

СКОРОСТЬ СХОДИМОСТИ АЛГОРИТМОВ РЕШЕНИЯ ЛИНЕЙНОГО УРАВНЕНИЯ МЕТОДОМ КВАНТОВОГО ОТЖИГА¹⁾

© 2024 г. С. Б. Тихомиров^{1,*}, В. С. Шалгин^{2,**}

¹⁾ Pontifícia Universidade Católica do Rio de Janeiro – PUC-Rio, Rua Marquês de São Vicente, 225, Gávea – Rio de Janeiro, RJ – Brasil Cep: 22451-900 – Сх. Postal: 38097

²⁾ 199034 Санкт-Петербург, Университетская наб., 7/9, Санкт-Петербургский государственный университет, Россия

*e-mail: sergey.tikhomirov@gmail.com

**e-mail: st086496@student.spbu.ru

Поступила в редакцию 06.11.2023 г.

Переработанный вариант 26.12.2023 г.

Принята к публикации 06.02.2024 г.

Рассмотрены различные итеративные алгоритмы решения линейного уравнения $ax = b$ с помощью квантового вычислительного устройства, работающего по принципу квантового отжига. В предположении, что результат работы компьютера описывается распределением Больцмана, показано, при каких условиях алгоритмы решения уравнения сходятся, и дана оценка скорости их сходимости. Рассмотрено применение данного подхода для алгоритмов, использующих как бесконечное количество кубитов, так и малое количество кубитов. Библ. 31. Фиг. 2.

Ключевые слова: адиабатические квантовые вычисления, квантовый отжиг, линейное уравнение, распределение Больцмана, усеченное нормальное распределение.

DOI: 10.31857/S0044466924050061, EDN: YDHDHX

1. ВВЕДЕНИЕ

Квантовые вычисления представляют собой новую парадигму выполнения вычислений, предложенную Ю.И. Маниным (см. [1]) и Р. Фейнманом (см. [2]). Функционирование таких вычислительных устройств основано на квантовой механике. В основе вычислений лежат квантовые биты (кубиты), которые могут находиться не только в состоянии “0” или “1”, но и в их суперпозиции. Что более важно, квантовые биты могут находиться в запутанном состоянии. Таким образом, система из n кубитов описывается 2^n комплексными числами, более того операция над одним кубитом “меняет состояние” всех запутанных кубитов, что является основой квантового параллелизма — одновременного проведения вплоть до $O(2^n)$ операций над числами, описывающими состояние системы (см. [3], [4]). В квантовых вычислениях появляются и ограничения, не свойственные классическим вычислениям, например, невозможность копирования состояния и считывания состояния без его изменения. Для множества задач разработаны алгоритмы для квантовых компьютеров, работающие быстрее, чем их классические аналоги, вплоть до экспоненциального ускорения: например, алгоритм поиска (алгоритм Гровера) (см. [5]), разложение на множители (алгоритм Шора) (см. [6]), приближенное решение систем линейных уравнений (алгоритм ННЛ) (см. [7]).

Есть две основные модели квантовых вычислений: универсальная схемная модель (circuit based) (см. [3], [4]) и адиабатическая модель (см. [8]). В схемной модели операции выполняются одна за другой, как в классических вычислениях. Операции — квантовые вентили — представляют собой унитарные операторы, действующие на состояние системы кубитов. На основе этой модели функционируют квантовые компьютеры таких компаний,

¹⁾ Авторы благодарят Михаила Скопенкова за внимание к работе и полезные замечания. Исследование разд. 1, 2 и п. 3.1 выполнено при финансовой поддержке гранта РНФ № 21-11-00047. Исследование пп. 3.3 и 3.4 выполнено в Санкт-Петербургском международном математическом институте имени Леонарда Эйлера при финансовой поддержке Минобрнауки РФ (соглашение № 075–15–2022–287 от 06.04.2022). Исследование п. 3.2 выполнено при поддержке Projeto Paz и Coordenação de Aperfeiçoamento de Pessoal de Nível Superior — Brasil (CAPES) — Finance Code 001.

как IBM, Google, Intel. Основными ограничениями для практического использования таких квантовых компьютеров в настоящее время является небольшой размер (порядка 100 кубитов) и низкая точность выполнения операций.

Принцип работы адиабатических квантовых компьютеров основан не на последовательном выполнении операций, а на адиабатической теореме (см. [8]). Если изначально система находилась в состоянии минимальной энергии для гамильтониана H_1 , то при достаточно медленной эволюции гамильтониана:

$$H(t) = (1 - vt)H_1 + vtH_2, \quad t \in [0, 1/v],$$

в конечный момент времени система будет находиться в состоянии минимальной энергии для гамильтониана H_2 . Гамильтониан H_2 строится таким образом, что состояние минимальной энергии для него будет решением некоторой задачи. Такой подход позволяет эффективно решать задачи дискретной оптимизации, например, задачу коммивояжера (см. [9]) и задачу разрешимости булевых функций (см. [10]). Известно, что адиабатическая модель эквивалентна универсальной схемной модели квантовых вычислений (см. [11]).

С адиабатическими квантовыми вычислениями тесно связан квантовый отжиг (quantum annealing) — квантовый аналог алгоритма имитации отжига (см. [12]). Устройство, работающее по принципу квантового отжига, носит название “quantum annealer” (QA). Процесс поиска также начинается с состояния минимальной энергии системы для гамильтониана H_1 . Однако структура целевого гамильтониана H_2 более ограничена по сравнению с той, что фигурирует в общей адиабатической модели. А именно, квантовый отжиг нацелен на поиск точки минимума целевой функции модели Изинга (см. [12], [13]). В упрощенном виде ее можно представить так:

$$F(\sigma) = \sum_i h_i \sigma_i + \sum_{i < j} J_{ij} \sigma_i \sigma_j, \tag{1}$$

где $\sigma_i \in \{-1, 1\}$ представляют собой спины кубитов, а h_i и J_{ij} — коэффициенты линейных и квадратичных слагаемых соответственно. Результат работы квантового отжига — набор спинов кубитов $\{\sigma_i\}$, которые доставляют минимум функции $F(\sigma)$.

Точный результат может быть получен только в случае нулевой абсолютной температуры у QA, что в текущих реализациях является недостижимым. На практике такое устройство будет выдавать “сэмпл” (sample) из распределения Больцмана (см. [14]). Вероятность получить состояние σ зависит от значения функции F и параметра β :

$$P(\sigma) \propto e^{-\beta^2 F(\sigma)},$$

параметр $\beta \rightarrow \infty$ при стремлении температуры устройства к нулю. Ввиду этого квантовый отжиг является неточным, эвристическим алгоритмом. Кроме того, наличие шума также может внести помехи в работу компьютера.

Высокий интерес к модели квантового отжига обусловлен наличием реализации устройства, работающего по данной модели с большим количеством кубитов. Соответствующая реализация представлена устройствами компании D-Wave Systems (см. [15]), количество кубитов в которых достигает 5000. Дополнительными ограничениями в работе компьютера являются граф связности между кубитами и неточность выполнения операций (квантовый шум). Перед тем, как решить задачу с помощью компьютера D-Wave, ее необходимо перевести в термины модели Изинга. Согласованность распределения Больцмана и результата работы D-Wave достаточно хорошо продемонстрирована в [13], [16]–[18] (для обсуждения вопроса о наличии превосходства QA над классическими компьютерами см., например, [19], [20]).

В нашей работе мы будем опираться на модель квантовых вычислений, работающую по принципу квантового отжига. Одной из важных для приложений задач является решение систем линейных алгебраических уравнений. Задача решения системы $Ax = b$ эквивалентна задаче минимизации функции $\|Ax - b\|^2$, часто называемой в литературе “linear least squares problem” (LLS). Данная задача может быть решена с помощью QA путем ее перевода в целевую функцию (1) модели Изинга. Заметим, что линейность по переменной x является необходимым условием, так как в противном случае целевая функция не будет иметь форму (1).

Во многих работах исследуется решение задачи LLS с помощью QA. В [21] предлагается подход к переформулировке задачи LLS в форму, эквивалентную модели Изинга. Также в работе авторы предположили, что QA лучше всего подходит для решения задачи LLS в случае, если матрица A разреженная, или когда компоненты вектора x бинарные. Этот подход получил дальнейшее развитие. В [22] предлагается подход к решению произвольной системы линейных уравнений, а также приводятся условия, при которых возможно получить ускорение по сравнению с лучшим из известных классических алгоритмов решения произвольных систем линейных уравнений. В [23] рассматривается задача решения одного уравнения с одной неизвестной и задача LLS, подробно излагается процесс переформулировки исходной задачи в форму модели Изинга и встраивания полного графа задачи в граф компьютера D-Wave. В [24] исследуется вопрос о целесообразности использования

QA для решения систем уравнений и предлагается гибридный алгоритм решения линейных систем. Подход решения линейных систем с помощью QA нашел применение во многих задачах, в которых возникает необходимость решения системы уравнений: оценке линейной регрессии (см. [25]), для задач сейсмической томографии (см. [26]), в задаче определения преобразования из точечного множества (см. [27]), решения краевой задачи для эллиптических уравнений (см. [28]).

Во всех упомянутых работах были предложены алгоритмы решения линейных уравнений и систем, включая и итеративные алгоритмы (см. [23], [26]–[28]). Однако рассматривалась только экспериментальная постановка задачи, теоретические вопросы сходимости подобных итеративных алгоритмов не исследовались.

В настоящей работе рассматривается подход к решению задачи LLS, аналогичный [23], [26]–[28], для случая одного уравнения с одной неизвестной:

$$ax = b.$$

В рамках данного подхода мы учитываем подверженность квантового компьютера ошибкам и вероятностную природу QA. Нас будет интересовать вопрос устойчивости итеративного алгоритма к погрешностям, связанным с вероятностной обусловленностью результата работы компьютера. Ввиду этого простейшее уравнение является подходящей моделью для анализа сходимости и устойчивости итеративных алгоритмов. Так как результат работы QA подчиняется распределению Больцмана, это позволяет проводить оценку работы алгоритмов на основе нормальных распределений и их модификаций.

Мы рассматриваем различные итеративные алгоритмы, которые работают как для большого (стремящегося к бесконечности) количества кубитов, так и для малого числа кубитов. Характерной чертой рассмотренных алгоритмов является адаптация размера поправки на каждом шаге в зависимости от текущего значения невязки, что аналогично подходам из [23], [26]–[28] и расширяет предложенные ранее подходы из [21], [22], [25]. Мы доказываем, что предложенные алгоритмы сходятся к точному значению при достаточно малых ошибках в квантовом компьютере, и оцениваем скорость сходимости (см. теоремы 1, 2). Для реализации алгоритмов адаптации используем сложение, умножение и домножение на целые степени двойки, что соответствует сдвигу битов, и не используем деление на произвольное число.

В разд. 2 мы даем предварительные определения, переводим задачу решения уравнения в термины модели, эквивалентной модели Изинга, и устанавливаем вероятностную модель вычислений. В разд. 3 рассматриваем итеративные алгоритмы, основанные на последовательном улучшении приближенных решений уравнения. В п. 3.1 мы рассматриваем идеализированный случай, при котором результат работы QA подчиняется закону нормального распределения. В п. 3.2 рассматриваем общий подход к изучению скорости сходимости итеративных алгоритмов. Случай усеченного нормального распределения решений, соответствующий бесконечному количеству кубитов, представлен в п. 3.3, случай распределения Больцмана, соответствующий конечному количеству кубитов, — в п. 3.4.

2. ПРЕДВАРИТЕЛЬНЫЕ ПОСТРОЕНИЯ

Модель Изинга эквивалентна так называемой модели QUBO (quadratic unconstrained binary optimization) (см. [29]):

$$H(q_1, \dots, q_n) = \sum_{i=1}^n \sum_{j=i}^n Q_{ij} q_i q_j, \quad (2)$$

где $(Q_{ij})_{i,j=1}^n$ — верхнетреугольная квадратная матрица порядка n , $q_i \in \{0, 1\}$. Для переформулировки задачи решения уравнения $ax = b$ в задачу QUBO мы используем функцию

$$H(x) = (ax - b)^2. \quad (3)$$

Переменную x представляем с конечной точностью в виде

$$x = \vartheta q_p + \sum_{i=r}^{p-1} 2^i q_i, \quad (4)$$

где $p, r \in \mathbb{Z}$, $r < p$, $\vartheta = -2^p + 2^r$, $q_i \in \{0, 1\}$. Бит q_p отвечает за знак переменной x . Роль константы ϑ заключается в том, что набор битов q_r, \dots, q_{p-1} для отрицательных значений x представляет собой дополнительный код к аналогичному набору битов для положительных значений x . Отметим, что можно использовать и другие представления переменной (см. [22], [23], [28]). Обозначим через $\Omega_{r,p}$ множество чисел вида (4). Тогда

$$\Omega_{r,p} = \left\{ \pm \sum_{i=r}^{p-1} q_i 2^i : q_i \in \{0, 1\} \right\}.$$

После подстановки (4) в (3) и отбрасывания постоянного слагаемого получаем целевую функцию вида (2):

$$H(q_r, \dots, q_p) = \sum_{i=r}^p \sum_{j=i}^p Q_{ij} q_i q_j,$$

где

$$Q_{ii} = \begin{cases} a^2 \vartheta^2 - 2ab\vartheta, & i = p, \\ 2^{2i} a^2 - 2^{i+1} ab, & r \leq i \leq p-1, \end{cases} \quad Q_{ij} = \begin{cases} 2^{i+1} a^2 \vartheta, & j = p, r \leq i \leq p-1, \\ 2^{i+j+1} a^2, & r \leq i < j \leq p-1. \end{cases}$$

Как было сказано ранее, ошибки в работе QA имеют вероятностный характер, и мы считаем, что они подчиняются распределению Больцмана. Будем использовать вариацию этого распределения, определенную ниже.

Определение 1. Пусть Ω — конечное подмножество вещественных чисел, $H : \mathbb{R} \rightarrow [0, +\infty)$, $\beta > 0$. *Распределением Больцмана* $B(\beta, \Omega, H(x))$ на множестве Ω с параметром β и целевой функцией $H(x)$ назовем вероятностное распределение на Ω , в котором вероятность элемента $x \in \Omega$ определяется как

$$P(x) = \frac{1}{Z} e^{-\beta^2 H(x)}, \quad \text{где } Z = \sum_{y \in \Omega} e^{-\beta^2 H(y)}.$$

Чем меньше значение $H(x)$, тем больше вероятность получить x в качестве результата работы компьютера. Параметр β отражает точность работы компьютера. Чем больше значение β , тем более вероятно результат работы компьютера будет близок к точке минимума функции $H(x)$ на множестве Ω . Если $\beta \rightarrow +\infty$, то компьютер будет работать без ошибок.

Вернемся к целевой функции (3). Если предположить, что количество кубитов в QA стремится к бесконечности, и для двоичного представления (4) переменной x мы используем как все положительные степени двойки, так и все отрицательные, то распределение решений будет стремиться к нормальному, что показывает следующее очевидное предложение.

Предложение 1. Пусть $r, p \in \mathbb{Z}$, $r < p$, $\Omega_{r,p} = \left\{ \pm \sum_{i=r}^{p-1} q_i 2^i : q_i \in \{0, 1\} \right\}$, $\beta > 0$, $a, b \in \mathbb{R}$, $a \neq 0$. Тогда

$$\lim_{\substack{p \rightarrow +\infty \\ r \rightarrow -\infty}} B(\beta, \Omega_{r,p}, (ax - b)^2) = \mathcal{N}\left(\frac{b}{a}, \frac{1}{2a^2\beta^2}\right)$$

по распределению.

Таким образом, в качестве приближения к распределению Больцмана мы можем использовать нормальное распределение.

3. УЛУЧШЕНИЕ РЕШЕНИЯ УРАВНЕНИЯ

3.1. Модель улучшения решения, основанная на нормальном распределении

В этом пункте мы будем предполагать, что результат работы QA по решению уравнения $ax = b$ имеет распределение $\mathcal{N}\left(\frac{b}{a}, \frac{1}{2a^2\beta^2}\right)$ согласно предложению 1. Ниже рассмотрим, как будут распределены ошибки приближения к решению, если мы будем итерировать алгоритм. Пусть x_n — n -ое фиксированное приближение к решению уравнения $ax = b$. Точное решение x мы можем представить как сумму x_n и поправки: $x = x_n + \Delta_n$. Подставляя ее в исходное уравнение, мы получаем уравнение относительно поправки Δ_n :

$$a\Delta_n = b - ax_n. \tag{5}$$

Пусть целое число l_n такое, что

$$\frac{1}{2^{l_n+1}} < |b - ax_n| \leq \frac{1}{2^{l_n}}. \tag{6}$$

Вместо уравнения (5) будем решать на QA уравнение

$$a\tilde{\Delta}_n = 2^{l_n}(b - ax_n) \tag{7}$$

с неизвестной $\tilde{\Delta}_n$. По предположению имеем, что

$$\tilde{\Delta}_n \sim \mathcal{N}\left(\frac{2^{l_n}(b - ax_n)}{a}, \frac{1}{2a^2\beta^2}\right). \tag{8}$$

Так как $\Delta_n = \frac{1}{2^{l_n}} \tilde{\Delta}_n$, то по свойствам нормального распределения получаем, что

$$\Delta_n \sim \mathcal{N}\left(\frac{b}{a} - x_n, \frac{1}{2^{2l_n}} \cdot \frac{1}{2a^2\beta^2}\right).$$

Пусть $\xi_n \sim \mathcal{N}(0, 1)$, тогда

$$\Delta_n \stackrel{d}{=} \frac{b}{a} - x_n + \frac{\xi_n}{2^{l_n} \sqrt{2a\beta}}.$$

Следующее приближение к решению будем вычислять по формуле

$$x_{n+1} = x_n + \Delta_n.$$

Заметим, что следующая поправка Δ_{n+1} и число l_{n+1} зависят от предыдущей поправки Δ_n . Следующее предложение позволяет построить последовательность приближений x_n , учитывая эти зависимости.

Предложение 2. Пусть $a, b \in \mathbb{R} \setminus \{0\}$, $\sigma > 0$. Пусть $\xi_n \sim \mathcal{N}(0, 1)$, $n \geq 0$, — независимые в совокупности случайные величины. Построим последовательности случайных величин $x_n, x'_n, l_n, l'_n, \Delta_n, \Delta'_n$ при $n \geq 0$. Положим $x_0 = x'_0 = l_0 = l'_0 = 0$. Дадим дальнейшие определения при $n \geq 0$. Пусть условное распределение случайной величины Δ_n при условии x_n равно $\mathcal{N}\left(\frac{b}{a} - x_n, \frac{\sigma^2}{2^{2l_n}}\right)$. Положим

$$\Delta'_n = \frac{b}{a} - x'_n + \frac{\sigma \xi_n}{2^{l'_n}},$$

$$x_{n+1} = x_n + \Delta_n,$$

$$x'_{n+1} = x'_n + \Delta'_n.$$

Определим l_{n+1} и l'_{n+1} следующим образом: $l_{n+1}, l'_{n+1} \in \mathbb{Z}$ и

$$2^{l_{n+1}} |b - ax_{n+1}|, 2^{l'_{n+1}} |b - ax'_{n+1}| \in (1/2, 1].$$

Тогда $(\Delta_0, \dots, \Delta_n) \stackrel{d}{=} (\Delta'_0, \dots, \Delta'_n)$ для любого $n \geq 0$.

Доказательство. Доказательство будем вести индукцией по n . При $n = 0$ имеем $\Delta_0 \sim \mathcal{N}\left(\frac{b}{a}, \sigma^2\right)$ и $\Delta'_0 = \frac{b}{a} + \sigma \xi_0 \sim \mathcal{N}\left(\frac{b}{a}, \sigma^2\right)$. Значит, $\Delta_0 \stackrel{d}{=} \Delta'_0$.

Пусть $(\Delta_0, \dots, \Delta_{n-1}) \stackrel{d}{=} (\Delta'_0, \dots, \Delta'_{n-1})$, то есть для любого борелевского множества $A \subset \mathbb{R}^n$ выполнено $P((\Delta_0, \dots, \Delta_{n-1}) \in A) = P((\Delta'_0, \dots, \Delta'_{n-1}) \in A)$. Покажем, что тогда для любого борелевского множества $A \subset \mathbb{R}^{n+1}$ будет выполнено $P((\Delta_0, \dots, \Delta_n) \in A) = P((\Delta'_0, \dots, \Delta'_n) \in A)$. Достаточно доказать, что

$$P((\Delta_0, \dots, \Delta_n) \in A_0 \times \dots \times A_n) = P((\Delta'_0, \dots, \Delta'_n) \in A_0 \times \dots \times A_n)$$

для любых борелевских $A_i \subset \mathbb{R}$, $i = 0, 1, \dots, n$. По формуле полной вероятности получаем, что вероятность $P((\Delta'_0, \dots, \Delta'_n) \in A_0 \times \dots \times A_n)$ равна

$$\int_{\mathbb{R}^n} P((\Delta'_0, \dots, \Delta'_n) \in A_0 \times \dots \times A_n \mid (\Delta'_0, \dots, \Delta'_{n-1}) = (r_0, \dots, r_{n-1})) dP'_{n-1},$$

где P'_{n-1} — распределение вектора $(\Delta'_0, \dots, \Delta'_{n-1})$ и интегрирование ведется по переменным r_0, \dots, r_{n-1} . Если $(r_0, \dots, r_{n-1}) \notin A_0 \times \dots \times A_{n-1}$, то $(\Delta'_0, \dots, \Delta'_n) \notin A_0 \times \dots \times A_n$, и вероятность под знаком интеграла равна нулю. Поэтому последний интеграл равен

$$\int_{A_0 \times \dots \times A_{n-1}} P(\Delta'_n \in A_n \mid \Delta'_0 = r_0, \dots, \Delta'_{n-1} = r_{n-1}) dP'_{n-1}. \quad (9)$$

Зафиксируем числа r_0, \dots, r_{n-1} . Положим $S = r_0 + \dots + r_{n-1}$ и возьмем целое l таким, что $2^l |b - aS| \in (1/2, 1]$. Покажем, что условное распределение случайной величины Δ'_n при условии $\Delta'_0 = r_0, \dots, \Delta'_{n-1} = r_{n-1}$ равно условному распределению случайной величины Δ_n при условии $\Delta_0 = r_0, \dots, \Delta_{n-1} = r_{n-1}$.

При этих условиях $x_n = S$ и $x'_n = S$. Тогда по определению l'_n получим, что $l'_n = l$. По определению Δ'_n имеем

$$\Delta'_n = \frac{b}{a} - S + \frac{\sigma \xi_n}{2^l} \sim \mathcal{N}\left(\frac{b}{a} - S, \frac{\sigma^2}{2^{2l}}\right),$$

что есть условное распределение случайной величины Δ_n при условии $x_n = S$.

По индукционному предположению мы имеем $P'_{n-1} = P_{n-1}$, где P_{n-1} — распределение вектора $(\Delta_0, \dots, \Delta_{n-1})$. Значит, интеграл (9) равен

$$\begin{aligned} & \int_{A_0 \times \dots \times A_{n-1}} P(\Delta_n \in A_n | \Delta_0 = r_0, \dots, \Delta_{n-1} = r_{n-1}) dP_{n-1} = \\ & = \int_{\mathbb{R}^n} P((\Delta_0, \dots, \Delta_n) \in A_0 \times \dots \times A_n | (\Delta_0, \dots, \Delta_{n-1}) = (r_0, \dots, r_{n-1})) dP_{n-1}. \end{aligned}$$

Последний интеграл равен вероятности $P((\Delta_0, \dots, \Delta_n) \in A_0 \times \dots \times A_n)$, что и требовалось показать.

Полагая, что $\sigma^2 = \frac{1}{2a^2\beta^2}$ в предложении 2, получаем, что $x'_{n+1} = \frac{b}{a} + \frac{\xi_n}{2^{l_n}\sqrt{2a\beta}}$.

Следующая теорема показывает, что эта последовательность при определенных условиях сходится к решению уравнения $ax = b$.

Теорема 1. Пусть $a, b \in \mathbb{R}$, $a \neq 0$, $\beta > 0$, γ — постоянная Эйлера–Маскерони. Зафиксируем последовательность независимых в совокупности случайных величин $\xi_n \sim \mathcal{N}(0, 1)$, $n \geq 0$. Построим последовательности случайных величин l_n и x_n по правилу $l_0 = 0$, $x_0 = 0$, и при $n \geq 0$ положим $x_{n+1} = \frac{b}{a} + \frac{\xi_n}{2^{l_n}\sqrt{2a\beta}}$, целое число l_{n+1} таково, что $2^{l_{n+1}}|b - ax_{n+1}| \in (\frac{1}{2}, 1]$. Тогда

- 1) если $s \in [1, \beta e^{\gamma/2})$, то $s^n (x_n - \frac{b}{a}) \xrightarrow[n \rightarrow \infty]{n.н.} 0$,
- 2) если $s > 2\beta e^{\gamma/2}$, то $s^n (x_n - \frac{b}{a}) \xrightarrow[n \rightarrow \infty]{n.н.} \infty$,
- 3) если $\beta < \frac{1}{2}e^{-\gamma/2}$, то $x_n \xrightarrow[n \rightarrow \infty]{n.н.} \infty$.

Замечание 1. Процесс, описанный в теореме, соответствует процессу работы квантового компьютера. Процесс последовательного улучшения решения описывался в статьях [23], [26]–[28], однако теоретические вопросы сходимости подобных итеративных алгоритмов не исследовались.

Из п. 1) теоремы 1 следует, что если $\beta > e^{-\gamma/2} \approx \frac{3}{4}$, то последовательность x_n будет сходиться к решению уравнения $ax = b$ почти наверное и притом с экспоненциальной скоростью. Пункт 2) устанавливает верхнюю границу скорости сходимости. Пункт 3) устанавливает достаточное условие расходимости последовательности x_n : если β мало (точность работы QA слишком плоха), то последовательность x_n будет расходиться.

Для доказательства теоремы 1 нам понадобится следующая лемма, непосредственно следующая из усиленного закона больших чисел Колмогорова (см. [30]).

Лемма 1. Пусть $\delta > 0$, и пусть $(X_i)_{i=1}^\infty$ — независимые в совокупности случайные величины такие, что для любого натурального i существуют $\mathbb{E} \ln |X_i|$ и $\mathbb{E} \ln^2 |X_i|$. Пусть дисперсии $\text{Var}(\ln |X_i|)$ ограничены в совокупности. Тогда

- 1) если $\mathbb{E} \ln |X_i| < -\delta$ для любого i , то $X_1 \cdots X_n \xrightarrow[n \rightarrow \infty]{n.н.} 0$,
- 2) если $\mathbb{E} \ln |X_i| > \delta$ для любого i , то $X_1 \cdots X_n \xrightarrow[n \rightarrow \infty]{n.н.} \infty$.

Доказательство теоремы 1. Проведем предварительные построения. Рассмотрим случайную величину $z_n = \frac{b}{a} - x_n$. Оценим сверху $|z_{n+1}|$, используя определение случайных величин x_n и l_n :

$$|z_{n+1}| = \left| \frac{\xi_n}{2^{l_n}\sqrt{2a\beta}} \right| = \left| \frac{(b - ax_n)\xi_n}{2^{l_n}(b - ax_n)\sqrt{2a\beta}} \right| < \left| \frac{\sqrt{2}(b - ax_n)\xi_n}{a\beta} \right| = \frac{\sqrt{2}}{\beta} |z_n \xi_n|.$$

Так как $|z_1| = \frac{|\xi_0|}{\sqrt{2a\beta}}$, то

$$|z_{n+1}| < \left| \frac{1}{2a} \right| \left(\frac{\sqrt{2}}{\beta} \right)^{n+1} |\xi_0 \cdots \xi_n|. \tag{10}$$

Аналогично получаем оценку $|z_{n+1}|$ снизу:

$$|z_{n+1}| \geq \left| \frac{1}{a} \right| \left(\frac{1}{\sqrt{2\beta}} \right)^{n+1} |\xi_0 \cdots \xi_n|. \quad (11)$$

Найдем математическое ожидание случайной величины $\ln |\xi_0|$:

$$\mathbb{E} \ln |\xi_0| = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-x^2/2} \ln |x| dx = \frac{2}{\sqrt{\pi}} \int_0^{\infty} e^{-x^2} \ln(\sqrt{2}x) dx = \ln \frac{1}{\sqrt{2}e^{\gamma/2}},$$

где мы воспользовались соотношением для γ из [31]:

$$\int_0^{\infty} e^{-x^2} \ln x dx = -\frac{\sqrt{\pi}}{4} (\gamma + \ln 4).$$

Тогда

$$\mathbb{E} \ln \frac{\sqrt{2}|\xi_0|}{\beta} = \ln \frac{1}{\beta e^{\gamma/2}}, \quad \mathbb{E} \ln \frac{|\xi_0|}{\sqrt{2\beta}} = \ln \frac{1}{2\beta e^{\gamma/2}}. \quad (12)$$

Докажем п. 1). Достаточно показать, что $\left(\frac{s\sqrt{2}}{\beta}\right)^n |\xi_0 \cdots \xi_{n-1}| \xrightarrow[n \rightarrow \infty]{\text{п.н.}} 0$, исходя из неравенства (10). Ввиду (12) и того, что $s \in [1, \beta e^{\gamma/2})$, имеем

$$\mathbb{E} \ln \frac{s\sqrt{2}|\xi_0|}{\beta} = \ln \frac{1}{\beta e^{\gamma/2}} + \ln s < 0.$$

Значит, по лемме 1 произведение $\left(\frac{s\sqrt{2}}{\beta}\right)^n |\xi_0 \cdots \xi_{n-1}|$ сходится к нулю почти наверное.

Докажем п. 2). Достаточно показать, что $\left(\frac{s}{\sqrt{2\beta}}\right)^n |\xi_0 \cdots \xi_{n-1}| \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \infty$, исходя из неравенства (11). Ввиду (12) и того, что $s > 2\beta e^{\gamma/2}$, имеем

$$\mathbb{E} \ln \frac{s|\xi_0|}{\sqrt{2\beta}} = \ln \frac{1}{2\beta e^{\gamma/2}} + \ln s > 0.$$

Значит, по лемме 1 произведение $\left(\frac{s}{\sqrt{2\beta}}\right)^n |\xi_0 \cdots \xi_{n-1}|$ сходится к бесконечности почти наверное.

Докажем п. 3). Достаточно показать, что $\left(\frac{1}{\sqrt{2\beta}}\right)^n |\xi_0 \cdots \xi_{n-1}| \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \infty$, исходя из неравенства (11). Ввиду (12) и того, что $\beta < \frac{1}{2}e^{-\gamma/2}$, имеем $\mathbb{E} \ln \frac{|\xi_0|}{\sqrt{2\beta}} = \ln \frac{1}{2\beta e^{\gamma/2}} > 0$. Значит, по лемме 1 произведение $\left(\frac{1}{\sqrt{2\beta}}\right)^n |\xi_0 \cdots \xi_{n-1}|$ сходится к бесконечности почти наверное.

3.2. Общий подход к исследованию сходимости алгоритмов решения линейного уравнения

В этом пункте рассмотрим общий подход к построению и исследованию сходимости алгоритма, решающего уравнение $ax = b$. Мы рассмотрим общую схему последовательных приближений и докажем теорему 2, позволяющую оценить скорость сходимости подобных алгоритмов. Домножая a и b на одинаковую степень двойки, можно добиться выполнения неравенства

$$1/2 \leq a < 1. \quad (13)$$

В дальнейшем в статье будем предполагать, что выполнено неравенство (13).

Вернемся к построению последовательности приближений к решению уравнения $ax = b$. Имея очередное фиксированное приближение x_n , решаем на QA уравнение (7) относительно $\tilde{\Delta}_n$, где целое l_n выбирается в соответствии с (6). Следующее приближение вычисляется как

$$x_{n+1} = x_n + 2^{-l_n} \tilde{\Delta}_n. \quad (14)$$

Распределение случайной величины $\tilde{\Delta}_n$ зависит от выбранного нами алгоритма и определяется значениями c_n , $\text{sign}(b - ax_n)$, a , β , где

$$c_n = \frac{1}{2^{l_n} |b - ax_n|}. \quad (15)$$

Заметим, что $c_n \in [1, 2)$. В этом пункте мы не фиксируем конкретный алгоритм и соответствующее распределение $\tilde{\Delta}_n$. В дальнейшем будем предполагать, что зависимость от $\text{sign}(b - ax_n)$ имеет специальный вид, а именно,

существует такая функция $q(u, c, a, \beta)$, что если η — случайная величина, равномерно распределенная на $[0, 1]$, то

$$\tilde{\Delta}_n | x_n \stackrel{d}{=} \text{sign}(b - ax_n) q(\eta, c_n, a, \beta) | x_n. \tag{16}$$

Условие (16) означает, что распределение поправок $\tilde{\Delta}_n$ в случае положительных и отрицательных невязок отличается лишь знаком.

Заметим, что если функция $q(\cdot, c_n, a, \beta)$ равна обратной функции распределения закона $\mathcal{N}\left(\frac{1}{ac_n}, \frac{1}{2a^2\beta^2}\right)$, то $q(\eta, c_n, a, \beta) \sim \mathcal{N}\left(\frac{1}{ac_n}, \frac{1}{2a^2\beta^2}\right)$, и поправка $\tilde{\Delta}_n$ распределена, как в (8).

По аналогии с предложением 2 верно следующее

Предложение 3. Пусть $b \neq 0, \beta > 0$. Пусть $\eta_n, n \geq 0$, — независимые в совокупности случайные величины, равномерно распределенные на промежутке $[0, 1]$. Зафиксируем функцию $q(\eta, c, a, \beta)$, определяемую выбранным алгоритмом последовательных приближений.

Введем функции

$$G_1(u_0) = q(u_0, 1/b, a, \beta) - \frac{b}{a},$$

$$G_{n+1}(u_0, \dots, u_n) = G_n(u_0, \dots, u_{n-1}) (1 - ac'_n q(u_n, c'_n, a, \beta)), \quad n \geq 1, \tag{17}$$

где $u_i \in [0, 1], c'_n = \frac{1}{2^{l'_n} |aG_n|}$, и целое l'_n выбрано так, что $2^{l'_n} |aG_n| \in (\frac{1}{2}, 1]$.

Пусть $x_0 = l_0 = 0$. Пусть $\tilde{\Delta}_0 \stackrel{d}{=} q(\eta_0, 1/b, a, \beta)$ и при $n \geq 1$ выполнено (16) для $\eta = \eta_n$, где l_n и c_n определяются в (6) и (15). При $n \geq 0$ положим

$$x_{n+1} = x_n + 2^{-l_n} \tilde{\Delta}_n,$$

$$x'_{n+1} = \frac{b}{a} + G_{n+1}(\eta_0, \dots, \eta_n). \tag{18}$$

Тогда $(x_1, \dots, x_n) \stackrel{d}{=} (x'_1, \dots, x'_n)$ для любого $n \geq 1$.

Здесь и далее будем рассматривать последовательность x_n , заданную равенством (18). Последовательность x_n определяется выбором функции q . В следующей теореме мы будем использовать обозначения и определения из предложения 3.

Теорема 2. Введем функцию

$$r(u, a, \beta) = \max_{c \in [1, 2]} |1 - c \cdot a \cdot q(u, c, a, \beta)|. \tag{19}$$

Пусть $E(a, \beta)$ — математическое ожидание случайной величины $\ln r(\eta, a, \beta)$, где η — случайная величина, равномерно распределенная на $[0, 1]$:

$$E(a, \beta) = \int_0^1 \ln r(u, a, \beta) du.$$

Тогда

- 1) если $E(a, \beta) < 0$, то $x_n \xrightarrow[n \rightarrow \infty]{\text{п.н.}} \frac{b}{a}$,
- 2) если $\ln s + E(a, \beta) < 0$, то $s^n (x_n - \frac{b}{a}) \xrightarrow[n \rightarrow \infty]{\text{п.н.}} 0$.

Доказательство. Докажем п. 1). Исходя из (18), достаточно показать, что $G_{n+1}(\eta_0, \dots, \eta_n) \xrightarrow{\text{п.н.}} 0$. Используя (17), получаем

$$|G_{n+1}(\eta_0, \dots, \eta_n)| = |G_1(\eta_0)| \prod_{i=1}^n |1 - ac_i q(\eta_i, c_i, a, \beta)| \leq |G_1(\eta_0)| \prod_{i=1}^n r(\eta_i, a, \beta).$$

Так как математические ожидания случайных величин $\ln r(\eta_i, a, \beta)$ меньше нуля, то по лемме 1 получаем требуемое.

Докажем п. 2). Достаточно показать, что $s^{n+1} G_{n+1}(\eta_0, \dots, \eta_n) \xrightarrow{\text{п.н.}} 0$. Аналогично получаем

$$s^{n+1} |G_{n+1}(\eta_0, \dots, \eta_n)| \leq s |G_1(\eta_0)| \prod_{i=1}^n (s r(\eta_i, a, \beta)).$$

Из неравенства $\mathbb{E} \ln(sr(\eta_i, a, \beta)) < 0$ следует требуемое.

Замечание 2. В теореме 2 условие сходимости алгоритма зависит от a и β , в то время как в теореме 1 условие сходимости зависит только от β . Это порождает вопрос: от чего может зависеть скорость сходимости аналогичных итеративных алгоритмов для систем линейных уравнений – размер матрицы, число обусловленности и др.? Рассмотрение систем линейных уравнений выходит за рамки работы и нуждается в отдельном исследовании.

3.3. Модели вычислений, основанные на усеченном нормальном распределении

В разд. 2 и п. 3.1 мы рассматривали представление (4) переменной x по положительным и отрицательным степеням двойки. В текущем пункте мы рассмотрим более “гибкое” представление:

$$x = (d_2 - d_1) \sum_{i=1}^r q_i 2^{-i} + d_1, \quad (20)$$

где $q_i \in \{0, 1\}$, $d_1 < d_2$, $r \in \mathbb{N}$. В таком представлении x принимает значения в промежутке $[d_1, d_2]$. Коэффициенты Q_{ij} в модели QUBO (2) находятся из подстановки представления (20) в функцию (3).

Мы будем рассматривать несколько различных алгоритмов поиска поправки, отличающихся значениями d_1, d_2 . На каждом шаге будем искать поправку $\tilde{\Delta}_n$ в шаге (14) алгоритма как решение уравнения (7) на промежутке $\text{sign}(b - ax_n)[d_1, d_2]$, представляя $\tilde{\Delta}_n$ по аналогии с формулой (20):

$$\tilde{\Delta}_n = \text{sign}(b - ax_n) \left((d_2 - d_1) \sum_{i=1}^r q_i 2^{-i} + d_1 \right). \quad (21)$$

Таким образом, числа d_1, d_2 определяют алгоритм и могут влиять на характер его сходимости. В дальнейшем рассмотрим четыре алгоритма с различными значениями d_1, d_2 .

Предположим, что количество кубитов в QA стремится к бесконечности, т.е. $r \rightarrow \infty$. Посмотрим, как при этом ведет себя распределение Больцмана на множестве, состоящем из чисел вида (20), с целевой функцией $(ax - b)^2$. Для начала введем следующее определение.

Определение 2. Пусть $\sigma > 0$, $\mu, d_1, d_2 \in \mathbb{R}$, $d_1 < d_2$. Усеченное нормальное распределение $\mathcal{N}(\mu, \sigma^2, d_1, d_2)$ – это распределение с функцией плотности

$$f(t) \propto e^{-\frac{(t-\mu)^2}{2\sigma^2}} \mathbf{1}_{(d_1, d_2)}(t).$$

Если $d_1 > d_2$, то обозначение $\mathcal{N}(\mu, \sigma^2, d_1, d_2)$ будет пониматься как усеченное нормальное распределение $\mathcal{N}(\mu, \sigma^2, d_2, d_1)$, а обозначение (d_1, d_2) будет пониматься как интервал (d_2, d_1) .

Предложение 4. Пусть $\beta > 0$, $a, b \in \mathbb{R}$, $a \neq 0$, $d_1 < d_2$, $r \in \mathbb{N}$. Тогда

$$\mathbb{V} \left(\beta, \left\{ (d_2 - d_1) \sum_{i=1}^r q_i 2^{-i} + d_1 : q_i \in \{0, 1\} \right\}, (ax - b)^2 \right) \xrightarrow[r \rightarrow \infty]{d} \mathcal{N} \left(\frac{b}{a}, \frac{1}{2a^2\beta^2}, d_1, d_2 \right).$$

Таким образом, мы можем использовать усеченное нормальное распределение в качестве приближения к распределению Больцмана.

Согласно представлению (21) и предложению 4 будем считать, что

$$\tilde{\Delta}_n \sim \text{sign}(b - ax_n) \mathcal{N} \left(\frac{1}{ac_n}, \frac{1}{2a^2\beta^2}, d_1, d_2 \right).$$

Обозначим через $q(u, c, a, \beta)$ такие функции, что если c, a, β фиксированы и η равномерно распределена на промежутке $[0, 1]$, то

$$q(\eta, c, a, \beta) \sim \mathcal{N} \left(\frac{1}{ac}, \frac{1}{2a^2\beta^2}, d_1, d_2 \right).$$

Функция q представляет собой обратную функцию распределения соответствующего закона распределения и в явном виде записывается следующим образом:

$$q(u, c, a, \beta) = \frac{1}{ac} + \frac{1}{a\beta} \text{erf}^{-1} \left((1 - u) \text{erf} \left(d_1 a \beta + \frac{\beta}{c} \right) + u \text{erf} \left(d_2 a \beta - \frac{\beta}{c} \right) \right),$$

где $\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$ – функция ошибок.

Из неравенств (6),(13) следует, что точное решение уравнения (7) удовлетворяет неравенствам

$$|\tilde{\Delta}_n| \leq 2, \quad \text{sign}(b - ax_n)\tilde{\Delta}_n \in [1/2, 2]. \tag{22}$$

Числа d_1, d_2 определяют функцию q и соответственно задают алгоритм улучшения решения. Ниже приводятся четыре алгоритма для соответствующих значений d_1, d_2 , три из них учитывают правильный знак поправки на каждом шаге. Ввиду (22) мы ограничиваемся значениями d_1, d_2 такими, что $|d_1|, |d_2| \leq 2$.

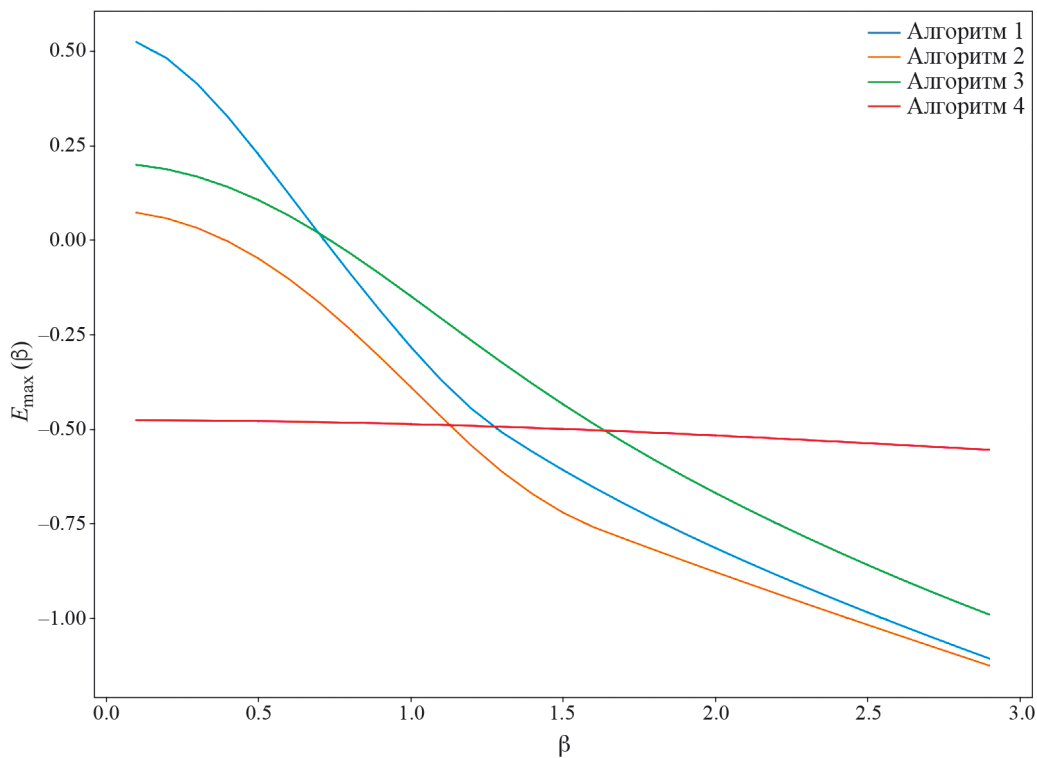
Алгоритм 1: $d_1 = -2, d_2 = 2$. Не учитывает знак поправки, а лишь наибольшее значение модуля $|\tilde{\Delta}_n|$.

Алгоритм 2: $d_1 = 0, d_2 = 2$. Учитывает знак поправки и наибольшее значение модуля $|\tilde{\Delta}_n|$.

Алгоритм 3: $d_1 = 1/2, d_2 = 2$. Учитывает знак поправки и наибольшее и наименьшее значения модуля $|\tilde{\Delta}_n|$.

Алгоритм 4: $d_1 = 1/2, d_2 = 1$. Консервативный алгоритм, при котором гарантированно выполняется неравенство $r(u, a, \beta) \leq 1$, где r определено в (19). По теореме 2 такой алгоритм сходится для любых a, β . Отметим, что при этом точное значение $\frac{1}{ac_n}$ поправки $\tilde{\Delta}_n$ не всегда лежит в интервале $[d_1, d_2]$.

На фиг. 1 приведены сравнительные графики для соответствующих функций $E_{\max}(\beta) = \max_{a \in [1/2, 1]} E(a, \beta)$, дающие пессимистичную оценку на скорость сходимости алгоритма. Если $E_{\max}(\beta) < 0$, то алгоритм сходится при любых $a \in [0.5, 1), b \in \mathbb{R}$, чем меньше значение $E_{\max}(\beta)$, тем сходимость быстрее.



Фиг. 1. Графики функций $E_{\max}(\beta)$ для различных алгоритмов.

Поскольку алгоритм 4 при любом выборе $\tilde{\Delta}_n$ уменьшает значение невязки, то он сходится в любом случае, что отражено в отрицательности функции $E_{\max}(\beta)$ для всех значений β . При этом, поскольку включение $\frac{1}{ac} \in [0.5, 1]$ не всегда выполнено, алгоритм показывает не быструю скорость сходимости даже при больших значениях β . Алгоритмы 1–3 ведут себя приблизительно одинаково при больших значениях β , это объясняется высокой вероятностью получить значение $\tilde{\Delta}_n$, близкое к точному решению уравнения (7). Лучшие показатели сходимости наблюдаются у алгоритма 2, учитывающего знак, но разрешающего малые значения поправки на каждом шаге. Его преимущество над алгоритмом 3 вероятно объясняется уменьшением веса хвоста усеченного нормального распределения при котором $r(u, a, \beta) > 1$.

3.4. Модели вычислений, основанные на распределении Больцмана

В пп. 3.1, 3.3 мы рассматривали непрерывные распределения, приближающие распределение Больцмана. Здесь мы непосредственно рассмотрим модель вычислений, основанную на распределении Больцмана, и будем считать, что количество кубитов в QA конечно.

Будем рассматривать несколько различных алгоритмов поиска поправки $\tilde{\Delta}_n$ в шаге (14) алгоритма, отличающихся представлениями переменной $\tilde{\Delta}_n$ в уравнении (7). Используемое представление переменной определяет алгоритм и может влиять на характер его сходимости.

В одной группе алгоритмов мы не будем учитывать правильный знак поправки и будем искать решение уравнения (7), представляя $\tilde{\Delta}_n$ как в формуле (4):

$$\tilde{\Delta}_n = \text{sign}(b - ax_n) \left(\vartheta q_p + \sum_{i=r}^{p-1} 2^i q_i \right), \quad (23)$$

где $r, p \in \mathbb{Z}$, $r < p$, $\vartheta = -2^p + 2^r$.

В другой группе алгоритмов будем учитывать знак поправки и искать $\tilde{\Delta}_n$, представляя его как

$$\tilde{\Delta}_n = \text{sign}(b - ax_n) \sum_{i=r}^{p-1} 2^i q_i. \quad (24)$$

Заметим, что количество n_q участвующих в представлении $\tilde{\Delta}_n$ кубитов в группе алгоритмов, не учитывающих знак, равно $p - r + 1$, а в группе, учитывающих знак, равно $p - r$.

Обозначим

$$\Omega_{r,p}^{\pm} = \left\{ \pm \sum_{i=r}^{p-1} q_i 2^i : q_i \in \{0, 1\} \right\}, \quad \Omega_{r,p}^+ = \Omega_{r,p} \cap [0, \infty).$$

Таким образом, алгоритм определяется выбором $\Omega_{r,p}^{\pm}$ или $\Omega_{r,p}^+$ в качестве множества, на котором ищется поправка $\tilde{\Delta}_n$. Ввиду (22) мы будем ограничиваться значениями $p \leq 1$.

Распределение поправки на n -м шаге задается соотношением

$$\tilde{\Delta}_n \sim \text{sign}(b - ax_n) \text{B} \left(\beta, \Omega_{r,p}, \left(ax - \frac{1}{c_n} \right)^2 \right),$$

где $\Omega_{r,p}$ равно либо $\Omega_{r,p}^{\pm}$, либо $\Omega_{r,p}^+$.

Обозначим через $q_{r,p}(u, c, a, \beta)$, такие функции, что если c, a, β фиксированы и η равномерно распределена на промежутке $[0, 1]$, то

$$q_{r,p}(\eta, c, a, \beta) \sim \text{B} \left(\beta, \Omega_{r,p}, \left(ax - \frac{1}{c} \right)^2 \right),$$

где $\Omega_{r,p}$ определяется в соответствии с выбранным алгоритмом. Функции $q_{r,p}$ представляют собой обратные функции распределения соответствующих законов распределения и определяются как

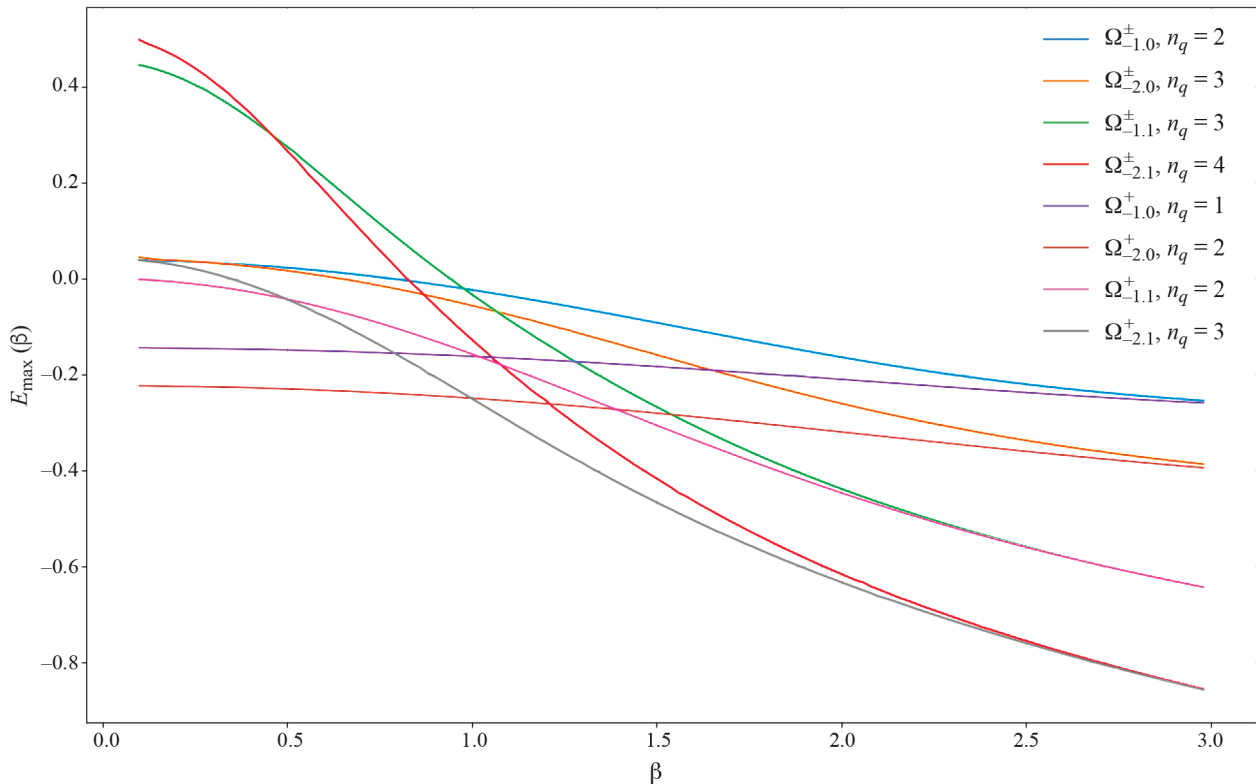
$$q_{r,p}(u, c, a, \beta) = \inf \{ t \mid F_{r,p}(t) \geq u \}, \quad u \in (0, 1],$$

где $F_{r,p}(t)$ — функция распределения закона $\text{B} \left(\beta, \Omega_{r,p}, \left(ax - \frac{1}{c} \right)^2 \right)$. Отметим, что функция $q_{r,p}(\cdot, c, a, \beta)$ кусочно-постоянная. Дополнительно определим

$$q_{r,p}(0, c, a, \beta) = \lim_{u \rightarrow 0^+} q_{r,p}(u, c, a, \beta).$$

На фиг. 2 приведены сравнительные графики для соответствующих функций $E_{\max}(\beta) = \max_{a \in [1/2, 1]} E(a, \beta)$, дающие пессимистичную оценку на скорость сходимости алгоритма. Если $E_{\max}(\beta) < 0$, то алгоритм сходится при любых $a \in [0.5, 1)$, $b \in \mathbb{R}$, чем меньше значение $E_{\max}(\beta)$, тем сходимость быстрее.

Из графиков следует, что при достаточно больших β сходятся все методы, включая основанные на одном кубите. Методы, учитывающие знак поправки, в которых $\Omega_{r,p} = \Omega_{r,p}^+$, сходятся быстрее, чем не учитывающие знак поправки, в которых $\Omega_{r,p} = \Omega_{r,p}^{\pm}$. При этом методы, учитывающие знак поправки используют меньшее



Фиг. 2. Графики функций $E_{\max}(\beta)$ для различных $\Omega_{r,p}$. Число n_q — количество кубитов, кодирующих поправку.

количество кубитов. Ожидаемо, с увеличением количества используемых кубитов скорость сходимости возрастает, но остается ниже, чем предельная скорость, соответствующая усеченному нормальному распределению на фиг. 1. По аналогии со сравнением алгоритмов 2 и 3 из п. 3.3 отметим, что методы с $p = 1$ включают больше значений поправок, при которых точность может ухудшиться, но при этом гарантированно содержат наилучшую возможную поправку, в то время как методы с $p = 0$ гарантированно не ухудшают точность приближения на каждом шаге, но при этом имеют меньшую вероятность для оптимальной поправки. Как и в случае бесконечного количества кубитов, при больших β методы с $p = 1$ оказываются более эффективными, чем при $p = 0$.

4. ВЫВОДЫ

В статье рассмотрены несколько адаптивных итеративных методов для поиска корня линейного уравнения $ax = b$ при помощи устройства, работающего по принципу квантового отжига. Результат работы QA моделируется распределением Больцмана. Для широкого класса алгоритмов предложен метод доказательства их сходимости и оценки скорости сходимости. Рассмотрены алгоритмы с бесконечным количеством кубитов и с малым количеством кубитов. Показано, что при достаточно малом шуме скорость сходимости экспоненциальная. При этом алгоритмы, учитывающие знак поправки сходятся быстрее, чем не учитывающие знак.

СПИСОК ЛИТЕРАТУРЫ

1. Манин Ю.И. Вычислимое и невычислимое. М.: Советское радио, 1980.
2. Feynman R.P. Simulating physics with computers // Int. J. Theor. Phys. 1982. V. 21. № 6. P. 467–488. <https://doi.org/10.1007/BF02650179>.
3. Williams C.P. Explorations in quantum computing. New York: Springer, 1998. <https://doi.org/10.1007/978-1-84628-887-6>.
4. Nielsen M.A., Chuang I.L. Quantum Computation and Quantum Information. Cambridge: Cambridge Univ. Press, 2010. <https://doi.org/10.1017/CB09780511976667>.

5. *Grover L.K.* A fast quantum mechanical algorithm for database search // *Proceed. of the twenty eighth Ann. ACM Symp. on Theory of Computing, Philadelphia, Pennsylvania, USA: Association for Computing Machinery.* 1996. P. 212–219. <https://doi.org/10.1145/237814.237866>.
6. *Shor P.W.* Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer // *SIAM J. on Comp.* 1997. V. 26. № 5. P. 1484–1509. <https://doi.org/10.1137/s0097539795293172>.
7. *Harrow A.W., Hassidim A., Lloyd S.* Quantum algorithm for linear systems of equations // *Phys. Rev. Lett.* 2009. V. 103. № 15. P. 150502. <https://doi.org/10.1103/physrevlett.103.150502>.
8. *Albasha T., Lidar D.A.* Adiabatic quantum computation // *Rev. Mod. Phys.* 2018. V. 90. № 1. P. 015002. <https://link.aps.org/doi/10.1103/RevModPhys.90.015002>.
9. *Kieu T.D.* The travelling salesman problem and adiabatic quantum computation: an algorithm // *Quant. Inform. Proces.* 2019. V. 18. № 3. P. 1–19. <https://doi.org/10.1007/s11128-019-2206-9>.
10. *Farhi E., Goldstone J., Gutmann S., Sipser M.* Quantum Computation by Adiabatic Evolution // *arXiv preprint quant-ph/0001106.* 2000. <https://doi.org/10.48550/arXiv.quant-ph/0001106>.
11. *Aharonov D., van Dam W., Kempe J., Landau Z., Lloyd S., Regev O.* Adiabatic quantum computation is equivalent to standard quantum computation // *SIAM Rev.* 2008. V. 50. № 4. P. 755–787. <https://doi.org/10.1137/080734479>.
12. *Kadowaki T., Nishimori H.* Quantum annealing in the transverse Ising model // *Phys. Rev. E.* 1998. V. 58. № 5. P. 5355–5363. <https://doi.org/10.1103/physreve.58.5355>.
13. *Bian Z., Chudak F., Macready W.G., Rose G.* The Ising model: teaching an old problem new tricks // *D-Wave Systems.* 2010.
14. *Albasha T., Martin-Mayor V., Hen I.* Temperature scaling law for quantum annealing optimizers // *Phys. Rev. Lett.* 2017. V. 119. № 11. P. 110502. <https://doi.org/10.1103/physrevlett.119.110502>.
15. D-Wave Systems. QPU Solver Datasheet. https://docs.dwavesys.com/docs/latest/doc_qpu.html, accessed 24 Oct 2023.
16. *Vinci W., Buffoni L., Sadeghi H., Khoshaman A., Andriyash E., Amin M.H.* A path towards quantum advantage in training deep generative models with quantum annealers // *Machine Learning: Science and Technology.* 2020. V. 1. № 4. P. 045028. <https://doi.org/10.1088/2632-2153/aba220>.
17. *Korenkevych D., Xue Y., Bian Z., Chudak F., Macready W., Rolfe J., Andriyash E.* Benchmarking quantum hardware for training of fully visible boltzmann machines // *arXiv preprint arXiv:1611.04528.* 2016. <https://doi.org/10.48550/arXiv.1611.04528>.
18. *Denil M., de Freitas N.* Toward the implementation of a quantum RBM // *NIPS Deep Learning and Unsupervised Feature Learning Workshop.* 2011.
19. *Albasha T., Lidar D.A.* Demonstration of a scaling advantage for a quantum annealer over simulated annealing // *Phys. Rev. X.* 2018. V. 8. № 3. P. 031016. <https://doi.org/10.1103/physrevx.8.031016>.
20. *King A.D., Raymond J., Lanting T., Harris R., Zucca A., Altomare F., Berkley A.J., Boothby K., Ejtemaee S., Enderud C., Hoskinson E., Huang S., Ladizinsky E., MacDonald A.J.R., Marsden G., Molavi R., Oh T., Poulin-Lamarre G., Reis M., Rich C., Sato Y., Tsai N., Volkmann M., Whittaker J.D., Yao J., Sandvik A.W., Amin M.H.* Quantum critical dynamics in a 5000-qubit programmable spin glass // *Nature.* 2023. V. 617. № 7959. P. 61–66. <https://doi.org/10.1038/s41586-023-05867-2>.
21. *O'Malley D., Vesselinov V.V.* ToQJl: A high-level programming language for D-Wave machines based on Julia // *2016 IEEE High Performance Extreme Computing Conference (HPEC), Waltham, MA, USA.* 2016. P. 1–7. [10.1109/HPEC.2016.7761616](https://doi.org/10.1109/HPEC.2016.7761616).
22. *Borle A., Lomonaco S.J.* Analyzing the quantum annealing approach for solving linear least squares problems // *Lect. Notes Comp. Sci.* 2018. P. 289–301. https://doi.org/10.1007/978-3-030-10564-8_23.
23. *Rogers M.L., Singleton R.L.* Floating-point calculations on a quantum annealer: Division and matrix inversion // *Front. Phys.* 2020. V. 8. <https://doi.org/10.3389/fphy.2020.00265>.

24. *Borle A., Lomonaco S.J.* How viable is quantum annealing for solving linear algebra problems? // arXiv preprint arXiv:2206.10576, 2022. <https://doi.org/10.48550/arXiv.2206.10576>.
25. *Date P., Potok T.* Adiabatic quantum linear regression // Sci. Rep. 2021. V. 11. № 1. <https://doi.org/10.1038/s41598-021-01445-6>.
26. *Souza A.M., Martins E.O., Roditi I., Sa N., Sarthour R.S., Oliveira I.S.* An application of quantum annealing computing to seismic inversion // Front. Phys. 2022. V. 9. <https://doi.org/10.3389/fphy.2021.748285>.
27. *Meli N.K., Mannel F., Lellmann J.* An Iterative Quantum Approach for Transformation Estimation from Point Sets // 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA. 2022. P. 519–527. <https://doi.org/10.1109/CVPR52688.2022.00061>.
28. *Conley R., Choi D., Medwig G., Mroczko E., Wan D., Castillo P., Yu K.* Quantum optimization algorithm for solving elliptic boundary value problems on D-Wave quantum annealing device // Proc. SPIE 12446, Quantum Computing, Communication, and Simulation III, 124460A. 2023. <https://doi.org/10.1117/12.2649076>.
29. *Lewis M., Glover F.* Quadratic Unconstrained Binary Optimization Problem Preprocessing: Theory and Empirical Analysis // Networks. 2017. V. 70. № 2. P. 79–97. <https://doi.org/10.1002/net.21751>.
30. *Ширяев А.Н.* Вероятность. М.: Наука, 1980.
31. *Lagarias J.C.* Euler's constant: Euler's work and modern developments // Bull. Am. Math. Soc. 2013. V. 50. № 4. P. 527–628. <https://doi.org/10.1090/s0273-0979-2013-01423-x>.

CONVERGENCE RATE OF ALGORITHM FOR SOLVING LINEAR EQUATIONS BY QUANTUM ANNEALING

S. B. Tikhomirov^{a,*}, V. S. Shalgin^{b,**}

^a Pontifícia Universidade Católica do Rio de Janeiro – PUC-Rio, Rua Marquês de São Vicente, 225, Gávea – Rio de Janeiro, RJ - Brazil Zip: 22451-900 - P.O. Box: 38097

^b St. Petersburg State University, 7/9 Universitetskaya nab., St. Petersburg, 199034, Russia

*e-mail: sergey.tikhomirov@gmail.com

**e-mail: st086496@student.spbu.ru

Received 06 November, 2023

Revised 26 December, 2023

Accepted 06 February, 2024

Abstract. Various iterative algorithms for solving the linear equation $ax = b$ using a quantum computer operating on the principle of quantum annealing are studied. Assuming that the result produced by the computer is described by the Boltzmann distribution, conditions under which these algorithms converge are obtained and an estimate of their convergence rate is provided. Application of this approach for algorithms that use an infinite number of qubits and a small number of qubits is considered.

Keywords: adiabatic quantum computing, quantum annealing, linear equation, Boltzmann distribution, truncated normal distribution.