## Reversible computation

Each classical algorithm is Boolean function

$$\{0,1\}^n \to \{0,1\},$$

which assigns each input the output value. If the output is longer (which it usually is), the algorithm is

The complexity of the algorithm, when viewed as a Boolean function, corresponds to the size of the smallest Boolean circuit that computes the function. The circuit can consist of several simple, usually one- or two-value gates. The construction of the circuit is thus a kind of *decomposition* of a Boolean function into some suitable set of simple functions.

On a similar basis, it is possible to talk about the complexity of quantum algorithms: we decompose the relevant unitary transformation into simple transformations and ask about the size of the decomposition, i.e. the number of gates used.

To do this, it is necessary to choose a suitable set of basic operators that we can use in the decomposition and with which any operator can be constructed. Such a set is called a *universal set of gates*.

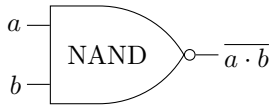
In the classical case, the AND, OR and NOT functions, which are the Boolean operators $\wedge$, $\vee$ and $\overline{\ }$, form a natural universal set of gates. Any function can be straightforwardly decomposed into it using, for example, the disjunctive normal form of the function $f$:

$$f(x_1, x_2, \ldots, x_n) = \bigvee_{f(\mathbf{z})=1} (y_1 \wedge y_2 \wedge \cdots \wedge y_n)\,,$$

where $\mathbf{z} = (z_1, z_2, \ldots, z_n)$ runs through $\{0,1\}^n$ and

$$y_i = \begin{cases} x_i, & \text{pokud } z_i = 1, \\ \overline{x_i}, & \text{pokud } z_i = 0. \end{cases}$$

However, there are also several gates that are universal in themselves, such as NAND:




Universality of NAND follows from rules $\overline{a} = \mathrm{NAND}(a, a)$ and $a \vee b = \mathrm{NAND}(\overline{a}, \overline{b})$.

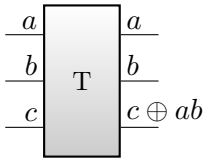
If we expect quantum circuits to be more powerful than classical ones, we should at least be able to implement quantum classical algorithms. To do this, it would be enough to implement a NAND quantum circuit. However, this is not immediately possible, because all quantum operators are reversible, while NAND is not.

This opens the issue of reversible calculation of a Boolean functions. A universal reversible function must be at least three-bit. Such a universal function is, for example, the Toffoli function defined as

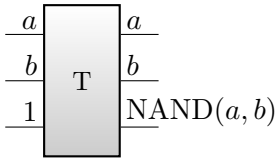
$$T : (a, b, c) \mapsto (a, b, c \oplus ab),$$

where the binary multiplication corresponds to AND and the binary addition $\oplus$ corresponds to the logical gate XOR, that is the "exclusive or". The Toffoli function

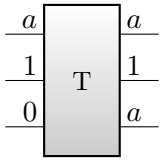
is reversible, and it an its own inverse. We will depict the corresponding gate as




the operator NAND is obtained using the Toffoli gate as follows:




For quantum computing, it is important that we can copy input basis states:

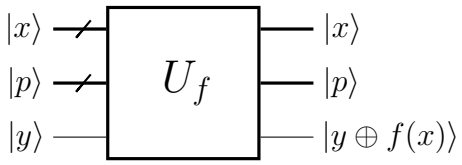



Note that this only copies the basis states. For the general state $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$ we get

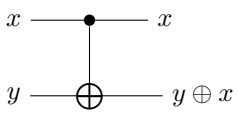
$$T|\varphi\rangle|1\rangle|0\rangle = \alpha \cdot T|010\rangle + \beta \cdot T|110\rangle = \alpha|010\rangle + \beta|111\rangle,$$

which is an entangled state, certainly not equal to $|\varphi\rangle|1\rangle|\varphi\rangle$. (The "non-cloning theorem" shows that the general state can not be copied.)

It can be seen from the figures that calculations composed of Toffoli gates will need auxiliary bits in addition to the input. We have also seen that the auxiliary bits are intertwined with the input bits, which is undesirable because we have to take them into account, for example, when estimating the measurement result. The quantum calculation of the Boolean function $f$ should correspond to the form we saw in the Deutsch-Jozsa algorithm. It should consist of an input register $|x\rangle$, which is not changed by calculation, an auxiliary register $|p\rangle$, which also does not change, and an output qubit $|y\rangle$, to which the value $f(x)$ is added. Schematically:




where the auxiliary register is naturally omitted from the picture. We will realize the result of the operation by the following circuit



which is called CNOT, or *controlled negation*, because it is actually an instruction: if $x$, negate $y$.

The desired form of calculation can now be achieved by the following steps:

- reversible quantum calculation of the function $f$ (e.g. using Toffoli gates) in the input and auxiliary registers;
- adding the result to the output qubit using the CNOT function;
- reversed calculation in the input and auxiliary register leading to the original states.

The procedure is illustrated by the following scheme calculating the function $a \vee b = \overline{(\overline{a} \wedge \overline{b})}$.