Quantum Turing machine

A **quantum Turing machine** (**QTM**), also a **universal quantum computer**, is an abstract machine used to model the effect of a quantum computer. It provides a very simple model which captures all of the power of quantum computation. Any quantum algorithm can be expressed formally as a particular quantum Turing machine. Such Turing machines were first proposed in a 1985 article written by Oxford University physicist David Deutsch suggesting quantum gates could function in a similar fashion to traditional digital computing binary logic gates.[1]

Quantum Turing machines are not always used for analyzing quantum computation; the quantum circuit is a more common model. These models are computationally equivalent.[2]

Quantum Turing machines can be related to classical and probabilistic Turing machines in a framework based on transition matrices, shown by Lance Fortnow.[3]

Iriyama, Ohya, and Volovich have developed a model of a *linear quantum Turing machine* (LQTM). This is a generalization of a classical QTM that has mixed states and that allows irreversible transition functions. These allow the representation of quantum measurements without classical outcomes.[4]

A *quantum Turing machine with postselection* was defined by Scott Aaronson, who showed that the class of polynomial time on such a machine (PostBQP) is equal to the classical complexity class PP.[5]

> **Unsolved problem in physics**:
>
> ? *Is a universal quantum computer sufficient to efficiently simulate an arbitrary physical system?*
>
> (more unsolved problems in physics)

Informal sketch

A way of understanding the quantum Turing machine (QTM) is that it generalizes the classical Turing machine (TM) in the same way that the quantum finite automaton (QFA) generalizes the deterministic finite automaton (DFA). In essence, the internal states of a classical TM are replaced by pure or mixed states in a Hilbert space; the transition function is replaced by a collection of unitary matrices that map the Hilbert space to itself.[1]

That is, a classical Turing machine is described by a 7-tuple $M = \langle Q, \Gamma, b, \Sigma, \delta, q_0, F \rangle$.

For a three-tape quantum Turing machine (one tape holding the input, a second tape holding intermediate calculation results, and a third tape holding output):

- The set of states Q is replaced by a Hilbert space.
- The tape alphabet symbols Γ are likewise replaced by a Hilbert space (usually a different Hilbert space than the set of states).
- The blank symbol $b \in \Gamma$ corresponds to the zero-vector
- The input and output symbols Σ are usually taken as a discrete set, as in the classical system; thus, neither the input nor output to a quantum machine need be a quantum system itself.
- The transition function $\delta : \Sigma \times Q \otimes \Gamma \rightarrow \Sigma \times Q \otimes \Gamma \times \{L, R\}$ is a generalization of a transition monoid, and is understood to be a collection of unitary matrices that are automorphisms of the Hilbert space Q.
- The initial state $q_0 \in Q$ may be either a mixed state or a pure state.
- The set F of *final* or *accepting states* is a subspace of the Hilbert space Q.

The above is merely a sketch of a quantum Turing machine, rather than its formal definition, as it leaves vague several important details: for example, how often a measurement is performed; see for example, the difference between a measure-once and a measure-many QFA. This question of measurement affects the way in which writes to the output tape are defined.

References

1. Deutsch, David (July 1985). "Quantum theory, the Church-Turing principle and the universal quantum computer" (https://web.archive.org/web/20081123183419/http://www.ceid.upatras.gr/tech_news/papers/quantum_theory.pdf) (PDF). *Proceedings of the Royal Society A* **400** (1818): 97–117. Bibcode:1985RSPSA.400...97D (http://adsabs.harvard.edu/abs/1985RSPSA.400...97D). doi:10.1098/rspa.1985.0070 (https://doi.org/10.1098%2Frspa.1985.0070). Archived from the original (http://www.ceid.upatras.gr/tech_news/papers/quantum_theory.pdf) (PDF) on 2008-11-23.
2. Andrew Yao (1993). *Quantum circuit complexity*. 34th Annual Symposium on Foundations of Computer Science. pp. 352–361.
3. Lance Fortnow (2003). "One Complexity Theorist's View of Quantum Computing". *Theoretical Computer Science* **292** (3): 597–610. doi:10.1016/S0304-3975(01)00377-2 (https://doi.org/10.1016%2FS0304-3975%2801%2900377-2).
4. Simon Perdrix; Philippe Jorrand (2007-04-04). "Classically-Controlled Quantum Computation". *Math. Struct. In Comp. Science*. **16** (4): 601–620. arXiv:quant-ph/0407008 (https://arxiv.org/abs/quant-ph/0407008). doi:10.1017/S096012950600538X (https://doi.org/10.1017%2FS096012950600538X) also Simon Perdrix and Philippe Jorrand (2006). "Classically-Controlled Quantum Computation" (http://dcm-workshop.org.uk./2005/dcm-draft-proceedings.pdf) (PDF). *Math. Struct. In Comp. Science* **16** (4): 601–620. doi:10.1017/S096012950600538X (https://doi.org/10.1017%2FS096012950600538X)
5. Aaronson, Scott (2005). "Quantum computing, postselection, and probabilistic polynomial-time". *Proceedings of the Royal Society A* **461** (2063): 3473–3482. arXiv:quant-ph/0412187 (https://arxiv.org/abs/quant-ph/0412187). Bibcode:2005RSPSA.461.3473A (http://adsabs.harvard.edu/abs/2005RSPSA.461.3473A). doi:10.1098/rspa.2005.1546 (https://doi.org/10.1098%2Frspa.2005.1546) Preprint available at [1] (https://arxiv.org/abs/quant-ph/0412187)

Further reading

- Satoshi Iriyama; Masanori Ohya; Igor Volovich (2004). "Generalized Quantum Turing Machine and its Application to the SAT Chaos Algorithm". arXiv:quant-ph/0405191
- Abstract of Deutsch's paper
- The quantum computer – history