Some Recent Research Issues in Quantum Logic

Marek Perkowski

Part one

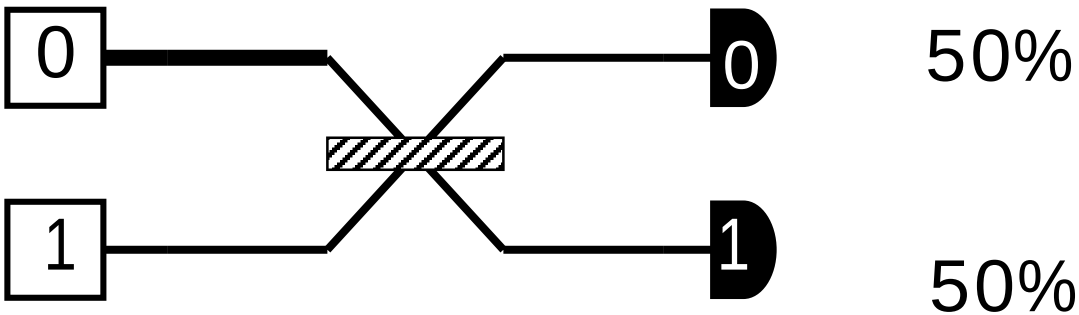
What will be discussed?

- 1. Background
- 2. Quantum circuits synthesis
- 3. Quantum circuits simulation
- 4. Quantum logic emulation and evolvable hardware
- 5. Quantum circuits verification
- 6. Quantum-based robot control

Quantum Logic Circuits

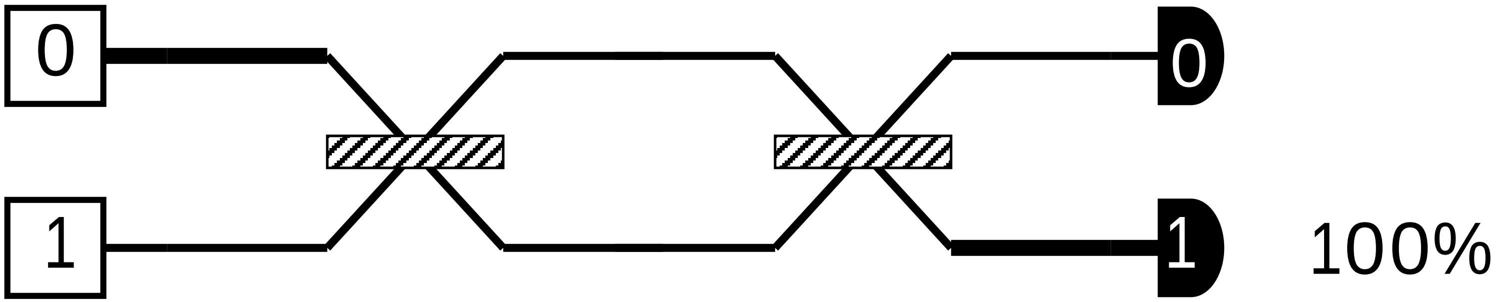
Origin of slides: John Hayes, Peter Shor, Martin Lukac, Mikhail Pivtoraiko, Alan Mishchenko, Pawel Kerntopf.

A beam-splitter



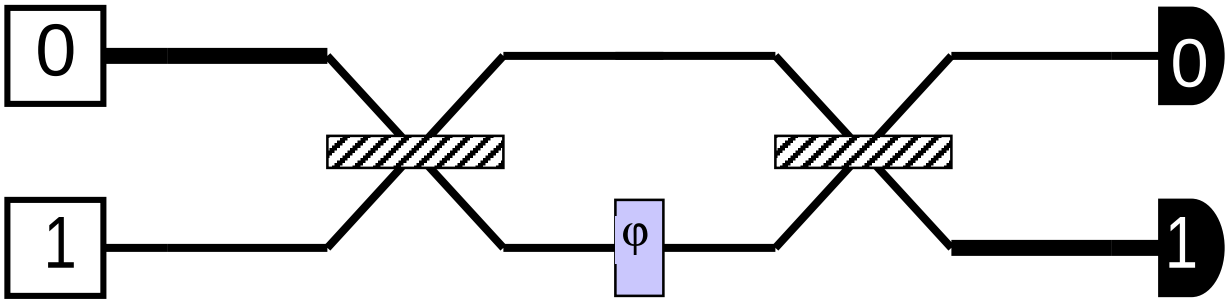
The simplest explanation is that the beam-splitter acts as a classical coin-flip, randomly sending each photon one way or the other.

Quantum Interference



The simplest explanation must be wrong, since it would predict a 50-50 distribution.

More experimental data

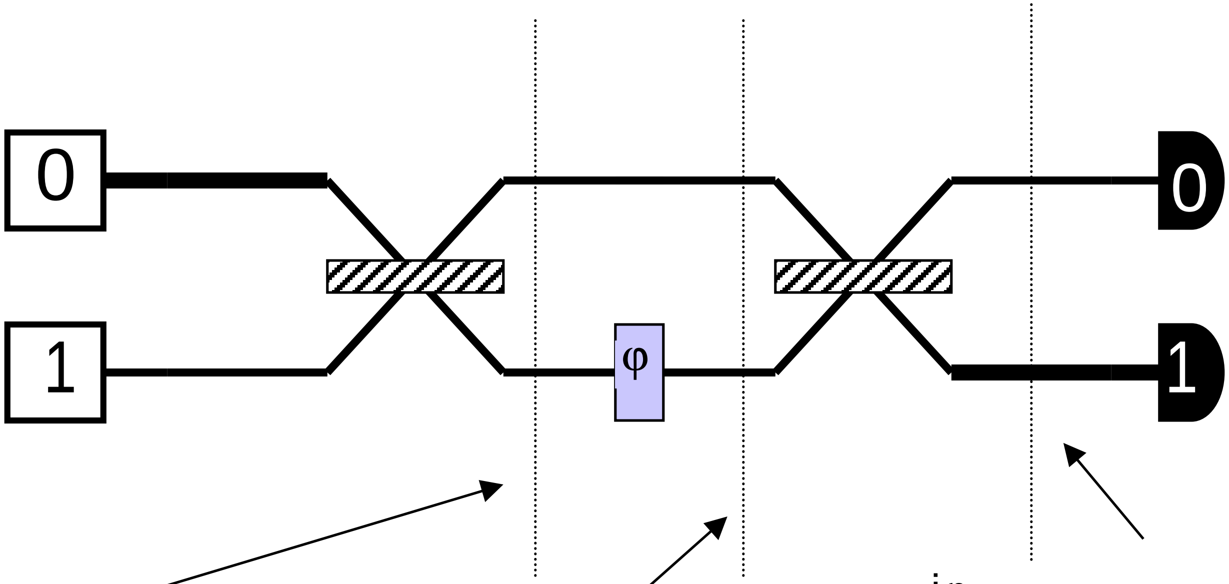


$\sin^2\left(\frac{\varphi}{2}\right)$

$\cos^2\left(\frac{\varphi}{2}\right)$

A new theory

The particle can exist in a linear combination or *superposition* of the two paths



$$\sin^2\left(\frac{\varphi}{2}\right)$$

$$\cos^2\left(\frac{\varphi}{2}\right)$$

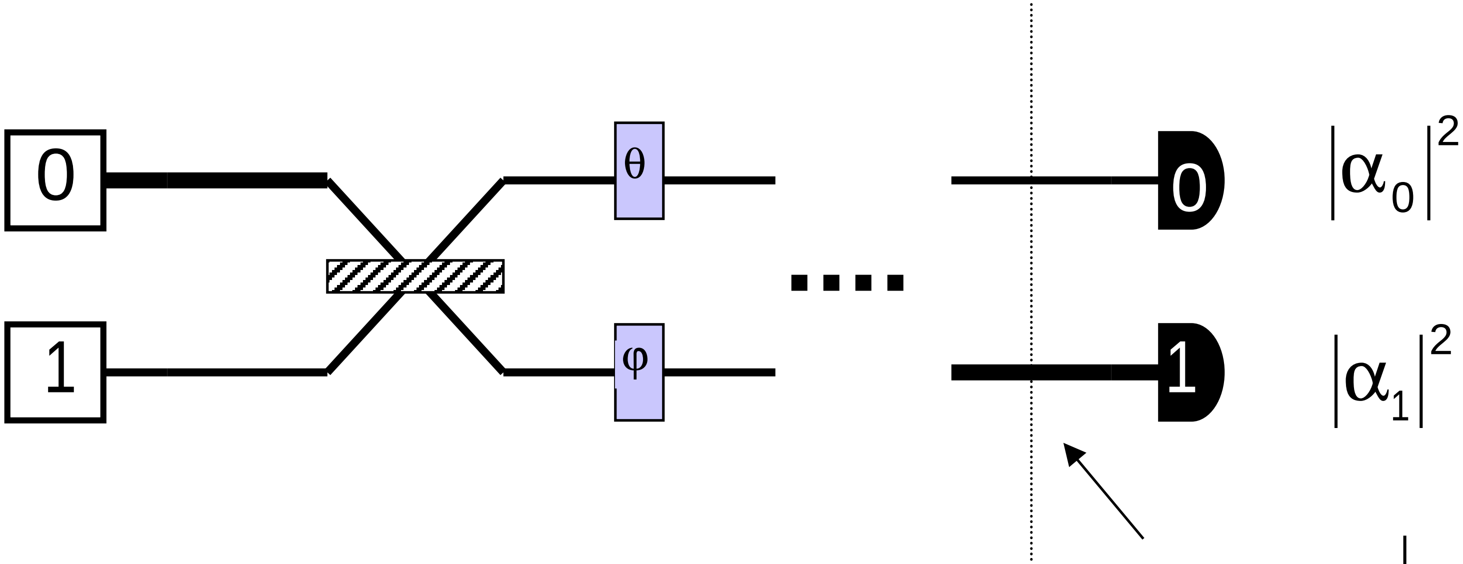
$$\frac{i}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

$$\frac{i}{\sqrt{2}}|0\rangle + \frac{e^{i\varphi}}{\sqrt{2}}|1\rangle$$

$$\frac{e^{i\varphi}-1}{2}|0\rangle + \frac{i(e^{i\varphi}+1)}{\sqrt{2}}|1\rangle$$

Probability Amplitude and Measurement

If the photon is measured when it is in the state $\alpha_0|0\rangle + \alpha_1|1\rangle$ then we get $|0\rangle$ with probability $|\alpha_0|^2$



$$|\alpha_0|^2 + |\alpha_1|^2 = 1$$

Quantum Operations

The operations are induced by the apparatus *linearly*, that is, if

$$|0\rangle \to \frac{i}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

and

$$|1\rangle \to \frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle$$

then

$$\alpha_0|0\rangle + \alpha_1|1\rangle \to \alpha_0\left(\frac{i}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) + \alpha_1\left(\frac{1}{\sqrt{2}}|0\rangle + \frac{i}{\sqrt{2}}|1\rangle\right)$$

$$= \left(\alpha_0\frac{i}{\sqrt{2}} + \alpha_1\frac{1}{\sqrt{2}}\right)|0\rangle + \left(\alpha_0\frac{1}{\sqrt{2}} + \alpha_1\frac{i}{\sqrt{2}}\right)|1\rangle$$

Quantum Operations

Any linear operation that takes states

$$\alpha_0|0\rangle + \alpha_1|1\rangle \quad \text{satisfying} \quad |\alpha_0|^2 + |\alpha_1|^2 = 1$$

and maps them to states

$$\alpha_0^{'}|0\rangle + \alpha_1^{'}|1\rangle \quad \text{satisfying} \quad |\alpha_0^{'}|^2 + |\alpha_1^{'}|^2 = 1$$

must be UNITARY

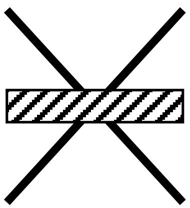
Linear Algebra

$|0\rangle$ corresponds to $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$

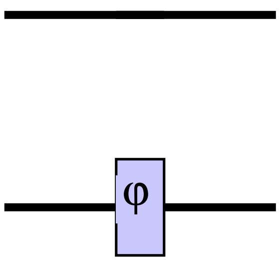
$|1\rangle$ corresponds to $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$

$\alpha_0|0\rangle + \alpha_1|1\rangle$ corresponds to $\alpha_0\begin{pmatrix} 1 \\ 0 \end{pmatrix} + \alpha_1\begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}$

Linear Algebra

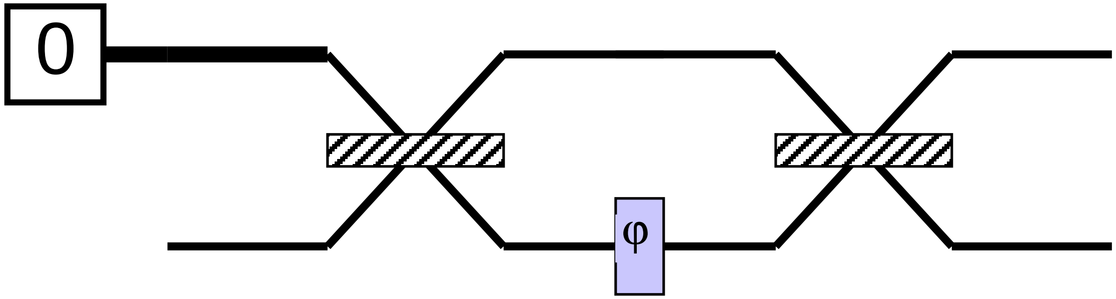


corresponds to $\begin{pmatrix} \frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \end{pmatrix}$



corresponds to $\begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix}$

Linear Algebra



corresponds to

$$\begin{pmatrix} \frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix} \begin{pmatrix} \frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

Linear Algebra

$$U = \begin{bmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{bmatrix}$$

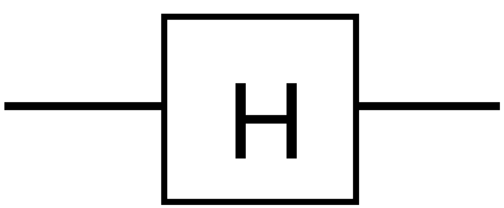
is unitary if and only if

$$UU^t = \begin{bmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{bmatrix} \begin{bmatrix} u^*_{00} & u^*_{10} \\ u^*_{01} & u^*_{11} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I$$

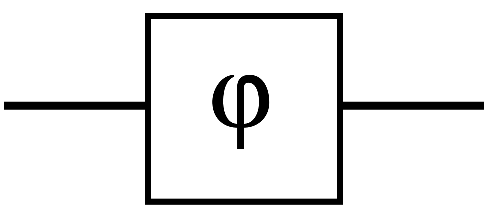
Abstraction

The two position states of a photon in a Mach-Zehnder apparatus is just one example of a quantum bit or ***qubit***

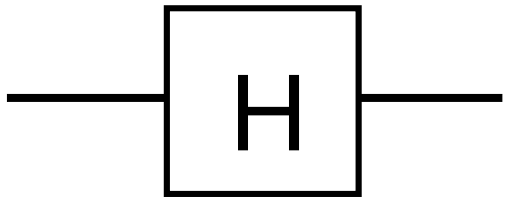
Except when addressing a particular physical implementation, we will simply talk about "basis" states $|0\rangle$ and $|1\rangle$
and **unitary operations** like



and



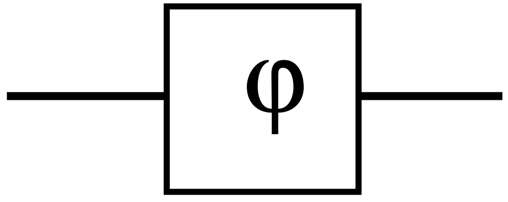
where



corresponds to

$$\begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

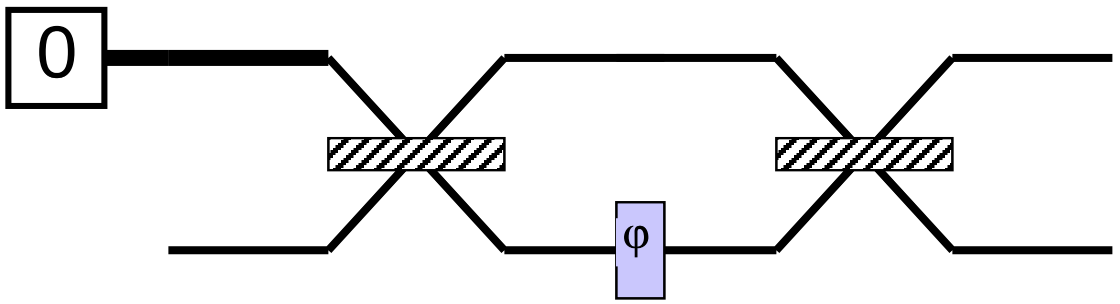
and



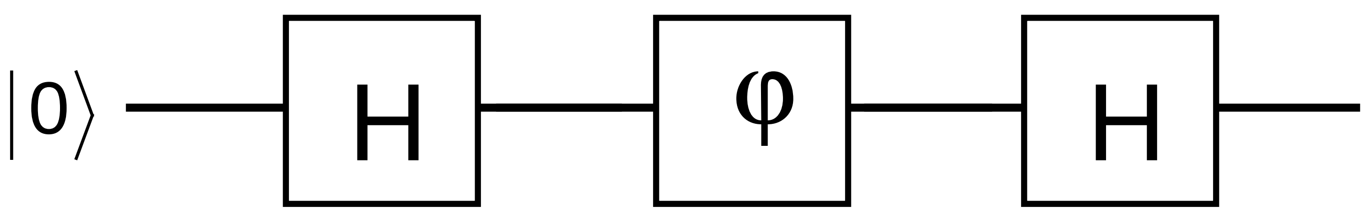
corresponds to

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{pmatrix}$$

An arrangement like



is represented with a network *like*



More than one qubit

If we concatenate two qubits

$$\left(\alpha_0|0\rangle + \alpha_1|1\rangle\right)\ \left(\beta_0|0\rangle + \beta_1|1\rangle\right)$$

we have a 2-qubit system with 4 basis states

$$|0\rangle|0\rangle = |00\rangle \qquad |0\rangle|1\rangle = |01\rangle \qquad |1\rangle|0\rangle = |10\rangle \qquad |1\rangle|1\rangle = |11\rangle$$

and we can also describe the state as

$$\alpha_0\beta_0|00\rangle + \alpha_0\beta_1|01\rangle + \alpha_1\beta_0|10\rangle + \alpha_1\beta_1|11\rangle$$

or by the vector
$$\begin{pmatrix} \alpha_0\beta_0 \\ \alpha_0\beta \\ \alpha_1\beta_0 \\ \alpha_1\beta_1 \end{pmatrix} = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix} \otimes \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix}$$

More than one qubit

In general we can have arbitrary superpositions

$$\alpha_{00}|0\rangle|0\rangle + \alpha_{01}|0\rangle|1\rangle + \alpha_{10}|1\rangle|0\rangle + \alpha_{11}|1\rangle|1\rangle$$

$$|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$$

where there is no factorization into the tensor product of two independent qubits.

These states are called *entangled.*

Measuring multi-qubit systems

If we measure both bits of

$$\alpha_{00}|0\rangle|0\rangle + \alpha_{01}|0\rangle|1\rangle + \alpha_{10}|1\rangle|0\rangle + \alpha_{11}|1\rangle|1\rangle$$

we get $|x\rangle|y\rangle$ with probability $|\alpha_{xy}|^2$

Classical

Versus

Quantum

Classical vs. Quantum Circuits

- Goal: Fast, low-cost implementation of useful algorithms using standard components (gates) and design techniques
- Classical Logic Circuits
  - Circuit behavior is governed implicitly by classical physics
  - Signal states are simple bit vectors, e.g. $X = 01010111$
  - Operations are defined by Boolean Algebra
  - No restrictions exist on copying or measuring signals
  - Small well-defined sets of universal gate types, e.g. {NAND}, {AND,OR,NOT}, {AND,NOT}, etc.
  - Well developed CAD methodologies exist
  - Circuits are easily implemented in fast, scalable and macroscopic technologies such as CMOS

Classical vs. Quantum Circuits

- Quantum Logic Circuits
  - Circuit behavior is governed explicitly by quantum mechanics
  - Signal states are vectors interpreted as a superposition of binary "qubit" vectors with complex-number coefficients

$$|\Psi\rangle = \sum_{i=0}^{2^n-1} c_i \left| i_{n-1} i_{n-1} \ldots i_0 \right\rangle$$

  - Operations are defined by linear algebra over Hilbert Space and can be represented by unitary matrices with complex elements
  - Severe restrictions exist on copying and measuring signals
  - Many universal gate sets exist but the best types are not obvious
  - Circuits must use microscopic technologies that are slow, fragile, and not yet scalable, e.g., NMR

Quantum Circuit Characteristics

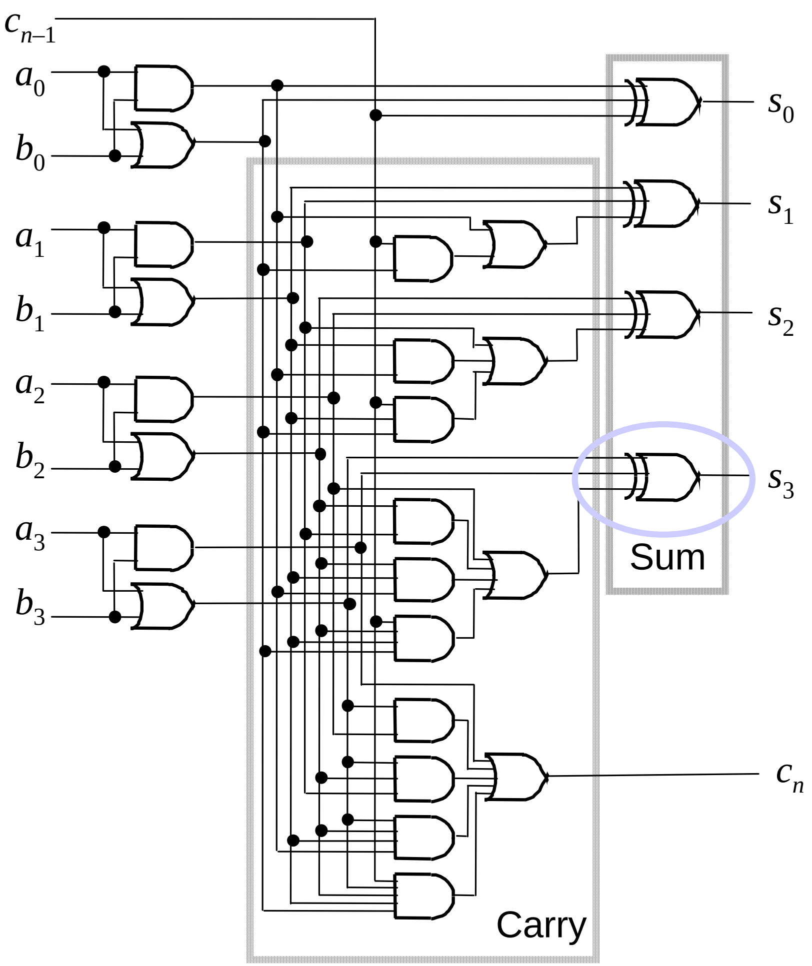
- Unitary Operations
  - Gates and circuits must be reversible (information-lossless)
    - Number of output signal lines = Number of input signal lines
    - The circuit function must be a bijection, implying that output vectors are a permutation of the input vectors
  - **Classical** logic behavior can be represented by permutation matrices
  - **Non-classical** logic behavior can be represented including state sign (phase) and entanglement

Quantum Circuit Characteristics

- Quantum Measurement
  - Measurement yields only one state *X* of the superposed states
  - Measurement also makes *X* the new state and so ***interferes with computational processes***
  - *X* is determined with some **probability**, implying uncertainty in the result
  - States cannot be copied (“cloned”), implying that **signal fanout is not permitted**
  - Environmental interference can cause a measurement-like state collapse (**decoherence**)

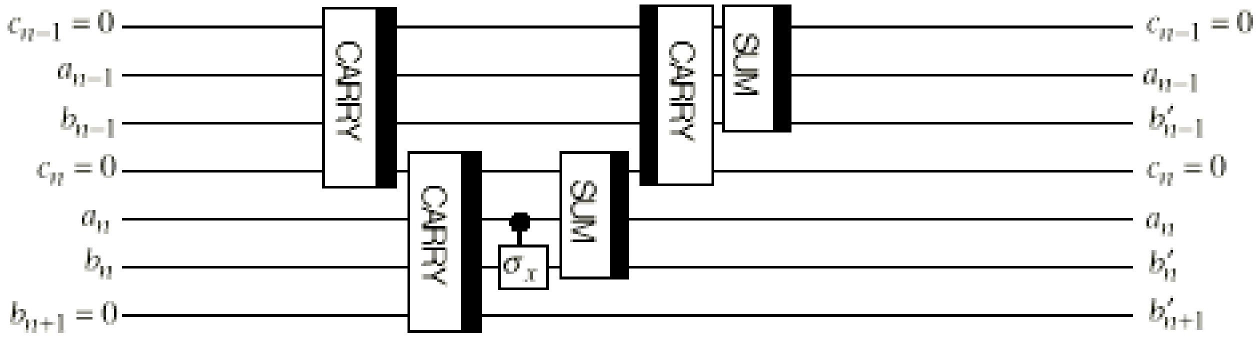
Classical vs. Quantum Circuits

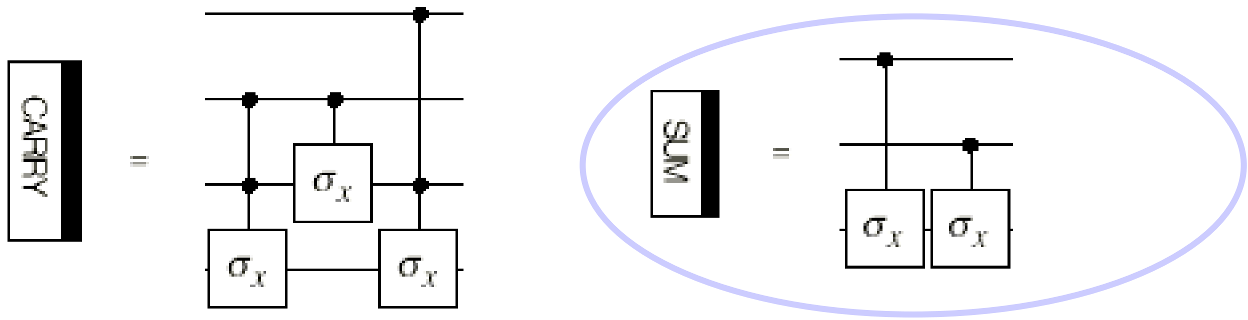
Classical adder



Classical vs. Quantum Circuits

Quantum adder

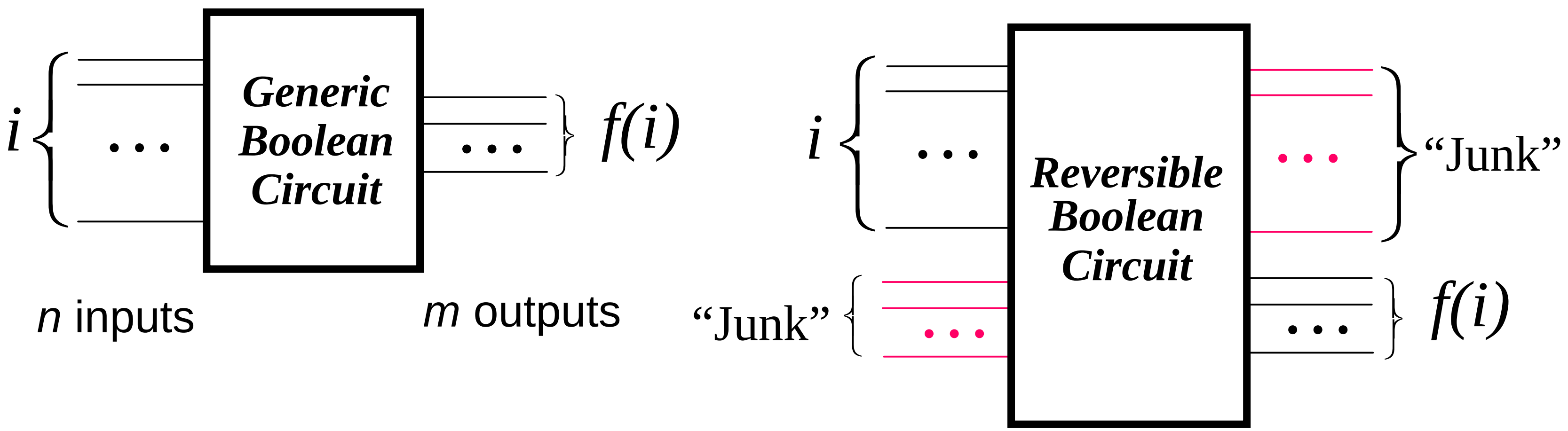




Reversible Circuits

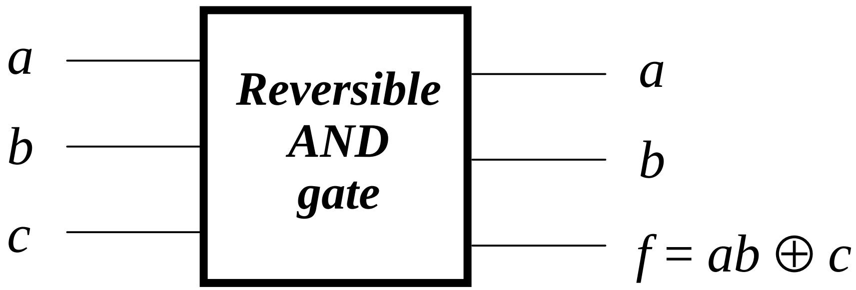
Reversible Circuits

- Reversibility was studied around 1980 motivated by power minimization considerations
- Bennett, Toffoli et al. showed that any classical logic circuit *C* can be made reversible with modest overhead



Reversible Circuits

- How to make a given f reversible
  - Suppose $f: i \rightarrow f(i)$ has n inputs m outputs
  - Introduce n extra outputs and m extra inputs
  - Replace f by $f_{\text{rev}}: i, j \rightarrow i, f(i) \oplus j$ where $\oplus$ is XOR
- Example 1: $f(a,b) = \text{AND}(a,b)$

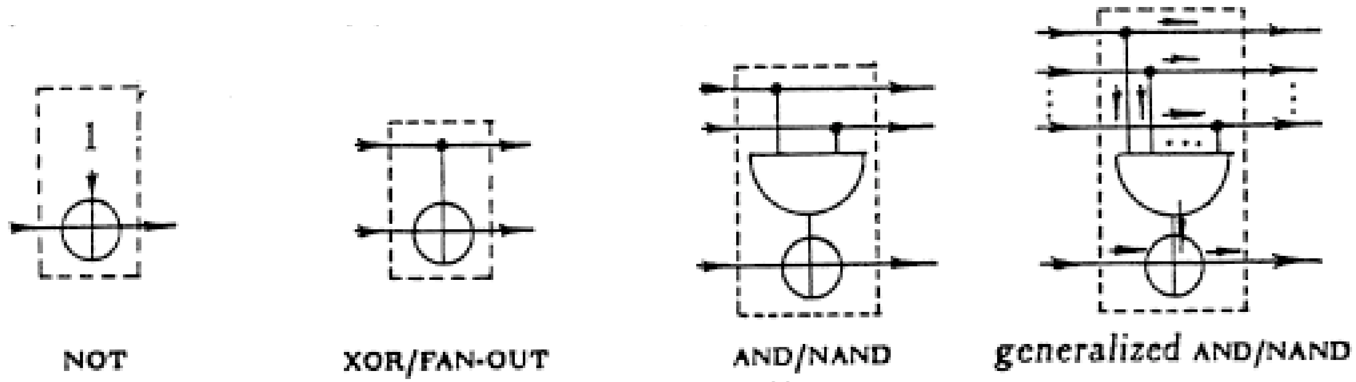


a	b	c	a	b	f
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

- This is the well-known Toffoli gate, which realizes AND when $c = 0$, and NAND when $c = 1$.

Reversible Circuits

- Reversible gate family [Toffoli 1980]



(Toffoli gate)

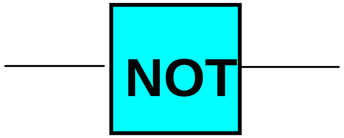
- Every Boolean function has a reversible implementation using Toffoli gates.
- There is no universal reversible gate with fewer than three inputs

Quantum

Gates

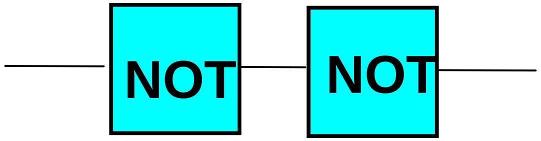
Quantum Gates

- **One-Input gate:** NOT
  - Input state: $c_0|0\rangle + c_1|1\rangle$
  - Output state: $c_1|0\rangle + c_0|1\rangle$



  - Pure states are mapped thus: $|0\rangle \rightarrow |1\rangle$ and $|1\rangle \rightarrow |0\rangle$
  - Gate operator (matrix) is $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ $\quad |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
  - As expected:

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$



Quantum Gates

- **One-Input gate:** "Square root of NOT"
  - Some matrix elements are imaginary
  - Gate operator (matrix):

$$\begin{pmatrix} i/\sqrt{1/2} & 1/\sqrt{1/2} \\ 1/\sqrt{1/2} & i/\sqrt{1/2} \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} i & 1 \\ 1 & i \end{pmatrix}$$

  - We find:

$$\frac{1}{\sqrt{2}}\begin{pmatrix} i & 1 \\ 1 & i \end{pmatrix}\begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} i \\ 1 \end{pmatrix}$$

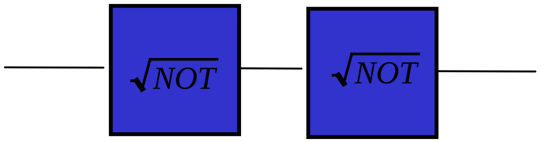
so $|0\rangle \rightarrow |0\rangle$ with probability $|i/\sqrt{2}|^2 = 1/2$

and $|0\rangle \rightarrow |1\rangle$ with probability $|1/\sqrt{2}|^2 = 1/2$

Similarly, this gate randomizes input $|1\rangle$

  - But concatenation of two gates eliminates the randomness!

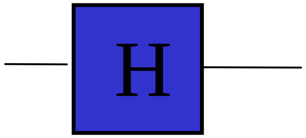
$$\frac{1}{2}\begin{pmatrix} i & 1 \\ 1 & i \end{pmatrix}\begin{pmatrix} i & 1 \\ 1 & i \end{pmatrix} = \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix}$$



Quantum Gates

- **One-Input gate:** Hadamard

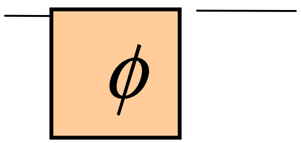
$$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$



  - Maps $|0\rangle \rightarrow 1/\sqrt{2}\,|0\rangle + 1/\sqrt{2}\,|1\rangle$ and $|1\rangle \rightarrow 1/\sqrt{2}\,|0\rangle - 1/\sqrt{2}\,|1\rangle$.
  - Ignoring the normalization factor $1/\sqrt{2}$, we can write
    $|x\rangle \rightarrow (-1)^x\,|x\rangle - |1-x\rangle$

- **One-Input gate:** Phase shift

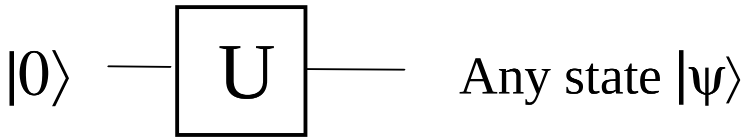
$$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}$$



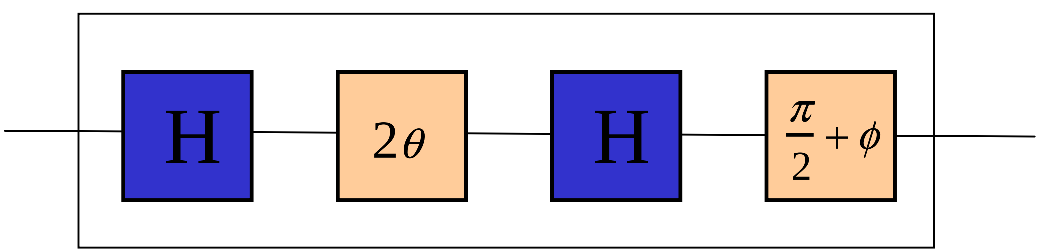
Quantum Gates

Universal One-Input Gate Sets

- Requirement:

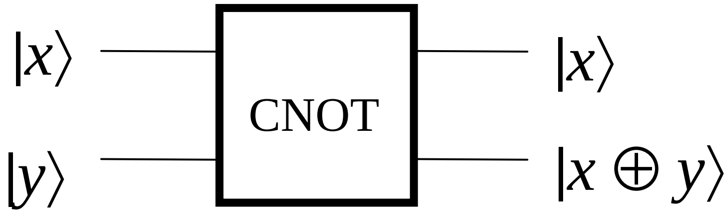


- **Hadamard** and **phase-shift** gates form a universal gate set
- *Example*: The following circuit generates
  $|\psi\rangle = \cos\theta\ |0\rangle + e^{i\phi} \sin\theta\ |1\rangle$ up to a global factor

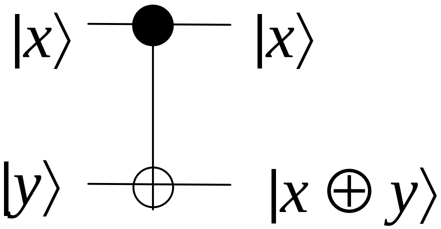


Quantum Gates

- **Two-Input Gate:** Controlled NOT (CNOT)



$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$



  – CNOT maps $|x\rangle|0\rangle \rightarrow |x\rangle||x\rangle$ and $|x\rangle|1\rangle \rightarrow |x\rangle||\text{NOT } x\rangle$

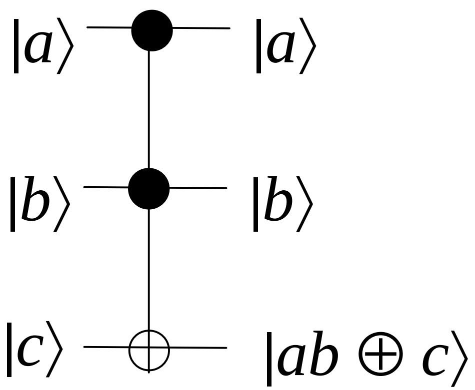
    $|x\rangle|0\rangle \rightarrow |x\rangle||x\rangle$ *looks like cloning*, <u>but it's not</u>. These mappings are **<u>valid only for the pure states</u>** $|0\rangle$ and $|1\rangle$

  – Serves as a "non-demolition" measurement gate

Quantum Gates

- **3-Input gate:** Controlled CNOT (C^2NOT or Toffoli gate)

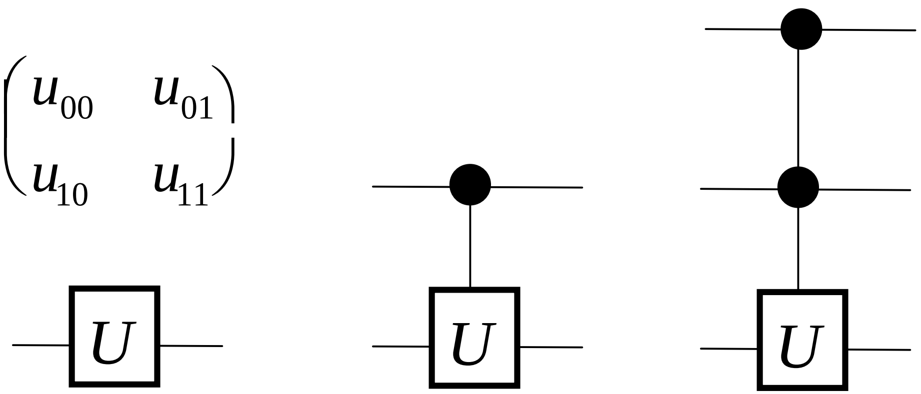
$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Quantum Gates

- General controlled gates that control some 1-qubit unitary operation U are useful

$$\begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix}$$



etc.

U $\quad$ $C(U)$ $\quad$ $C^2(U)$

Quantum Gates

Universal Gate Sets

- To implement any unitary operation on *n* qubits exactly requires an infinite number of gate types
- The (infinite) set of all 2-input gates is universal
  - Any *n*-qubit unitary operation can be implemented using $\Theta(n^3 4^n)$ gates [Reck et al. 1994]
- CNOT and the (infinite) set of all 1-qubit gates is universal

Quantum Gates

Discrete Universal Gate Sets

- The **error** on implementing *U* by *V* is defined as

$$E(U,V) = \max_{|\Psi\rangle} \left\| (U - V)|\Psi\rangle \right\|$$

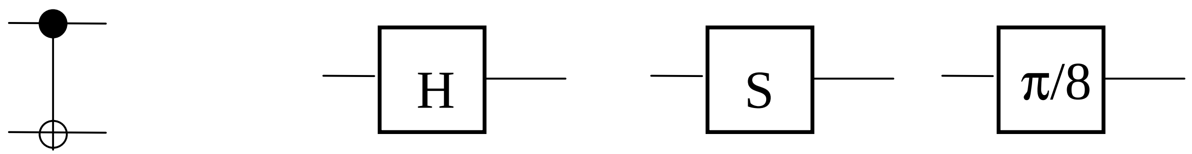
- If *U* can be implemented by *K* gates, we can simulate *U* with a total error less than ε with a gate overhead that is polynomial in log(*K*/ε)
- A discrete set of gate types ***G*** is universal, if we can approximate any *U* to within any ε > 0 using a sequence of gates from ***G***

Quantum Gates

Discrete Universal Gate Set

- **Example 1**: Four-member “standard” gate set

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$$



CNOT Hadamard Phase π/8 (T) gate

- **Example** 2: {CNOT, Hadamard, Phase, Toffoli}

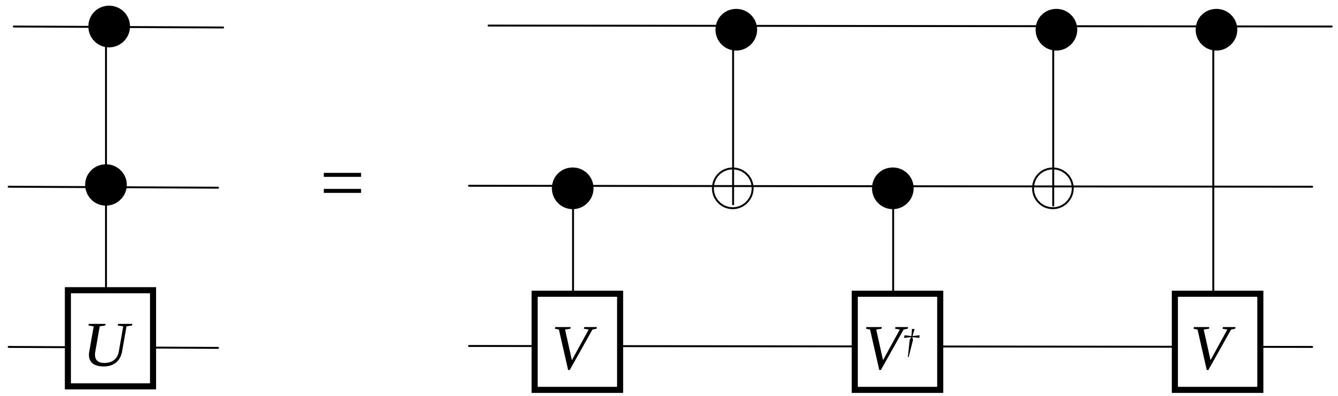
Quantum Circuits

Quantum Circuits

- A quantum (combinational) circuit is a sequence of quantum gates, linked by “wires”
- The circuit has fixed “width” corresponding to the number of qubits being processed
- Logic design (classical and quantum) attempts to find circuit structures for needed operations that are
  - Functionally correct
  - Independent of physical technology
  - Low-cost, e.g., use the minimum number of qubits or gates
- Quantum logic design is not well developed!

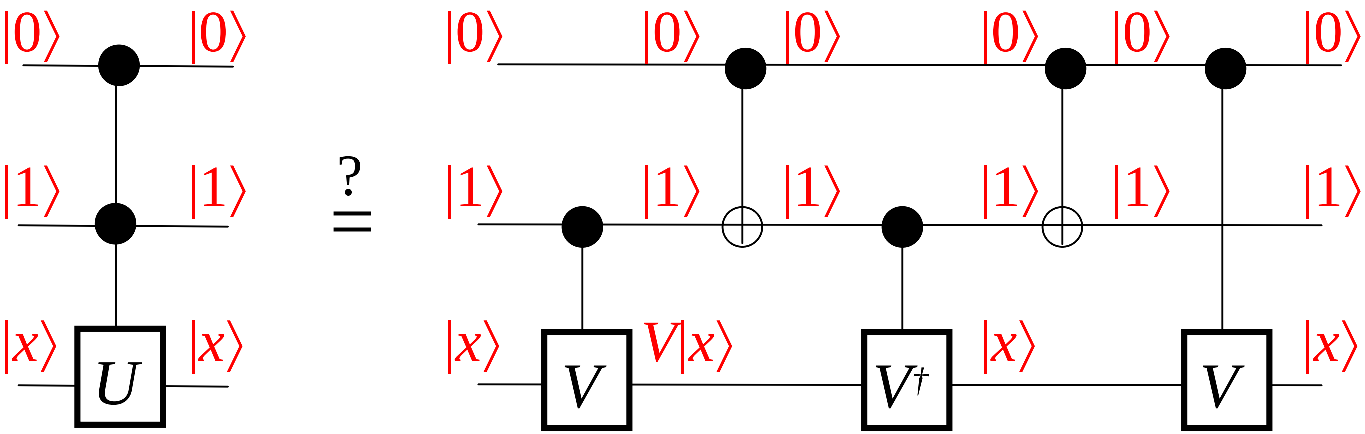
Quantum Circuits

- Ad hoc designs known for many specific functions and gates
- **Example 1** illustrating a theorem by [Barenco et al. 1995]: Any $C^2(U)$ gate can be built from CNOTs, $C(V)$, and $C(V^\dagger)$ gates, where $V^2 = U$



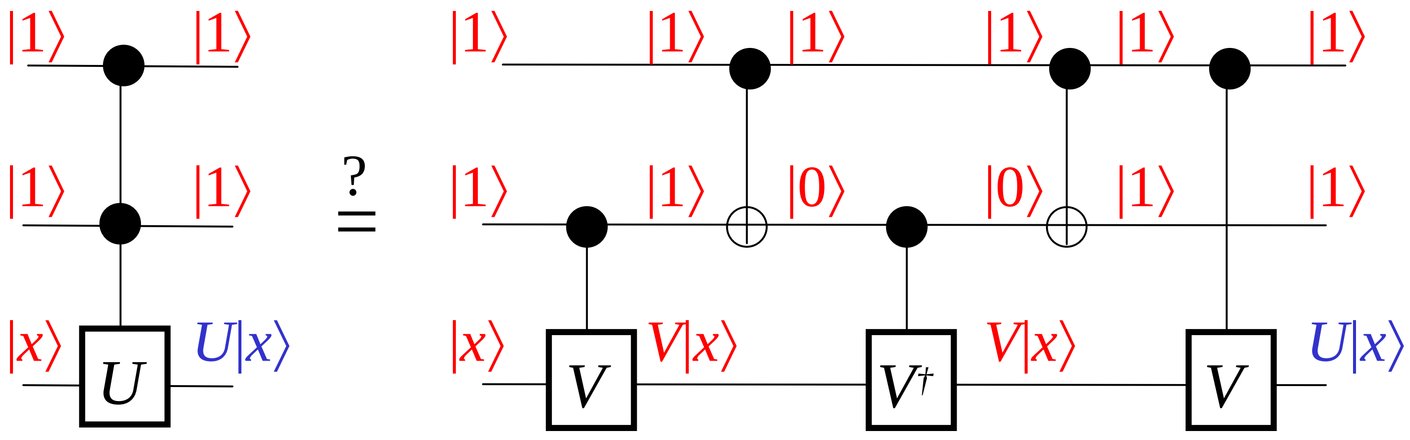
Quantum Circuits

Example 1: Simulation



Quantum Circuits

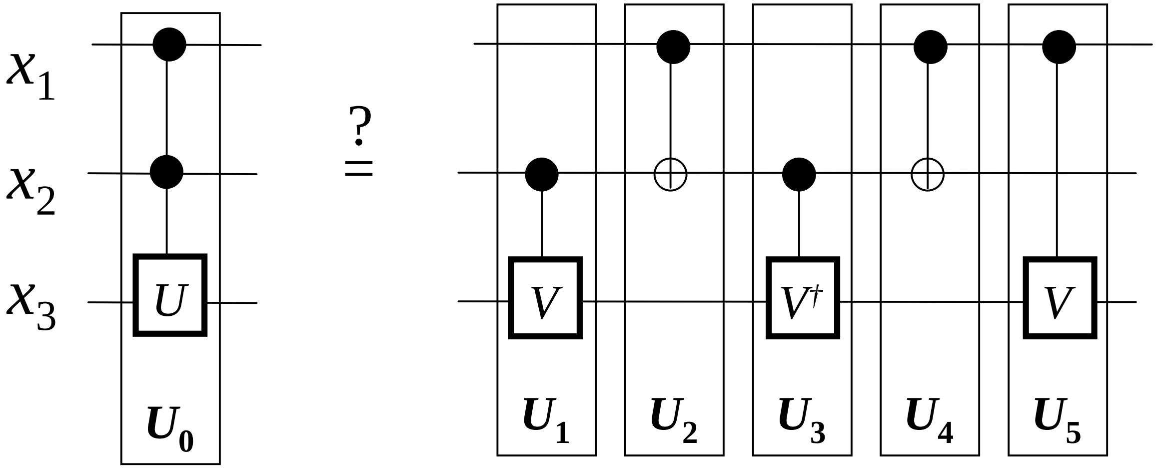
Example 1: Simulation (contd.)



- *Exercise*: Simulate the two remaining cases

Quantum Circuits

Example 1: Algebraic analysis



- Is $U_0(x_1, x_2, x_3) = U_5U_4U_3U_2U_1(x_1, x_2, x_3)$

  $= (x_1, x_2, x_1x_2 \oplus U(x_3))$?

Quantum Circuits

Example 1 (contd);

$$U_1 = I_1 \otimes C(V)$$

$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & v_{00} & v_{01} \\ 0 & 0 & v_{10} & v_{11} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & v_{00} & v_{01} & 0 & 0 & 0 & 0 \\ 0 & 0 & v_{10} & v_{11} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & v_{00} & v_{01} \\ 0 & 0 & 0 & 0 & 0 & 0 & v_{10} & v_{11} \end{pmatrix}$$

Quantum Circuits

Example 1 (contd);

$$U_2 = U_4 = CNOT(x_1, x_2) \otimes I_1$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \otimes \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Quantum Circuits

Example 1 (contd);

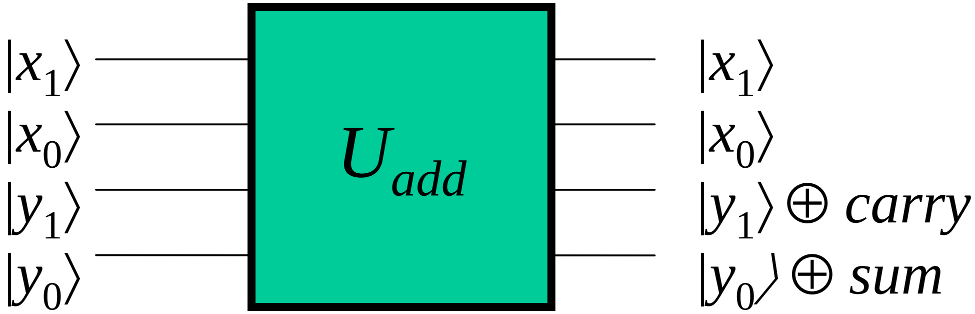
- U_5 is the same as U_1 but has $x_1$and x_2 permuted (tricky!)
- It remains to evaluate the product of five 8 x 8 matrices $U_5U_4U_3U_2U_1$ using the fact that $VV^\dagger = I$ and $VV = U$

$$
\begin{pmatrix}
1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0\\
0&0&0&0&v_{00}&v_{01}&0&0\\
0&0&0&0&v_{10}&v_{11}&0&0\\
0&0&0&0&0&0&v_{00}&v_{01}\\
0&0&0&0&0&0&v_{10}&v_{11}
\end{pmatrix}
\begin{pmatrix}
1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&1\\
0&0&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0
\end{pmatrix}
\begin{pmatrix}
1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0\\
0&0&\mathbf{v}_{00}&\mathbf{v}_{10}&0&0&0&0\\
0&0&\mathbf{v}_{01}&\mathbf{v}_{11}&0&0&0&0\\
0&0&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&\mathbf{v}_{00}&\mathbf{v}_{10}\\
0&0&0&0&0&0&\mathbf{v}_{01}&\mathbf{v}_{11}
\end{pmatrix}
\begin{pmatrix}
1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&1\\
0&0&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0
\end{pmatrix}
\begin{pmatrix}
1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0\\
0&0&v_{00}&v_{01}&0&0&0&0\\
0&0&v_{10}&v_{11}&0&0&0&0\\
0&0&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&v_{00}&v_{01}\\
0&0&0&0&0&0&v_{10}&v_{11}
\end{pmatrix}
$$

$$
= \begin{pmatrix}
1&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0\\
0&0&0&0&1&0&0&0\\
0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&\mathbf{v}_{00}v_{00}+\mathbf{v}_{10}v_{10}&\mathbf{v}_{00}v_{01}+\mathbf{v}_{10}v_{11}\\
0&0&0&0&0&0&\mathbf{v}_{01}v_{00}+\mathbf{v}_{11}v_{10}&\mathbf{v}_{01}v_{01}+\mathbf{v}_{11}v_{11}
\end{pmatrix} = U_0
$$

Quantum Circuits

- **Implementing a Half Adder**
  - *Problem:* Implement the classical functions $sum = x_1 \oplus x_0$ and $carry = x_1 x_0$
- Generic design:



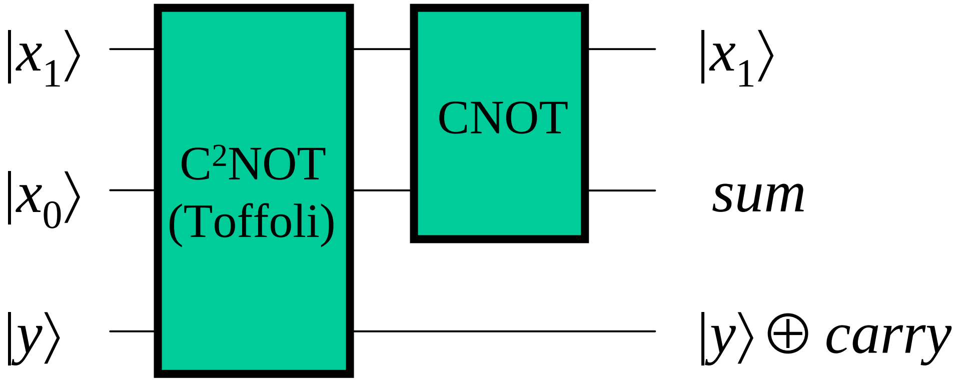
Quantum Circuits

- **Half Adder**: Generic design (contd.)

$$
U_{ADD} = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0
\end{pmatrix}
$$

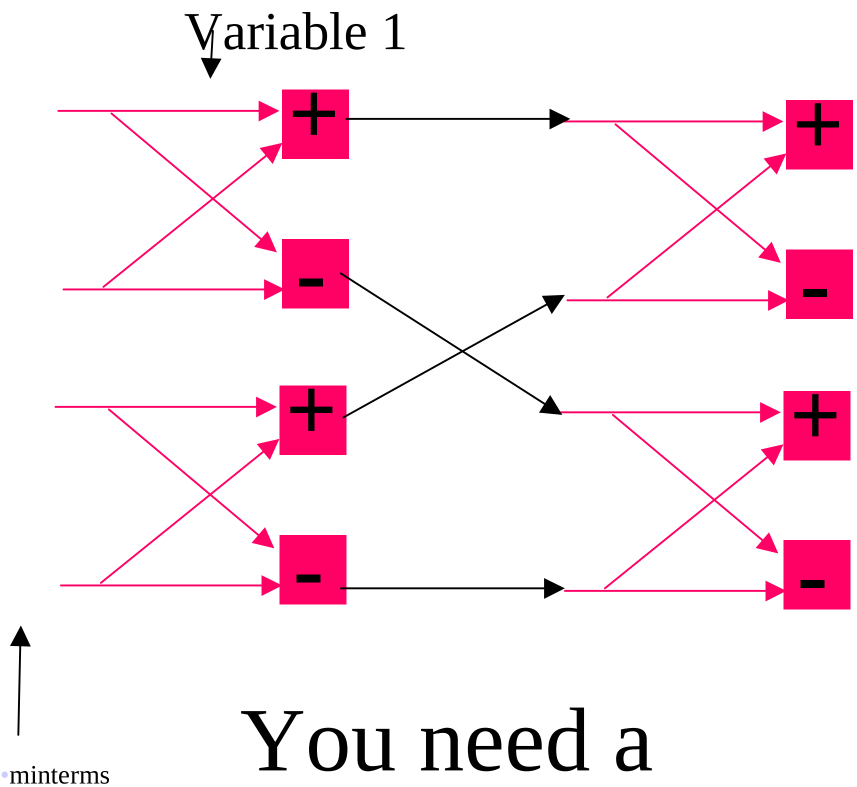
Quantum Circuits

- **Half Adder**: Specific (reduced) design



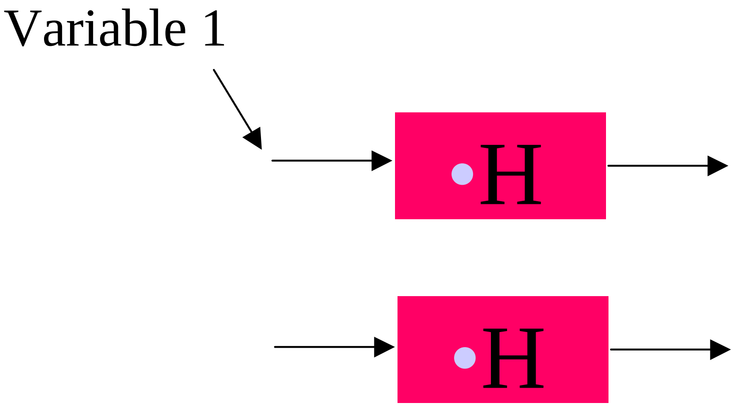
Walsh Transform for two binary-input many-valued variables

Classical logic



You need a butterfly

Quantum logic



Butterfly is created automatically by tensor product corresponding to superposition

Portland Quantum Logic Group (PQLG)

What we do?

People at PSU and collaborators

- Marek Perkowski
- Martin Zwick
- Xiaoyu Song
- William Hung
- Anas Al-Rabadi
- Martin Lukac
- Mikhail Pivtoraiko
- Andrei Khlopotine
- Alan Mishchenko (University of California, Berkeley, USA)
- Bernd Steinbach (Technical University of Freiberg, Germany)
- Pawel Kerntopf (Technical University of Warsaw, Poland)
- Mitch Thornton (Southern Methodist University, Dallas, USA)
- Lech Jozwiak (Technical University of Eindhoven, The Netherlands)
- Andrzej Buller (ATR, Kansai Science City, Japan)
- Tsutomu Sasao (Kyushu University of Technology, Iizuka, Japan).

Current Projects

- Logic Synthesis for Reversible Logic
  - decomposition
  - Decision Diagram Mapping
  - composition
  - regular structures - lattices, PLAs, nets
- Logic Synthesis for Quantum Logic
- Quantum Simulation using new Decision Diagrams

4 papers published

paper submitted

Current Projects

- FPGA-based model of Quantum Computer
- Reversible FPGA using CMOS.
- Realization of new spectral transforms using quantum logic.
- Non-linear Quantum Logic solves NP problems in polynomial time.
- Quantum-inspired search algorithms for robotics

Where to learn more

- Web Page of Marek Perkowski
  - class 572 - *see description of student projects*
  - Portland Quantum Logic Group

We are open to collaboration and we want to grow

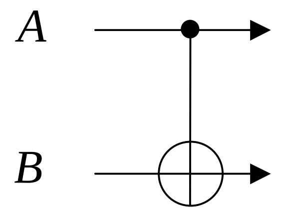
Automated Synthesis of Generalized Reversible Cascades using Genetic Algorithm

Agenda

- Introduction and history
- Reversible Logic and Reversible Gates
- Genetic algorithms
- The Model
- Simulation
- Conclusion

Reversible gates…

Feynman, Toffoli, Fredkin, …

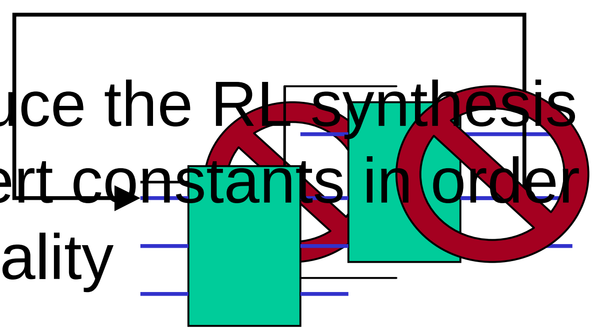


$$P = A$$

$$Q = P \oplus B$$

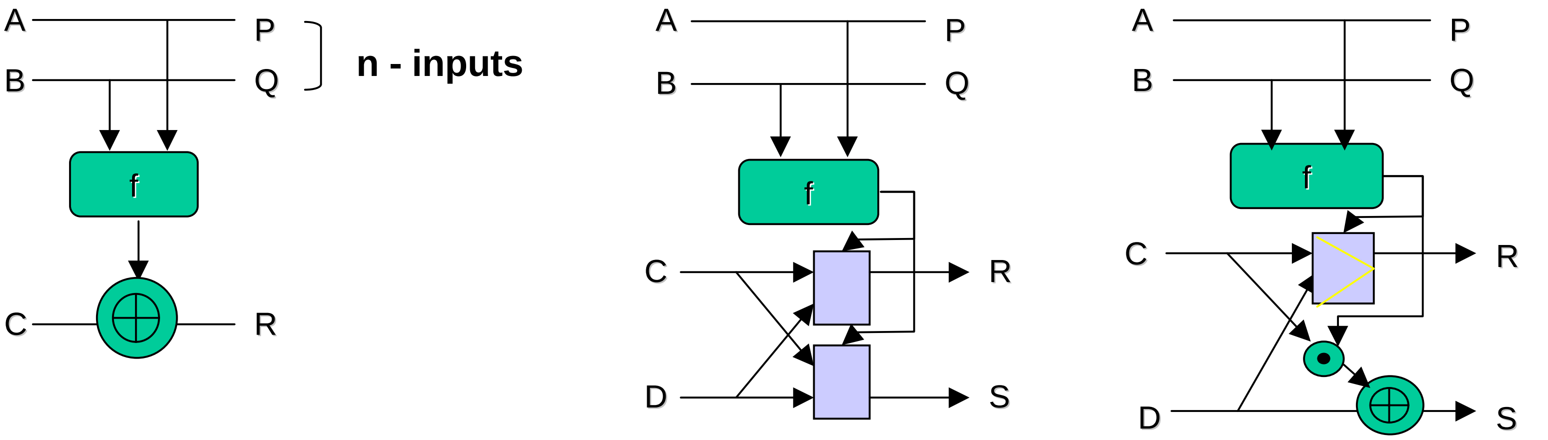
Mapping of I/O allows unique (P,Q)$\Rightarrow$(A,B)

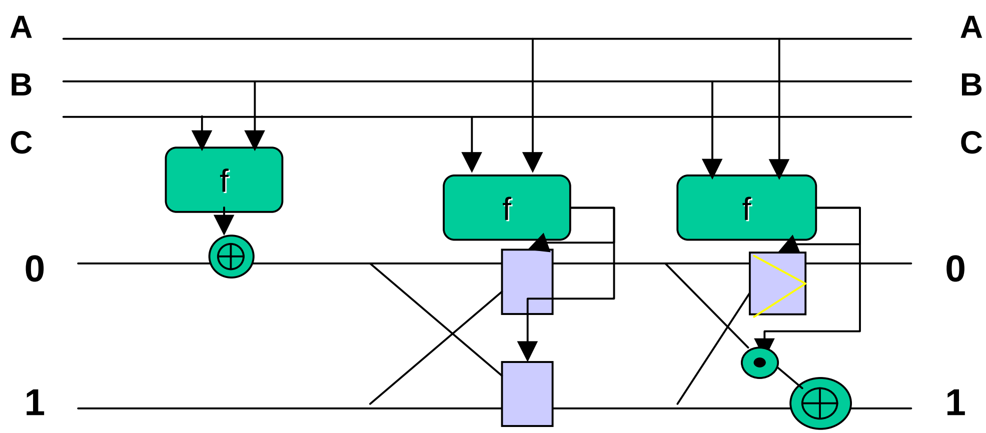
and Reversible Circuits



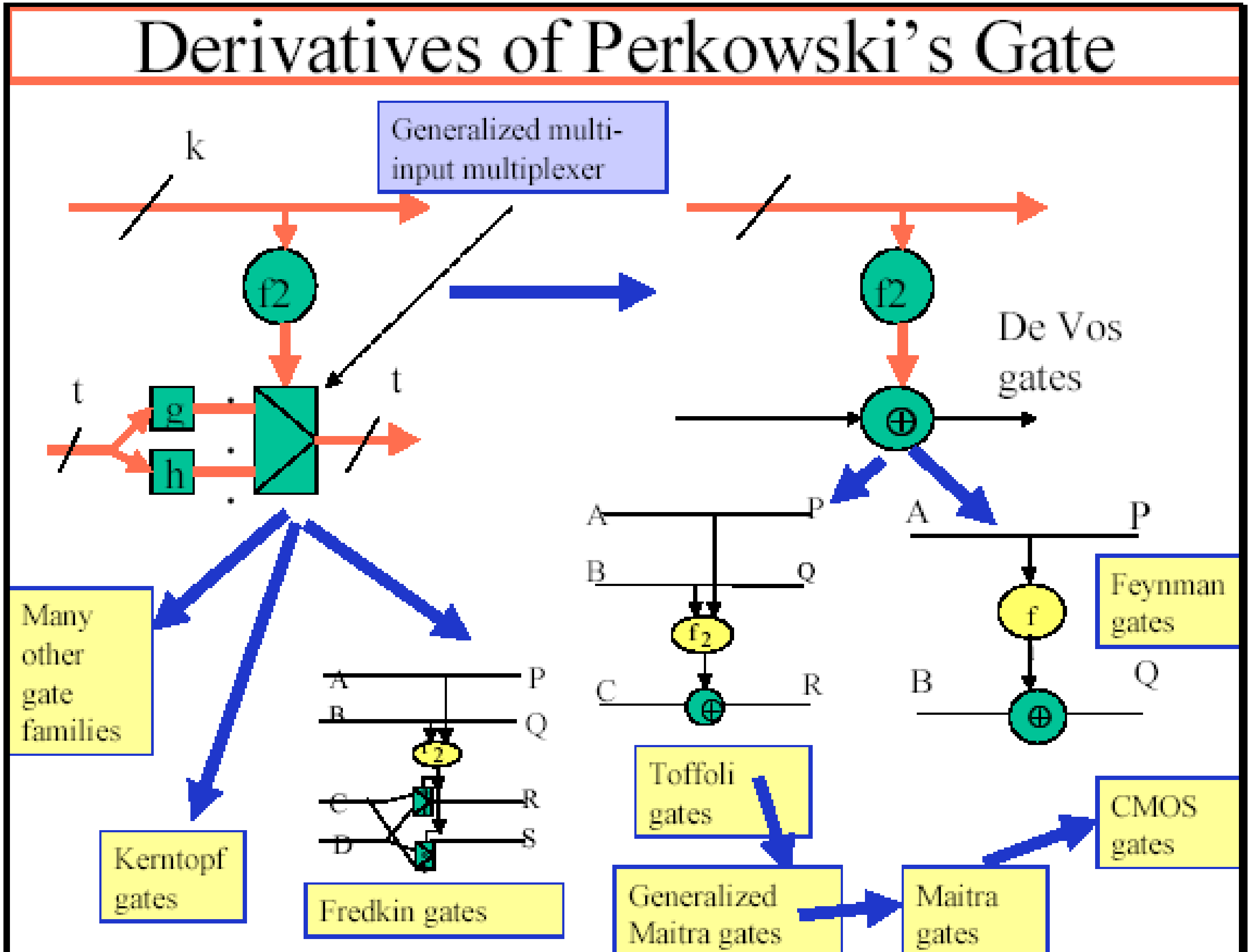
- To reduce the RL synthesis limitations one can insert constants in order to modify the functionality

Generalized Reversible Gates





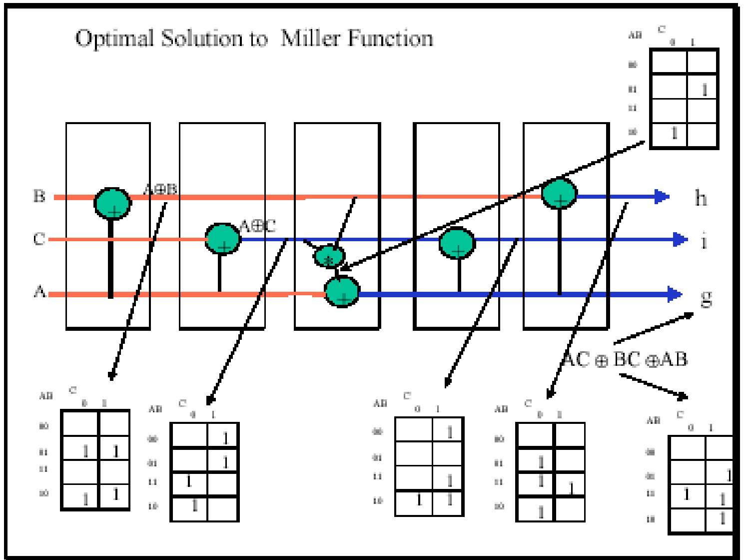
Perkowski gates family

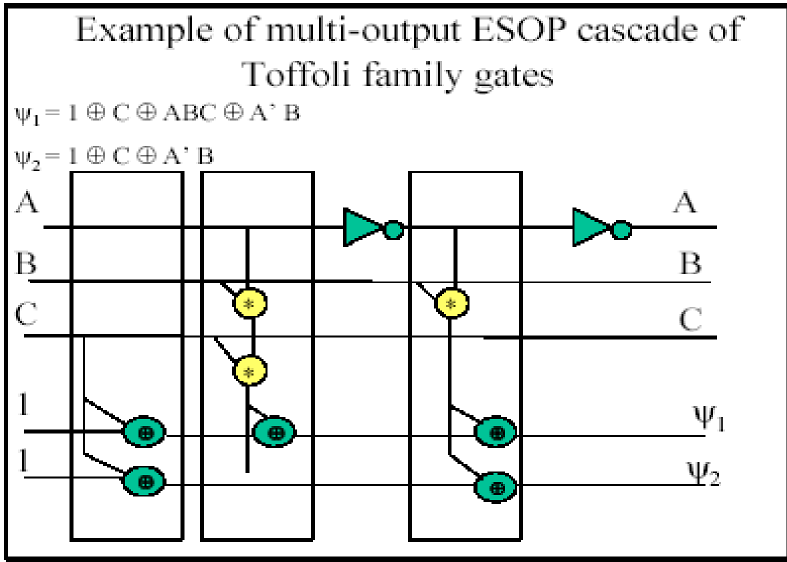


Cascades

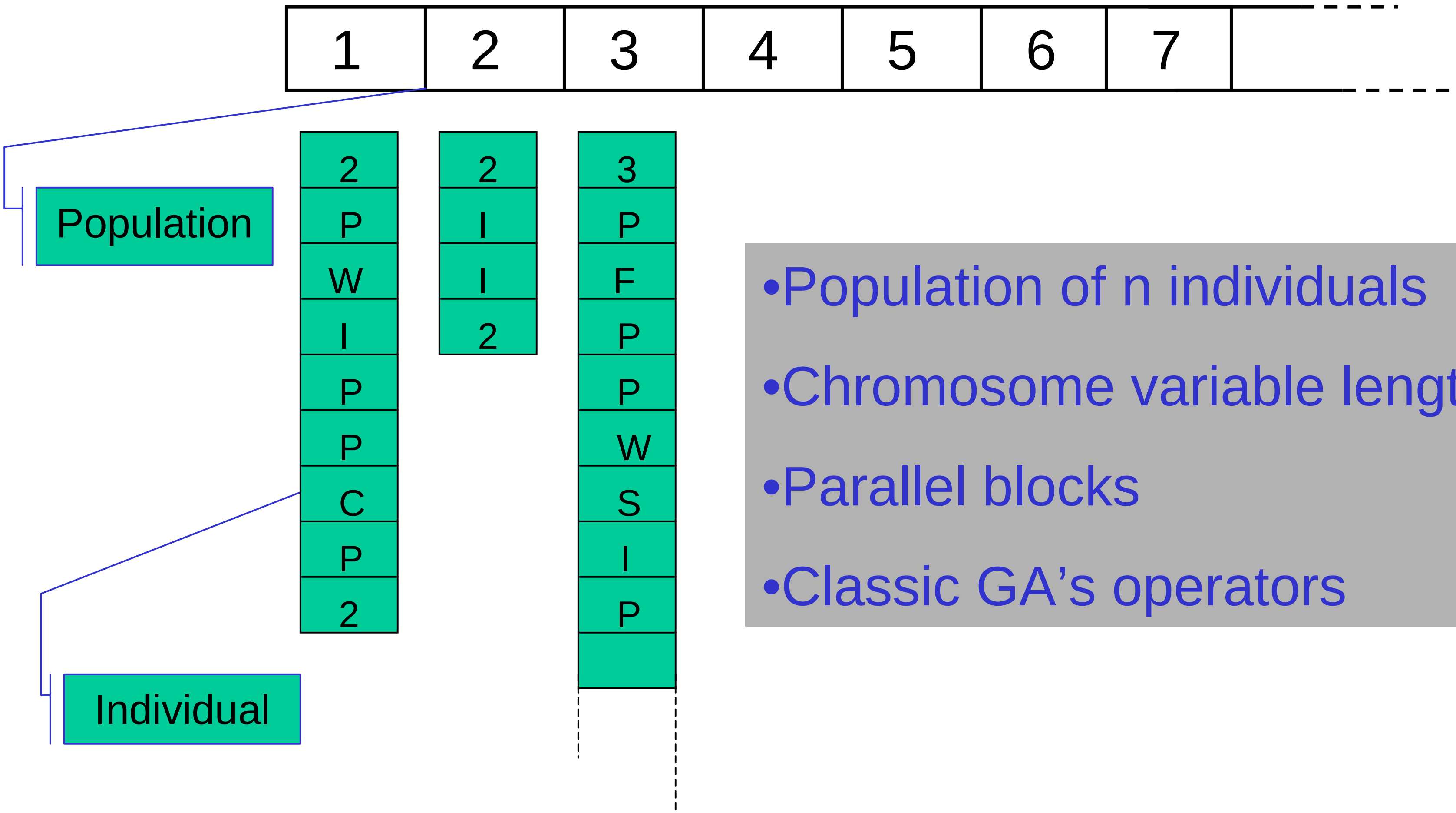
- ***Mixed data/control inputs (generalized complex control gates)***
- ***All :***
  - ESOP
  - Factorized-ESOP
  - MV Complex Terms
  - XOR family

- Example:



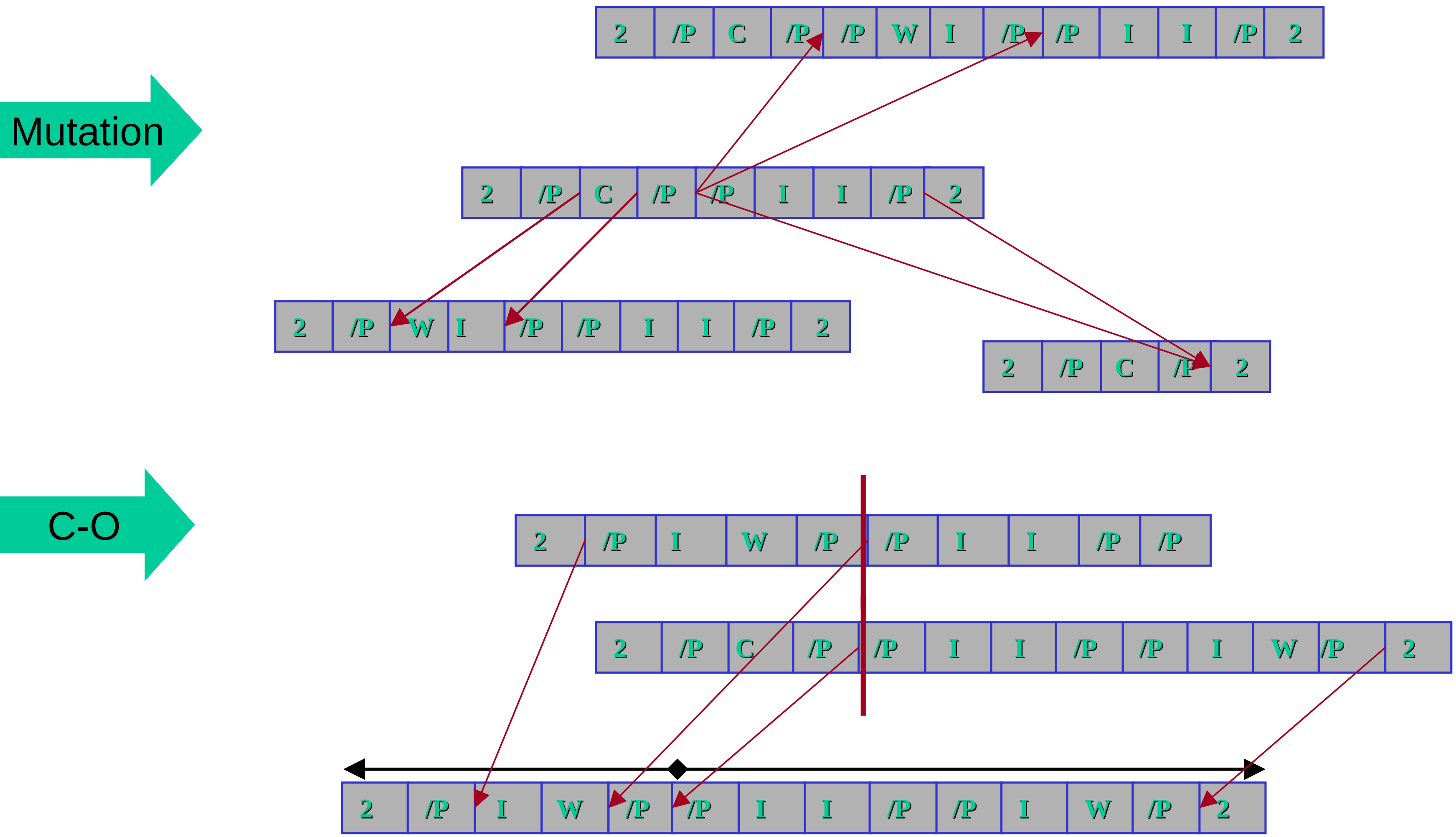


Genetic algorithms

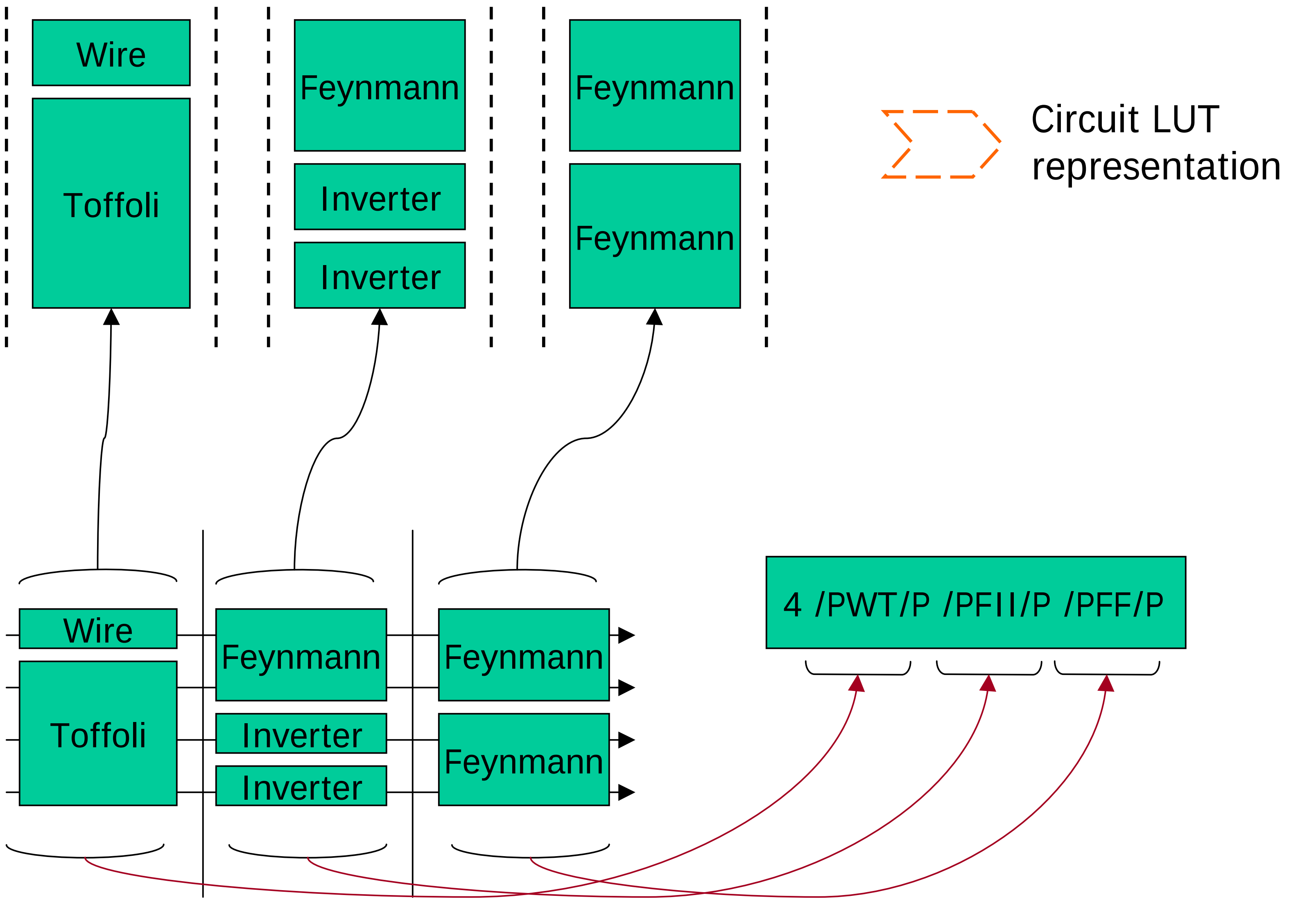


- Population of n individuals
- Chromosome variable length
- Parallel blocks
- Classic GA’s operators

Encoding & operations



Circuit Encoding



GA's settings

- Stochastic universal sampling
- Fitness:

$$F_i = \frac{1}{1 + error_i} - \Lambda_i$$

- Error:

$$error = \sum_{i=1}^{n} \sum_{j=1}^{2^n} \left| U_{ij} - S_{ij} \right| \quad S, U \in U(2^n)$$

- LUT for Fredkin gate:

A, B, D	A'	B'	C'
000	0	0	0
001	0	0	1
010	0	1	0
011	0	1	1
100	1	0	0
101	1	0	1
110	1	1	1
111	1	1	0

- Error evaluation:
  - -comparison outputs / LUT
  - -Permutations of all constants and inputs
  - -Normalization of error by wires and patterns
  - -Penalization for length

Overview

Mutation	Gates Blocks	Position (block/circuit)
Cross-Over*	Segments	Experimental (unitary matrices)
Reproduction	Circuits	Best gates Best Circuits

* - for circuits having only same number of I/O

Experimental settings

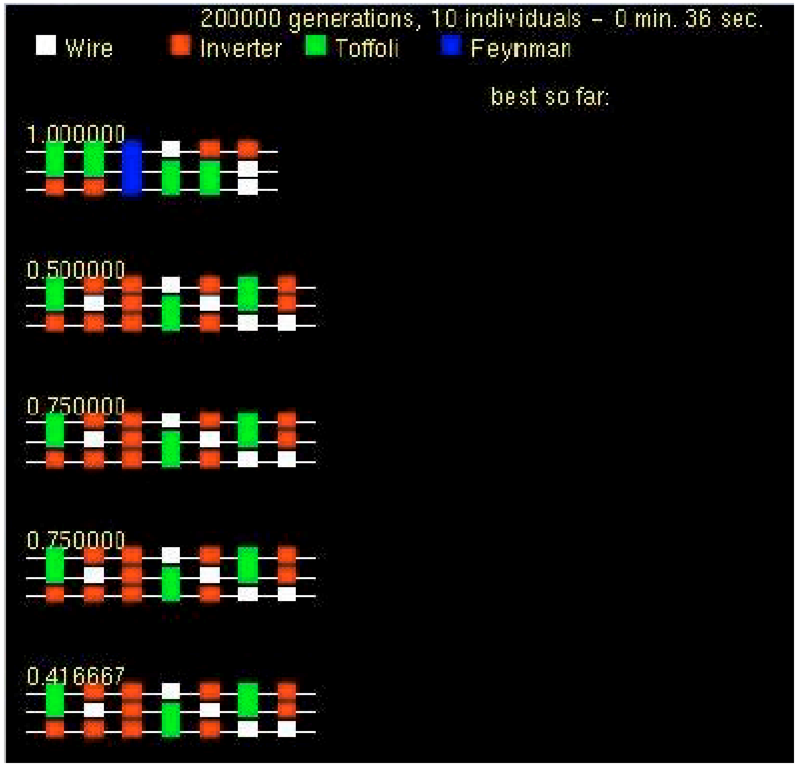
- Each input is equivalent with any other
- Evolving new circuits by recombination
- Non specific conditions
- Population 100-150
- Mutation = 0.01 –1
- Crossover = 0.3 – 0.8
- Specifications:
  - Genetic operations based on RCB > minimal element
  - The noise in these experiments is not only a mutation but an random operator on random blocks !!!

Number of wires	**Gates**
1	Wire, Inverter
2	Feynman, Swap
3	Fredkin, Toffoli
4	Margolus

Testing

-No starting set restriction

-Mutation only on blocks



Unitary gate search

# of inputs	Number of individuals	Number of generations	Real gate found	Real Time
2	10/50	10/1	*	< 1 Min
3	10/50	10/1	*	< 1 Min
4	10/50	10/1	*	< 1 Min

Random function search

Improvements

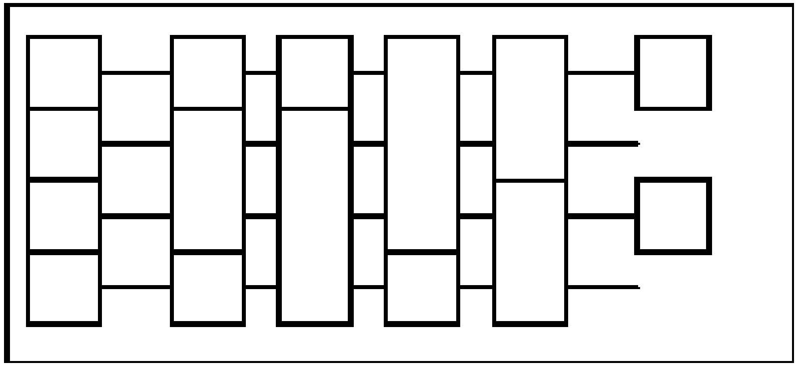
- Using min(ESOP(F⊕G)) for fitness
- Lamarckian learning
  - One genotype ⇒ multiple possibilities of phenotype
  - Using to minimize Exorcism-4

Circuit search

-Starting set restriction

-Mutation all levels (0.01 – 0.1)

Circuit/Gate	# of Gen.	R.T.	Exact/similar
Toffoli	5/1	0	*/*
Fredkin	5/1	0	*/*
Adder	?/200,000	120 sec	0/*



Conclusion

- Ideas:
  - Using GA to evolve arbitrary Reversible Circuit
  - Specific Encoding helps the evolution
  - Alternative encoding presented
- Future works:
  - Apply Lamarckian GA and other new variants of evolutionary approaches
  - Create hybrid algorithms by mixing evolutionary and logic-symbolic methods
  - Use new representations such as permutations and decision diagrams
  - Use Logic minimizer to minimize the ESOP expression of the circuit
  - **THIS IS WORK IN PROGRESS, EVERYBODY IS WELCOME TO JOIN.**
  - **Publishing**