12th Week

BQP, NQP, Quantum NP, and Quantum Finite Automata

Synopsis.

- Quantum Turing Machines
- Quantum Finite Automata
- BQP, NQP, FBQP, and #QP
- Quantum NP

June 25, 2018. 23:59

Course Schedule: 16 Weeks

Subject to Change

- Week 1: Basic Computation Models
- Week 2: NP-Completeness, Probabilistic and Counting Complexity Classes
- Week 3: Space Complexity and the Linear Space Hypothesis
- Week 4: Relativizations and Hierarchies
- Week 5: Structural Properties by Finite Automata
- Week 6: Stype-2 Computability, Multi-Valued Functions, and State Complexity
- Week 7: Cryptographic Concepts for Finite Automata
- Week 8: Constraint Satisfaction Problems
- Week 9: Combinatorial Optimization Problems
- Week 10: Average-Case Complexity
- Week 11: Basics of Quantum Information
- Week 12: BQP, NQP, Quantum NP, and Quantum Finite Automata
- Week 13: Quantum State Complexity and Advice
- Week 14: Quantum Cryptographic Systems
- Week 15: Quantum Interactive Proofs
- Week 16: Final Evaluation Day (no lecture)

YouTube Videos

- This lecture series is based on numerous papers of T. Yamakami. He gave conference talks (in English) and invited talks (in English), some of which were video-recorded and uploaded to YouTube.
- Use the following keywords to find a playlist of those videos.
- YouTube search keywords:

  Tomoyuki Yamakami conference invited talk playlist

Conference talk video

Main References by T. Yamakami I

- T. Yamakami. A foundation of programming a multi-tape quantum Turing machine. In Proc. of MFCS 1999, LNCS, Vol.1672, pp.430-441 (1999)
- T. Yamakami and A. C. Yao. NQP_C=co-$C_=P$. Information Processing Letters 71, 63-69 (1999)
- T. Yamakami. Quantum NP and a quantum hierarchy. In Proc. of IFIP TCS 2002, Kluwer Academic Press, Vol.96 (Track 1), pp.323-336 (2002)
- T. Yamakami. Analysis of quantum functions. International Journal of Foundations of Computer Science 14, 815-852 (2003)

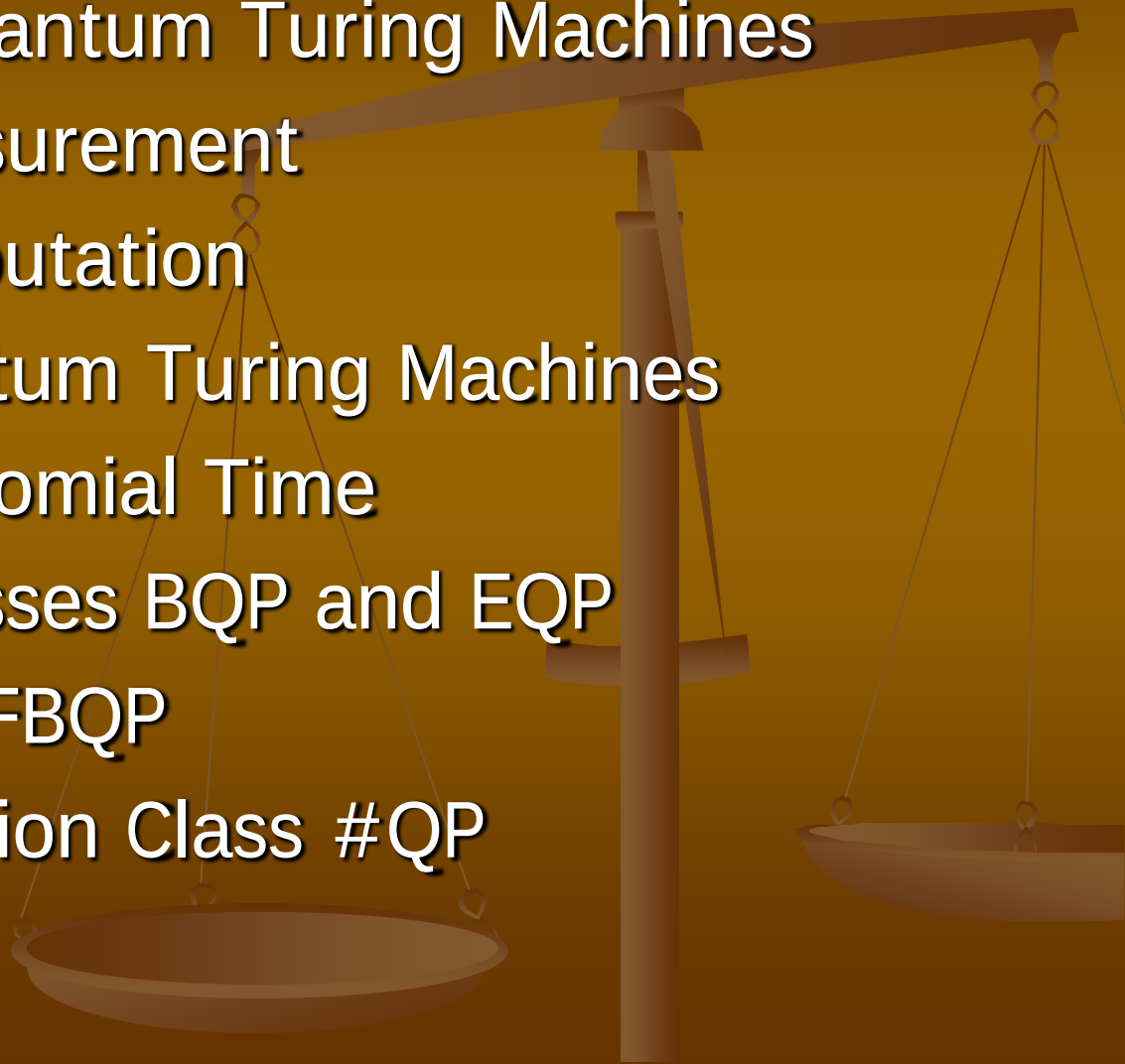
(To be continued)

Main References by T. Yamakami II

- H. Nishimura and T. Yamakami. An application of quantum finite automata to interactive proof systems. Journal of Computer and System Sciences 75, 255-269 (2009)
- Tomoyuki Yamakami. One-way reversible and quantum finite automata with advice. Information and Computation 239, 122-148 (2014)
- Tomoyuki Yamakami. Complexity bounds of constant-space quantum computation. In Proc. of DLT 2015, Lecture Notes in Computer Science, vol. 9168, pp.426-438 (2015)

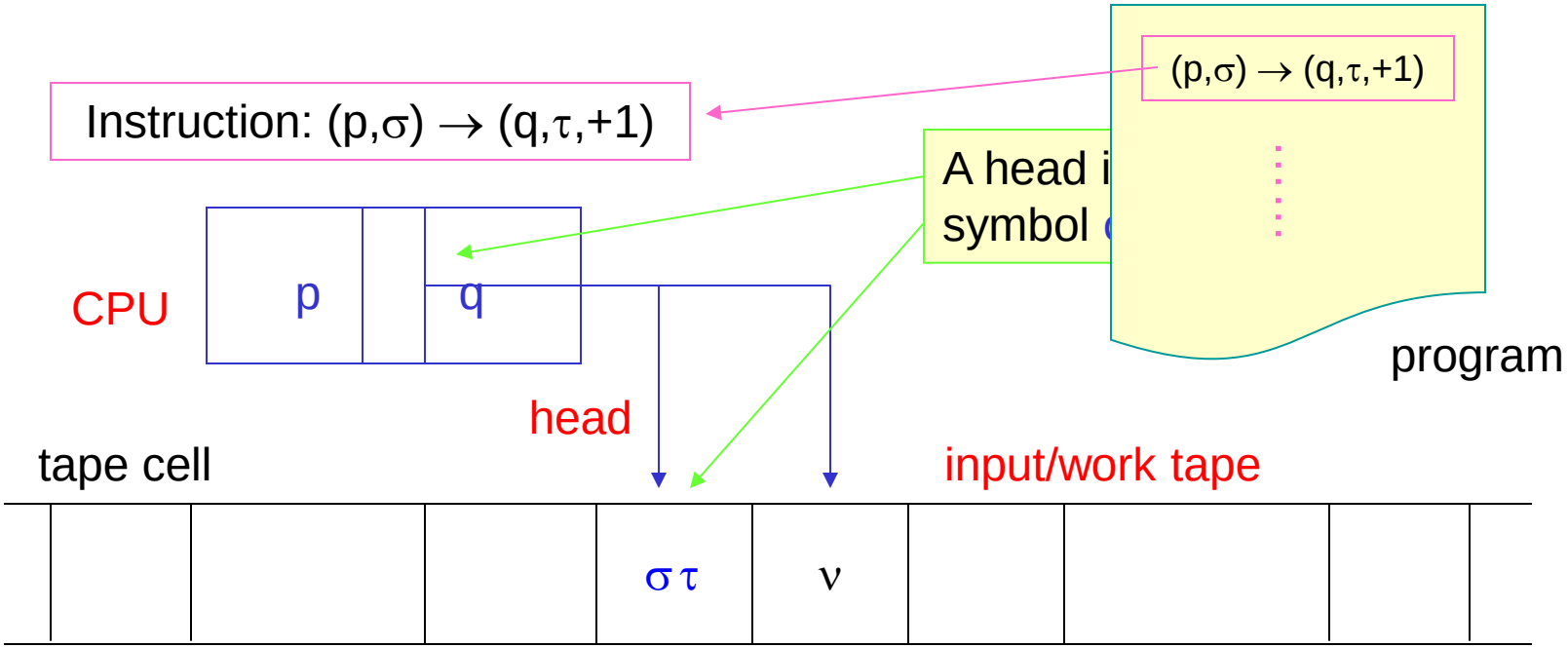
I. Quantum Polynomial-Time Computation

1. Single-Tape Quantum Turing Machines
2. Projection Measurement
3. Quantum Computation
4. Universal Quantum Turing Machines
5. Quantum Polynomial Time
6. Complexity Classes BQP and EQP
7. Function Class FBQP
8. Quantum Function Class #QP



Turing Machine (revisited)

- A Turing machine consists of tape, head, and CPU.
- An input is written on the tape.
- The machine scans a symbol on the tape, follows a program, rewrites the tape symbol and the inner state of the CPU, and then moves the head.



The Notion of Quantum Turing Machines

- The notion of quantum Turing machine has developed by many researchers.
- Benioff (1980) first considered a quantum analogue of Turing machine.
- Deutsch (1985) and Yao (1993) further developed a model of quantum Turing machines (or QTMs).
- Bernstein and Vazirani (1997) gave a modern definition to a single-tape QTMs.
- Yamakami (1999) studied multi-tape model of QTMs and presented the well-formedness condition.
- Nishimura and Ozawa (2000,2002) studied properties of multi-tape QTMs.

Single-Tape Quantum Turing Machines

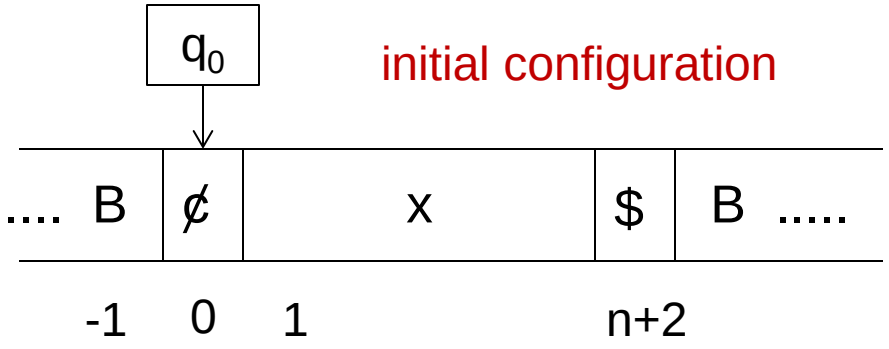
- Similar to classical Turing machines, a quantum Turing machine (QTM) consists of tape, tape head, and CPU.
- An input is written on the tape between two endmarkers.
- The QTM scans a classical symbol on the tape, follows a classical program, rewrites the tape symbol and the inner state of the CPU, and then moves the head in superpositions.
- A quantum transition function

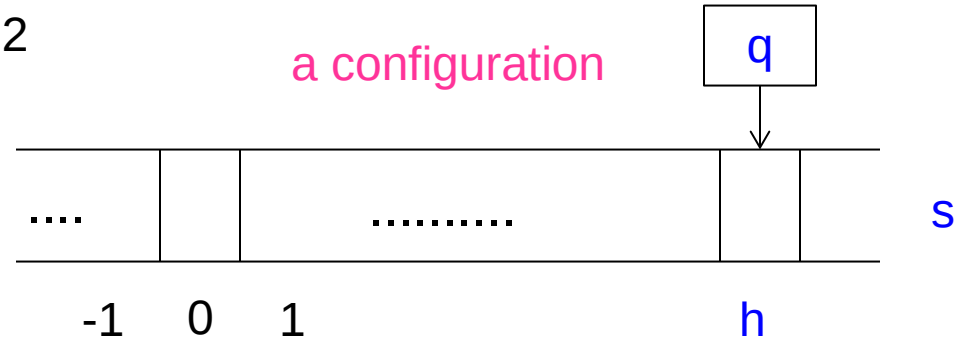
$$\delta : Q \times \breve{\Sigma} \times Q \times \breve{\Sigma} \times D \to C$$

  induces a time-evolution matrix U_δ, which must be unitary.
- NOTE: Hereafter, we use only QTMs whose time-evolution matrices are unitary.

Configurations

- Given a QTM M, its configuration is of the form (q,h,s), which means that M is in inner state q, scanning the h-th tape cell, and the content of the tape is s.
- E denotes the configuration space of M on input x, which is a vector space spanned by all configurations of M on x.





Decomposing a Configuration Space

- Recall that the configuration spaces of M on input x is:
  E = span{ c | c is any configuration of M on input x }

- E can be decomposed of three vector spaces:
  $E = E_{acc} \oplus E_{rej} \oplus E_{non}$, where

  1. E_{acc} = span{ c | c is an accepting configuration of M on x}
  2. E_{rej} = span{ c | c is a rejecting configuration of M on x}
  3. E_{non} = span{ c | c is a non-halting configuration of M on x}

Formal Definition of Single-Tape QTMs

A QTM $M = (Q,\Sigma,\{¢,\$\},\Gamma,\delta,q_0,Q_{acc},Q_{rej})$ has an input tape and a quantum transition function δ:

$$\delta : Q \times \Gamma \times Q \times \Gamma \times D \to C$$

D = { -1, 0, +1 }

$¢,\$ \in \Gamma$

s' is obtained from s by changing symbol s_h to s'_h.

- Time-evolution matrix (or operator) U_δ: E→ E

$$U_\delta \left| q,h,s \right\rangle = \sum_{(p,d)} \delta(q,s_h,p,s_h^{'}d) \left| p,h+d,s^{'} \right\rangle$$

- Unitary Requirement: U_δ is a unitary matrix.

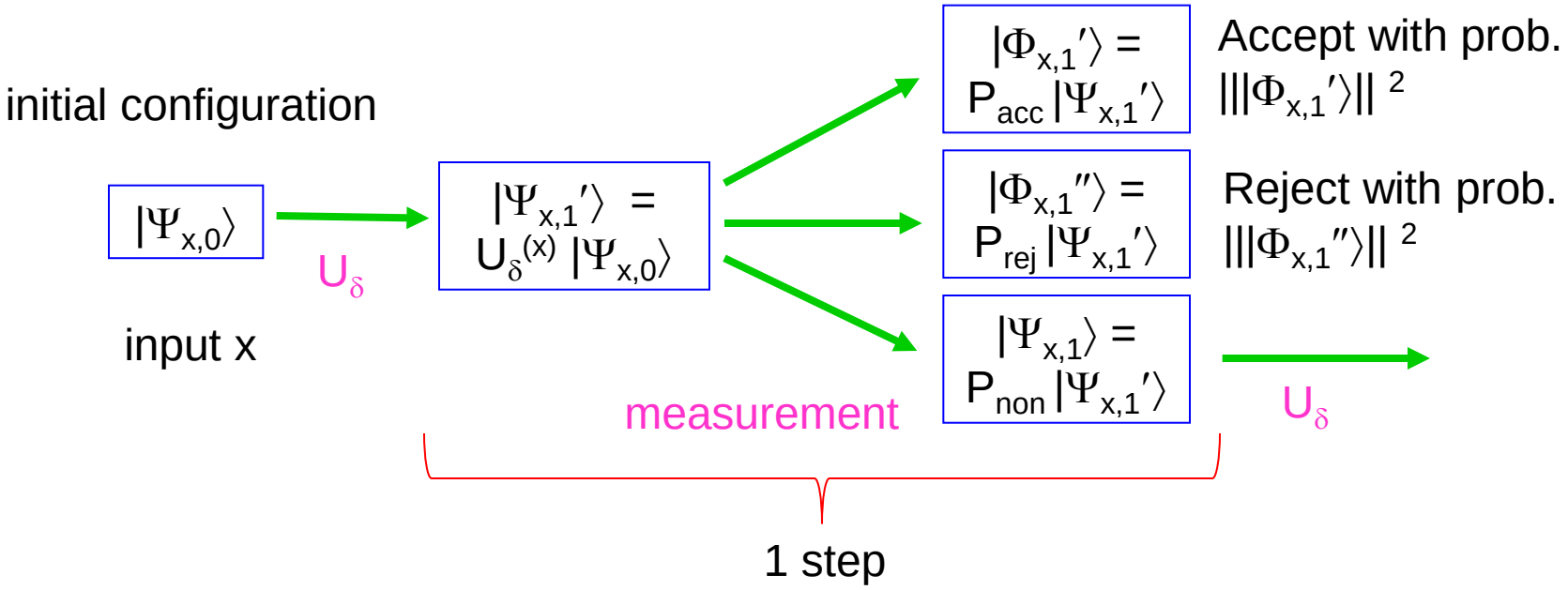
Projection Measurement

- Measurement makes quantum configuration to collapse, losing quantum information and to halt.
- Here, we consider only the following types of projection measurements.

1. P_{acc} = projection operator that projects onto the accepting configuration space E_{acc}
2. P_{rej} = projection operator that projects onto the rejecting configuration space E_{rej}
3. P_{non} = projection operator that projects onto the non-halting configuration space E_{non}

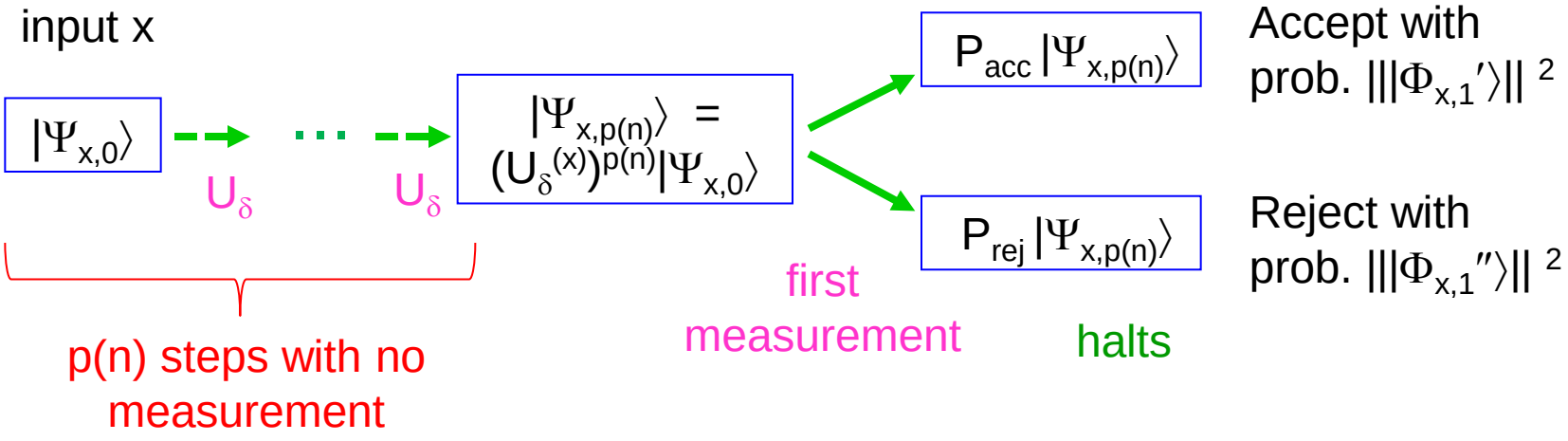
Quantum Computation

- A QTM $M = (Q, \Sigma, \{¢,\$\}, \Gamma, \delta, q_0, Q_{acc}, Q_{rej})$
- A QTM works as follows.
  - U_δ is a time-evolution operator
  - P_{acc}, P_{rej}, P_{non} are (projection) measurement operators.
  - $T_\delta = P_{non}U_\delta$ is a transition operator.



Postponing Measurement

- It is possible to postpone the measurement to the end of computation by remembering all potentially-lost information by measurement.
- Hence, we can adjust the timing of entering halting states.
- In particular, we can make all computation paths terminate at once (at the end of the whole computation).



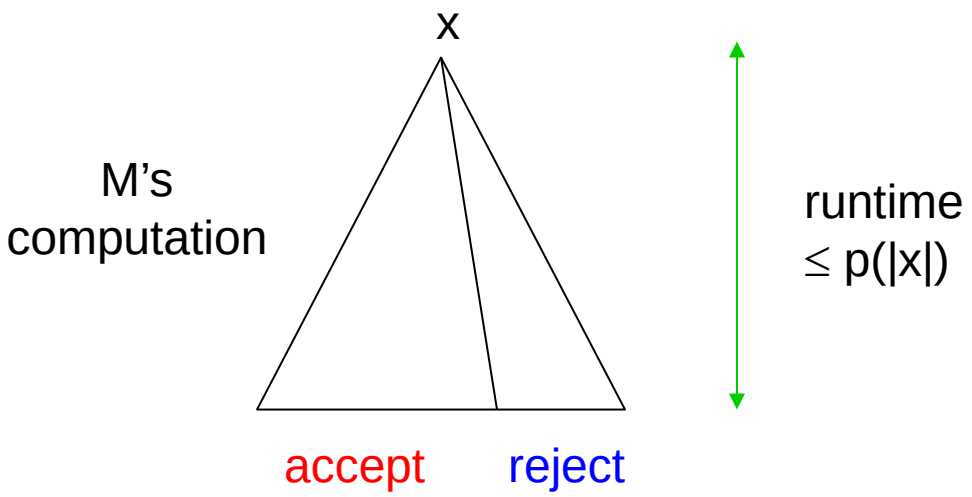
Universal Quantum Turing Machines

- Based on a classical model of Turing machine, we have discussed universal Turing machines in Week 2.
- Similarly to the fact that there is a finite set of universal gates, it is possible to restrict the use of amplitudes by QTMs.
- This helps us enumerate all QTMs with those restricted amplitudes. Such an enumerate makes us construct a so-called universal QTM.
- We say that a QTM U is universal if, for any QTM M and any input $x \in \{0,1\}^*$, U takes $\langle M,x,1^t \rangle$ as input and approximates the acceptance probability generated by M on input x within t steps.

Quantum Polynomial Time

- We say that a QTM M runs in polynomial time if there is a polynomial p such that, for all inputs x, M takes x as an input and halts in at most p(|x|) steps.

- A QTM M recognizes a language L in polynomial time if M recognizes L and M runs in polynomial time.



Complexity Class BQP

- Bernstein and Vazirani (1997) introduced the complexity class BQP (quantum polynomial time).
- Let K be any amplitude set.
- A language L is in BQP_K if there is a K-amplitude polynomial-time QTM M such that, for any input x,
  - If $x \in L$, then M accepts x with probability $\geq 2/3$,
  - If $x \notin L$, then M rejects x with probability $\geq 2/3$.
- When K = PC, we write BQP.
- (Claim) $BPP \subseteq BQP \subseteq PSPACE$

bounded-error probability

Here, "PC" denotes the set of polynomial-time approximable complex numbers

Complexity Class EQP

- Adleman, DeMarrais, and Huang (1997) introduced the complexity class EQP (exact or error-free quantum polynomial time).

- A language L is in EQP_K if there is a polynomial-time K-amplitude QTM M such that, for any input x,
  - If $x \in L$, then M accepts x with probability 1,
  - If $x \notin L$, then M rejects x with probability 1.

error-free

- (Claim) $P \subseteq EQP_K \subseteq BQP_K$
- When K = PC, we omit K and write EQP instead of EQP_{PC}.

Power of Amplitudes

- Adleman, DeMarrais, and Huang (1997) proved the following statements.

- (Claim) $BQP_Q = BQP_{PC}$.
- (Claim) BQP_C contains non-recursive languages.
- (Claim) $BQP_{PC} \neq BQP_C$. (From the above results.)

- (Claim) $EQP_C = EQP_{PC}$.
- (Claim) $EQP_C \neq BQP_C$. (From the above result.)

Function Class FBQP I

- Recall from Week 1 the function class FP, each element of which can be computed by a certain DTM with a write-only output tape in polynomial time.
- Similarly, to compute functions, we equip each QTM with a write-only output tape.
- Let K be any amplitude set.
- A function $f : \Sigma^* \to \Sigma^*$ (where Σ is an alphabet) is in $\mathrm{FBQP}_K \Leftrightarrow$ there are a polynomial p and a K-amplitude QTM M that, on each input $x \in \Sigma^*$, M produces f(x) on its output tape and halts with probability $\geq 3/4$ in time at most p(|x|).
- When K = PC, we omit K and write FBQP.

Function Class FBQP II

- Recall FPSPACE from Week 3.
- (Claim) $FP \subseteq FBQP \subseteq FPSPACE$
- Proposition: [Yamakami (2002)]
  $FBQP = FP^{BQP}$.

Quantum Function Class #QP I

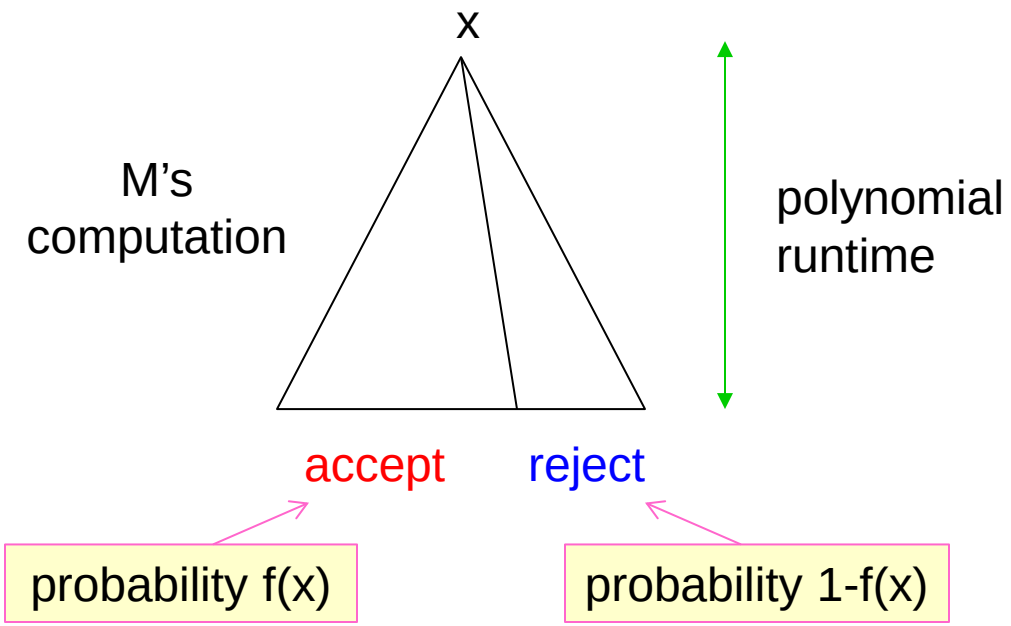
- Yamakami (2003) studied special quantum functions, which output acceptance probabilities of QTMs.
- Let K be any amplitude set.
- A function $f : \Sigma^* \to R$ (where Σ is an alphabet) is in $\#QP_K$ $\Leftrightarrow$ there exits a K-amplitude polynomial-time QTM M that, on each input $x \in \Sigma^*$, M accepts x with probability exactly f(x).
- When K = PC, we write #QP instead of $\#QP_{PC}$.
- NOTE: This notion of "quantum function" is quite different from the one that we will discuss in Week 14.

Quantum Function Class #QP II

- In other words,
  - $f \in \#QP \Leftrightarrow \exists$ M: polynomial-time QTM $\forall x$

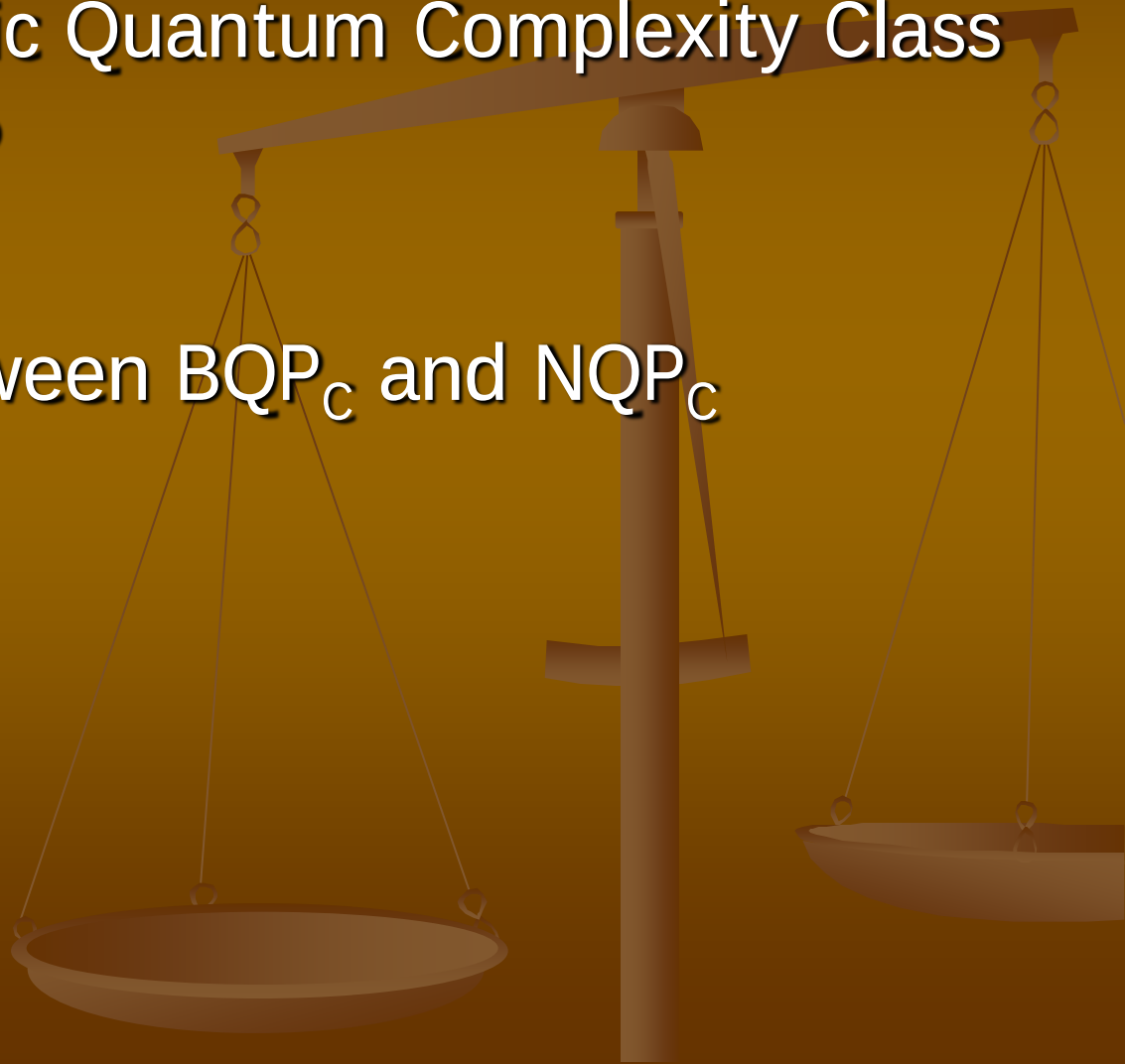
    $f(x)$ = the probability that M accepts x

    $= \text{Prob}_M[M(x) = 1]$



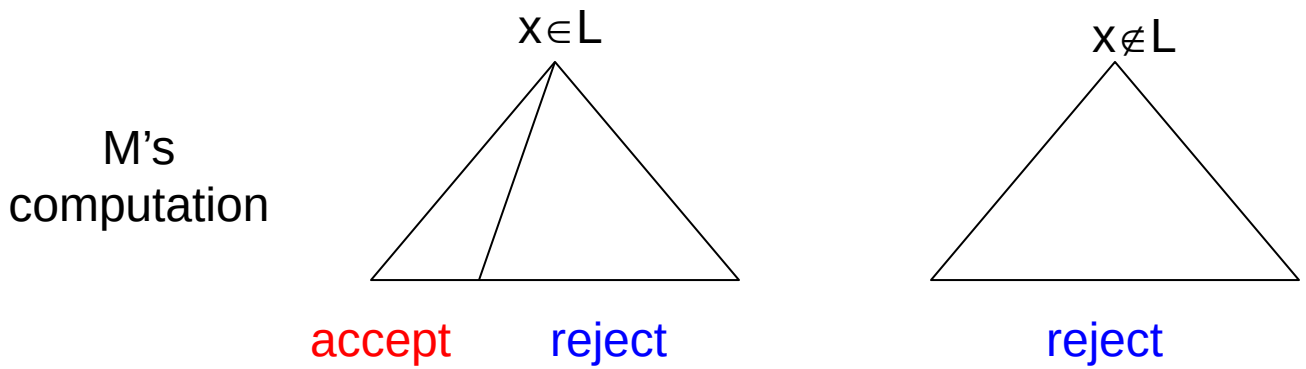
II. Characterization of NQP

1. Nondeterministic Quantum Complexity Class
2. $C_{=}P$ and co-$C_{=}P$
3. NQP_C = co-$C_{=}P$
4. Separation between BQP_C and NQP_C
5. Proof Sketch



Nondeterministic Quantum Complexity Class I

- The quantum complexity class NQP, which is a natural extension of NP, was introduced by Adleman, DeMarrais, and Huang (1997).
- Let $K \subseteq C$ be an amplitude set.
- A language L is in NQP_K if there is a polynomial-time K-amplitude QTM M such that, for every input x,
  - If $x \in L$, then M accepts x with probability >0, and
  - If $x \notin L$, then M accepts x with probability =0.



Nondeterministic Quantum Complexity Class II

- There is another way to define NQP_K using $\#QP_K$, .
- Here, we re-define NQP_K as follows.
- (Claim) $L \in NQP_K \Leftrightarrow$ there exits a function $f \in \#QP_K$ s.t., for any x, $x \in L \leftrightarrow f(x) > 0$.

- (Claim) $NQP_Q \subseteq NQP_A \subseteq NQP_C$ (since $Q \subseteq A \subseteq C$)
- (Claim) $NP \subseteq NQP_Q \subseteq PSPACE$ [Adleman-DeMarrais-Huang (1997)].

In Week 15, NQP will appear again.

$C_{=}P$ and co-$C_{=}P$ (revisited)

- Recall from Week 2 the counting complexity class $C_{=}P$ and its complement class co-$C_{=}P$.
- A language L is in $C_{=}P$ if there exists a polynomial-time probabilistic Turing machine M such that, for every x,
  - If $x \in L$, then M accepts x with probability = 1/2,
  - If $x \notin L$, then M accepts x with probability $\neq$ 1/2.
- co-$C_{=}P$ = { L^c | L is in $C_{=}P$ }
- (Claim) $P \subseteq C_{=}P \cap \text{co-}C_{=}P$ and $NP \subseteq \text{co-}C_{=}P \subseteq PP$

$NQP_C = co\text{-}C_=P$

- What is the computational complexity of NQP_K?
- (Claim) $NQP_A \subseteq PP$ [Adleman- DeMarrais-Huang (1997)]
- (Claim) $NQP_Q \subseteq co\text{-}C_=P$ [Fortnow-Rogers (1998)]
- (Claim) $NQP_A = co\text{-}C_=P$ [Fenner-Green-Homer-Pruim (1998)]
- This last result was significantly improved as follows.
- Theorem: [Yamakami-Yao (1999)]

  $NQP_K = co\text{-}C_=P$ for any set K with $Q \subseteq K \subseteq C$.
- This theorem intuitively indicates that nondeterministic quantum computation can be simulated by classical counting computation.

Separation between BQP_C and NQP_C

- Corollary: [Yamakami-Yao (1999)]
  $BQP_C \neq NQP_C$.

❑ Proof: As seen before, Adleman et. al. (1997) demonstrated that BQP_C contains non-recursive languages. By contrast, NQP_C is recursive.

- For the theorem, it suffices to show the following.
- (Claim) $\text{co-}C_=P \subseteq NQP_T$ ← T = { 0, ±3/5, ±4/5, ±1 }
- (Claim) $NQP_C \subseteq \text{co-}C_=P$
- (*) In the next slides, we will give a sketch of the proof of $\text{co-}C_=P \subseteq NQP_T$.

Proof Sketch I

- Proof Sketch for co-$C_{=}P \subseteq NQP_T$:

- Let $S \in$ co-$C_{=}P$. Take an $f \in$ GapP s.t. for all x, $x \in S \leftrightarrow f(x) \neq 0$. Choose a poly-time computable predicate R and a polynomial p s.t., for all x,

$$f(x) = \left|\left\{y \in \{0,1\}^* \mid R(x,y) = 1\right\}\right| - \left|\left\{y \in \{0,1\}^* \mid R(x,y) = 0\right\}\right|$$

- For a,b$\in${0,1,2,3}, we define a new operator H[a,b|α] as

$$H[a,b \mid \alpha] = \sum_{y \in \{a,b\}} \sum_{u \in \{a,b\}} (-1)^{[y=u=b]} \alpha^{[y=z]} (1-\alpha)^{[y \neq u]}$$

- For simplicity, we define H = H[0,1|4/5], J = H[0,1|3/5], and K= H[0,2|3/5] + H[1,3|4/5].

Proof Sketch II

- Consider the following QTM M:
  1. Take an input x (|x|=n).
  2. Generate $|\varphi_0\rangle = |0^{p(n)}\rangle|0\rangle$.
  3. Apply $H^{p(n)}\otimes I$ to $|\varphi_0\rangle$.
  4. Change the last qubit $|0\rangle$ to $|R(x,y)\rangle$.
  5. Apply $J^{p(n)}\otimes(JK)$ and obtain $|\varphi\rangle$.
  6. Apply observable $|\varphi_1\rangle=|0^{p(n)}\rangle|1\rangle$ and we obtain

     $$\langle\varphi_1|\varphi\rangle = -\varepsilon^{p(n)+1} f(x).$$

- It is easy to see that this QTM M satisfies:

  $x \in S \leftrightarrow$ M accepts x with probability > 0.
- Thus, S is in NQP_T.

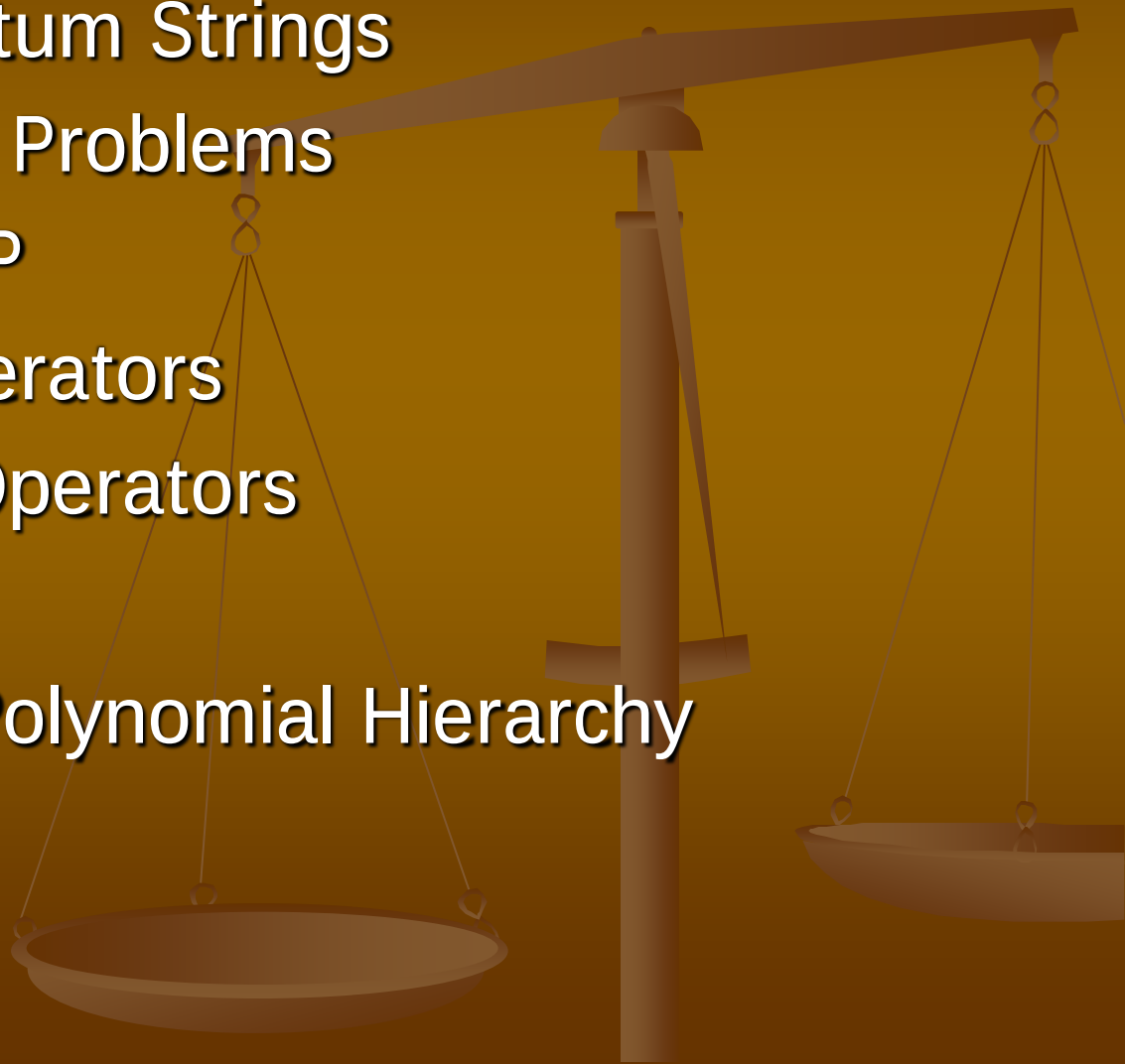
QED

Open Problems

- Prove or disprove the following statements.
  1. BQP = NQP.
  2. NP = NQP.
  3. NQP = co-NQP.

III. Quantum NP

1. Tuples of Quantum Strings
2. Partial Decision Problems
3. *#QP and *BQP
4. *∃- and *∀-Operators
5. *∃^{Q}- and *∀^{Q}-Operators
6. Quantum NP
7. The Quantum Polynomial Hierarchy



Tuples of Quantum Strings

- Let n be any number in N.
- Recall from Week 11 the space:

  H_n = Hilbert space of dimension n

$$H_\infty = \bigcup_{n \geq 1} H_{2^n}$$

- Consider a tuple $|\varphi^{vec}\rangle = (|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle)$ of qustrings.
- Let $\ell(|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle) = \Sigma_{i=1}^{m} \ell(|\varphi_i\rangle)$. (total length)
- Φ^m_n = the set of all m-tuples $|\varphi^{vec}\rangle = (|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle)$ of qustrings of length n

$$\Phi^m_\infty = \bigcup_{n \geq 1} \Phi^m_n \quad \text{and} \quad \Phi^*_\infty = \bigcup_{m \geq 1} \Phi^m_\infty$$

Partial Decision Problems

- We introduce the notion of partial decision problems.
- A partial decision problem is a pair (A,B) of sets with $A \cap B = \emptyset$. Intuitively speaking, A consists of YES instances and B consists of NO instances. This problem is also called a promise problem.
- When $A \cup B = \Sigma^*$, we call (A,B) total. In this case, we obtain $B = \Sigma^* - A$.

*#QP

- Earlier, we have discussed #QP as the class of all real-valued functions computing acceptance probabilities of polynomial-time QTMs.
- We want to extend this function class #QP as follows.
- Let f be a quantum function from $\in\Phi_\infty^*$ to [0,1].
- f $\in$ *#QP $\Leftrightarrow$ there is a polynomial-time QTM M s.t., for all $|\varphi^{vec}\rangle = (|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle) \in\Phi_\infty^*$, M starts with input $|\varphi^{vec}\rangle$ on $\ell(|\varphi^{vec}\rangle)$-tapes and accepts with probability exactly $f(|\varphi^{vec}\rangle)$.

*BQP

- We have already discussed NQP, which was introduced as nondeterministic quantum polynomial-time complexity class.
- From a different approach, we define a quantum analogue of NP, called quantum NP.
- Let a and b be two functions from N to [0,1] such that $a(n)+b(n)=1$ for all $n \in N$.

- $(A,B) \in$ *BQP(a,b) $\Leftrightarrow$ there is a function $f \in$ *#QP s.t., for all $|\varphi^{vec}\rangle = (|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle) \in \Phi_\infty^*$,
  - $|\varphi^{vec}\rangle \in A \rightarrow f(|\varphi^{vec}\rangle) \geq a(\ell(|\varphi^{vec}\rangle))$, and
  - $|\varphi^{vec}\rangle \in B \rightarrow f(|\varphi^{vec}\rangle) \leq b(\ell(|\varphi^{vec}\rangle))$.

- We simply write *BQP for *BQP(3/4,1/4).

*∃- and *∀-Operators

- Let *C be any class of partial decision problems.
- Let (A,B) be any partial decision problem.

- $(A,B) \in {}^*\exists \cdot {}^*C \Leftrightarrow$ there is a partial decision problem $(C,D) \in {}^*C$ s.t., for all $|\varphi^{vec}\rangle = (|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle) \in \Phi_\infty{}^*$,
  - $|\varphi^{vec}\rangle \in A \rightarrow \exists x \in \Sigma^{p(\ell(|\varphi vec\rangle))} [\ (|x\rangle,|\varphi^{vec}\rangle) \in C\ ]$, and
  - $|\varphi^{vec}\rangle \in B \rightarrow \forall x \in \Sigma^{p(\ell(|\varphi vec\rangle))} [\ (|x\rangle,|\varphi^{vec}\rangle) \in D\ ]$.

- $(A,B) \in {}^*\forall \cdot {}^*C \Leftrightarrow$ there is a partial decision problem $(C,D) \in {}^*C$ s.t., for all $|\varphi^{vec}\rangle = (|\varphi_1\rangle,|\varphi_2\rangle,...,|\varphi_m\rangle) \in \Phi_\infty{}^*$,
  - $|\varphi^{vec}\rangle \in A \rightarrow \forall x \in \Sigma^{p(\ell(|\varphi vec\rangle))} [\ (|x\rangle,|\varphi^{vec}\rangle) \in C\ ]$, and
  - $|\varphi^{vec}\rangle \in B \rightarrow \exists x \in \Sigma^{p(\ell(|\varphi vec\rangle))} [\ (|x\rangle,|\varphi^{vec}\rangle) \in D\ ]$.

$*\exists^{Q}$- and $*\forall^{Q}$-Operators

- Let *C be any class of partial decision problems.
- Let (A,B) be any partial decision problem.
- $(A,B) \in *\exists^{Q} \cdot *C \Leftrightarrow$ there is a partial decision problem $(C,D) \in *C$ s.t., for all $|\varphi^{vec}\rangle \in \Phi_{\infty}^{*}$,
  - $|\varphi^{vec}\rangle \in A \rightarrow \exists |\psi\rangle \in \Phi_{p(\ell(|\varphi vec\rangle))}^{*}$ [$(|\psi\rangle,|\varphi^{vec}\rangle) \in C$], and
  - $|\varphi^{vec}\rangle \in B \rightarrow \forall |\psi\rangle \in \Phi_{p(\ell(|\varphi vec\rangle))}^{*}$ [$(|\psi\rangle,|\varphi^{vec}\rangle) \in C$].
- $(A,B) \in *\forall^{Q} \cdot *C \Leftrightarrow$ there is a partial decision problem $(C,D) \in *C$ s.t., for all $|\varphi^{vec}\rangle \in \Phi_{\infty}^{*}$,
  - $|\varphi^{vec}\rangle \in A \rightarrow \forall |\psi\rangle \in \Phi_{p(\ell(|\varphi vec\rangle))}^{*}$ [$(|\psi\rangle,|\varphi^{vec}\rangle) \in C$], and
  - $|\varphi^{vec}\rangle \in B \rightarrow \exists |\psi\rangle \in \Phi_{p(\ell(|\varphi vec\rangle))}^{*}$ [$(|\psi\rangle,|\varphi^{vec}\rangle) \in C$].

Quantum NP

- Using *BQP(a,b) and *BQP, we define “quantum NP”, denoted by $^*\Sigma_1^{QP}(a,b)$ and $^*\Sigma_1^{QP}$ as follows.

$$^*\Sigma_{1,m}^{QP}(a,b) = {}^*\exists_m^Q \cdot {}^*BQP(a,b)$$

$$^*\Sigma_1^{QP}(a,b) = \bigcup_{m\geq 1} {}^*\Sigma_{1,m}^{QP}(a,b)$$

$$^*\Sigma_1^{QP} = {}^*\Sigma_1^{QP}\left(3/4, 1/4\right)$$

- NOTE: The “Quantum NP” makes it possible to introduce a hierarchy, similar to the polynomial(-time) hierarchy given in Week 4.

The Quantum Polynomial Hierarchy I

- The quantum polynomial hierarchy (or QP hierarchy) consists of partial decision problems such that

$$^{*}\Sigma_{0,m}^{QP}(a,b) = {}^{*}\Pi_{0,m}^{QP}(a,b) = {}^{*}BQP(a,b)$$

$$^{*}\Sigma_{k+1,m}^{QP}(a,b) = {}^{*}\exists_{m}^{Q} \cdot {}^{*}\Pi_{k,m}^{QP}(a,b)$$

$$^{*}\Pi_{k+1,m}^{QP}(a,b) = {}^{*}\forall_{m}^{Q} \cdot {}^{*}\Sigma_{k,m}^{QP}(a,b)$$

$$^{*}\Sigma_{k}^{QP}(a,b) = \bigcup_{m\geq 1} {}^{*}\Sigma_{k,m}^{QP}(a,b)$$

$$^{*}\Pi_{k}^{QP}(a,b) = \bigcup_{m\geq 1} {}^{*}\Pi_{k,m}^{QP}(a,b)$$

$$^{*}QPH_{m}(a,b) = \bigcup_{k\geq 0} \left({}^{*}\Sigma_{k,m}^{QP}(a,b) \cup {}^{*}\Pi_{k,m}^{QP}(a,b) \right)$$

$$^{*}QPH(a,b) = \bigcup_{k\geq 0} {}^{*}QPH_{m}(a,b)$$

$$^{*}\Sigma_{k}^{QP} = {}^{*}\Sigma_{k}^{QP}\left(3/4, 1/4\right)$$

$$^{*}\Pi_{k}^{QP} = {}^{*}\Pi_{k}^{QP}\left(3/4, 1/4\right)$$

$$^{*}QPH = {}^{*}QPH\left(3/4, 1/4\right)$$

The Quantum Polynomial Hierarchy II

- The total classical parts of $*\Sigma_k^{QP}(a,b)$, $*\Pi_k^{QP}(a,b)$, and *QPH(a,b) are denoted by

$$\Sigma_k^{QP}(a,b),\ \ \Pi_k^{QP}(a,b),\ \ \mathrm{QPH}(a,b)$$

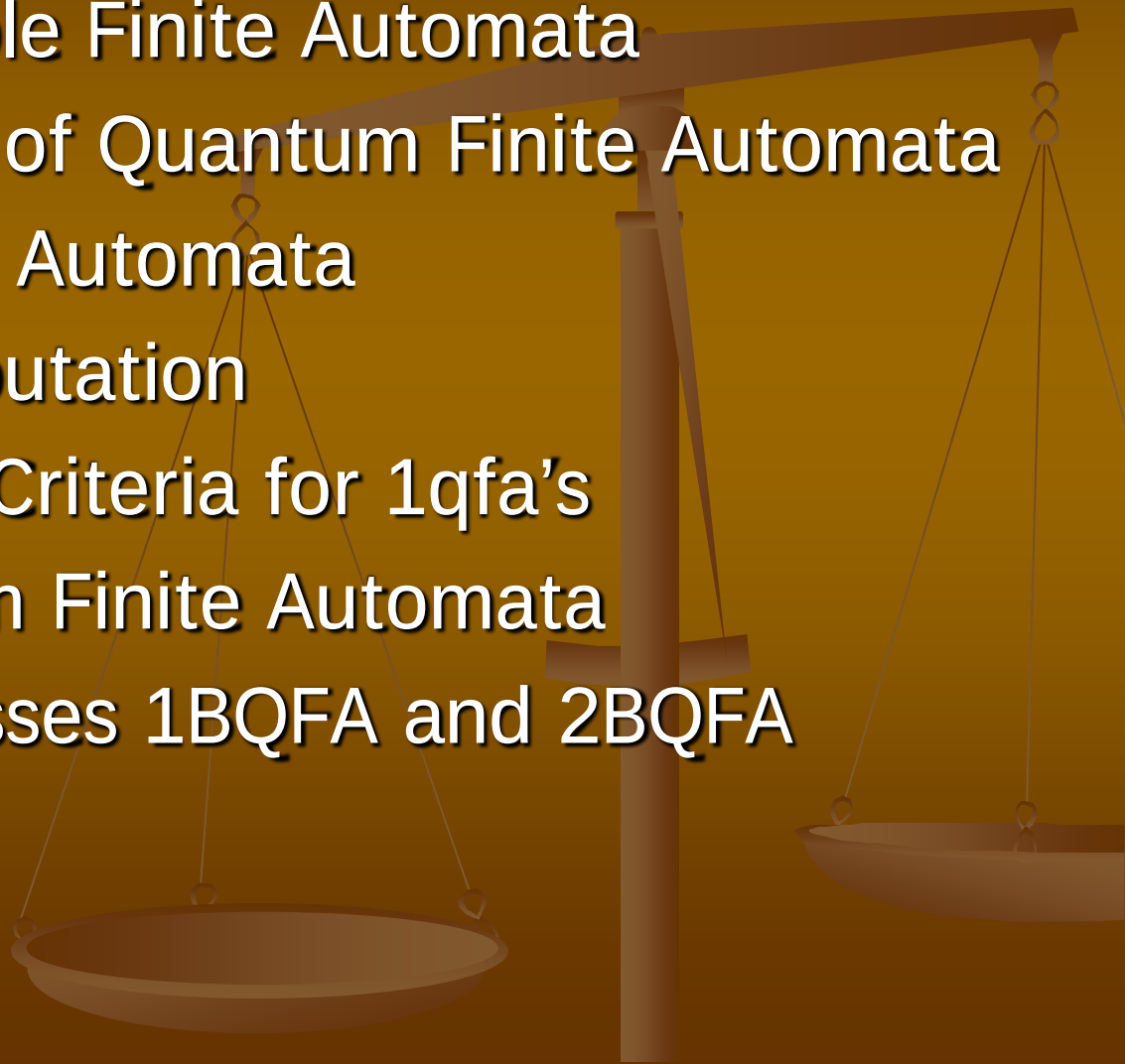
- Let Σ_k^{exp} be th k-th level of the exponential hierarchy, defined similarly to the polynomial hierarchy, using "exponential time" instead of "polynomial time."

- Proposition: [Yamakami (2002)]

  For any $k \geq 1$, $\Sigma_k^p \subseteq *\Sigma_k^{QP} \subseteq \Sigma_k^{exp}$ and $PH \subseteq QPH \subseteq EXPH$.

IV. Basics of Quantum Finite Automata

1. 1-Way Reversible Finite Automata
2. Various Models of Quantum Finite Automata
3. Quantum Finite Automata
4. Quantum Computation
5. Bounded-Error Criteria for 1qfa's
6. 2-Way Quantum Finite Automata
7. Complexity Classes 1BQFA and 2BQFA



1-Way Reversible Finite Automata I

- A 1-way (deterministic) reversible finite automaton (1rfa) has a read-only input tape and a transition function.

$M = (Q, \Sigma, \{ ¢, \$ \}, \delta, q_0, Q_{acc}, Q_{rej})$

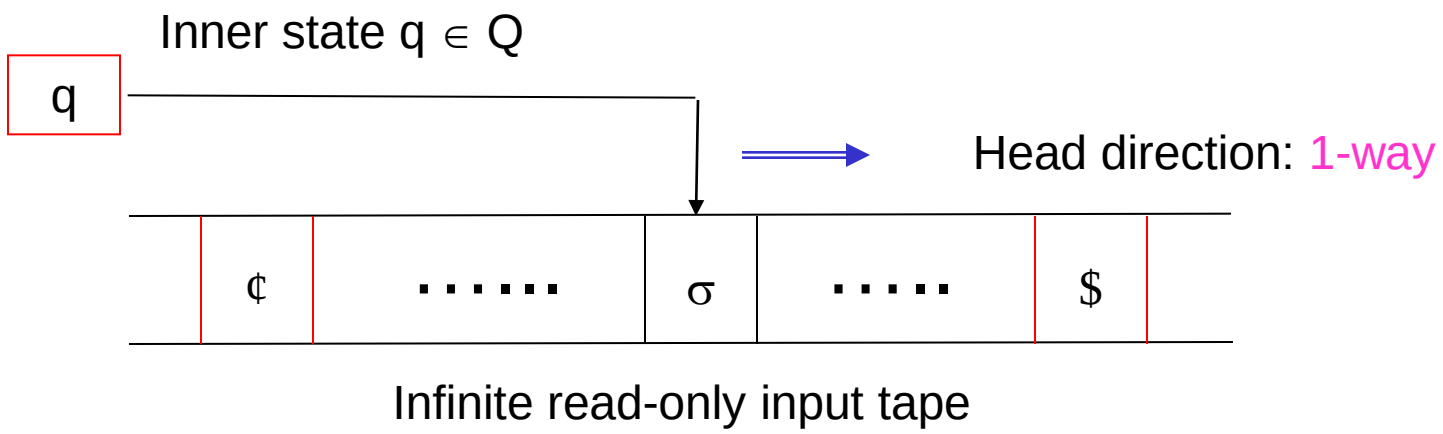
$\breve{\Sigma} = \Sigma \cup \{ ¢, \$ \}$

Σ = input alphabet

$Q_{halt} = Q_{acc} \cup Q_{rej} \subseteq Q$

δ : a transition function

$$\delta : Q \times \breve{\Sigma} \to Q \text{ or } \delta : Q \times \breve{\Sigma} \times Q \to \{0,1\}$$



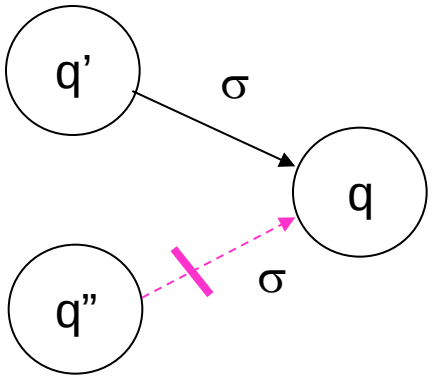
1-Way Reversible Finite Automata II

- Each 1rfa must satisfy the reversibility condition

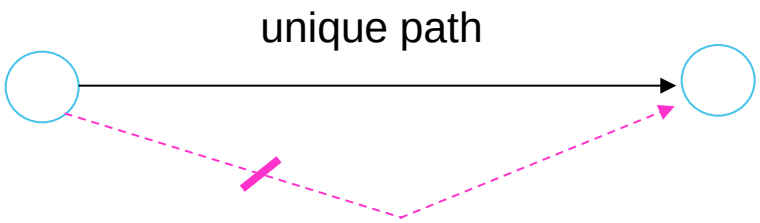
$M = (Q,\Sigma,\delta,q_0,Q_{acc},Q_{rej})$ Σ = input alphabet $\breve{\Sigma} = \Sigma \cup \{ ¢, \$ \}$

Reversibility condition:

$\forall q \in Q \ \forall \sigma \in \Sigma \ \exists$ at most one $q' \in Q$ s.t. $\delta(q',\sigma) = q$.



Property: If there is a computation path from q_0 to $q \in Q_{acc}$ (or Q_{rej}), such a path should be unique.

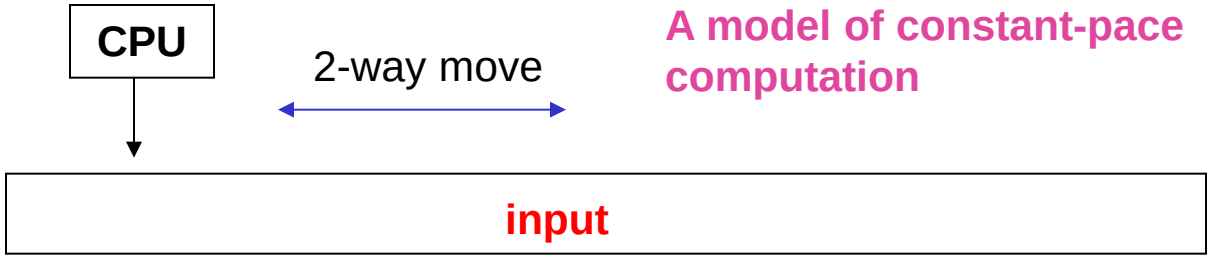


Various Models of Quantum Finite Automata

- Moore and Crutchfield (2000) introduced measure-once 1-way quantum finite automata, which conduct measurement only at the end of computations.
- Kondacs and Watrous (1997) studied (measure-many) 1-way quantum finite automata and 2-way quantum finite automata, in which machines make measurements at every step. (KWQFAs)
- Ambainis, Beaudry, Golovkins, Ķikusts, Mercer, and Thérien (2006) proposed a Latvian quantum finite automata. (LaQFAs)
- Nayak (1999) combined KWQFAs and LaQFAs.
- Hirvensalo (2010) and Yakaryılmaz and Say (2011) considered general quantum finite automata.

Quantum Finite Automata

- Hereafter, we will use the models of (measure-many) 1-way quantum finite automata (1qfa) and 2-way quantum finite automata (2qfa).
- 1qfa's and 2qfa's are considered as constant-space quantum machines.
- A configuration of a qfa is a superposition of all possible classical information about state, head position, and tape content.



A model of constant-pace computation

1-Way Quantum Finite Automata

- A (measure-many) one-way quantum finite automaton (or 1qfa) M:

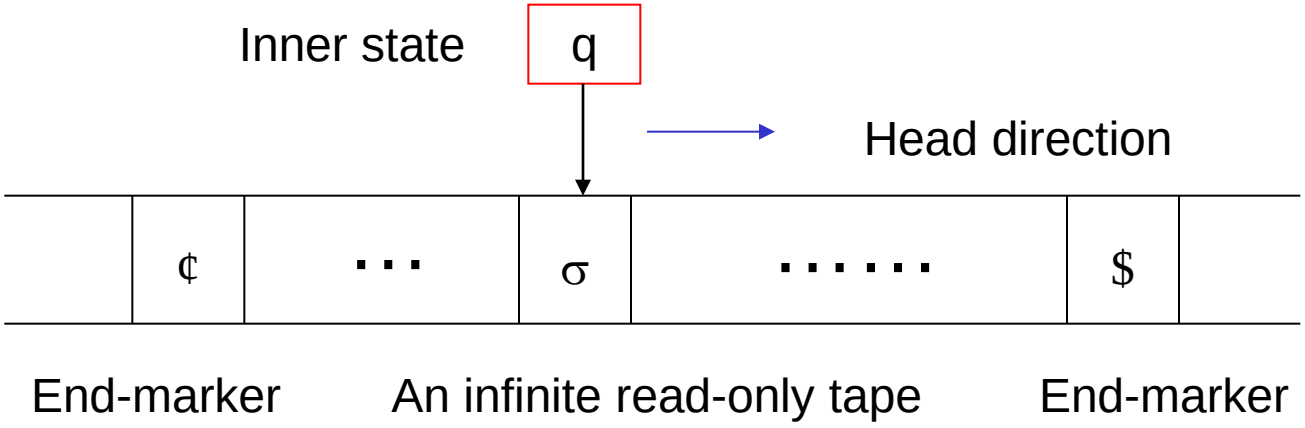
$M = (Q,\Sigma,\{¢,\$\},\delta,q_0,Q_{acc},Q_{rej})$ Σ = input alphabet

Time-evolution operator U_σ: a unitary matrix over $E_Q = \text{span}\{ |q\rangle \mid q \in Q\}$

P_{acc} : a projection onto space $E_{acc} = \text{span}\{ |q\rangle \mid q \in Q_{acc}\}$

P_{rej} : a projection onto space $E_{rej} = \text{span}\{ |q\rangle \mid q \in Q_{rej}\}$

Transition operator $T_\sigma = P_{non}U_\sigma$, where $P_{non} = I - (P_{acc} + P_{rej})$



Quantum Transition Functions

A 1qfa M = $(Q,\Sigma,\{¢,\$\},\delta,q_0,Q_{acc},Q_{rej})$ has a read-only input tape and a special quantum transition function δ:

$$\delta : Q \times \breve{\Sigma} \times Q \to C$$

$\breve{\Sigma} = \Sigma \cup \{ ¢, \$ \}$

- Time-evolution matrix:

$$U_\sigma |q\rangle = \sum_{p \in Q} \delta(q,\sigma,p) |p\rangle$$

- Unitary Requirement: $U_\delta^{(x)}$ is a unitary matrix.

U is unitary $\Leftrightarrow$ $U(U^*)^T = (U^*)^T U = I$

Another Way to Describe 1qfa's

- In the case of 1qfa's, a configuration is just an inner state.
- Therefore, instead of using δ, we can use a set of time-evolution operator U_σ ($\sigma \in \Sigma \cup \{ ¢, \$ \}$), where

$$U_\sigma |q\rangle = \sum_{p \in Q} \delta(q, \sigma, p) |p\rangle$$

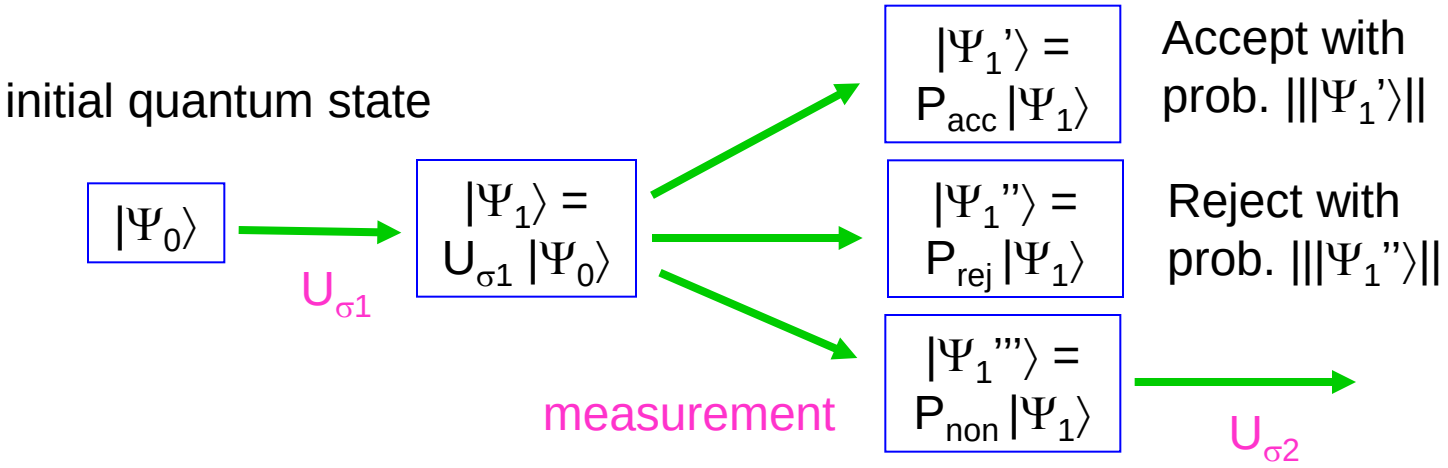
  to express the transition of 1qfa's.
- This makes us define a 1qfa M as

$$M = (Q, \Sigma, \{U_\sigma\}_\sigma, q_0, Q_{acc}, Q_{rej}).$$

Time-evolution operators

Quantum Computation I

- A 1qfa works as follows.
- Let $M = (Q, \Sigma, \{U_\sigma\}_\sigma, q_0, Q_{acc}, Q_{rej})$ be a 1qfa with
  - U_σ is a time-evolution operator
  - P_{acc}, P_{rej}, P_{non} are (projection) measurement operators.
  - $T_\sigma = P_{non} U_\sigma$ is a transition operator.
  - $T_x = T_{\sigma n}\, T_{\sigma(n-1)} \ldots\ldots T_{\sigma 2}\, T_{\sigma 1}$ if $x = \sigma_1\sigma_2\ldots\sigma_n$



Quantum Computation II

- A computation of a 1qfa M is defined as follows.

$M = (Q, \Sigma, \{U_\sigma\}_\sigma, q_0, Q_{acc}, Q_{rej})$: 1qfa Σ = input alphabet

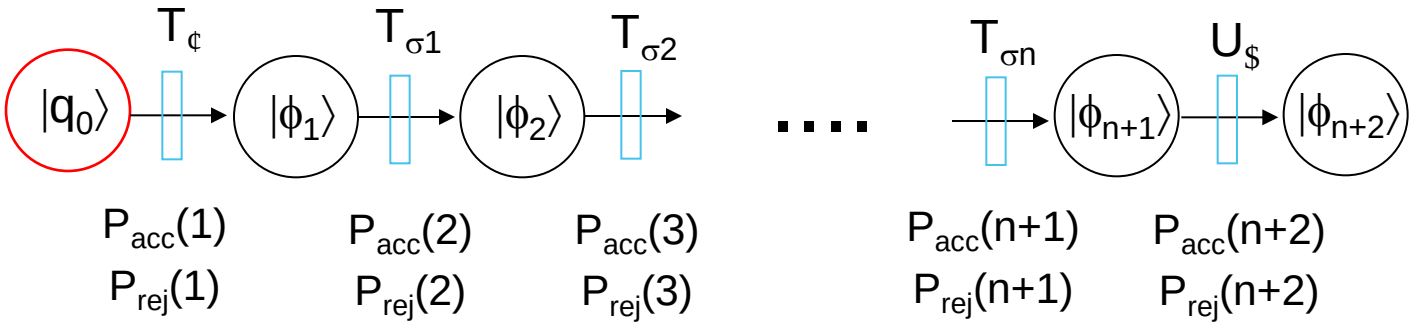
$x = \sigma_1\sigma_2\sigma_3.... \sigma_n$ ($\in\Sigma^n$) : input of length n

$x_i = \sigma_1\sigma_2\sigma_3.... \sigma_i$: the first i symbols of x

Extended transition operator $T_{¢xi} = T_{\sigma i}.....T_{\sigma 3}T_{\sigma 2}T_{\sigma 1}T_{¢}$

Acceptance probability at step i+1 $p_{acc}(i+1)$ = the squared norm of $P_{acc}U_{\sigma i}T_{¢xi-1}|q_0\rangle$

Rejection probability at step i+1 $p_{rej}(i+1)$ = the squared norm of $P_{rej}U_{\sigma i}T_{¢xi-1}|q_0\rangle$



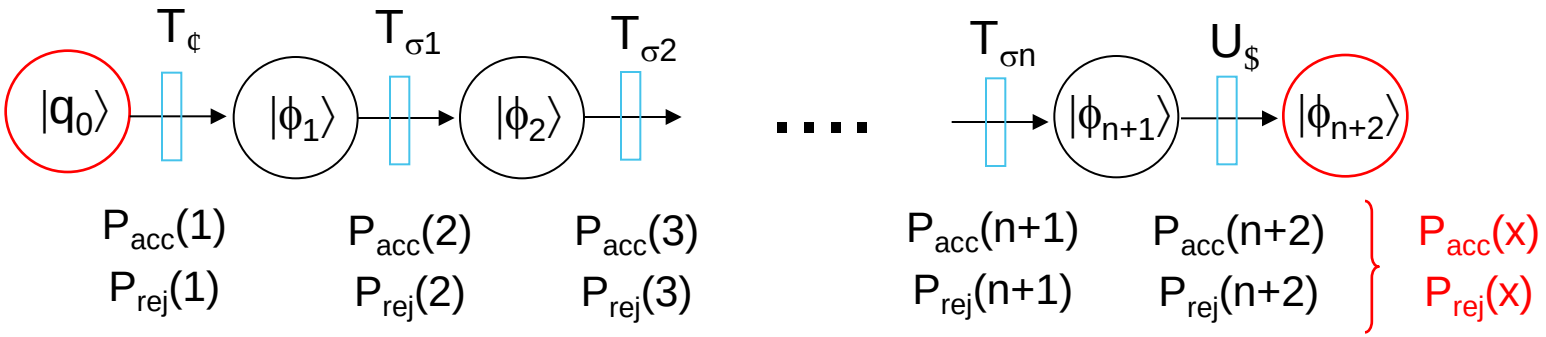
Quantum Computation III

- The probability of accepting/rejecting input x by M is defined as the sum of all probabilities at any step.

That is,

Acceptance probability of x : $p_{acc}(x) = p_{acc}(1) + p_{acc}(2) + ... + p_{acc}(n+2)$

Rejection probability of x : $p_{rej}(x) = p_{rej}(1) + p_{rej}(2) + ... + p_{rej}(n+2)$



Bounded Error Criteria for 1qfa's

- Let M be a 1qfa, let $\eta \in [0,1]$, and $L \subseteq \Sigma^*$.
  - $p_{M,acc}(x)$ = acceptance probability of M on input x
  - $p_{M,rej}(x)$ = rejection probability of M on input x

- A 1qfa recognizes language L with bounded error probability $\Leftrightarrow$ There is a constant $\varepsilon \in [0,1/2)$ s.t.,
  1. for all $x \in L$, $p_{M,acc}(x) \geq 1-\varepsilon$
  2. for all $x \in \Sigma^*-L$, $p_{M,rej}(x) \geq 1-\varepsilon$

- These criteria are similar to the isolated cut-point criteria of Rabin (1963).

Complexity Class 1BQFA

- L : language over alphabet Σ, K : amplitude set $\subseteq$ C

- $L \in 1BQFA_K \Leftrightarrow$

  $\exists M$: 1qfa $\exists \varepsilon \in [0,1/2)$ s.t.

  1. M has K-amplitudes
  2. $\forall x \in L$ [M accepts x with prob. $\geq 1-\varepsilon$]
  3. $\forall x \in \Sigma^* - L$ [M rejects x with prob. $\geq 1-\varepsilon$]

- We omit K if K = C.
- (Claim) 1BQFA $\subseteq$ REG [Kondacs-Watrous (1997)]

2-Way Quantum Finite Automata

- A 2-way quantum finite automaton (2qfa) is similar to a 2pfa with a read-only input tape but with a quantum transition function.

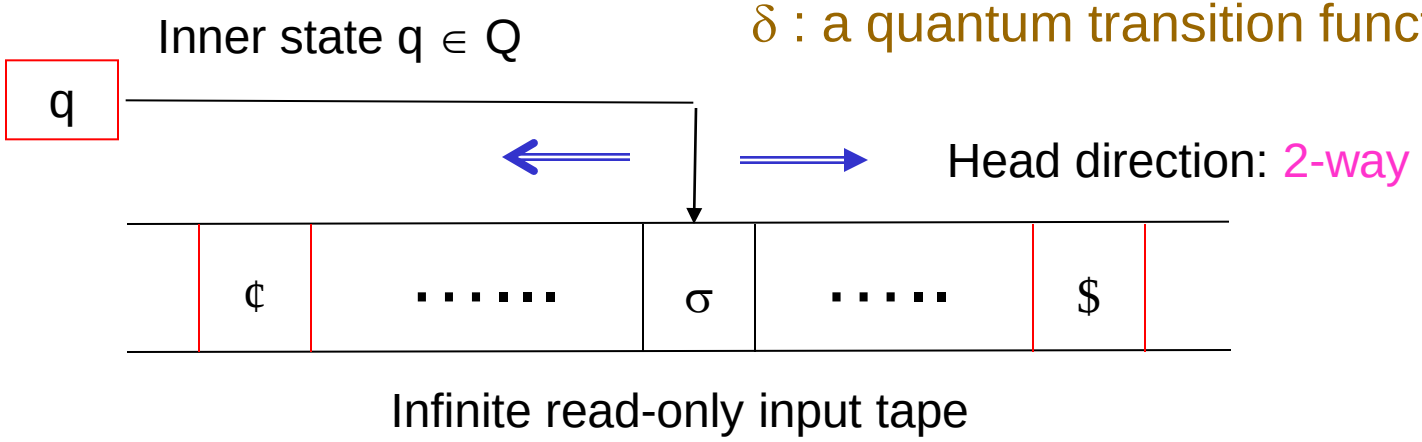
$M = (Q, \Sigma, \{¢, \$\}, \delta, q_0, Q_{acc}, Q_{rej})$

Σ = input alphabet

$\breve{\Sigma} = \Sigma \cup \{ ¢, \$ \}$

$Q_{halt} = Q_{acc} \cup Q_{rej} \subseteq Q$

δ : a quantum transition function



❖ For simplicity, the input tape is assumed to be circular.

Formal Definition of 2qfa's

A 2qfa M = $(Q,\Sigma,\{¢,\$\},\delta,q_0,Q_{acc},Q_{rej})$ has a read-only input tape and a special quantum transition function δ:

$$\delta : Q \times \breve{\Sigma} \times Q \times D \to C$$

$\breve{\Sigma} = \Sigma \cup \{ ¢, \$ \}$ D = { -1, 0, +1 }

- Time-evolution matrix (or operator):

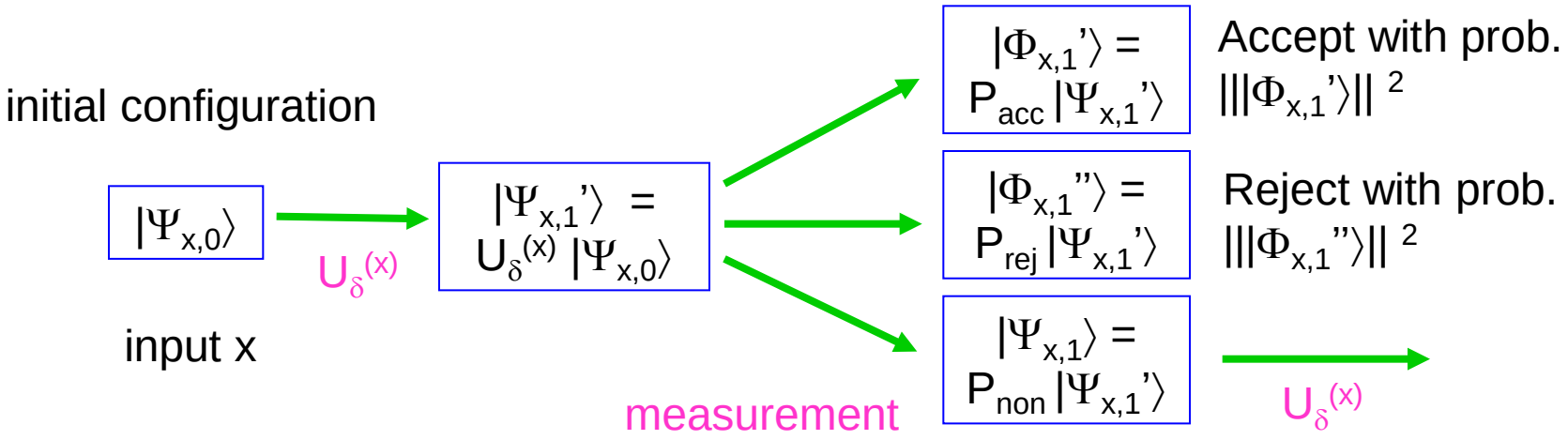
$$U_\delta^{(x)} \left| q,h \right\rangle = \sum_{(p,d)} \delta(q, x_h, p, d) \left| p, h+d \,(\mathrm{mod}\, n+1) \right\rangle$$

- Unitary Requirement: $U_\delta^{(x)}$ is a unitary matrix.

U is unitary $\Leftrightarrow$ $U(U^*)^T = (U^*)^T U = I$

Quantum Computation of 2qfa's

- A 2qfa works as follows.
- A 2qfa $M = (Q, \Sigma, \{¢,\$\}, \delta, q_0, Q_{acc}, Q_{rej})$
  - $U_\delta^{(x)}$ is a time-evolution operator
  - P_{acc}, P_{rej}, P_{non} are (projection) measurement operators.
  - $T_\delta^{(x)} = P_{non} U_\delta^{(x)}$ is a transition operator.



Bounded Error Criteria for 2qfa's

- Let M be a 2qfa, let $\eta \in [0,1]$, and let $L \subseteq \Sigma^*$.
  - $p_{M,acc}(x)$ = acceptance probability of M on input x
  - $p_{M,rej}(x)$ = rejection probability of M on input x

- A 2qfa M recognizes language L with bounded error probability $\Leftrightarrow$ There is a constant $\varepsilon \in [0,1/2)$ s.t.,
  1. for all $x \in L$, $p_{M,acc}(x) \geq 1-\varepsilon$
  2. for all $x \in \Sigma^*-L$, $p_{M,rej}(x) \geq 1-\varepsilon$

- These criteria are similar to the isolated-cut-point criteria of Rabin (1963).

Complexity Class: 2BQFA

- L : language over alphabet Σ, K : amplitude set $\subseteq$ C

- $L \in 2BQFA_K \Leftrightarrow$ L is recognized by K-amplitude 2qfa's with bounded error probability; namely,

  $\exists M$: 2qfa $\exists \varepsilon \in [0,1/2)$ s.t.

  1. M has K-amplitudes
  2. $\forall x \in L$ [M accepts x with prob. $\geq 1-\varepsilon(n)$]
  3. $\forall x \in \Sigma^* - L$ [M rejects x with prob. $\geq 1-\varepsilon(n)$]

bounded-error probability

- (Claim) 1BQFA $\subseteq$ REG $\subseteq$ 2BQFA [Kondacs-Watrous (1997)]

2EQFA, 2RQFA, $2C_{=}QFA$, 2PQFA

- Let $L \subseteq \Sigma^*$ and $K \subseteq C$
- $L \in 2C_{=}QFA_K \Leftrightarrow$

  $\exists M$: 2qfa s.t.

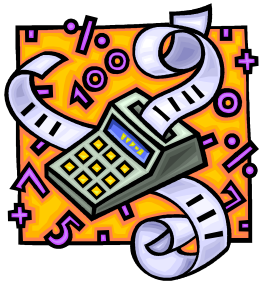
  1. M has K-amplitudes
  2. $\forall x \in \Sigma^* [\, x \in L \leftrightarrow p_{M,acc}(x) = 1/2 \,]$

- Similarly, define:
  - 2EQFA …. error-free 2qfa's (i.e., error prob. = 0)
  - 2PQFA …. unbounded-error 2qfa's (i.e., >1/2)
  - 2RQFA …. one-sided error 2qfa's
  - 2NQFA …. 2qfa's with cut point 0

V. Absolutely Halting QFAs

1. Absolutely Halting 2qfa's
2. Worst-case Linear Time
3. Time-Bounded 2qfa's
4. Relationships to 2qfa's

Absolutely Halting 2qfa's



- A 2qfa is said to halt absolutely $\Leftrightarrow$ all computation paths (both accepting and rejecting ones) of M on all inputs terminate within a finite number of steps.
- $2EQFA_K$(abs-halt) = class of languages recognized by K-amplitude error-free 2qfa's that halt absolutely.
- Similarly, define $2RQFA_K$(abs-halt), $2BQFA_K$(abs-halt), etc.
- Proposition: [Yamakami (2015)]

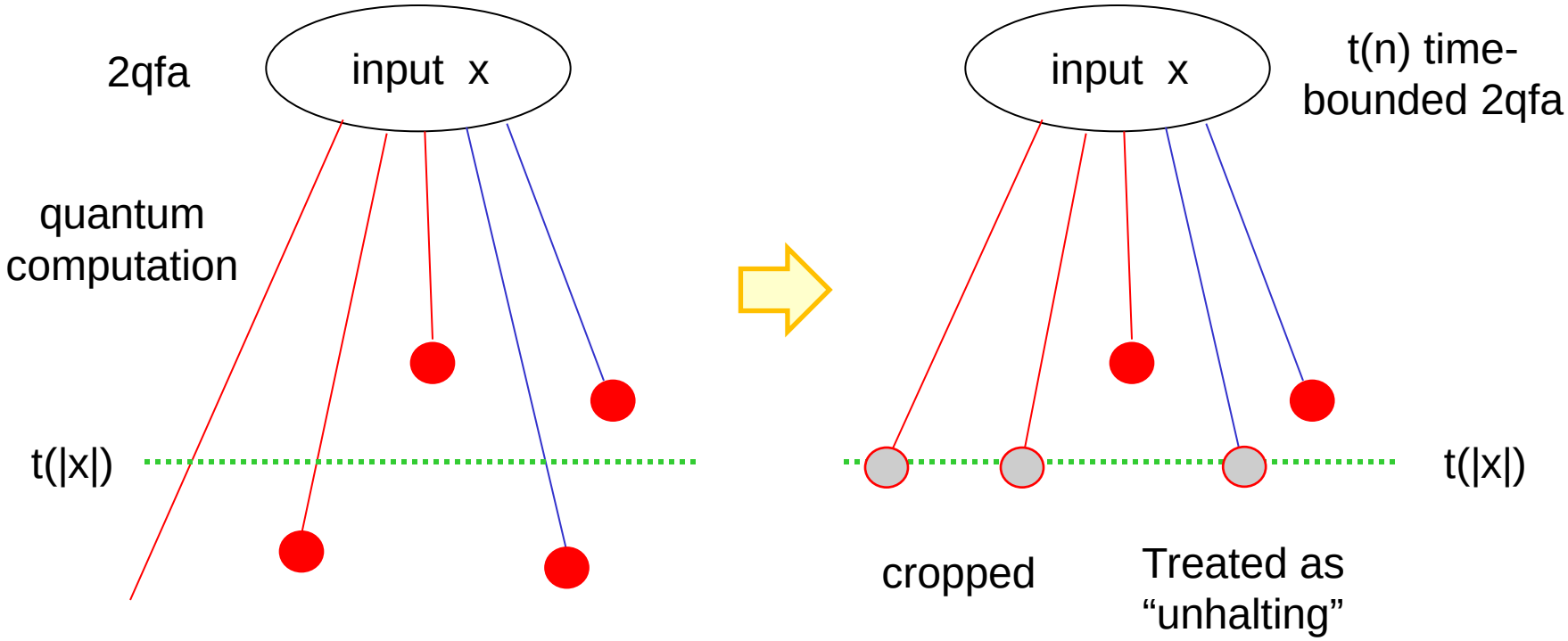
$$\mathrm{REG} \subseteq 2\mathrm{EQFA}_Q(\text{abs-halt}) \subseteq 2\mathrm{RQFA}_Q(\text{abs-halt})$$

Worst-Case Linear Time

- 2qfa halts in worst-case linear time $\Leftrightarrow$ all computation paths of M on each input x halt within $a|x|+b$ steps, where a,b are constants independent of x.
- $2BQFA_K$[lin-time] = class of languages recognized by K-amplitude bounded-error 2qfa's that halt in worst-case linear time.
- Similarly, define $2EQFA_K$[lin-time], $2RQFA_K$[lin-time], $2C_=QFA_K$[lin-time], and $2PQFA_K$[lin-time].
- Theorem: [Yamakami (2015)]
  - $2BQFA_K$(abs-halt) = $2BQFA_K$[lin-time].
  - The same equality holds for 2EQFA, 2RQFA, $2C_=QFA$, and 2PQFA.

Time-Bounded 2qfa's

- A t(n) time-bounded 2qfa M is obtained from a 2qfa by cropping its computation paths after exactly t(n) steps. Cropped paths are considered as "unhalting".



Relationships to 2qfa's

- Nice relationships hold between 2qfa's and time-bounded 2qfa's.
  - All languages recognized by t(n) time-bounded 2qfa's with (un)bounded-error probability are also recognized by 2qfa's with (un)bounded-error probability.
  - The converse also holds.

- Theorem: [Yamakami (2015)]

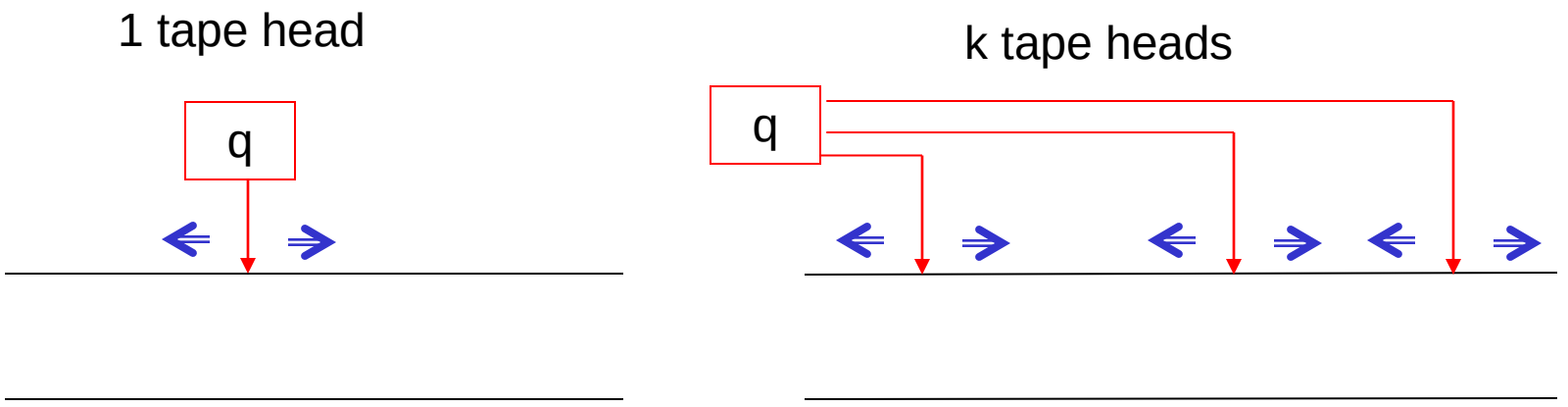
  Any language L in $2BQFA_A$ can be recognized by a certain $2^{O(n)}$ time-bounded 2qfa with bounded-error probability.

VI. Classical Simulation of 2QFAs

1. Multi-Head 2pfa's
2. Classical Simulations
3. Class Separation

Multi-Head 2pfa's

- A k-head 2pfa is a variant of a 2pfa that uses k tape heads moving separately on a read-only input tape.
- $2PPFA_K$(k-head)[poly-time] = class of languages recognized with "cut points" in $K \cap [0,1]$ by k-head-2pfa's in worst-case polynomial-time.

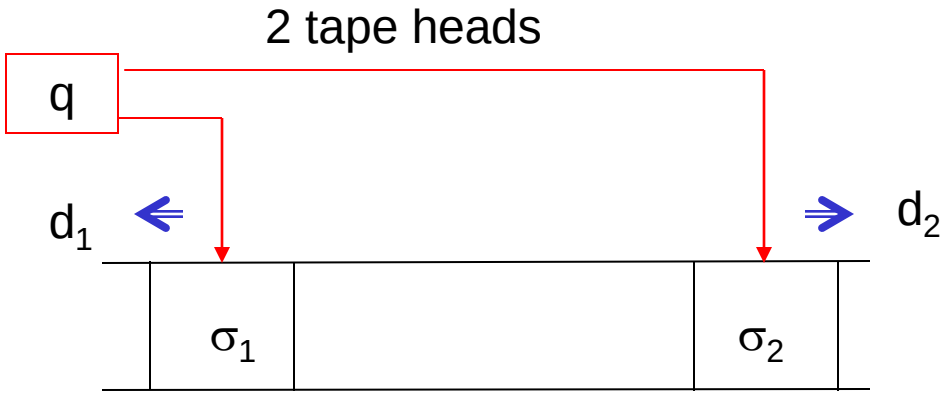


Probabilistic Transition Functions

- A 2-head probabilistic transition function δ is of the form

$$\delta : Q \times \breve{\Sigma} \times \breve{\Sigma} \times Q \times D_1 \times D_2 \to [0,1]$$

- "$\delta(q,\sigma_1,\sigma_2,p,d_1,d_2)=\gamma$" means that, when M is in inner state q, scanning (σ_1,σ_2), M changes q to p and moves its tape heads in direction (d_1,d_2) with probability γ.



Classical Simulations

- Given an amplitude set H, $\hat{H}$ denotes the minimal set that contains H and is closed under multiplication and addition.

- Theorem: [Yamakami (2015)]
  - There exists an integer $k \geq 2$ s.t. for any $H \subseteq R$, the following statements hold.
  - $2PQFA_H \subseteq 2PPFA_{\hat{H}}$(k-head)[poly-time].
  - $2C_=QFA_H \subseteq 2C_=PFA_{\hat{H}}$(k-head)[poly-time].

- Corollary: [Yamakami (2015)]
  - $2EQFA_H \subseteq 2C_=PFA_{\hat{H}}$(k-head)[poly-time] $\cap$ co-$2C_=PFA_{\hat{H}}$(k-head)[poly-time].
  - $2BQFA_H \subseteq 2PPFA_{\hat{H}}$(k-head)[poly-time] $\cap$ co-$2PPFA_{\hat{H}}$(k-head)[poly-time].

Corollary: Class Separation

- PL = unbounded-error probabilistic log-space complexity class (unbounded-error probabilistic version of L)
- Nishimura and Yamakami (2009) stated that $2BQFA_A$ is contained within PL (using a result of [Watrous (2003)]).
- The previous theorem implies the following corollary.

- Corollary: [Nishimura-Yamakami (2009)]
  - $2BQFA_Q$ is properly contained in PL.
- In particular, $2BQFA_Q \neq PL$ holds.

Stochastic Functions vs. Quantum Functions

- A stochastic function f is of the form $p_{M,acc}$ for a certain multi-head 2pfa M.
- $\#2PFA_K$ (k-head)[poly-time] = class of all stochastic functions defined by k-head 2pfa's running in worst-case polynomial-time
- As seen before, a quantum function f is of the form $p_{M,acc}$ for a certain target machine M
- $\#2QFA_K$ = class of all quantum functions defined by 2qfa's

Corollary: Stochastic vs. Quantum Functions

- The following corollary can be also obtained.

- Corollary: [Yamakami (2015)]
  - There is an integer $k \geq 2$ s.t., $\forall\, f \in \#2QFA_H$,
    $\exists\, g_1, g_2, h_1, h_2 \in \#2PFA_{\hat{H}}$(k-head)[poly-time] $\forall\, x$

$$\left(g_1(x) - g_2(x)\right) f(x) = h_1(x) - h_2(x)$$

Open Problems

- Find the exact complexity of 2BQFA and #2QFA.
- Characterize those quantum complexity classes in terms of classical complexity classes.
- Find more interesting features of quantum functions.

Thank you for listening

Q & A

I’m happy to take your question!

END