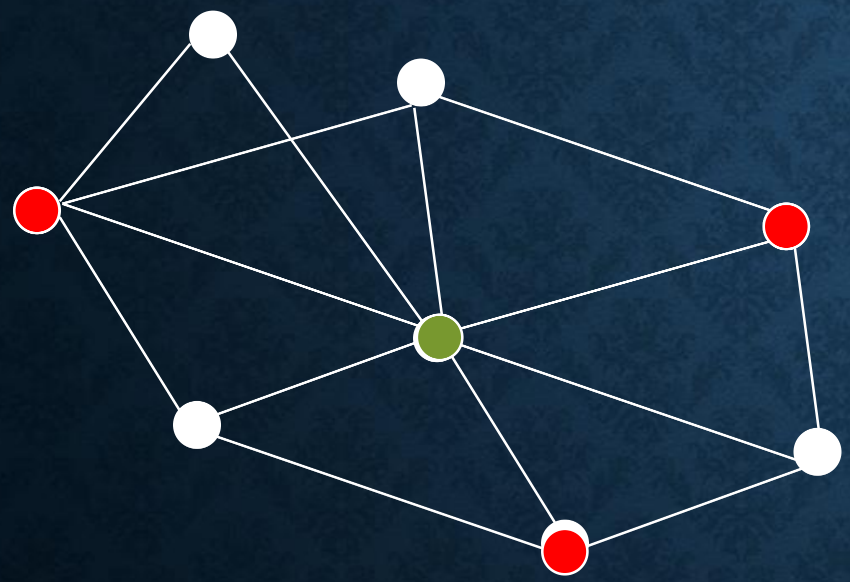
University of Latvia

SEARCH TREE OF UNKNOWN STRUCTURE



- We are given:
  - Root r;
  - Black box which takes vertex v, outputs all children of v.
  - Black box for testing if v is a leaf.

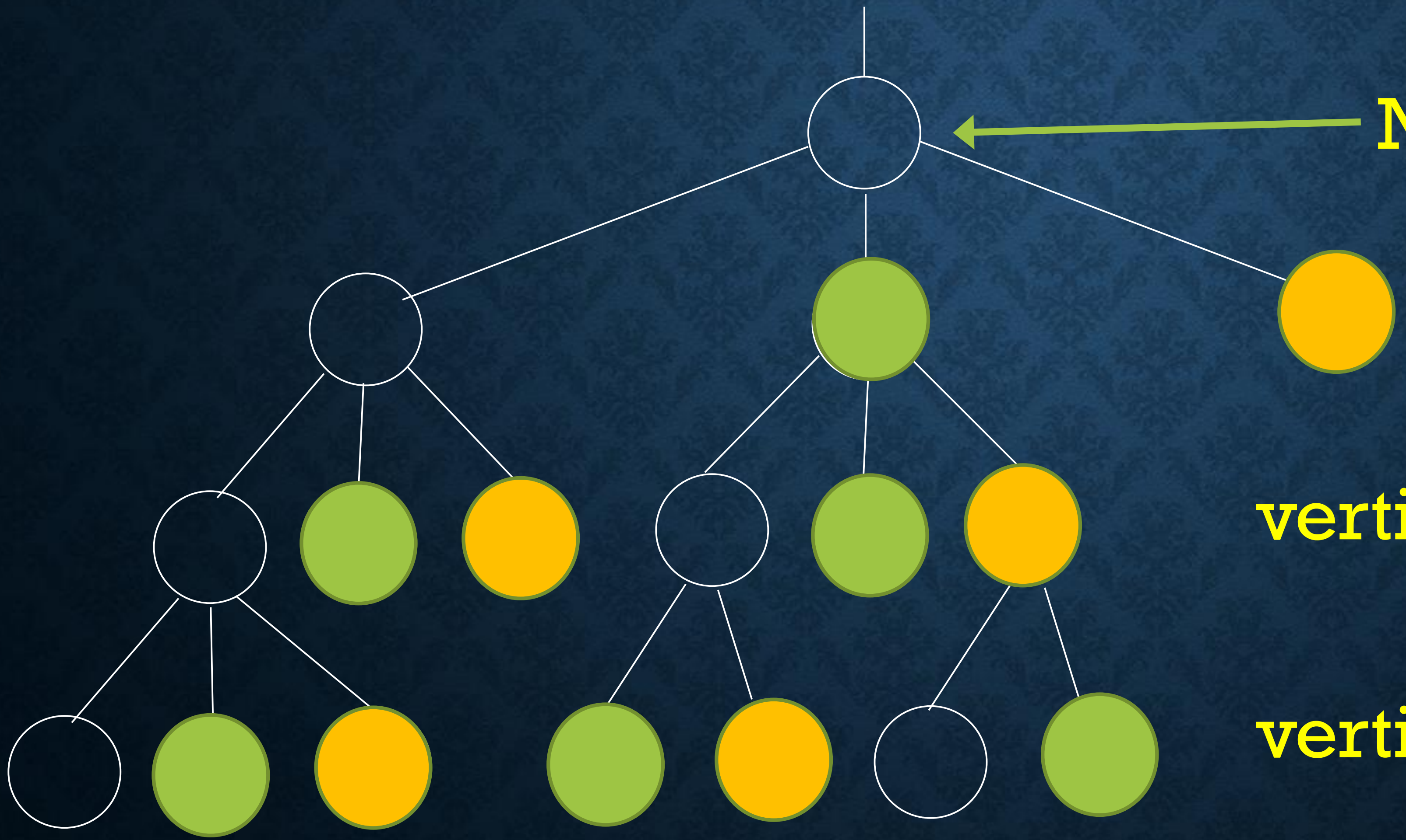
APPLICATION 1



- 3-COLORING: Can we colour vertices with 3 colours so that no edge is monochromatic?
- NP-complete.

Algorithm: attempt to colour vertices one by one.

TREE OF PARTIAL COLORINGS



No vertices coloured

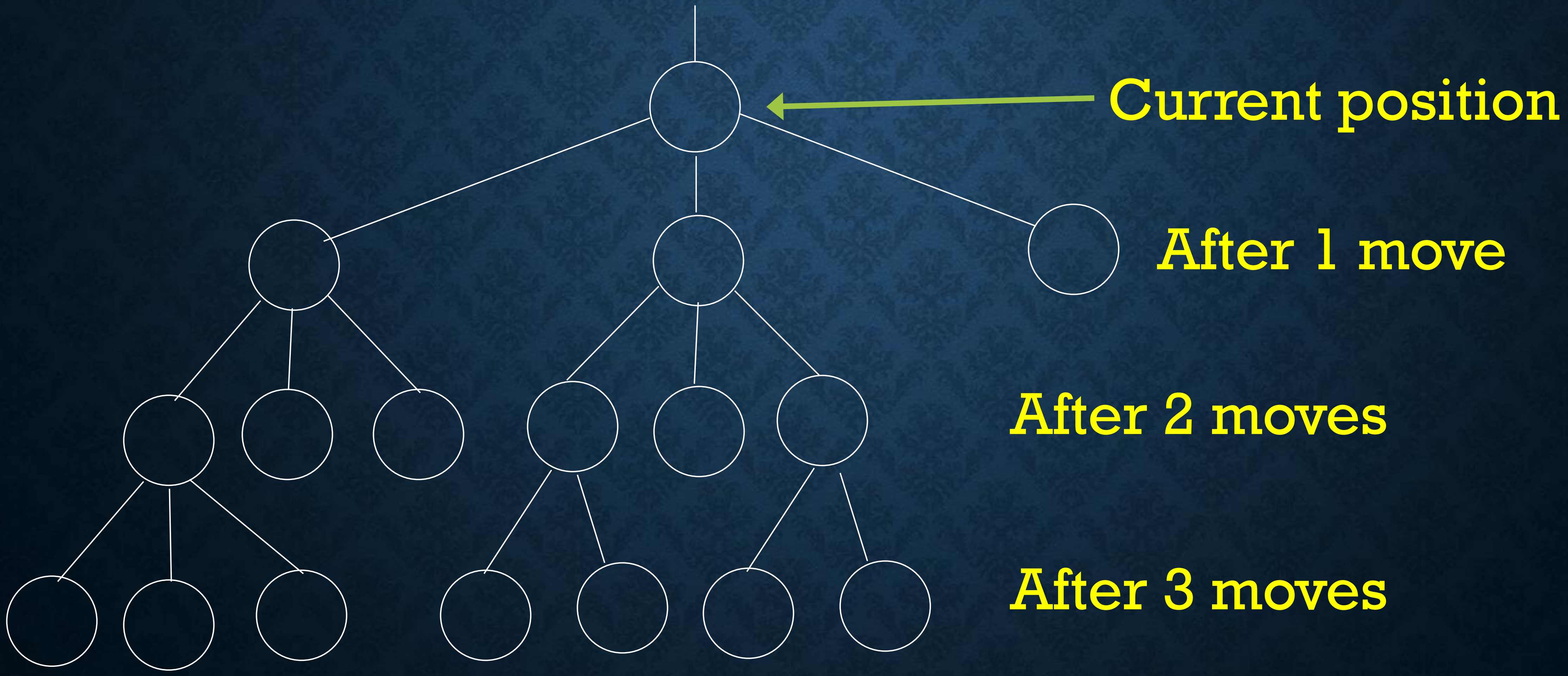
vertex 1 coloured

vertices 1, 2 coloured

vertices 1, 2, 3 coloured

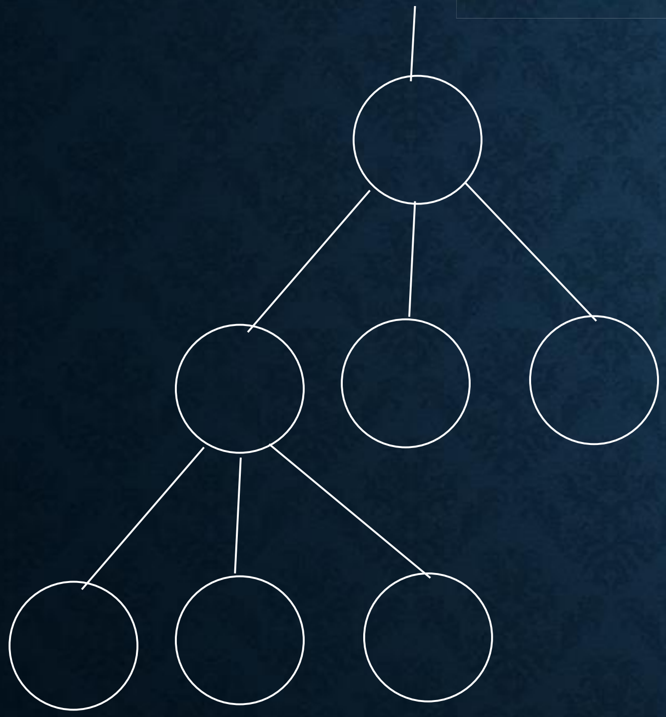
APPLICATION 2

TREE OF POSITIONS



How large is this tree (up to 1%)?

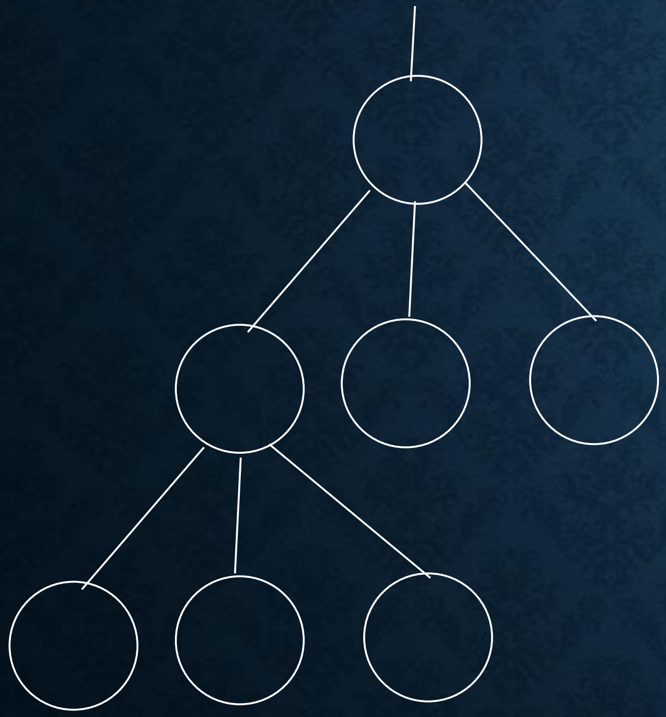
OUR QUANTUM ALGORITHM



- T – size of the tree.
- Produces an estimate T' such that $|T-T'| \leq \varepsilon T$.
- Time: $O\left(\sqrt{Tn}\right)$, n – depth of the tree.

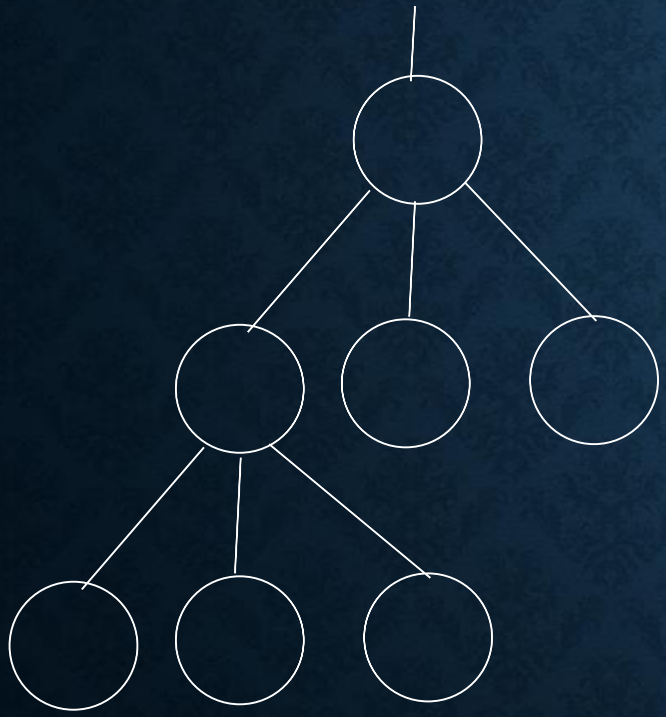
APPLICATION 1: BACKTRACKING

MONTANARO, 2015



- Tree of unknown structure.
- Some leaves marked.
- Quantum algorithm for finding a marked leaf in time $O\left(\sqrt{Tn}\right)$.
- Useful for speeding up backtracking (e.g., 3-colouring).

OPEN PROBLEM



- Classical algorithm may examine the most promising branches first.
- Running time T' much smaller than tree size T.

OUR RESULT

- Quantum algorithm with running time $O\left(\sqrt{T'}n^{1.5}\right)$ where
  - T' - number of vertices visited by classical search algorithm;
  - n - depth of the tree.

OUR ALGORITHM

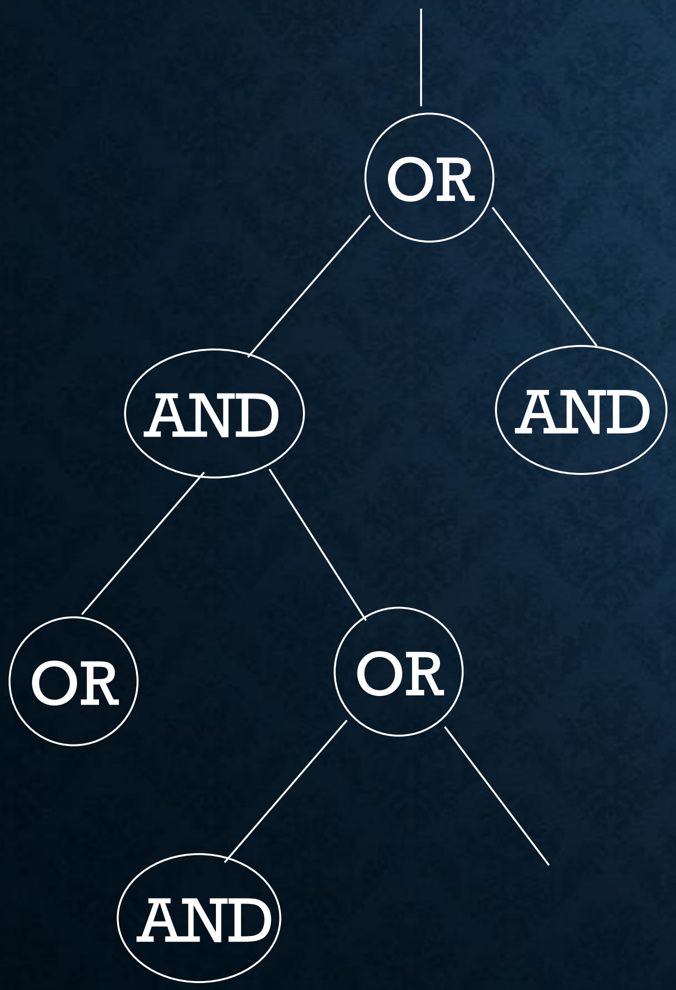
- T_i – subtree consisting of first 2^i vertices visited by the classical algorithm.
- Montanaro's algorithm on $T_1, T_2, \ldots, T_k$ until $T' \leq 2^k$.
- Running time: $O\left(\sqrt{2^k n}\right) = O\left(\sqrt{T' n}\right)$.

CONSTRUCTING SUBTREES

- Example: need subtree with 16 first vertices.
- Tree size estimation on every child of the root.

APPLICATION 2: 2-PLAYER GAMES

POSITION TREE



1st player

2nd player

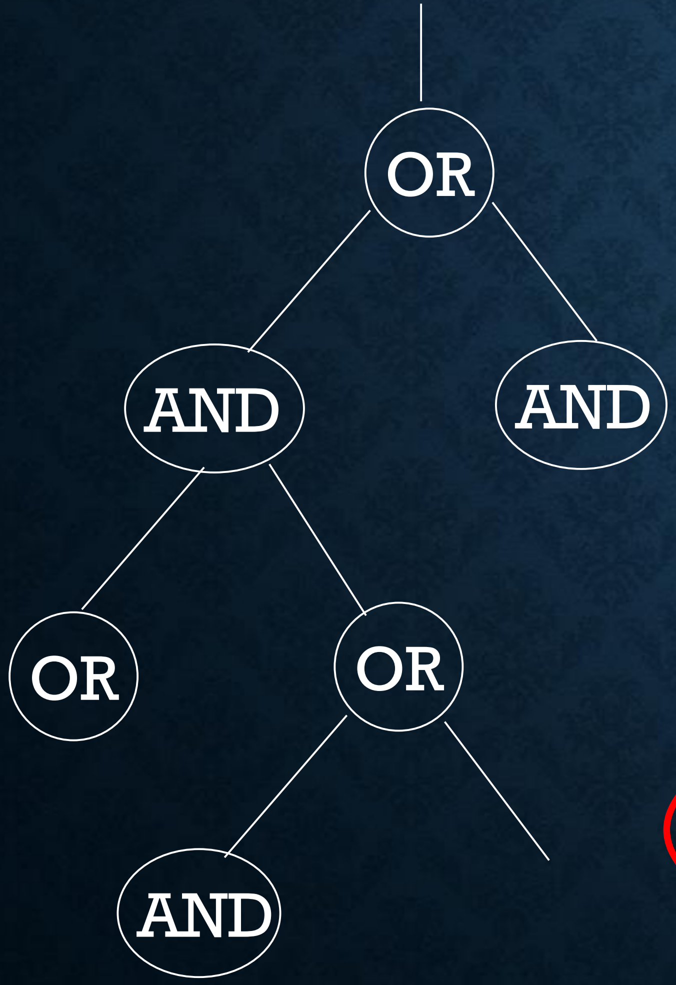
- Position tree = formula;
- Output = YES if 1st player wins;

EVALUATING BOOLEAN FORMULAS

- AND/OR formula of size T can be evaluated by evaluating $O(\sqrt{T})$ leaves [Reichardt, 2010].

Not applicable to game trees!

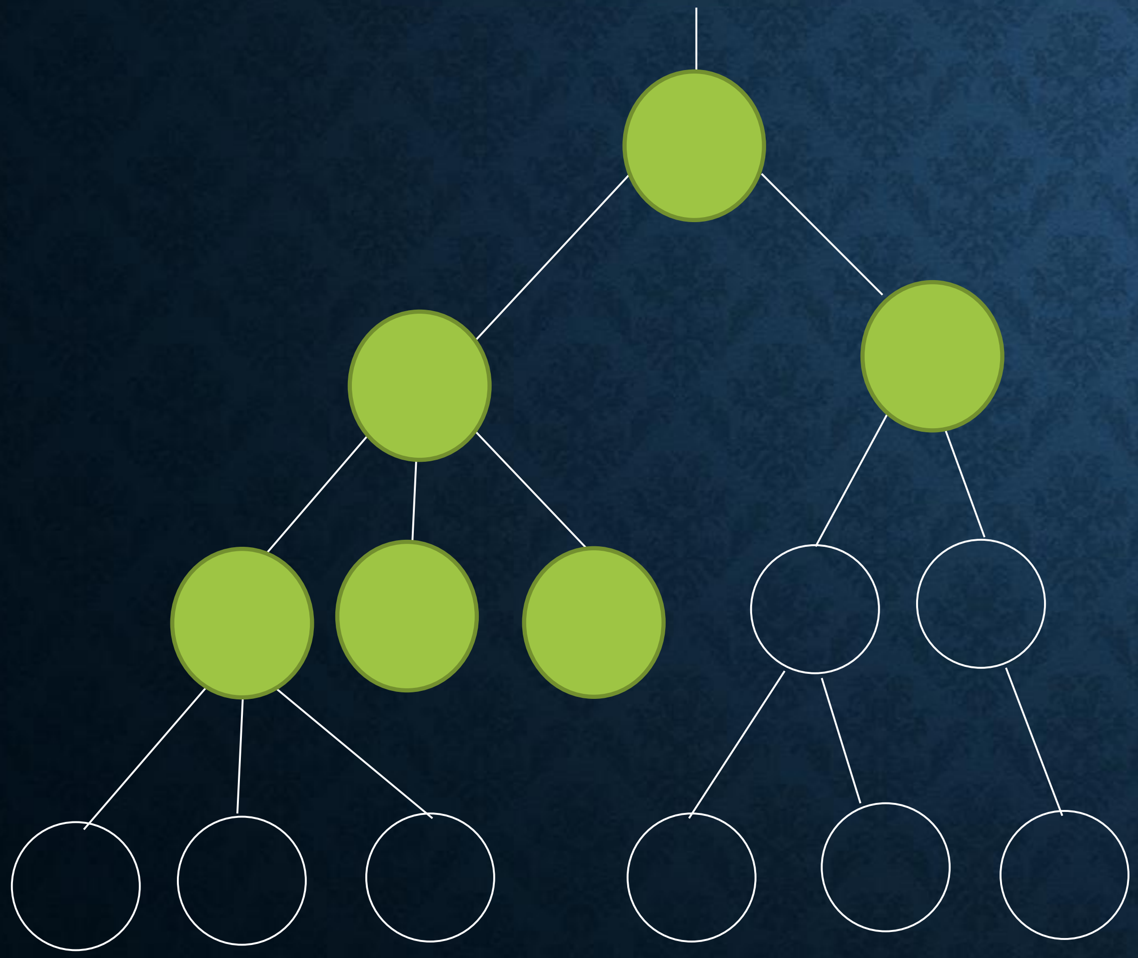
A, CHILDS, ŠPALEK, REICHARDT, ZHANG, 2007



- Basis states: $|u, v\rangle$, uv – edge.
- Coin flip transformation C_u on $|u, v_1\rangle, \ldots, |u, v_k\rangle$ with the same u.
- C_u depends on sizes of subtrees rooted at $v_1, \ldots, v_k$.

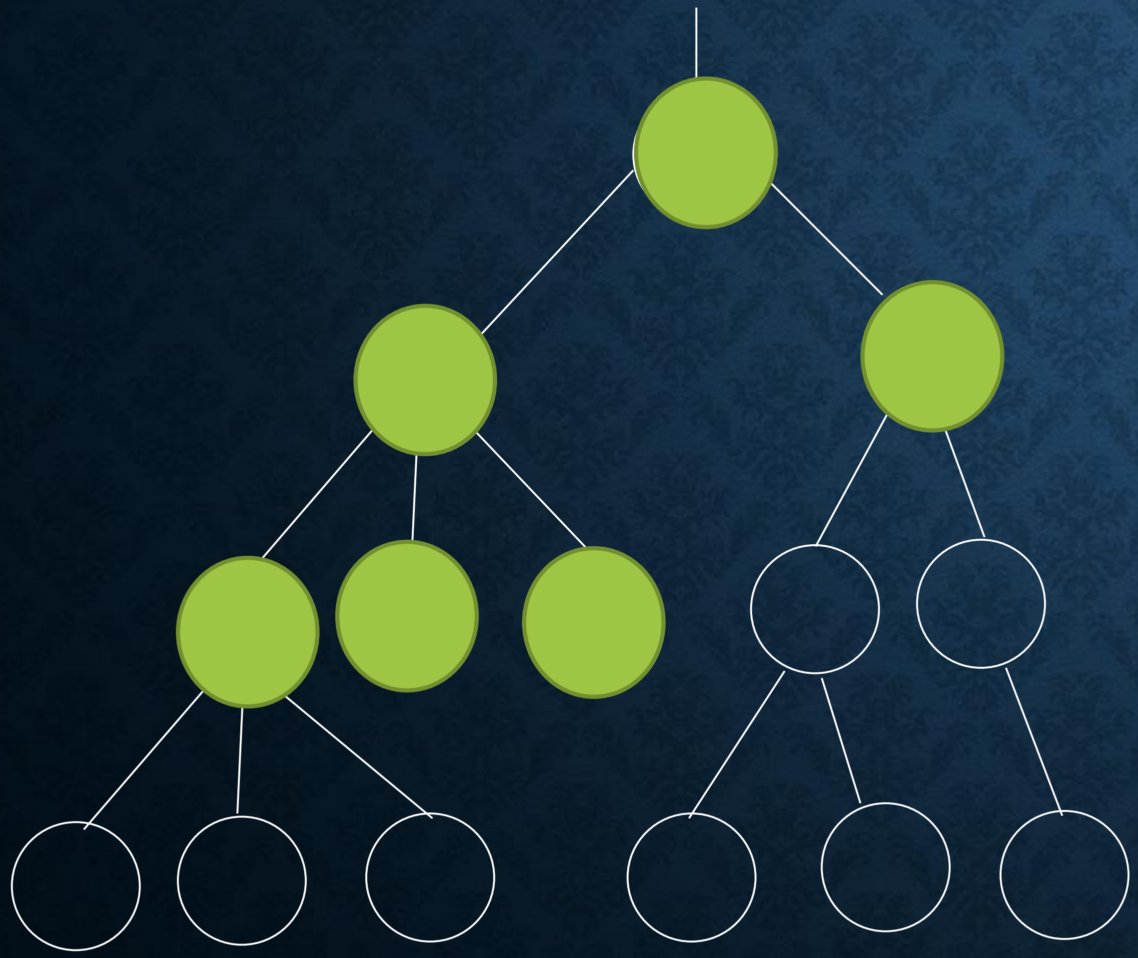
Trim the tree!

EVALUATING UNKNOWN FORMULAS



- T – size of the tree;
- Vertex v – heavy if S_v contains $\geq T/c$ vertices.
- Subtree T' – heavy vertices and their children.

ALGORITHM A



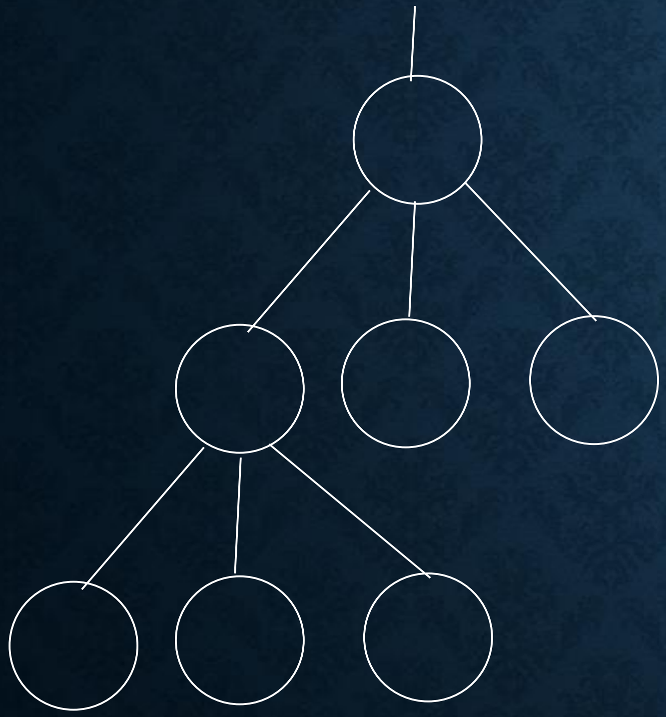
- Explore tree with tree size estimation to determine T'.
- Run AC+ algorithm on T', with recursive calls to A at the leaves.

OUR RESULT

- AND-OR trees of unknown structure with size T, depth $d=T^{o(1)}$ can be evaluated in $O(T^{1/2+o(1)})$ quantum steps.

TREE SIZE ESTIMATION

OUR RESULT

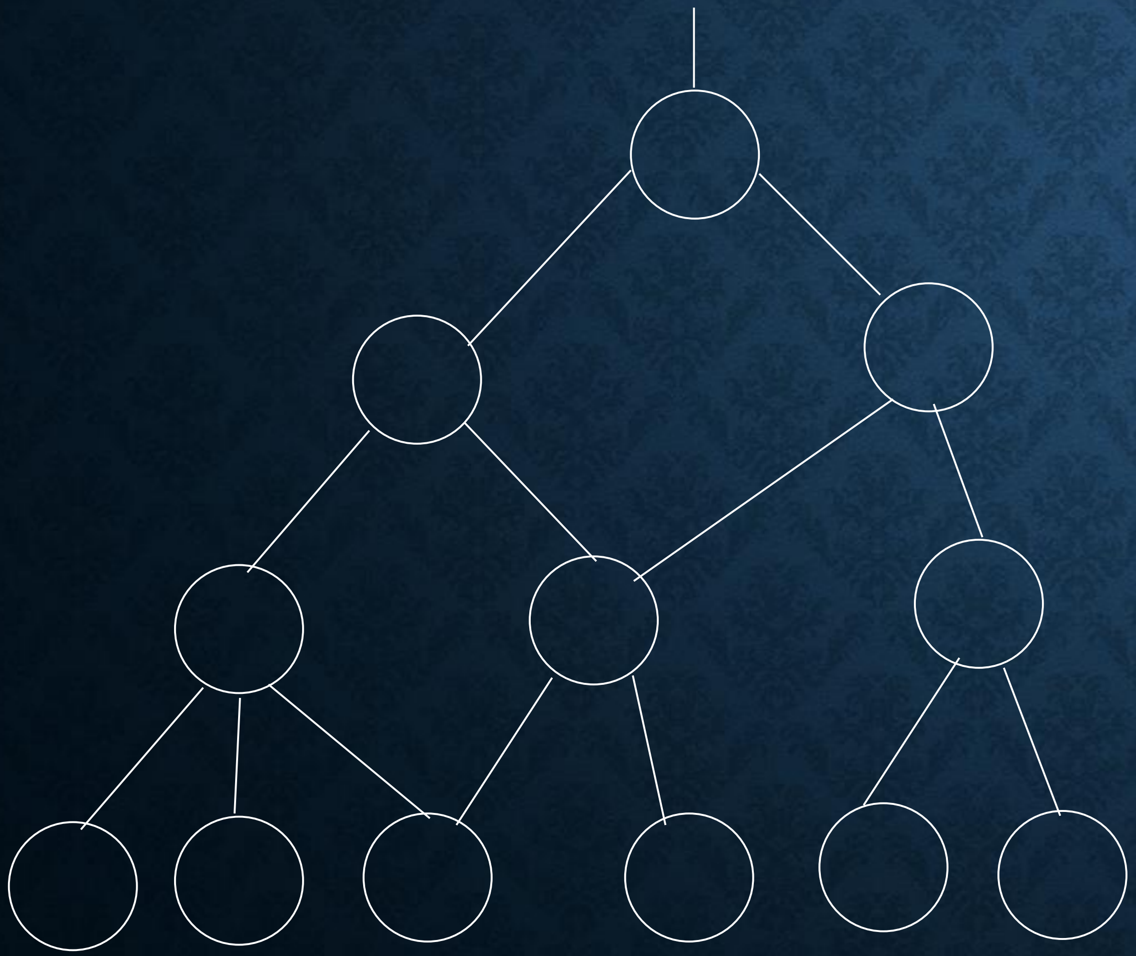


- T – size of the tree.
- Estimate T' such that

  $$|T-T'| \leq \varepsilon T.$$

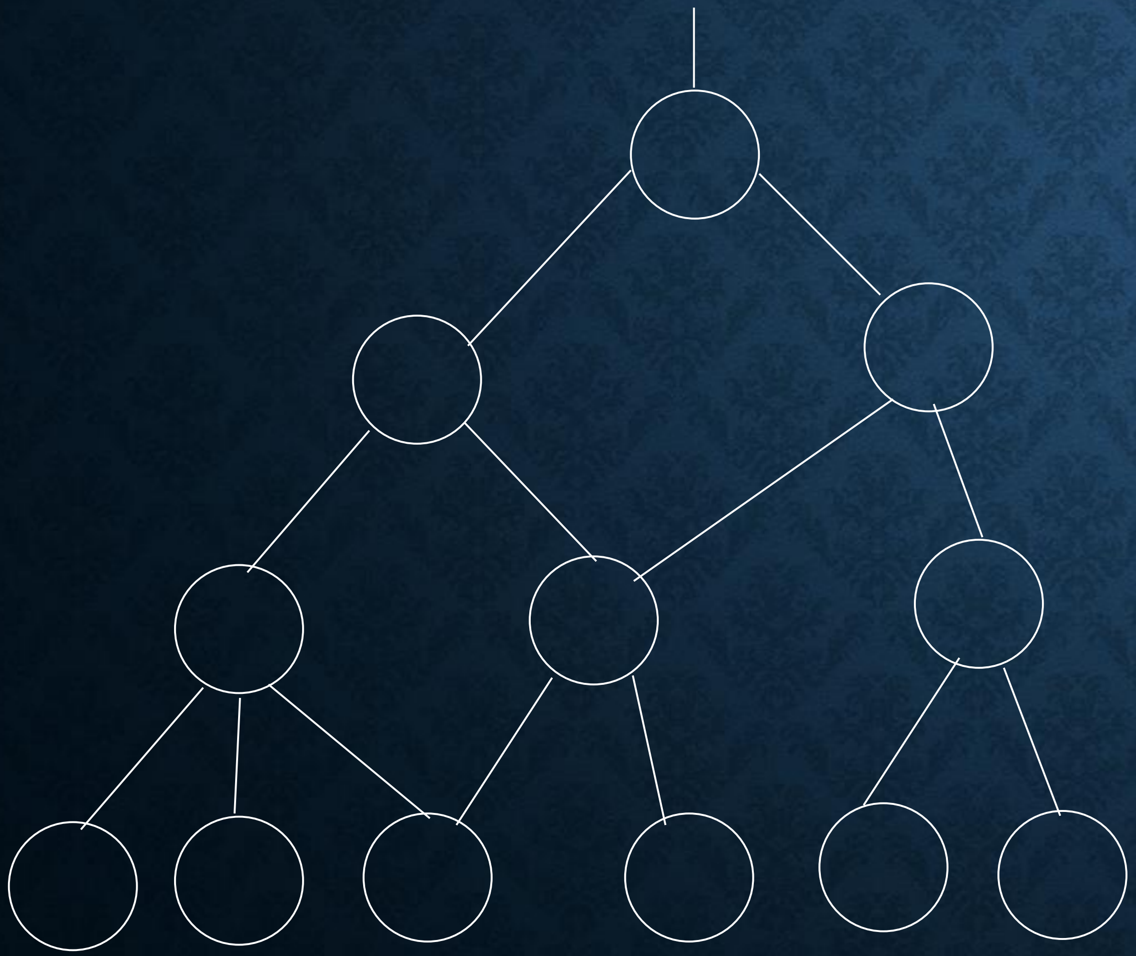
- Running time: $O\left(\sqrt{Tn}\right)$, n – depth of the tree.

GENERALIZATION



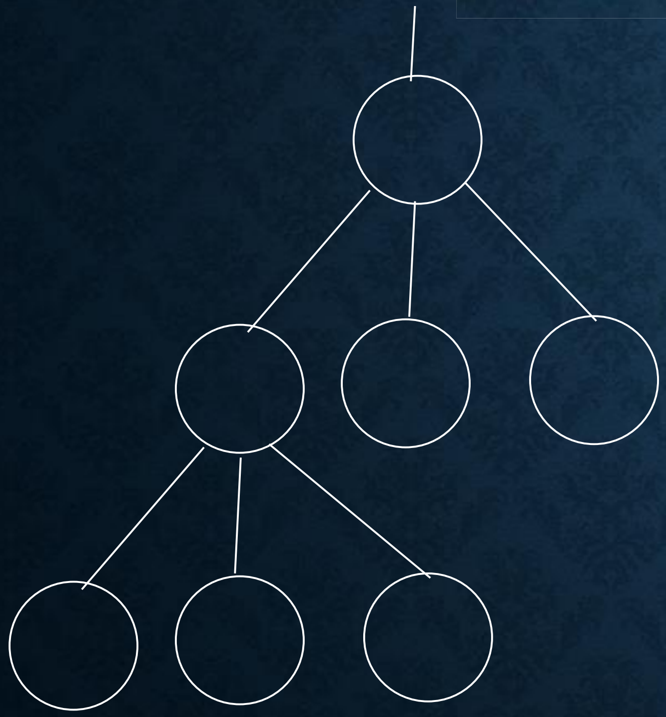
- Directed acyclic graph.
- All edges from level i to $i+1$.
- Estimate number of edges.

OUR RESULT



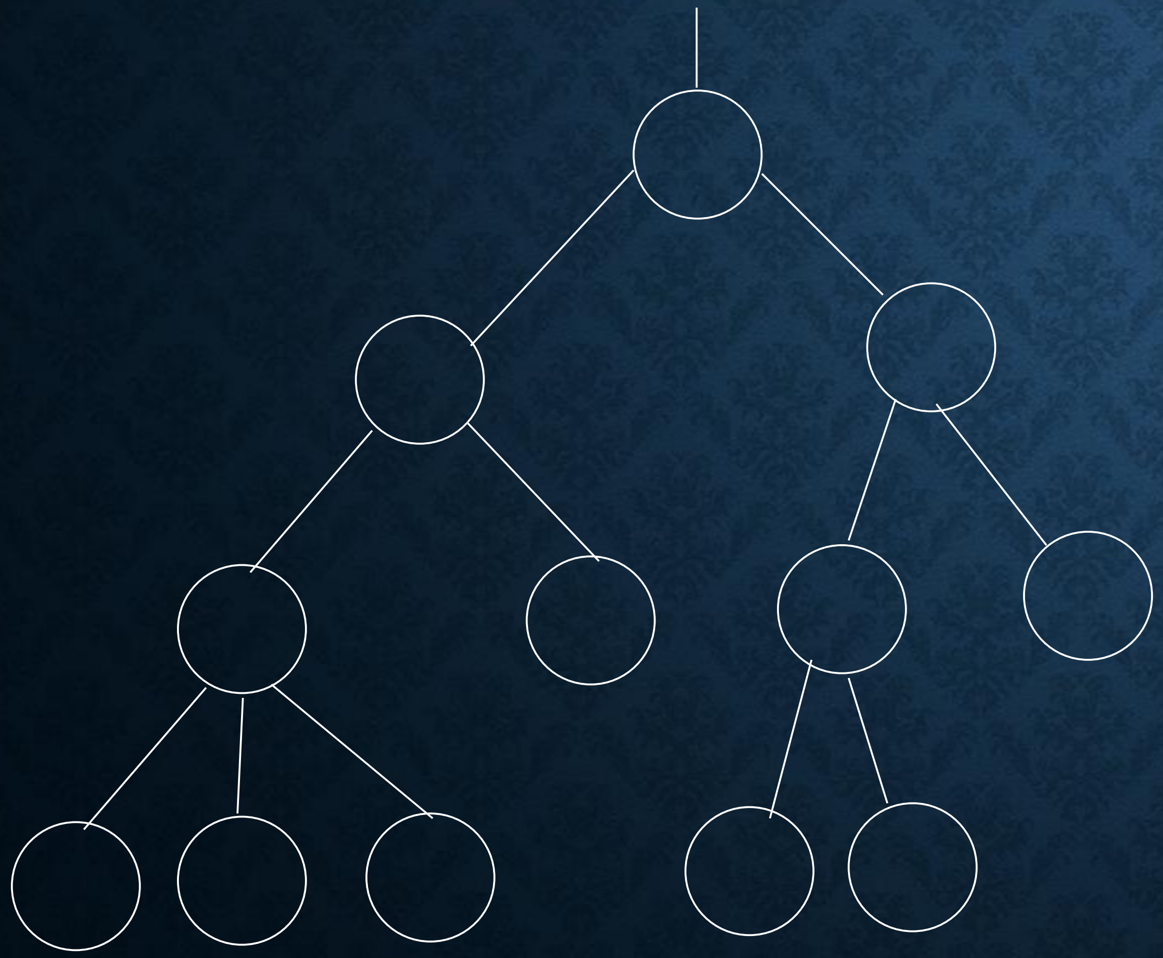
- Can estimate number of edges T within a factor of 1±ε.
- Time: $O\left(\sqrt{Tn}\right)$.

OUR QUANTUM ALGORITHM



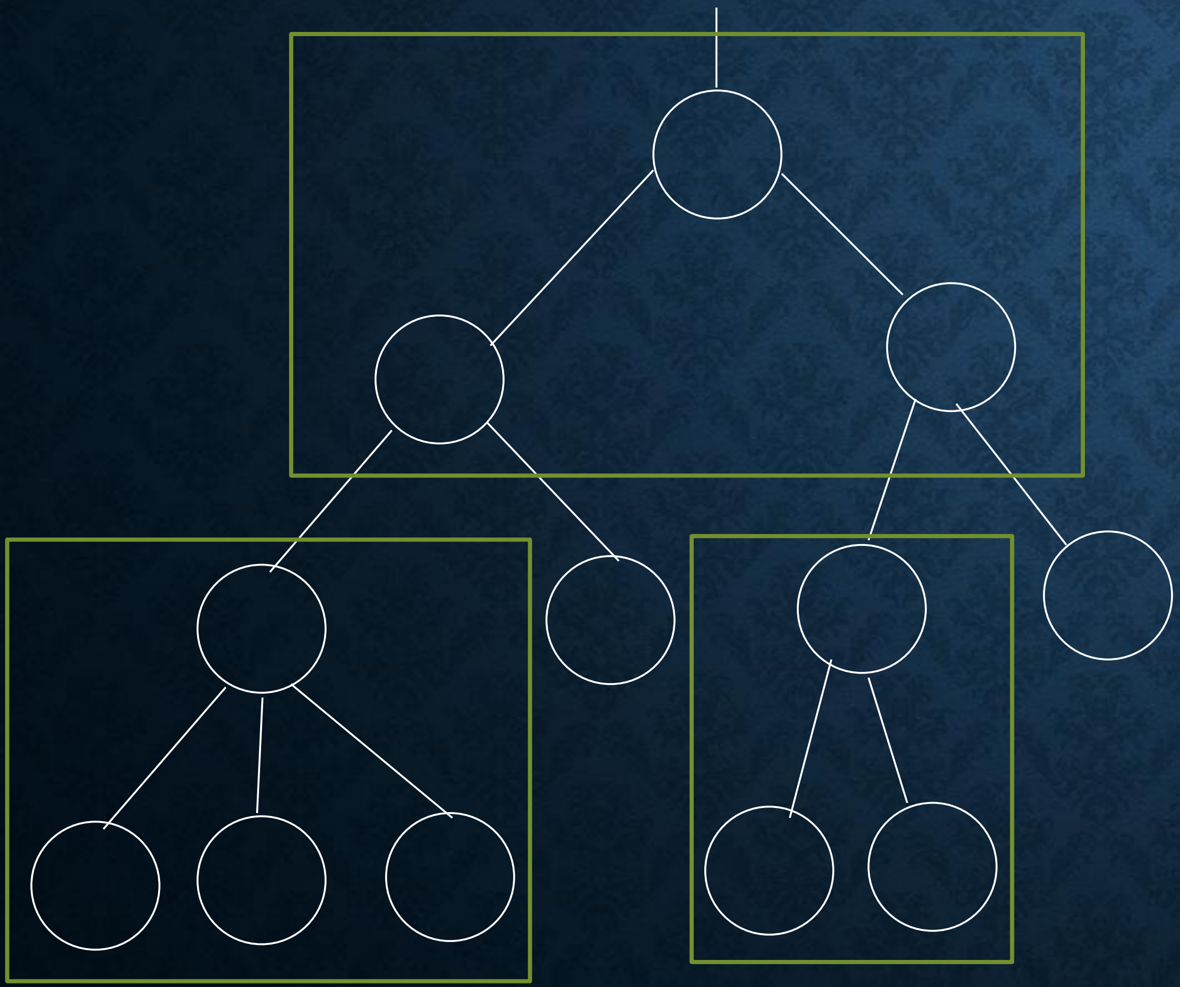
- Quantum walk on the tree/DAG.
- Eigenvalues closest to 1: $e^{\pm i\theta}$,
  $\theta \in \left[(1-\varepsilon)\frac{c}{\sqrt{Tn}}, (1+\varepsilon)\frac{c}{\sqrt{Tn}}\right]$.
- Eigenvalue estimation.

QUANTUM WALK (MONTANARO, 2015)



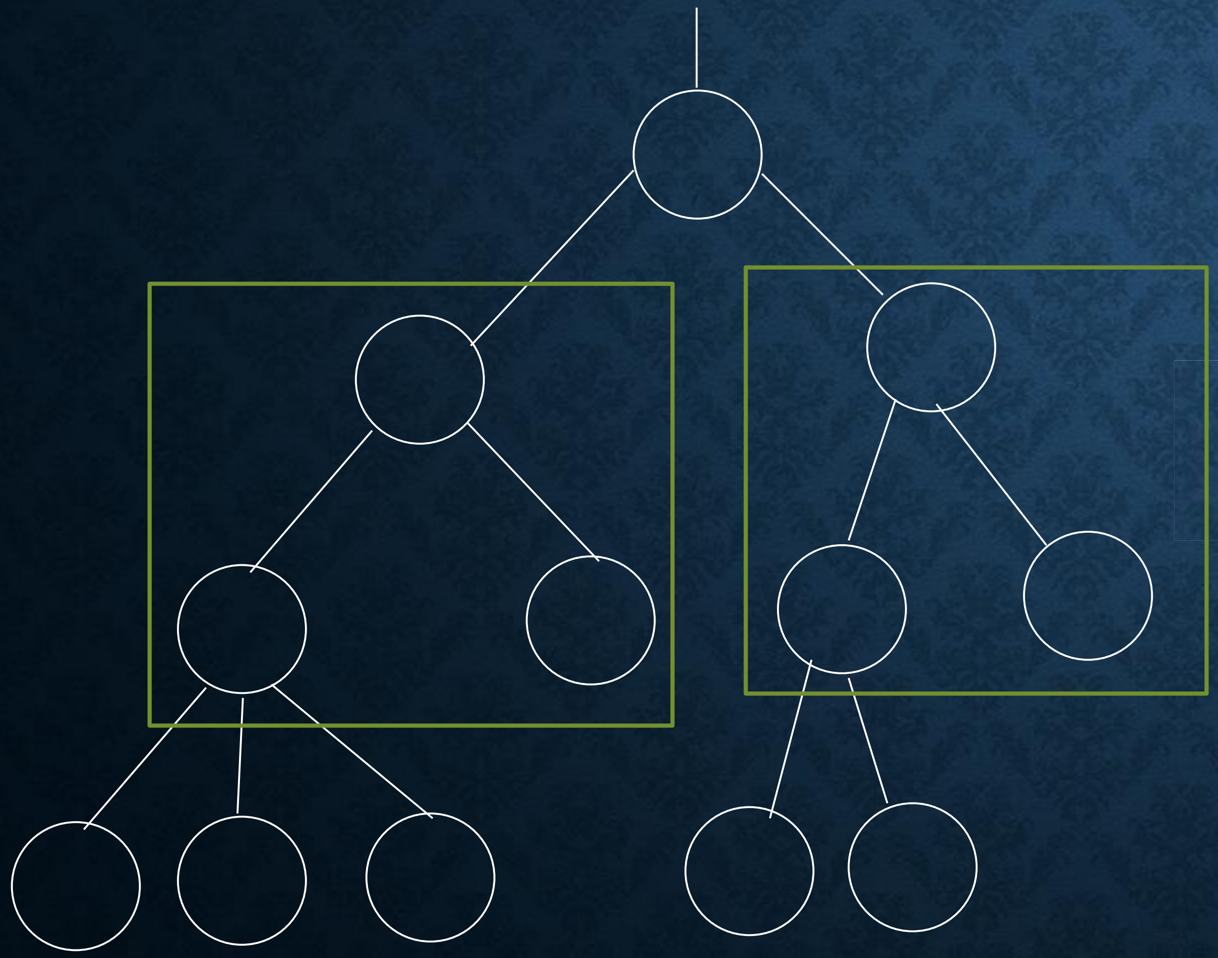
- Basis states: $|u\rangle$.
- Different transformations at odd, even steps.

ODD STEPS



- S_v : odd-level vertex v with all its children.
- Transformation C_v on $|u\rangle$, $u \in S_v$.

EVEN STEPS



- S_v : even-level vertex v with all its children.
- Transformation C_v on $|u\rangle$, $u \in S_v$.

ANALYSIS

- Reduce quantum walk to a classical random walk.
- Matrix of quantum walk $\rightarrow$ Fundamental matrix of classical walk.
- Bound matrix entries using electric resistances.
- Result: exact expression for elements of the matrix.

SUMMARY

- Quantum algorithm for estimating size of a tree/DAG.
- Applications:
  - Backtracking;
  - Game trees;

OPEN QUESTIONS (GAME TREES)

Our algorithm: trees of size T, depth $T^{o(1)}$ in time $O(T^{1/2+o(1)})$.

1. Algorithm for trees of larger depth?
2. Algorithm with small memory?

OPEN QUESTIONS (GENERAL)

1. Is time $O\left(\sqrt{Tn}\right)$ for tree size estimation optimal?
2. Applications for evaluating size of DAGs?
3. Other algorithms for «estimating size of ...»?