Classical Simulation of Quantum Systems via Tensor Networks

Robert Špalek

UC Berkeley

Quantum simulation

- quantum systems have complex behavior
- want to *simulate* them, i.e. compute the outcome classically without actually building the system
- hard in the worst case, but there are systems for which this is feasible

Tensors

- are multi-linear operators
- generalize vectors and matrices
- *dimension* is the number of indices
  rank of an index denotes its domain

Tensors

- are multi-linear operators
- generalize vectors and matrices
- *dimension* is the number of indices
  rank of an index denotes its domain
- drawn as a creature with a number of legs

Tensor networks

- *tensor network* is a collection of possibly connected tensors

Tensor networks

- *tensor network* is a collection of possibly connected tensors
- connecting two legs = contracting a common index *i*

$$R^{a,b,c}_{x,y,z} = \sum_i P_{a,b,c,i} Q_{x,y,z,i}$$

  - requires equal rank
  - generalizes matrix multiplication
  - can contract more than 1 leg at the same time

Quantum states as tensors

- don’t think of them as vectors (with 1 leg of rank 2^n)

Quantum states as tensors

- don't think of them as vectors (with 1 leg of rank 2^n)
- instead, an n-qubit state will be a tensor with n legs of rank 2, i.e. it is specified by 2^n complex coefficients
  - an arbitrary quantum state is one fat spider with many legs
  - product states can be drawn as a group of skinnier creatures
    $\Longrightarrow$ fewer coefficients are needed!

Quantum states as tensors

- don't think of them as vectors (with 1 leg of rank 2^n)
- instead, an n-qubit state will be a tensor with n legs of rank 2, i.e. it is specified by 2^n complex coefficients
  - an arbitrary quantum state is one fat spider with many legs
  - product states can be drawn as a group of skinnier creatures
    $\Longrightarrow$ fewer coefficients are needed!
- is there anything between?
  - for larger family of states
  - stil efficient

Schmidt decomposition

- every bipartite quantum state can be written as

$$|\phi\rangle = \sum_{i=1}^{r} |\psi_{A,i}\rangle|\psi_{B,i}\rangle,$$

where r is the *Schmidt rank* of the bipartition
the states $|\psi_{B,i}\rangle$ need not be normalized

Schmidt decomposition

- every bipartite quantum state can be written as

$$|\phi\rangle = \sum_{i=1}^{r} |\psi_{A,i}\rangle |\psi_{B,i}\rangle,$$

  where r is the *Schmidt rank* of the bipartition
  the states $|\psi_{B,i}\rangle$ need not be normalized
- hence we can slash any creature into two smaller ones connected by just one leg
  - notice that these legs may be longer

Quantum states as tensor networks

- [Vidal] we can split tensors iterativaly until we end up, say, with a 3-regular tree with leaves corresponding to the original qubits and a couple of added internal vertices containing the intermediate Schmidt coefficients

Quantum states as tensor networks

- [Vidal] we can split tensors iterativaly until we end up, say, with a 3-regular tree with leaves corresponding to the original qubits and a couple of added internal vertices containing the intermediate Schmidt coefficients
- can this description possibly be efficient?
  - yes as long as the Schmidt ranks are not too high
  - then we need at most $n \cdot R^3$ coefficients, where $R = \max_e r_e$ is the maximal rank
  - not every possible tensor networks yields efficient ranks!

Quantum states as tensor networks

- [Vidal] we can split tensors iterativaly until we end up, say, with a 3-regular tree with leaves corresponding to the original qubits and a couple of added internal vertices containing the intermediate Schmidt coefficients
- can this description possibly be efficient?
  - yes as long as the Schmidt ranks are not too high
  - then we need at most $n \cdot R^3$ coefficients, where $R = \max_e r_e$ is the maximal rank
  - not every possible tensor networks yields efficient ranks!
- can apply unitaries and measurements fast on states with efficient networks
  - the tree structure is not altered much
  - hence we can simulate computation as long as all intermediate states are efficient

Effi cient tensor networks

- can we connect the n qubits by a 3-regular graph such that the rank of the worst bipartition is not too high?
- that is, optimize *Schmidt-rank width* defined as

$$\mathrm{rwd}(|\psi\rangle) = \log \min_{\text{tree } T} \max_{\text{edge } e \in T} \chi_{A_T^e, B_T^e}(|\psi\rangle),$$

where $\chi_{A_T^e, B_T^e}(|\psi\rangle)$ is the number of nonzero Schmidt coefficients of $|\psi\rangle$ corresponding to the bipartition induced by removing e from T

Efficient tensor networks

- can we connect the n qubits by a 3-regular graph such that the rank of the worst bipartition is not too high?
- that is, optimize *Schmidt-rank width* defined as

$$\mathrm{rwd}(|\psi\rangle) = \log \min_{\text{tree } T} \max_{\text{edge } e \in T} \chi_{A_T^e, B_T^e}(|\psi\rangle),$$

  where $\chi_{A_T^e, B_T^e}(|\psi\rangle)$ is the number of nonzero Schmidt coefficients of $|\psi\rangle$ corresponding to the bipartition induced by removing e from T
- [S.-I. Oum, PhD thesis] polynomial time constant approximation algorithm for the width of every sub-modal function χ (which is our case)
  - it is polynomial assuming that $\chi_{A_T^e, B_T^e}$ is an oracle whose computation takes constant time

Evaluating the Schmidt rank χ

- we cannot evaluate χ fast for an arbitrary state $|\psi\rangle$, because already the description of $|\psi\rangle$ is exponentially large

Evaluating the Schmidt rank χ

- we cannot evaluate χ fast for an arbitrary state $|\psi\rangle$, because already the description of $|\psi\rangle$ is exponentially large
- need an efficient description of $|\psi\rangle$ as an input
  - for example, $|\psi\rangle$ may be computed by a small quantum circuit
    this is hopeless, as it would solve factoring

Evaluating the Schmidt rank χ

- we cannot evaluate χ fast for an arbitrary state $|\psi\rangle$, because already the description of $|\psi\rangle$ is exponentially large
- need an efficient description of $|\psi\rangle$ as an input
  - for example, $|\psi\rangle$ may be computed by a small quantum circuit
    this is hopeless, as it would solve factoring
  - works when $|\psi\rangle$ is a cluster state, because then the Schmidt rank of a bipartition equals the $\mathbb{GF}(2)$ rank of the adjacency matrix of this bipartition

Cluster states

- *cluster state* corresponding to a graph $G = (V, E)$ is the (unique) state stabilized by

$$X^v \prod_{(v,w)\in E} Z^w$$

for every v

Cluster states

- *cluster state* corresponding to a graph $G = (V, E)$ is the (unique) state stabilized by

$$X^v \prod_{(v,w) \in E} Z^w$$

  for every v

- equivalently, start in the state $|+\rangle^{\otimes |V|}$ and apply CPHASE on every edge

Cluster states

- *cluster state* corresponding to a graph $G = (V, E)$ is the (unique) state stabilized by

$$X^v \prod_{(v,w)\in E} Z^w$$

  for every v
- equivalently, start in the state $|+\rangle^{\otimes |V|}$ and apply CPHASE on every edge
- [Raussendorf & Briegel] *one-way quantum computer*
  - start in a highly entangled cluster state
  - perform a sequence of adaptive one-qubit measurements
  - universal for quantum computation

How does cluster state computation work?

- if we have a chain (cluster state corresponding to a path), then left-to-right one-qubit measurements in a certain basis *teleport* quantum information to the right and one can also perform some unitaries along the way

How does cluster state computation work?

- if we have a chain (cluster state corresponding to a path), then left-to-right one-qubit measurements in a certain basis *teleport* quantum information to the right and one can also perform some unitaries along the way
- CPHASE gates can also be applied by incorporating them into the underlying cluster state

How does cluster state computation work?

- if we have a chain (cluster state corresponding to a path), then left-to-right one-qubit measurements in a certain basis *teleport* quantum information to the right and one can also perform some unitaries along the way
- CPHASE gates can also be applied by incorporating them into the underlying cluster state
- this set of gates is universal $\Longrightarrow$ every quantum circuit can be efficiently rewritten into this form
  - the cluster state basically resembles the shape of the circuit

Efficiency of quantum simulation

- [Markov & Shi] simulation in time $2^{\mathrm{twd}(G)}$, where twd is the *tree-width* of G

Efficiency of quantum simulation

- [Markov & Shi] simulation in time $2^{\mathrm{twd}(G)}$, where twd is the *tree-width* of G
- [Nest, Dür, Vidal & Briegel] simulation in time $2^{\mathrm{rwd}(G)}$
  - this is faster, because $\mathrm{rwd}(G) \leq 4 \cdot \mathrm{twd}(G) + 2$
  - on the other hand, there are graphs with constant rank width and tree width n

Efficiency of quantum simulation

- [Markov & Shi] simulation in time $2^{\mathrm{twd}(G)}$, where twd is the *tree-width* of G
- [Nest, Dür, Vidal & Briegel] simulation in time $2^{\mathrm{rwd}(G)}$
  - this is faster, because $\mathrm{rwd}(G) \leq 4 \cdot \mathrm{twd}(G) + 2$
  - on the other hand, there are graphs with constant rank width and tree width n
- this result also subsumes other similar results based on structural properties of the quantum circuit [Jozsa]

Efficiency of quantum simulation

- [Markov & Shi] simulation in time $2^{\mathrm{twd}(G)}$, where twd is the *tree-width* of G
- [Nest, Dür, Vidal & Briegel] simulation in time $2^{\mathrm{rwd}(G)}$
  - this is faster, because $\mathrm{rwd}(G) \leq 4 \cdot \mathrm{twd}(G) + 2$
  - on the other hand, there are graphs with constant rank width and tree width n
- this result also subsumes other similar results based on structural properties of the quantum circuit [Jozsa]
- when applied to factoring, the complexity lies in the modular exponentiation and the approximate QFT is easy

Summary

1. tensor networks
2. representation of quantum states
3. can find quickly the most efficient tensor network
   polynomial algorithm for representing cluster states
4. simulating general quantum circuits on cluster states
5. polynomial time simulation of quantum computation when
   the Schmidt-rank width is at most logarithmic