LATVIJAS UNIVERSITĀTE
UNIVERSITY OF LATVIA

EIROPAS SOCIĀLAIS FONDS

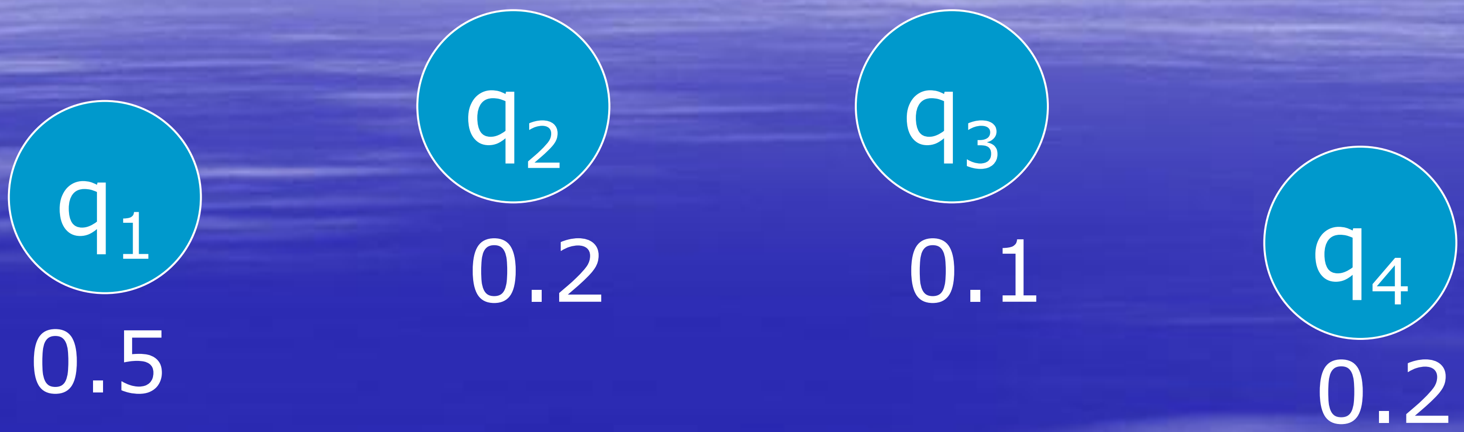
Datorzinātnes lietojumi un tās saiknes ar kvantu fiziku
Nr.2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044

Kvantu algoritmi lineāru vienādojumu sistēmu risināšanai

Andris Ambainis
LU Datorikas fakultāte

LU 69. konference, 2011. gada 4. februārī

Varbūtiska sistēma



$$p_i \geq 0$$

$$\sum_i p_i = 1$$

Kvantu sistēma

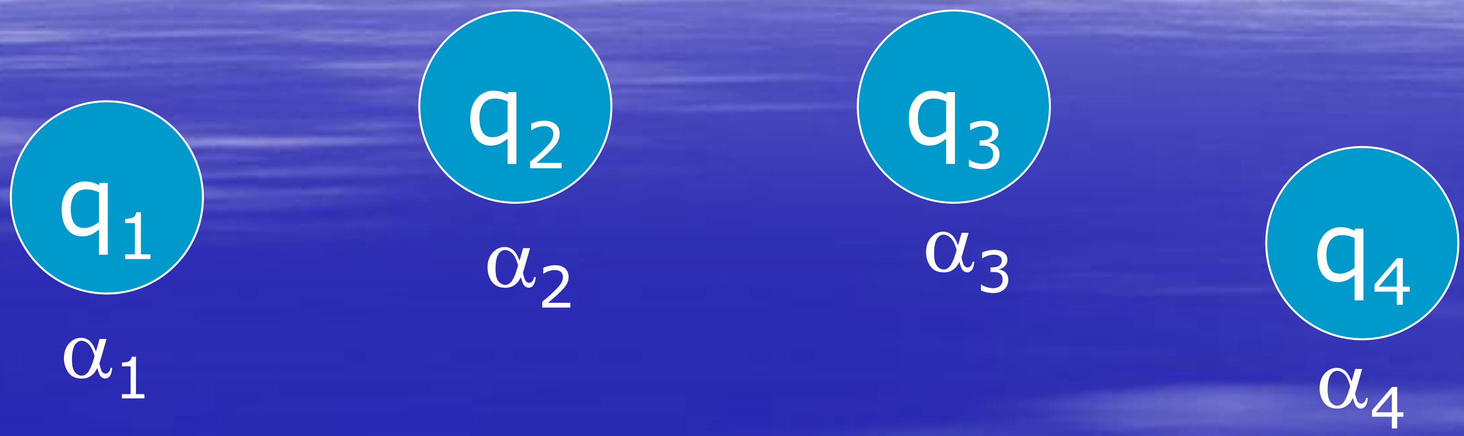
q_1 q_2 q_3 q_4

α_1 α_2 α_3 α_4

α_i – kompleksi skaitļi

$$\sum_i |\alpha_i|^2 = 1$$

Kvantu sistēma



Mērot iegūstam q_i
ar varbūtību $|\alpha_i|^2$.

Lineāras vienādojumu sistēmas

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N = b_2$$

$$\ldots$$

$$a_{N1}x_1 + a_{N2}x_2 + \ldots + a_{NN}x_N = b_N$$

Zināms: a_{11}, a_{12}, ..., a_{NN}, b_1, b_2, ..., b_N.
Jāatrod: x_1, x_2, ..., x_N.

Lineāras vienādojumu sistēmas

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1N}x_N = b_1$$
$$a_{21}x_1 + a_{22}x_2 + ... + a_{2N}x_N = b_2$$
$$...$$
$$a_{N1}x_1 + a_{N2}x_2 + ... + a_{NN}x_N = b_N$$

Klasiskais algoritms: $O(N^{2.38...})$.
Ieejas datu apjoms: N^2.
Izejas datu apjoms: N.

[Harrow-Hassidim-Lloyd, 2008]

- Algoritma rezultāts – kvantu stāvoklis:

$$x_1|1\rangle + x_2|2\rangle + \ldots + x_N|N\rangle$$

- N stāvokļi – log N kvantu biti.
- Stāvokli var radīt O(log N) laikā.

$O(N^{2.38\ldots}) \to O(\log N)$

Scientific American

- **Warp-Speed Algebra: New Algorithm Does Algebra in a Snap**

  New quantum algorithm can solve monster-size equations.

[Harrow-Hassidim-Lloyd, 2008]

- Algoritma rezultāts – kvantu stāvoklis:

$$x_1|1\rangle + x_2|2\rangle + ... + x_N|N\rangle$$

- Trūkums: no kvantu stāvokļa nevar nolasīt visu atrisinājumu x_1, x_2, ..., x_N.
- Var iegūt daļēju informāciju par atrisinājumu.

Pamatideja

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N = b_2$$

$$\ldots$$

$$a_{N1}x_1 + a_{N2}x_2 + \ldots + a_{NN}x_N = b_N$$

$$\sum_{i=1}^{N} b_i |i\rangle \longrightarrow \sum_{i=1}^{N} x_i |i\rangle$$

Zināms — Risinājums

Pamatidejas

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ a_{21} & a_{22} & \dots & a_{2N} \\ \dots & \dots & \dots & \dots \\ a_{N1} & a_{N2} & \dots & a_{NN} \end{pmatrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix} \qquad b = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_N \end{pmatrix} \qquad Ax = b$$

$$\sum_{i=1}^{N} b_i \left| i \right\rangle \longrightarrow \sum_{i=1}^{N} x_i \left| i \right\rangle$$

$$x = A^{-1} b$$

Algoritma darbības laiks

1. Atkarība no vienādojumu/nezināmo skaita:

   O(log N).

2. Kondīcijas skaitlis k.

Kondīcijas skaitlis

$$a_{11}x_1 + a_{12}x_2 + \ldots + a_{1N}x_N = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \ldots + a_{2N}x_N = b_2$$

$$\ldots$$

$$a_{N1}x_1 + a_{N2}x_2 + \ldots + a_{NN}x_N = b_N$$

Cik daudz mainās atrisinājums, nedaudz izmainot b_1, b_2, ..., b_N?

Kondīcijas skaitlis

$$\begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix} \longleftarrow \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_N \end{pmatrix}$$

Izmaiņa: δ_{max}.

$$\begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_N \end{pmatrix}$$

Izmaiņa: δ_{min}.

Izmaiņa: ε.

$$k = \frac{\delta_{\max}}{\delta_{\min}}$$

Algoritma ātrdarbība

- [Harrow, Hassidim, Lloyd, 08]: $O(k^2 \log N)$.
- [A, 2010]: $O(k \log N)$.

Algoritma pielietojumi

- Kā var izmantot atrisinājumu – kvantu stāvokli?

$$x_1|1\rangle + x_2|2\rangle + \ldots + x_N|N\rangle$$

- [Rivošs, 2010]: piemēri, kur no šāda atrisinājuma var iegūt lietderīgu informāciju (piemēram, x_i=1 vienam i, x_j=0 pārējiem j).