Pis'ma v ZhETF, vol.71, iss.5, pp.309 - 313

RELATIVE DIFFUSION TRANSFORM AND QUANTUM SPEED UP OF COMPUTATIONS

Yu.I.Ozhigov[1], *N.B.Victorova*[+]

Department of applied mathematics, "Stankin"
101472, Moscow, Russia

[+]*Department of Differential Equations and Functional Analysis,*
Russian University of Friendship
117198 Moscow, Russia

Submitted 26 January 2000

It is shown that every function computable in time $T(n)$ and space $S(n)$ on classical 1-dimensional cellular automaton can be computed with certainty in time $O(T^{1/2}S)$ and space $n\sqrt{T}$ on a quantum computer with relative diffusion transforms (RDTs) on parts of intermediate products of the classical computation. However, RDTs in general case cannot be implemented by a conventional quantum computer even with oracles for intermediate results. Such a function can be computed only in time $O(S4^{S/2}T/T_1)$ on a conventional quantum computer with oracles for intermediate results of classical computations with the time T_1.

PACS: 03.67.Lx

Quantum mechanical computations are distinct in nature from the classical ones (look at [1-3]). One of the most intriguing features of quantum computers is their ability to speed up searching. L.Grover in [4] constructed a quantum algorithm which for a given function $F: \{0,1\}^n \longrightarrow \{0,1\}$ finds a unique solution of equation $F(x) = 1$, after $O(\sqrt{N})$ quantum evaluations of F when every classical computer requires $\Omega(N)$ evaluations, $N = 2^n$.

Having Grover's algorithm for searching as a good precedent it is interesting to elucidate whether is it possible to speed up complicated classical algorithms on quantum computers transforming classical programs to quantum ones. Most likely, this cannot be done without some additional information about a classical algorithm. Such an information in its simplest form is an oracle testing intermediate results. Let a work of classical algorithm on an input word A have the form $x_0(A) \longrightarrow x_1(A) \longrightarrow \ldots \longrightarrow x_T(A)$. Given $T_1 < T$, the intermediate result is the set $\{\langle A, x_{T_1}(A)\rangle\}$. An oracle for this set is called a verifier.

The following Theorem 1 shows that verifier can speed up only sufficiently long computations. Theorem 2 exhibits the potentials of relative diffusion transform (RDT).

Theorem 1 . *Every function $F: \omega^* \longrightarrow \omega^*$ ($\mathrm{card}(\omega) = 4$), computable on a classical one dimensional cellular automaton with alphabet ω in time $T(n)$ and space $S(n)$ can be computed in time $T_q = O(S4^{S/2}T/T_1)$ and space S on a quantum computer with a verifier for intermediate results of F corresponding to the time T_1.*

Theorem 2 . *Every function $F(n)$ computable in time $T(n)$ and space $S(n)$ on a classical one dimensional cellular automaton can be computed in time and space $O(T^{1/2}S)$*

[1] e-mail: y@oz.msk.ru

on a quantum computer with RDT on parts of intermediate results of F corresponding to the time $T_1 = T^{1/2}S$.

Note, that in general case RDT cannot be localized (e.g. represented as a tensor product of small matrices) and cannot be replaced by computations on quantum query machine (see the work [5] for the definition) of polynomial time complexity. Therefore, generally speaking, the speeding up by Theorem 2 cannot be achieved on quantum query machine.

Quantum computations. We shall use the simple model of quantum computer with two parts: a classical part, which transforms by classical laws (say as a cellular automaton), and a quantum part which transforms by quantum mechanical principles.

A quantum part. It is a set $\mathcal{E} = \{\nu_1, \nu_2, \dots, \nu_r\}$ (r even), which elements are called qubits. Each qubit takes values from the set $\{z_0\mathbf{0} + z_1\mathbf{1} \mid z_1, z_2 \in \mathbb{C}, |z_0|^2 + |z_1|^2 = 1\}$. Here $\mathbf{0}$ and $\mathbf{1}$ are referred to as basic states of qubit. They form a basis of $\mathbb{C}^2$. It will be convenient to divide $\mathcal{E}$ into registers of 2 neighboring qubits each so that any register takes values from $\omega = \{0, 1, 2, 3\}$.

A basic state of the quantum part is a function of the form $e : \mathcal{E} \longrightarrow \{0, 1\}$. This state e may be encoded as $|e(\nu_1), e(\nu_2), \dots, e(\nu_r)\rangle$ and can be naturally identified with the corresponding word in alphabet ω.

Let $e_0, e_1, \dots, e_{K-1}$ be all basic states, taken in some fixed order, $\mathcal{H}$ be a K-dimensional Hilbert space with orthonormal basis $e_0, e_1, \dots, e_{K-1}$, $2^r = K$. This Hilbert space can be regarded as a tensor product $\mathcal{H}_1 \bigotimes \mathcal{H}_2 \bigotimes \cdots \bigotimes \mathcal{H}_r$ of 2-dimensional spaces, where $\mathcal{H}_i$ is generated by possible values of $e(\nu_i)$, $\mathcal{H}_i \cong \mathbb{C}^2$. A (pure) state of quantum part is such an element $x \in \mathcal{H}$ that $|x| = 1$. Thus, in contrast to classical devices, quantum device may be not only in basic states, but also in coherent states, and this imparts surprising properties to such devices.

Put $\mathcal{K} = \{0, 1, \dots, K-1\}$. For elements $x = \sum_{s \in \mathcal{K}} \lambda_s e_s$, $y = \sum_{s \in \mathcal{K}} \mu_s e_s \in \mathcal{H}$ their dot product $\sum_{s \in \mathcal{K}} \lambda_s \bar{\mu}_s$ is denoted by $\langle x|y\rangle$, where $\bar{\mu}$ means complex conjugation of $\mu \in \mathbb{C}$, hence $\langle x|y\rangle = \overline{\langle y|x\rangle}$.

Unitary transformations. Let $\{1, \dots, r\} = \bigcup_{i=1}^{l} L_i^s$, $L_i^s \cap L_j^s = \emptyset$ $(i \neq j)$, unitary transformation U_i^s acts on $\bigotimes_{j \in L_i^s} e_j$, then $U^s = \bigotimes_{i=1}^{l} U_i^s$ acts on $\mathcal{H}$, $s = 1, 2, \dots, M$. We require that all U_i^s belong to some finite set of transformations independent of $\mathcal{E}$ which can be easily performed by physical devices.

A computation is a sequence of unitary transformations: $U^1, U^2, \dots, U^M$. It is applied to some initial state χ_0.

A classical part of computer points partitions $\bigcup L_i$ and chooses transformations U_i^s sequentially for each s.

Observations. Let $\chi = \sum_{s \in \mathcal{K}} \lambda_s e_s$ be some fixed state of computer, often $\chi = \chi_M$. If $A \in \{0, 1\}^k$ is a list of possible values for the first k qubits, then we put

$$B_A = \{i \mid \exists a_{k+1}, a_{k+2}, \dots, a_r \in \{0, 1\} : \quad e_i = A a_{k+1} a_{k+2} \dots a_r\}.$$

A result of this observation is a new state

$$\chi^A = \sum_{i \in B_A} \frac{\lambda_i}{\sqrt{p_a}} e_i, \text{ where } p_A = \sum_{i \in B_A} \lambda_i^2.$$

An observation of the first register in a state χ is a procedure which gives a pair: $<$ classical word A, quantum state $\chi^A >$ with probability p_A for any possible $A \in \{0,1\}^k$. The only way to learn results of quantum computations is to obtain such words A.

Diffusion transform. Every unitary transformation $U : \mathcal{H} \longrightarrow \mathcal{H}$ can be represented by it's matrix $U = (u_{ij})$ where $u_{ij} = \langle U(e_j)|e_i\rangle$ so that for $x = \sum \lambda_p e_p$, $U(x) = = \sum \lambda'_p e_p$ we have $\bar{\lambda}' = U\bar{\lambda}$, where $\bar{\lambda}, \bar{\lambda}'$ are columns.

A diffusion transform D is defined by it's matrix D: $d_{ij} = 2/N$ if $i \neq j$, and $d_{ij} = = -1 + 2/N$ if $i = j$. Note that $D = WRW$, where R is a phase inversion of e_0, and W is Walsh – Hadamard transform, defined as a tensor product of n matrices

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

hence a diffusion transform can be performed on a quantum computer.

For any state $x = \sum\limits_{p \in \mathcal{K}} \lambda_p e_p$ an average amplitude is taken as $x_{av} = \sum\limits_{p \in \mathcal{K}} \lambda_p / N$. Hereafter $\mathcal{H}$ denotes a real Euclidean space.

Proposition 1 [4]. *For every state* x $\langle e_p|x\rangle - x_{av} = x_{av} - \langle e_p|D(x)\rangle$.

This means that D is an inversion about average. We need this property related to a subspace $\mathcal{H}_0 \subseteq \mathcal{H}$. Let $\mathcal{H}_0$ be a subspace of $\mathcal{H}$ with basis $e_0, \cdots, e_{M-1}$. Define the relative diffusion transform $D^{\mathcal{H}_0}$ by

$$d_{ij}^{\mathcal{H}_0} = \begin{cases} 2/M, & \text{if } i \neq j; i,j \in \mathcal{K}, \\ -1 + 2/M, & \text{if } i = j \in \{0, \ldots, M-1\}, \\ \delta_{ij}, & \text{in other cases.} \end{cases}$$

Given a state $x = \sum\limits_{p=0}^{M-1} \lambda_p e_p$ its average amplitude is taken as $x_{av}^{\mathcal{H}_0} = \sum\limits_{p=0}^{M-1} \lambda_p / M$. Proposition 1 can be easily extended to RDTs as follows.

Proposition 2 . *For every* $p = 0, 1, \cdots, M-1$ $\lambda_p - x_{av}^{\mathcal{H}_0} = x_{av}^{\mathcal{H}_0} - \langle e_p|D^{\mathcal{H}_0}(x)\rangle$.

Sequential applications of diffusion transforms and a simple transformation changing a sign of the target state are used for a fast quantum search in [4]. Any iteration increases amplitude of the target state (initially taken as $1/\sqrt{N}$) approximately by $1/\sqrt{N}$. Therefore Grover's algorithm requires $O(\sqrt{N})$ steps to make the target state really observable.

$Q - M$ speeding up. Suppose, we have a function $F : \omega^* \longrightarrow \omega^*$ computable in time $T(n) > n^2$ and space $S(n) = n$ on a classical Turing machine or a cellular automaton. Our goal is to compute F faster than $\Omega(T(n))$ on a quantum computer with RDTs.

Let $f : \omega^* \longrightarrow \omega^*$ denote one step of a classical algorithm computing F. In case F is one dimensional cellular automaton with radius R a neighborhood of radius R of each i-th letter in $\bar{a}$ determines i-th letter in $f(\bar{a})$. Without loss of generality we can assume that $\mathrm{card}(\omega) = 4$, because every cellular automaton can be simulated without slowdown by such cellular automaton with appropriate radius. Define for every $a \in \omega^*$ $f^{(0)}(a) = a$, $f^{(m)}(a) = f(f^{(m-1)}(a))$, so that $f^{(m)}$ is m-iteration of f, $f^{(C)} = F$.

Proof of Theorem 1. We can assume that $T > n4^{n/2}$ because otherwise $T = O(T_q)$. Prepare the state $\frac{1}{4^{n/2}} \sum\limits_x |x\rangle$. Now, using an oracle and applying Grover's algorithm

we obtain $f^{(T_1)}(x_0)$ in time $4^{n/2}n$. Then iterate this procedure and obtain sequentially $f^{(2T_1)}(x_0), f^{(3T_1)}(x_0), \ldots, f^{(T)}(x_0)$, which requires the time T_q. Theorem 1 is proved.

Proof of Theorem 2. Fix integers n, $T_1, T_2 : T_1T_2 = T(n)$. Let T_1 independent processors be given: $P_1, P_2, \ldots, P_{T_1}$, every P_i with the quantum part $B_i = \{1, 2, \ldots, 3n\}$. The pure states of all P_i will have the form: $|a_1, \ldots, a_{3n}\rangle$, where all $a_i \in \omega$.

At first prepare the state $\frac{1}{k^{n/2}} \sum\limits_{\bar{a}} |\bar{0}, \bar{a}, \bar{0}\rangle$ in each processor applying Walsh – Hadamard transformation to all states of the form $|\bar{0}, 0, \ldots, 0, a, 0, \ldots, 0, \bar{0}\rangle$, where $\bar{0} = 0^n$, $\bar{a} = (a_{i_1}, \ldots, a_{i_n})$. Then calculate T_2-iteration of f in the last registers to obtain the state $X_0 = \frac{1}{k^{n/2}} \sum\limits_{\bar{a}} |\bar{0}, \bar{a}, f^{(T_2)}(\bar{a})\rangle$ in all processors. This takes $O(T_2)$ steps. Denote the state $|f^{(iT_2)}(x_0), \bar{a}, f^{(T_2)}(\bar{a})\rangle$ by $\xi_i(\bar{a})$.

Then the processors work in a serial mode computing sequentially the intermediate results $\mathrm{tar}_1, \mathrm{tar}_2, \ldots, \mathrm{tar}_{T_1}$, where $\mathrm{tar}_i = \frac{1}{k^{n/2}} \sum\limits_{\bar{a}} \xi_i(\bar{a})$, x_0 is some fixed input word of the length n.

Beginning with tar_i the processor P_i achives the pure state

$$\mathrm{tar}^*_{i+1} = |f^{(iT_2)}(x_0), f^{(iT_2)}(x_0), f^{(iT_2+T_2)}(x_0)\rangle$$

in time $O(n^2)$. Then the state tar_{i+1} is prepared for the following processor P_{i+1} which is initially set to the state X_0. The last passage is quite clear, it takes one instant of time, and we need only to describe the first passage: $\mathrm{tar}_i \longrightarrow \mathrm{tar}^*_{i+1}$.

We omit indices, now tar^* is our target state. Let $\mathcal{H}_0$ be Euclidean space with orthonormal basis $\mathcal{B}_0$ consisting of all vectors of the form $\xi_i(\bar{a})$. We have:
$\mathrm{tar}^* = |\alpha_1, \ldots, \alpha_n, \alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n\rangle$. $\mathcal{B}_s$ denotes the set of all vectors of the form $|\alpha_1, \ldots, \alpha_n, \alpha_1, \ldots, \alpha_s, \gamma_{s+1}, \ldots, \gamma_{2n-s}\rangle$ from $\mathcal{B}_0$, $s = 0, \ldots, n$.

Define $\mathcal{H}_s$ as subspace of $\mathcal{H}_0$ spanned by all vectors from $\mathcal{B}_s$. Then

$$\dim \mathcal{H}_s = k^{n-s}; \quad \{\mathrm{tar}\} = \mathcal{H}_n \subset \mathcal{H}_{n-1} \subset \cdots \subset \mathcal{H}_1 \subset \mathcal{H}_0.$$

Now apply sequentially, for $j = 1, 2, \ldots, n$ the following procedure, beginning with tar. a) Rotation of all $\xi \in \mathcal{B}_j$. b) Following RDT $D^{\mathcal{H}_{j-1}}$.

Finally observe the quantum part. If $k = 4$, then in the instant of observation "tar^*" has amplitude 1. To show this we need the following Lemma. Let χ_j be the result of j-th step of our procedure a), b), $\chi_0 = \mathrm{tar}$.

Lemma 1 . *For all* $\xi \in \mathcal{B}_j$, $j = 0, 1, \ldots, n$ $\langle \chi_j | \xi \rangle = (3 - 4/k)^j / k^{n/2}$.

Sketch of the proof. Induction on j. Basis follows from the choise of χ_0. Step follows from Proposition 2. Now put $k = 4$, Lemma 1 yields $\langle \chi_j | \xi \rangle = 2^j / k^{n/2}$. Consequently, $\langle \chi_n | \mathrm{tar}^* \rangle = 1$.

This computation of F requires the time $O(T^{1/2}n)$ if we put $T_1 = O(T^{1/2}/n)$. Lemma 1 is proved. Theorem 2 is proved.

The power of RDT.

Theorem 3 . *1) RDTs cannot be implemented on a quantum query machine in polynomial time.*

2) Any device which is able to perform RDTs on the sets, localized by arbitrary oracles can find a solution of equation $f(x) = 1$ for a given oracle f in polynomial time with high probability provided this solution is unique.

Proof of Theorem 3.

Lemma 2 . *Let f be one-to-one function $\omega^n \longrightarrow \omega^n$,*

$$k = \mathrm{card}(\omega) = 4, \quad \mathrm{x_0(f)} = \frac{1}{\mathrm{k^{n/2}}} \sum_{\bar{\mathrm{b}}} |\mathrm{f}(\bar{\mathrm{b}}), \bar{\mathrm{b}}\rangle, \quad \bar{\mathrm{a}} \in \omega.$$

Then the value $f(\bar{a})$ can be found with certainty in time $O(n^2)$ on a computer with RDTs on all sets of the form

$$N_{\varepsilon_1 \ldots \varepsilon_k} = \{\langle f(\bar{a}), \bar{a}\rangle | \exists \varepsilon_{k+1}, \ldots, \varepsilon_n : \ \bar{a} = \varepsilon_1 \ldots \varepsilon_k \varepsilon_{k+1} \ldots \varepsilon_n\}.$$

Proof. Let $\mathcal{B}_j$ be a set of such vectors of the form $|f(\bar{b}), \bar{b}\rangle$, that the first j components of $\bar{a}$ and $\bar{b}$ are equal, $\mathcal{H}_j$ be Euclidean space with the basis $\mathcal{B}_j$.

Then $\{|f(\bar{a}), \bar{a}\rangle\} = \mathcal{H}_n \subset \mathcal{H}_{n-1} \subset \cdots \subset \mathcal{H}_0$. Apply sequentially for $j = 1, 2, \ldots, n$ rotation of all $\xi \in \mathcal{B}_j$ and RDT $D^{\mathcal{H}_{j-1}}$. This results in $|f(\bar{a}), \bar{a}\rangle$ by Lemma 1. Lemma 2 is proved.

1) Consider the computation with RDTs from Lemma 2, depending on f: $x_0(f) \longrightarrow x_1(f) \longrightarrow \ldots \longrightarrow x_p(f)$, $p = O(n^2)$.

Consider some other function $\tilde{f}$ which differs from f only on two arguments including $\bar{a}$. Then $\|x_0(\tilde{f}) - x_0(f)\| \leq 2/\sqrt{N}$, $N = 4^n$. Assume that RDTs can be implemented in polynomial time with the corresponding oracles for $N_{\varepsilon_1 \ldots \varepsilon_k}$. By the definition of $\tilde{f}$ the corresponding post query states x_m $m = 1, 2, \ldots, p$ for computations of $f(\bar{a})$ and $\tilde{f}(\bar{a})$ differs on $mP(n)/\sqrt{N}$ where $P(n)$ is a polynomial, as it is shown in the article [5]. But it is impossible, because for the final pure states $x_p(f) = |\bar{a}, f(\bar{a})\rangle$, $x_p(\tilde{f}) = |\bar{a}, \tilde{f}(\bar{a})\rangle$ we have $\|x_p(f) - x_p(\tilde{f})\| = \sqrt{2}$. Point 1) is proved.

2) Consider an oracle for f and find a solution of equation $f(x) = 1$ with RDTs. Put $\mathcal{B}_m = \{\langle x, y\rangle | \ y = f(x) \& (y = 1 \ \vee \exists x' : \ x = (0,0)^m x')\}$, $(0,0) \in \omega$. Then we have: $|\mathrm{card}(\mathcal{B}_m) - 4^{n-m}| \leq 1$, because the solution is unique. Therefore, applying the algorithm from the proof of Lemma 2 we obtain the desired x with high probability in time $O(n^2)$. Theorem 3 is proved.

We are grateful to Peter Hoyer for his comments to the previous version of this work.

---

1. A.Barenco and A.Ekert, Acta phys. slovaca **45**, 1 (1993).
2. S.Lloyd, Science **261**, 1569 (1993).
3. A.Steane, http://xxx.lanl.gov/archive/quant-ph/9708022.
4. L.K.Grover, http://xxx.lanl.gov/archive/quant-ph/9605043.
5. C.H.Bennett, E.Bernstein, G.Brassard, and U.Vazirani, To appear in SIAM Journal on Computing (lanl e-print quant-ph/9701001).