Parallelizing Quantum Circuits

by Anne Broadbent
Université de Montréal
joint work with Elham Kashefi

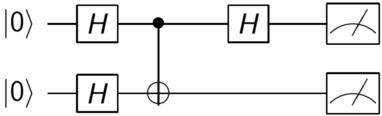
March 20 2007

Parallelising Quantum Circuits

by Anne Broadbent
Université de Montréal
joint work with Elham Kashefi

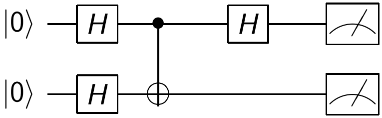
March 20 2007

Quantum Circuits



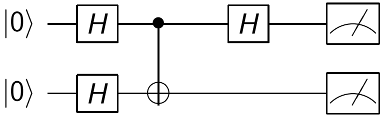
- *Size* of circuit: number of gates.

Quantum Circuits



- *Size* of circuit: number of gates.
- *Depth* of circuit: length of longest input-output path, with 2-qubit gates acting as "bridges" between wires.

Quantum Circuits



- *Size* of circuit: number of gates.
- *Depth* of circuit: length of longest input-output path, with 2-qubit gates acting as "bridges" between wires.
- (equivalently) the minimum number of layers required for the execution of the circuit, where each qubit interacts at most once in each layer.

Importance of the study of depth complexity

- The depth of a circuit thus gives the number of steps required for the parallel execution of the circuit.

Importance of the study of depth complexity

- The depth of a circuit thus gives the number of steps required for the parallel execution of the circuit.
- In computer science, one way to define classically *efficiently computable* problems is the class P which contains all problems computable in polynomial time. Another way to define *efficiently computable* is the class NC, which contains all classical polylogarithmic ($(\log n)^c$) depth circuits.

Importance of the study of depth complexity

- The depth of a circuit thus gives the number of steps required for the parallel execution of the circuit.
- In computer science, one way to define classically *efficiently computable* problems is the class P which contains all problems computable in polynomial time. Another way to define *efficiently computable* is the class NC, which contains all classical polylogarithmic ($(\log n)^c$) depth circuits.
- The power of parallel computers is thus characterized by NC, while P characterizes the power of sequential computers.

Importance of the study of depth complexity

- The depth of a circuit thus gives the number of steps required for the parallel execution of the circuit.
- In computer science, one way to define classically *efficiently computable* problems is the class P which contains all problems computable in polynomial time. Another way to define *efficiently computable* is the class NC, which contains all classical polylogarithmic ($(\log n)^c$) depth circuits.
- The power of parallel computers is thus characterized by NC, while P characterizes the power of sequential computers.
- NC $\subset$ P, but it is a long standing open question to determine whether or not P $=$ NC.

Importance of the study of depth complexity

- The depth of a circuit thus gives the number of steps required for the parallel execution of the circuit.
- In computer science, one way to define classically *efficiently computable* problems is the class P which contains all problems computable in polynomial time. Another way to define *efficiently computable* is the class NC, which contains all classical polylogarithmic ($(\log n)^c$) depth circuits.
- The power of parallel computers is thus characterized by NC, while P characterizes the power of sequential computers.
- NC $\subset$ P, but it is a long standing open question to determine whether or not P $=$ NC.
- We can define QNC and also ask whether or not P $\subset$ QNC or ultimately, BQP $=$ QNC.

Importance of the study of depth complexity

- For physical implementations, the number of steps required in the computation tells us how long the qubit must remain in a coherent state in order for the computation to have a good chance of success.

lower-depth $\Rightarrow$ more computing power

Importance of the study of depth complexity

- For physical implementations, the number of steps required in the computation tells us how long the qubit must remain in a coherent state in order for the computation to have a good chance of success.

$$\text{lower-depth} \Rightarrow \text{more computing power}$$

- Previous results obtained in the MBQC:
  - Clifford gates can be implemented in constant depth; so can Control-X and phase gates (R. Raussendorf, and H. J. Briegel 00)
  - Diagonal unitaries can be implemented in constant depth (D. E. Browne and H. J. Briegel 06)
  - Logarithmic classical depth is required for final corrections (Josza 05).

Result 1

All the models for QC are equivalent in computational power

Theorem

There exists a logarithmic separation in depth complexity between MBQC and the circuit model.

Result 2

An algorithm for parallelizing quantum circuits

1. Convert the circuit C to a pattern P.
2. Apply the tools of the measurement calculus to obtain a lower-depth pattern P'.
3. Convert P' back to a circuit, keeping all the auxiliary qubits.

Theorem

The above algorithm can decrease the depth of the circuit.

Result 3

Theorem

A pattern with flow $\mathcal{P}$ has depth $d+2$ if and only if on any influencing path we obtain $P^* N^{i \leq d} P^*$ $(\{Y, \emptyset\} N^{i \leq d} P^*)$, after applying the following rewriting rule:

$$N\, P_1^*\, \alpha_1 \beta_1\, P_2^*\, \alpha_2 \beta_2\, \cdots\, P_k^*\, N \begin{cases} NN & \textit{if} \quad \forall P_i^* \neq X(XY)^* \\ N & \textit{otherwise} \end{cases}$$

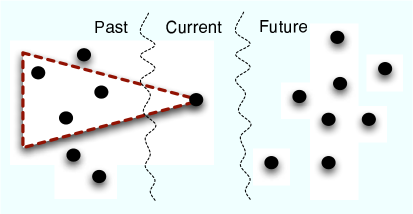
By understanding where the depth comes from, we can better understand how to design lower depth circuits and patterns.

Tools that we use

- measurement calculus
- standardization
- signal shifting
- flow
- influencing paths
- Pauli measurements

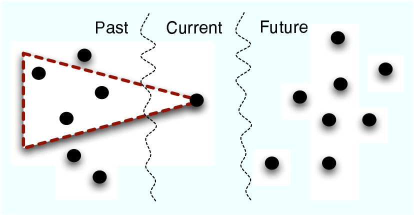
Depth

- Measurements and corrections depend on previous measurements; this is what creates depth in the measurement based model.



Depth

- Measurements and corrections depend on previous measurements; this is what creates depth in the measurement based model.



- Given qubit b, we give a method to list the qubits that b depends on. This is the "backward cone" of H. Briegel and R. Raussendorf.

Measurement Calculus

- V. Danos, E. Kashefi and P. Panangaden (2005) give us an algebraic formalism to reason about measurement patterns.

$$Z_k^s X_j^s \cdots {}_t[M_i^\alpha]^s \cdots E_G$$

Measurement Calculus

- V. Danos, E. Kashefi and P. Panangaden (2005) give us an algebraic formalism to reason about measurement patterns.

$$Z_k^s X_j^s \cdots {}_t[M_i^\alpha]^s \cdots E_G$$

- *Dependencies*: $s = \sum_{i \in I} s_i$ is a *signal*; measurements and corrections can depend on signals. ${}_t[M_i^\alpha]^s$, means $M_i^{(-1)^s \alpha + t\pi}$.

Standardization

- To obtain parallel computation, we commute all the entanglement commands to the beginning.

Standardization

- To obtain parallel computation, we commute all the entanglement commands to the beginning.
- The *standardization* rules do exactly this:

$$E_{ij} X_i^s \Rightarrow X_i^s Z_j^s E_{ij}$$
$$E_{ij} Z_i^s \Rightarrow Z_i^s E_{ij}$$
$${}_t[M_i^\alpha]^s X_i^r \Rightarrow {}_t[M_i^\alpha]^{s+r}$$
$${}_t[M_i^\alpha] Z_i^r \Rightarrow_{r+t} [M_i^\alpha]^s$$

Standardization

- To obtain parallel computation, we commute all the entanglement commands to the beginning.
- The *standardization* rules do exactly this:

$$
\begin{aligned}
E_{ij} X_i^s &\Rightarrow X_i^s Z_j^s E_{ij} \\
E_{ij} Z_i^s &\Rightarrow Z_i^s E_{ij} \\
{}_t[M_i^\alpha]^s X_i^r &\Rightarrow {}_t[M_i^\alpha]^{s+r} \\
{}_t[M_i^\alpha] Z_i^r &\Rightarrow_{r+t} [M_i^\alpha]^s
\end{aligned}
$$

- Every pattern has a unique standard form.

Standardization

- To obtain parallel computation, we commute all the entanglement commands to the beginning.
- The *standardization* rules do exactly this:

$$
\begin{aligned}
E_{ij} X_i^s &\Rightarrow X_i^s Z_j^s E_{ij} \\
E_{ij} Z_i^s &\Rightarrow Z_i^s E_{ij} \\
{}_t[M_i^\alpha]^s X_i^r &\Rightarrow {}_t[M_i^\alpha]^{s+r} \\
{}_t[M_i^\alpha] Z_i^r &\Rightarrow_{r+t} [M_i^\alpha]^s
\end{aligned}
$$

- Every pattern has a unique standard form.
- Could also be done using the biproduct method.

Standardization

Theorem

Standardization does not increase the depth of the pattern.

Signal Shifting

- A set of rules that say how to propagate the Z-dependency to the end.

Signal Shifting

- A set of rules that say how to propagate the Z-dependency to the end.
- If ${}_t[M_i^\alpha]^s$ appears in a pattern, replace it with $[M_i^\alpha]^s$.

Signal Shifting

- A set of rules that say how to propagate the Z-dependency to the end.
- If ${}_t[M_i^\alpha]^s$ appears in a pattern, replace it with $[M_i^\alpha]^s$.
- But for all future commands M, X or Z that have a dependency on s_i, we add dependency on t.

Signal Shifting

- A set of rules that say how to propagate the Z-dependency to the end.
- If ${}_t[M_i^\alpha]^s$ appears in a pattern, replace it with $[M_i^\alpha]^s$.
- But for all future commands M, X or Z that have a dependency on s_i, we add dependency on t.
- The formal rules

$$
\begin{aligned}
{}_t[M_i^\alpha]^s &\Rightarrow S_i^t \; [M_i^\alpha]^s \\
X_j^s S_i^t &\Rightarrow S_i^t \; X_j^{s[(t+s_i)/s_i]} \\
Z_j^s S_i^t &\Rightarrow S_i^t \; Z_j^{s[(t+s_i)/s_i]} \\
{}_t[M_j^\alpha]^s S_i^r &\Rightarrow S_i^r \; {}_{t[(r+s_i)/s_i]}[M_j^\alpha]^{s[(r+s_i)/s_i]}
\end{aligned}
$$

Signal shifting

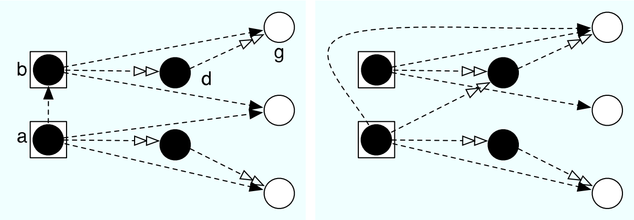
Theorem

Signal shifting never increases the computation depth.

Signal shifting

Theorem

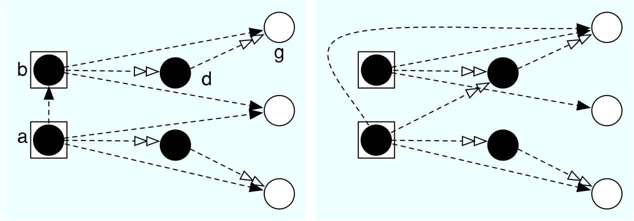
Signal shifting never increases the computation depth.



Signal shifting

Theorem

Signal shifting never increases the computation depth.



- Note: classical depth can increase

Classical depth

$$X_i^{s_1+s_2+\ldots+s_p}$$

The classical computation of the parity of the dependencies has depth in $\Theta(\log p)$.

Classical depth

$$X_i^{s_1+s_2+\ldots+s_p}$$

The classical computation of the parity of the dependencies has depth in $\Theta(\log p)$.

$${}_{s_1+s_2+\ldots+s_p}[M_i^{\alpha}]^{s_1+s_2+\ldots+s_q}$$

depth in $\Theta(\log(\max\{p,q\}))$.

Classical depth

$$X_i^{s_1+s_2+\ldots+s_p}$$

The classical computation of the parity of the dependencies has depth in $\Theta(\log p)$.

$${}_{s_1+s_2+\ldots+s_p}[M_i^{\alpha}]^{s_1+s_2+\ldots+s_q}$$

depth in $\Theta(\log(\max\{p, q\}))$.

- It's reasonable to consider that classical computation is cheap compared to quantum computation.

Flow

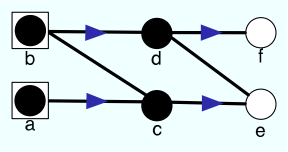
- Deterministic pattern: no matter what the measurements yield, the quantum output is always the same.

Flow

- Deterministic pattern: no matter what the measurements yield, the quantum output is always the same.
- The *flow* gives a recipe to construct the dependency structure.

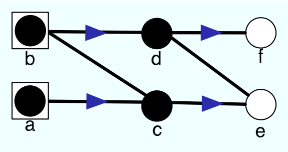
Flow

- Deterministic pattern: no matter what the measurements yield, the quantum output is always the same.
- The *flow* gives a recipe to construct the dependency structure.



Flow

- Deterministic pattern: no matter what the measurements yield, the quantum output is always the same.
- The *flow* gives a recipe to construct the dependency structure.

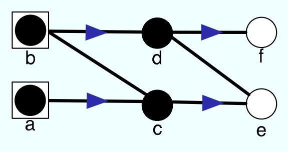


- From now on, we consider only patterns with flow (a sufficient condition for determinism, but not a necessary one!).

Influencing paths

We want to find the longest feed-forward path, which gives the depth. It's enough to consider only very specific paths in the graph:

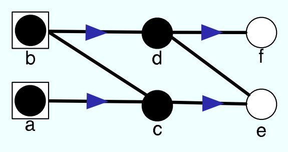
- An influencing path: any input-output path in G that starts with a flow edge and has no two consecutive non-flow edges.



Influencing paths

We want to find the longest feed-forward path, which gives the depth. It's enough to consider only very specific paths in the graph:

- An influencing path: any input-output path in G that starts with a flow edge and has no two consecutive non-flow edges.



Theorem

If qubit b has a dependency on qubit a then a and b are on the same influencing path.

Pauli measurements and dependencies

- All Pauli measurements can be done in the first layer:

$$[M^X]^s = M^X$$
$$[M^Y]^s =_s [M^Y]$$

Pauli measurements and dependencies

- All Pauli measurements can be done in the first layer:

$$[M^X]^s = M^X$$
$$[M^Y]^s =_s [M^Y]$$

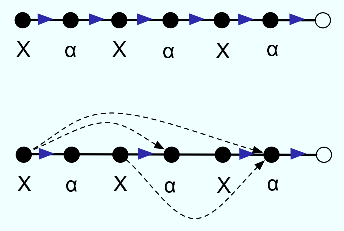
- Pauli measurements also act as a "reset" for the depth.

Example 1

$$\ldots M_6^{\alpha} M_5^{X} M_4^{\alpha} M_3^{X} M_2^{\alpha} M_1^{X}$$

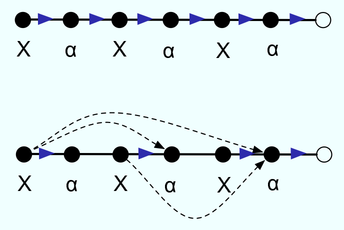
Example 1

$$\dots M_6^{\alpha} M_5^{X} M_4^{\alpha} M_3^{X} M_2^{\alpha} M_1^{X}$$



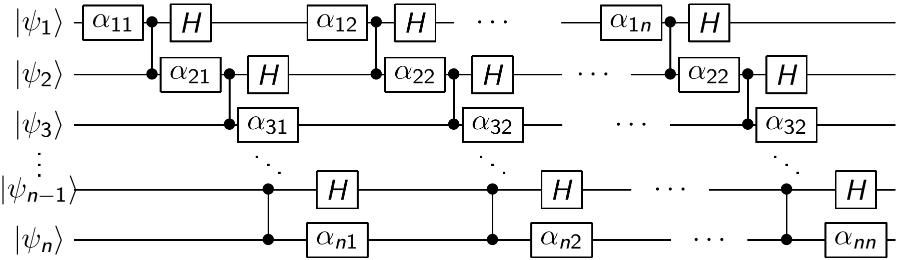
Example 1

$$\ldots M_6^{\alpha} M_5^{X} M_4^{\alpha} M_3^{X} M_2^{\alpha} M_1^{X}$$



Depth = 2

Example 2



- This polynomial-depth circuit can be implemented in depth 2 in the measurement-based model.

Depth characterisation

Theorem

A pattern with flow $\mathcal{P}$ has depth $d+2$ if and only if on any influencing path we obtain $P^* N^{i \leq d} P^*$ ($\{Y, \emptyset\} N^{i \leq d} P^*$), after applying the following rewriting rule:

$$N\, P_1^*\, \alpha_1 \beta_1\, P_2^*\, \alpha_2 \beta_2\, \cdots\, P_k^*\, N \begin{cases} NN & \text{if} \quad \forall P_i^* \neq X(XY)^* \\ N & \text{otherwise} \end{cases}$$

Separation Result

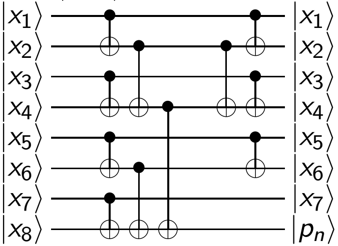
- For $x_1, x_2, \ldots, x_n \in \{0,1\}^n$, let $p_n = \bigoplus_{i=1}^{n} x_i$.

Separation Result

- For $x_1, x_2, \ldots, x_n \in \{0,1\}^n$, let $p_n = \bigoplus_{i=1}^{n} x_i$.
- Let parity $|x_1, x_2, \ldots, x_n\rangle = |x_1, x_2, \ldots, x_{n-1}, p_n\rangle$

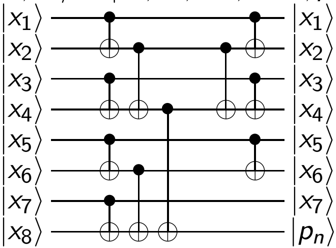
Separation Result

- For $x_1, x_2, \ldots, x_n \in \{0,1\}^n$, let $p_n = \bigoplus_{i=1}^{n} x_i$.
- Let parity $|x_1, x_2, \ldots, x_n\rangle = |x_1, x_2, \ldots, x_{n-1}, p_n\rangle$



Separation Result

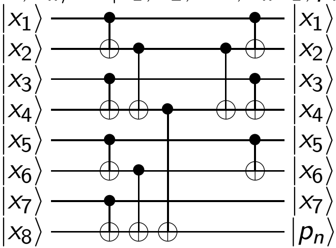
- For $x_1, x_2, \ldots, x_n \in \{0,1\}^n$, let $p_n = \bigoplus_{i=1}^{n} x_i$.
- Let parity $|x_1, x_2, \ldots, x_n\rangle = |x_1, x_2, \ldots, x_{n-1}, p_n\rangle$



- Lemma: Let C be any circuit (with one- and two-qubit gates) that computes parity. Then the depth of C is in $\Omega(\log n)$.

Separation Result

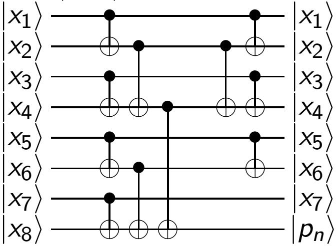
- For $x_1, x_2, \ldots, x_n \in \{0,1\}^n$, let $p_n = \bigoplus_{i=1}^{n} x_i$.
- Let parity $|x_1, x_2, \ldots, x_n\rangle = |x_1, x_2, \ldots, x_{n-1}, p_n\rangle$



- Lemma: Let C be any circuit (with one- and two-qubit gates) that computes parity. Then the depth of C is in $\Omega(\log n)$.
- Proof: The state of the last output qubit depends on every input qubit; this requires depth $\log n$.

Separation Result

- For $x_1, x_2, \ldots, x_n \in \{0,1\}^n$, let $p_n = \bigoplus_{i=1}^n x_i$.
- Let parity $|x_1, x_2, \ldots, x_n\rangle = |x_1, x_2, \ldots, x_{n-1}, p_n\rangle$



- Lemma: Let C be any circuit (with one- and two-qubit gates) that computes parity. Then the depth of C is in $\Omega(\log n)$.
- Proof: The state of the last output qubit depends on every input qubit; this requires depth $\log n$.
- But parity is in the Clifford group, so it has quantum depth 1 and classical depth in $O(\log n)$.

Conclusion

- We've shown how the tools of MBQC are useful in the design of parallel circuits.

Conclusion

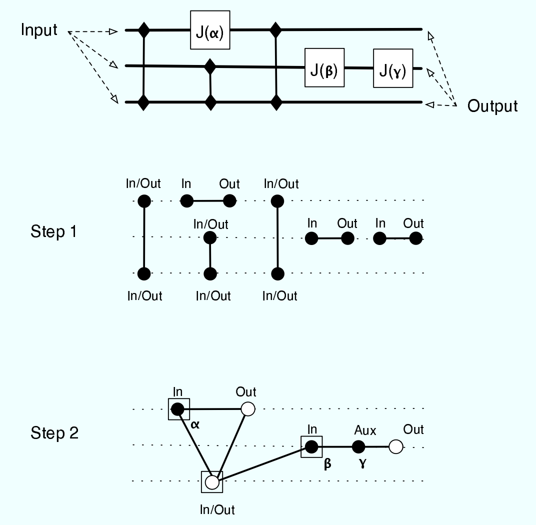
- We've shown how the tools of MBQC are useful in the design of parallel circuits.
- We've characterized classes of patterns of depth d.

Conclusion

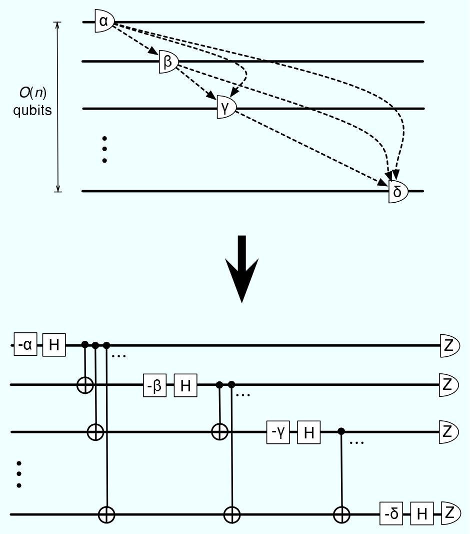
- We've shown how the tools of MBQC are useful in the design of parallel circuits.
- We've characterized classes of patterns of depth d.
- The MBQC gives a clear separation between quantum and classical depth, as illustrated by the parity example.

Thank you!

From circuits to patterns

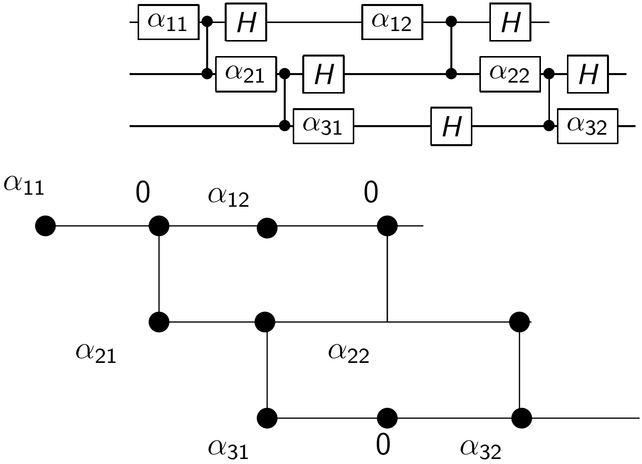


From patterns to circuits



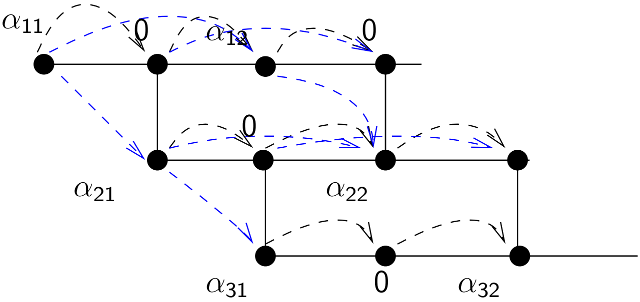
Example 2

- Convert circuit to pattern:



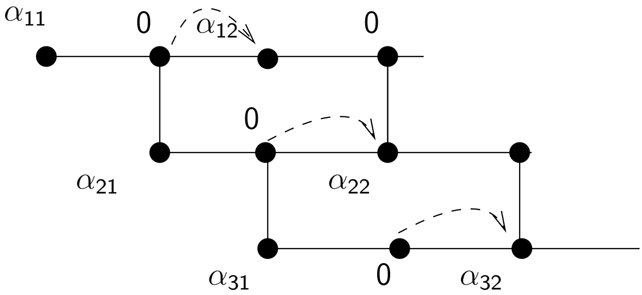
Example 2

- with dependencies



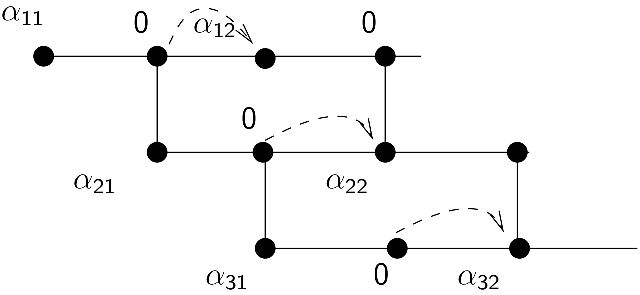
Example 2

- After pauli measurements and signal shifting



Example 2

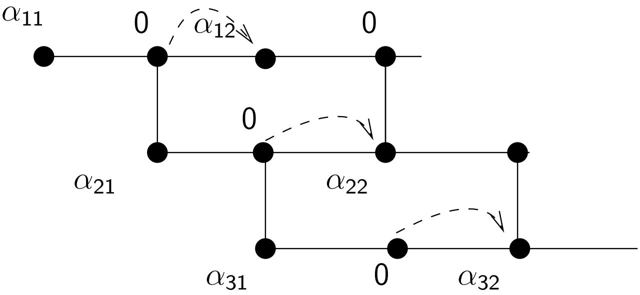
- After pauli measurements and signal shifting



- The 0 measurement have no dependencies and can all be performed in the first layer. The second layer contains all the α measurements.

Example 2

- After pauli measurements and signal shifting



- The 0 measurement have no dependencies and can all be performed in the first layer. The second layer contains all the α measurements.
- Classical depth is $O(\log n)$.