Математика и ее приложения : ЖИМО. 2013. Вып. 1 (10). С. 31–36

УДК 512.543

А. А. Толстопятов[1]

Применение квантового алгоритма Гровера для задания разбиения файла на буферы

Ключевые слова: разбиение файла, порождающие полиномы, алгоритм Гровера.

Квантовый алгоритм Гровера поиска в неупорядоченной базе данных применим к задаче о разбиении файла на буферы при булевом сжатии. Получена оценка эффективности такого применения.

Key words: boolean compress, boolean polynomials, Grover algorithm.

Grover's quantum search algorithm disordered database applicable to the problem of splitting the file into buffers at the Boolean compression. Obtain an estimate of the effectiveness of such application.

Задача разбиения файла на буферы [6] при булевом сжатии файлов [5] является экспоненциально сложной. Один из подходов к решению таких задач – это квантовые вычисления [1] – [3]. Использовать этот подход было предложено в [6], а в [7] было отмечено, что, если все разбиения файла на буферы рассматривать как неупорядоченную базу данных, то открывается возможность применить к этой задаче квантовый алгоритм Гровера [8]. Этот алгоритм не является самым эффективным, так как он снижает сложность задачи с 2^{N-1} до $2^{\frac{N-1}{2}}$, в то время как более эффективный алгоритм Шора [9] снижает ее до $(N-1)^3$, где $N-1$ – наименьшее число буферов в разбиении $L = 1, \ldots, N-1$. Однако, достоинством алгоритма Гровера может являться то, что его проще применить к указанной выше задаче. В настоящей работе исследуется эта возможность.

1. Алгоритм Гровера

Задача поиска в неупорядоченной базе данных может быть поставлена так [4]. Пусть $j = 0, 1, \ldots, 2^N - 1$. Тогда вектора

$$|j >= \sum_{k=0}^{2^N-1} C_k |c^{k-1} > \tag{1}$$

образуют векторное пространство $V_N(Z_2)$, если $C_k \in Z_2$. Это пространство, содержащее 2^N векторов $|j >$ будет представлять неупорядоченную

[1]Ивановский государственный университет; E-mail: khash2@mail.ru. Работа выполнена при финансовой поддержке РФФИ (проект 10-07-00350а)

базу данных. Если $|x_0 > \in V_N(Z_2)$, то задача поиска заключается в построении алгоритма, который за σ шагов найдет $|x_0>$. Число σ характеризует сложность алгоритма. Для решения задачи поиска необходим оператор U, действующий на $V_n(Z_2)$, называемый оракулом, который по-разному действует на векторах $|j> = |x_0>$ и $j> \neq |x_0 >$.

Для построения любого квантового алгоритма требуется построить регистр квантового компьютера. Если $V_2(C)$ – двумерное векторное пространство над полем комплексных чисел $\mathbb{C}$, – так называемый кубит, то $V_{2^N} = V_2(\mathbb{C})^{\otimes N}$ есть регистр. Классические векторы $|j> = V_N(Z_2)$ называемые чистыми состояниями, образуют базис $V_N(\mathbb{C})$. Тогда квантовый вектор $|\xi> \in V_{2^N}(\mathbb{C})$ будет записываться так:

$$|\xi> = \sum_{j=0}^{2^N-1} C_j|j>, \tag{2}$$

где $C_j \in \mathbb{C}$. Если обозначить $<\xi| = |\xi>^+$, то можно ввести на базисных векторах $|j>$ скалярное произведение

$$<k|j> = \delta_{kj}. \tag{3}$$

Тогда для любых $|\eta>, |\xi> \in V_{2^N}(\mathbb{C})$, где

$$|\eta> = \sum_{j=0}^{2^N-1} b_j^*|j >, \tag{4}$$

будем иметь

$$<\eta|\xi> = \sum_{j=0}^{2^N-1} b_j^* c_j. \tag{5}$$

Квантовые вычисления – это действия на $|\xi>$ унитарными операторами F – так называемыми гейтами. Результат квантовых вычислений должен быть выведен на классический компьютер. Для этого надо спроектировать квантовый вектор $|\xi>$ на один из классических $|j>$. Тогда вероятность того, что на k-м шаге алгоритма получится вектор $|j>$ есть

$$W_k(|j>) = |<j|F^k|\xi>|^2. \tag{6}$$

Алгоритм Гровера [8] содержит оракул U:

$$U = I - 2|x_0><x_0|, \tag{7}$$

где I – единичная матрица. Нетрудно видеть, что:

$$U|j> = \begin{cases} -|j>, & |j> = |x_0> \\ |j>, & |j> \neq |x_0> \end{cases} \tag{8}$$

т.е. действует по-разному на $|j> = |x_0>$ и $|j> \neq |x_0>$. Оператора U вещественный, симметричный, эрмитов и унитарный. Но поскольку это оператор отражения, то

$$U^2 = U \tag{9}$$

и нельзя взять $F = U$. Чтобы построить оператор F введем вектор $|\xi>$, задающий начальное состояние регистра:

$$|\xi> = 2^{-\frac{N}{2}} \sum_{j=0}^{2^N-1} |j> . \tag{10}$$

Действительно, вероятности того, что в классический компьютер будут выведен любой вектор $|j>$ равны друг другу:

$$W_0(|j>) = |<j|\xi>|^2 = 2^{-n} = \cos\varphi = \lambda^2 , \tag{11}$$

причем, если $N >> 1$, то $\lambda^2 << 1$. Через φ в (11) обозначен угол между векторами $|x_0>$ и $|\xi>$, причем $\varphi \approx \frac{\pi}{2} - \lambda$. Используя $|\xi>$ можно построить оператор V, согласно:

$$V = I - 2|\xi><\xi| . \tag{12}$$

Оператор V имеет те же свойства, что и оператор U. Из U и V можно построить оператор F, согласно:

$$F = VU \tag{13}$$

Оператор F из (13) можно представить в более удобной для оценки эффективности алгоритма форме, если ввести вектор $|\gamma>$, согласно:

$$|\gamma> = (1 - \lambda^2)^{-1/2}(|x_0> - \lambda|\xi>) . \tag{14}$$

Вектор $|\gamma>$ имеет единичную длину, ортогонален к $|\xi>$ и лежит в плоскости, натянутой на $|x_0>$ и $|\xi>$. Тогда, вводя операторы A и B, согласно

$$A = |\gamma><\gamma| + |\xi><\xi| \tag{15}$$

$$B = |\gamma><\xi| - |\xi><\gamma| \tag{16}$$

представим оператор F из (13) в форме

$$F = I - 2(1 - \lambda^2)A - 2\lambda(1 - \lambda^2)^{1/2}B . \tag{17}$$

Оператор F (17) является вещественным и унитарным, а значит – оператором вращения. Отметим, что A, B, I образуют 3-х мерную коммутативную алгебру с законом умножения:

$$A^2 = A,\ AB = BA = B,\ B^2 = -A, \tag{18}$$

причем $A^T = A$, $B^T = -B$.

Чтобы завершить построение алгоритма Гровера, нужно вычислить F^k и $W_k(|j>)$ из (6). Вводя обозначения $\cos\Theta = -(1-2\lambda^2)$, $\sin\Theta = -2\lambda(1-\lambda^2)^{1/2}$, выпишем действие F на $|\xi>$:

$$\begin{aligned} F|\xi> &= -2\lambda(1-\lambda^2)^{1/2}|\gamma> - (1-2\lambda^2)|\xi> = \\ &= \sin\Theta|\gamma> + \cos\Theta|\xi>. \end{aligned} \tag{19}$$

Используя (19), получим, что:

$$F|\xi> = \sin k\Theta|\gamma> + \cos k\Theta|\xi>. \tag{20}$$

Подставляя (20) в (6) найдем, что:

$$W_k|j> = \begin{cases} \cos^2\frac{(2k+1)\Theta}{2}, & |j> = |x_0> \\ \operatorname{ctg}^2\frac{\Theta}{2}\cdot\sin^2\frac{(2k+1)\Theta}{2}, & |j> \neq |x_0> \end{cases} \tag{21}$$

Вероятности $W_k|j>$ можно представить через полином Чебышева 1-го и 2-го рода $T_n(x)$ и $U_n(x)$ согласно:

$$W_k|j> = \begin{cases} T^2_{2k+1}(\lambda), & |j> = |x_0> \\ \lambda^2 U^2_{2k}(\lambda), & |j> \neq |x_0>. \end{cases} \tag{22}$$

Формулы (20) и (21) или (22) завершают построение алгоритма Гровера. Эффективность предложенного описания этого алгоритма будет изучена ниже.

2. Упорядочение разбиений файла на буферы и регистр квантового компьютера

Для того, чтобы использовать алгоритм Гровера для поиска нужного разбиения данного файла на буферы нужно связать регистр квантового компьютера с множеством всех возможных разбиений файла на буферы, т.е. построить пространство $V_{2^N-1} = V_2(\mathbb{C})^{\otimes N-1}$. Это фактически было сделано в работе [7]. Если обозначить через L – число буферов в разбиении файла, через m_l – число кортежей в l-м буфере, а через N – число кортежей в файле, то всего, с разными числами $L = 1, \ldots, N-1$ будет существовать 2^{N-1} разных разбиений:

$$\begin{array}{ll} 0 & m_1 = N, L = 1, \\ 1 & m_1 = 1, m_2 = N-1, m_3 = N-2, L = 2, \\ 2 & m_1 = 2, m_2 = N-2, m_3 = N-2, L = 2, \\ & \ldots \\ C^1_{N-1}+1 & m_1 = N-1, m_2 = 1, m_3 = N-2, \ldots, L = 3, \\ & \ldots \\ C^1_{N-1}+C^2_{N-1} & m_1 = N-2, m_2 = 1, m_3 = 1, \ldots, L = 3, \\ & \ldots \\ 2^{N-1}-1 & m_1 = 1, m_2 = 1, \ldots, m_N = 1, L = N. \end{array} \tag{23}$$

Если в $N-1$-мерном пространстве над Z_2 рассмотрим векторы $|j>$, причем компоненты вектора $|j>$, $j_j \in Z_2$ есть коэффициенты в двоичной записи натурального числа j:

$$j = \sum_{k=1}^{N-1} j_k 2^{k-1} \tag{24}$$

то каждой строке в (23) может быть поставлено во взаимно-однозначное соответствие число j из (24). Тем самым множество разбиений файла на буферы превращается в регистр квантового компьютера $V_2(\mathbb{C})^{\otimes N-1}$.

3. Оракул

Для построения оракула нужно знать вектор $|x_0>$. Как показано выше каждому разбиению файла соответствует вектор из $V_2(\mathbb{C})^{\otimes N-1}$. Поэтому достаточно определить нашим условия должен удовлетворять $|x_0>$. Разбиение файла на L буферов порождает L булевых полиномов $f_l(x_i), i = 1, \ldots, n$, где n – длина кортежа, а $l = 1, \ldots, L;\ L = 1, 2, \ldots, N$ таких, что кортежи, входящие в l-й буфер, являются решениями уравнения

$$f_l(x_i) = 0. \tag{25}$$

Если полиномы $f_l(x_i)$ могут быть получены из кодирующего уравнения

$$F(e_k^l) = f_l(x_i), \tag{26}$$

где аргументы $e_k, k = 1, \ldots, I$ булева полинома $F(e_k)$ заменяются на порождающие булевы полиномы $\mathfrak{\varphi}_p, p = 1, \ldots, P$. Существование кодирующего полинома $F(e_k)$, системы порождающих $\mathfrak{\varphi}_p$ и указанной выше замены $e_k \to \mathfrak{\varphi}_p$ и даст 1-е условие существования $|x_0>$. Вторым условием является то, что коэффициент сжатия файла k для $|x_0>$ должен удовлетворять условию $k > 1$. Коэффициент сжатия зависит не только от разбиения файла, которые заданы числом кортежей m_l, входящих в l-й буфер, но и параметрами I и P, характеризующими кодирующий полином F и систему порождающих $\mathfrak{\varphi}_p$, согласно:

$$k = \frac{n \sum_{l=1}^{L} m_l}{2^I + 2^n \cdot P + LI \log_2 P + \log_2 \prod_{l=1}^{L} \frac{C_{m_l-1}^{s_l-1}}{\prod_{k=1}^{s} n_k^l!}} \tag{27}$$

где n_k^l – числа повторов l-го буфера, $k = 1, \ldots, s_l$, а s_l – число разных кортежей в l-м буфере.

4. Эффективность

Для оценки эффективности алгоритма Гровера нужно оценить число $\boldsymbol{\sigma}$. Для этого заметим, что функции k из (21) или (22) являются периодическими с периодом $T = \frac{\pi}{\Theta}$. Так как $\Theta \approx \boldsymbol{\pi} + 2\boldsymbol{\lambda}$, то $T \approx 1 - \frac{2\lambda}{\pi}$. Тогда при $n >> 1$, $\boldsymbol{\lambda} << 1$, $T \approx 1$. Это значит, что алгоритм Гровера нужно остановить, когда он дойдет до первого локального максимума $W_k(|x_0>) = 1 - \boldsymbol{\varepsilon}$. Если положить $\boldsymbol{\varepsilon} << 1$, то получим, что $\boldsymbol{\sigma} \approx \frac{\pi + 2\varepsilon^{1/2}}{4\lambda} - \frac{1}{2}$, или, учитывая, что $\boldsymbol{\lambda} << 1$, будем иметь:

$$\boldsymbol{\sigma} \approx \frac{\boldsymbol{\pi}}{4} \cdot 2^{\frac{N}{2}} = 0.785 \cdot 2^{\frac{N}{2}} \approx 2^{\frac{N}{2}} \tag{28}$$

т. е. предложенный вариант квантового алгоритма поиска в неупорядоченной базе данных имеет ту же эффективность, что и полученная Гровером [8]. Однако его проще применить к задаче о разбиении файла на буферы. Главным недостатком предложенного подхода является то, что необходимо знать, что нужное разбиение существует и указать его номер x_0 в таблице (23). Если же такого разбиения нет, то невозможно построить оракул, а значит и сам алгоритм не существует.

Список литературы

1. *Валиев К. А., Канин А. А.* Квантовые компьютеры: надежды и реальность. Москва-Ижевск : НТЦ "Регулярная и хаотическая динамика 2001. 352 с.
2. *Китаев А., Шень А., Вялый М.* Классические и квантовые вычисления. М. : МЦНМО, ЧеРо, 1999. 192 с.
3. *Стин Э.* Квантовые вычисления. Москва-Ижевск : R&C Dynamica, 2000. 112 с.
4. *Толстопятов А. А.* Об алгоритмах решения задачи поиска в неупорядоченной базе данных на КЛАКах, КВАКах и БУЛЬКах // Вестник ИвГУ. 2000. Вып. 3. С. 93–100.
5. *Толстопятов А. А.* Возможные подходы к разбиению файла на буферы при булевом сжатии // Математика и ее приложения: журн. Иван. мат. о-ва. Иваново : ИвГУ, 2008. Вып. 1(6). С. 122–128.
6. *Толстопятов А. А.* Квантовый алгоритм разбиения файла на буферы // Математика и ее приложения: журн. Иван. мат. о-ва. Иваново : ИвГУ, 2011. Вып. 1(8). С. 113–120.
7. *Grover L.* A fast quantum mechanical algorithm for database search // STOC'28. 1996. P. 212–219.
8. *Shor P. W.* Algorithm for Quantum Computation // Discrete log and Factoring, FOCS'35. 1994. P. 124.

Поступила в редакцию 26.11.2012.