

Разработка и исследование квантовых моделей преобразования изображений

В.Ф. Гузик¹, С.М. Гушанский¹, В.С. Потапов¹^{*}

¹Южный федеральный университет,
Таганрог, 347939, Российская Федерация
^{*}Адрес для переписки: vpotapov@sfedu.ru

Информация о статье

Поступила в редакцию 18.08.2020

Принята к публикации 26.01.2021

Ссылка для цитирования: Гузик В.Ф., Гушанский С.М., Потапов В.С. Разработка и исследование квантовых моделей преобразования изображений // Труды учебных заведений связи. 2021. Т. 7. № 1. С. 20–30. DOI:10.31854/1813-324X-2021-7-1-20-30

Аннотация: В рамках статьи рассмотрена теория квантовой обработки изображений и квантовые манипуляции с изображениями (быстрые геометрические преобразования, включая переворот, ортогональные вращения и ограниченные геометрические преобразования). В работе проведено исследование области квантовой обработки изображений и протестирована модель ГПКИ. Были исследованы модели (разработанная в работе и др.) путем явной симуляции квантового поведения на большой квантовой системе. Создана базовая схема для кодирования 2×2 изображения, для получения более четкого представления о механике квантовых моделей изображений.

Ключевые слова: квантовая симуляция, квантовый алгоритм, квантовый бит, модель квантового вычислителя, квантовое запутывание, суперпозиция, квантовый параллелизм.

Введение

Квантовая обработка изображений посвящена использованию квантовых вычислений [1] и последующей обработке информации для создания и работы с изображениями. Из-за некоторых поразительных свойств, присущих квантовым вычислениям, в частности запутанности и параллелизма, ожидается, что технологии квантовой обработки информации будут предлагать возможности и характеристики, которые пока не имеют себе равных по своим традиционным эквивалентам. Эти улучшения могут быть связаны с быстродействием вычислений, гарантированной безопасностью, минимальными требованиями к хранилищу и т. д.

Работа А.Ю. Власова [2] в 1997 г. была сосредоточена на использовании квантовой системы для распознавания ортогональных изображений. За этим последовали попытки использования квантовых алгоритмов для поиска определенных паттернов в двоичных изображениях и определения положения определенных целей. Примечательно, что более основанная на оптике интерпретация для квантовой визуализации была первоначально экспериментально продемонстрирована в [3] и формализована в [4] через 7 лет. В 2003 г. Ш. Бозе представил «Qubit Lattice», первую опубликованную общую мо-

дель для хранения, обработки и извлечения изображений с использованием квантовых систем. Позже, в 2005 г., Д. Латторре предложил другой тип представления, называемый «истинный кет вектор», целью которого было кодирование квантовых изображений в качестве основы для дальнейшего применения. Кроме того, в 2010 г. С. Венегас-Андраца представил метод хранения и извлечения бинарных геометрических фигур в квантово-механических системах, в котором показано, что максимально запутанные кубиты могут использоваться для восстановления изображений без использования какой-либо дополнительной информации.

Основная идея квантовых вычислений – это кубит, квантовый аналог классического компьютерного бита. Классический бит способен хранить только определенное значение (0 или 1). Кубит может хранить и 0, и 1 в неопределенном состоянии, называемом суперпозицией:

$$\alpha|0\rangle + \beta|1\rangle,$$

где α и β – нормированные амплитуды вероятности.

Математический аппарат

Логика и математический аппарат квантовых вычислительных устройств в значительной мере отличается от логики и математики классической

вычислительной техники из-за природы и специфических свойств квантовых частиц (кубит), которые и позволяют проводить такие вычисления. Квантовый вычислитель оперирует, как было описано выше, с квантовыми битами, которые имеют два базисных состояния $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ и $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Состояние кубита $|0\rangle$ соответствует состоянию спина электрона в атоме «спин вверх», состояние «спин вниз» соответствует $|1\rangle$, а смешанное состояние (суперпозиция состояний) соответствует промежуточному положению спина. Спин – это собственный момент импульса элементарных частиц, но спин не связан с движением в пространстве, это собственное число квантовой частицы, которое нельзя объяснить с позиции классической механики. В общем случае состояние кубита описывается волновой функцией (или вектором состояния):

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle,$$

где $\alpha_0, \alpha_1 \in \mathbb{C}$, α_0 и α_1 – комплексные амплитуды вероятности чтения $|0\rangle$ или $|1\rangle$. Дело в том, что при измерении квантовых битов стандартном базисе с вероятностью $|\alpha_0|^2$ будет получен $|0\rangle$, а с вероятностью $|\alpha_1|^2$ – $|1\rangle$. При этом кубит перейдет в состояние квантового нуля или единицы. Эффект квантового смешанного состояния существует только до момента измерения кубита, то есть при чтении мы приводим кубит в одно из базисных состояний, а какое именно, зависит от вероятности, полученной при вычислениях. Данное явление представлено графически на рисунке 1. Один раз измерив квантовый бит, мы получим всего один бит информации, несмотря на то, что в состоянии суперпозиции в нем хранится гораздо больше. В совокупности с тем, что неизвестное квантовое состояние нельзя копировать, эта проблема в настоящий момент является одной из основных.

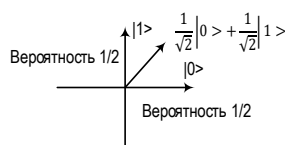


Рис. 1. Состояние кубита и вероятности измерения нуля и единицы

Fig. 1. The State of the Qubit and the Probabilities of Measuring Zeros and Units

α_0 и α_1 – представляют собой комплексные числа, они имеют следующее представление:

$$\alpha_0 = a + jb, \alpha_1 = c + jd,$$

где a, b, c, d – действительные числа; j – мнимая единица.

И следствием того, что $|\alpha_0|^2$ и $|\alpha_1|^2$ – вероятности чтения $|0\rangle$ и $|1\rangle$, является выражение:

$$|\alpha_0|^2 + |\alpha_1|^2 = 1.$$

Тогда уравнение $|\alpha_0|^2 + |\alpha_1|^2 = 1$ описывает сферу в двухмерном комплексном пространстве, а конкретное состояние – кольцо на этой сфере. Для

того, чтобы легче было представить, допустим, что α_0 и α_1 – вещественные числа (мнимые части равны нулю). В данном представлении состояние квантового бита отражает точка на окружности единичного радиуса. Еще одним визуальным вариантом представления вектора состояний кубита является так называемая сфера Блоха [5]. Она является отображением комплексной плоскости значений псифункции на сферу, фактически на ней можно визуальнo увидеть «спин вверх», «спин вниз». Однако это пространство значений состояния кубита, сам же псевдовектор спина не может быть отрицательным.

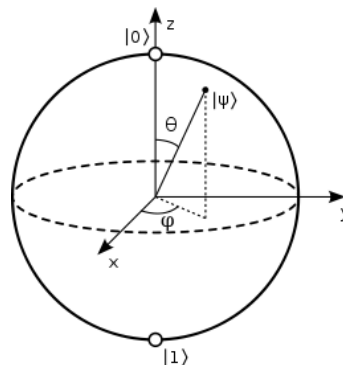


Рис. 2. Сфера Блоха

Fig. 2. Bloch Sphere

Анализируя рисунок 2, можно заметить, что умножение состояния кубита $|\psi\rangle$ на $e^{i\varphi}$ не меняет вероятности чтения нуля и единицы. Состояние системы из двух квантовых битов описывается относительно четырех базисных состояний (двумя битами в классическом компьютере можно закодировать 4 состояния): $|00\rangle, |01\rangle, |10\rangle, |11\rangle$.

$$|\psi\rangle = \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle, \\ |\alpha_0|^2 + |\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2 = 1.$$

То есть многокубитовая система вычисляемой функции $|\psi\rangle$ в формуле описывает все возможные состояния (комбинации нулей и единиц), количество комбинаций найдем по выражению:

$$N = 2^q,$$

где N – количество базисных состояний, которые описывает данная квантовая система кубитов; q – количество кубитов в системе. В начале анализа было указано, что состояния нуля и единицы кодируются как $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ и $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ функция состояния $|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$, если же раскрыть произведение $\alpha_0|0\rangle$, то получим следующее:

$$\alpha_0|0\rangle = \alpha_0 \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} \alpha_0 \\ 0 \end{pmatrix},$$

То есть первая строчка содержит в себе амплитуду вероятности считывания нуля. А для $\alpha_1|1\rangle$ получим, что вторая строчка содержит амплитуду вероятности считывания единицы:

$$\alpha_1|1\rangle = \alpha_1 \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ \alpha_1 \end{pmatrix}.$$

Тогда становится понятным, каким образом функция $|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$ содержит в себе вероятности всех состояний:

$$|\psi\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle = \begin{pmatrix} \alpha_0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}.$$

Первая строка содержит амплитуду вероятности – получить при чтении 0, а вторая – вероятность получить при чтении 1. Если рассмотреть систему, состоящую из двух кубитов, то ее вектор состояний описывается следующим образом:

$$\begin{aligned} |\psi\rangle &= \alpha_0|00\rangle + \alpha_1|01\rangle + \alpha_2|10\rangle + \alpha_3|11\rangle = \\ &= \alpha_0 \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \alpha_1 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \alpha_2 \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \alpha_3 \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \\ &= \begin{pmatrix} \alpha_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \alpha_1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \alpha_2 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix}. \end{aligned}$$

Для системы из трех кубит, вектор состояния будет состоять из $2^3 = 8$ строк, содержащих амплитуды для каждого возможного в классическом компьютере состояния. Проанализировав описанное выше, можно заключить, что пси-функция (вектор состояния) описывает все возможные состояния системы, а не одного кубита, и содержит в себе амплитуды вероятности чтения конкретного состояния, а не состояния для каждого кубита в отдельности. Также можно заключить, что, выполняя преобразования над кубитами, математически мы оперируем над состояниями самой системы. Дополнительной сложностью для моделирования является то, что амплитуды вероятностей являются комплексными числами.

Операции над кубитами можно отобразить с помощью квантовой схемы. Квантовая схема – это последовательность физических преобразований из конечного набора базисных элементарных преобразований – гейтов. На вход квантовой схемы подаются квантовые биты, а на выходе получают ее вероятностный результат работы. Вследствие того, что при измерении квантовых битов они переходят в состояния «0» или «1», то для получения точного результата можно использовать квантовую схему несколько раз для одинаковых входных данных. Физически можно реализовать только те операции, которые сохраняют постоянной сумму квадратов коэффициентов над небольшой группой кубитов. Исходя из этого, можно заключить, что любое преобразование над k квантовыми битами задается в виде матрицы $2^k \times 2^k$.

Теория квантовой обработки изображений

Рассмотрим случай использования квантовой информации кодированием 8 бит двоичной информации в 3 кубита. Ключевой особенностью кубита является способность хранить неопределенное

значение в виде суперпозиции «0» и «1». Поэтому любая операция будет по своей природе получать 0 или 1, как показано в таблице 1.

ТАБЛИЦА 1. Возможные состояния 3-кубитного регистра

TABLE 1. Possible States of a 3-Qubit Register

Кубиты	φ1	φ2	φ3	φ4	φ5	φ6	φ7	φ8
Ψ1	0	0	0	0	1	1	1	1
Ψ2	0	0	1	1	0	0	1	1
Ψ3	0	1	0	1	0	1	0	1

Трехбитный квантовый регистр ($\Psi = \psi_1\psi_2\psi_3$) может хранить 8 бит информации в суперпозиции, где каждой конфигурации назначается амплитуда вероятности, как показано ниже:

$$\varphi = \alpha_0|000\rangle + \alpha_1|001\rangle + \dots + \alpha_{2^n-1}|111\rangle.$$

Амплитуды ($\alpha_1, \alpha_2 \dots$) представляют собой вероятность того, что квантовый регистр будет в этой конфигурации. Также очень важно отметить, что квантовые логические вентили имеют некоторые уникальные достоинства, которыми не могут воспользоваться классические системы. Например, 2-кубитовый гейт CNOT обратимый. Это означает, что, если этот гейт применяется дважды, кубит вернется в исходное состояние. Для представления квантовых изображений можно использовать этот гейт, чтобы построить схему для подготовки квантового изображения. Гейт CNOT можно использовать как условный переключатель, поскольку он только переворачивает целевой бит, если на входе получаем 1. Возможно также прикрепить любую другую унитарную операцию к условному вентилю вместо гейта NOT для реализации условной операции.

Технически новаторские усилия в области квантового распознавания образов и объектов с последующими исследованиями, связанными с ними, можно разделить на три основные группы:

- 1) квантовая обработка изображений; эти приложения направлены на улучшение задач и приложений цифровой или классической обработки изображений;
- 2) оптическая квантовая визуализация;
- 3) квантовая обработка изображений, основанная на классических методах.

Обзор квантового представления изображения был опубликован в [6]. Кроме того, недавно опубликованная книга «Квантовая обработка изображений» [7] предоставляет исчерпывающее введение в квантовую обработку изображений, в которой основное внимание уделяется распространению традиционных задач обработки изображений на платформы квантовых вычислений [8]. В нем кратко представлены доступные представления квантовых изображений и их операции, рассматриваются возможные приложения квантовых изображений и их реализация, а также обсуждаются открытые вопросы и будущие тенденции развития.

Квантовые манипуляции с изображениями

Большая часть усилий была сосредоточена на разработке алгоритмов [9] для манипулирования информацией о положении и цвете, закодированной с использованием представления квантового изображения и его многочисленных вариантов. Например, изначально были предложены быстрые геометрические преобразования, включая переворот, ортогональные вращения и конечные геометрические преобразования, чтобы ограничить эти операции определенной областью изображения. Недавно была реализована довольно интересная идея квантовых изображений на основе нового расширенного представления для сопоставления положения каждого элемента изображения во входном изображении с новой позицией в выходном изображении и масштабирование квантового изображения для изменения его размера. В то время как основанная на представлении квантового изображения общая форма цветовых преобразований была впервые предложена с помощью однобитовых вентилей, таких как вентили X , Z и H [10].

Используя базовые квантовые элементы и вышеупомянутые операции, исследователи внесли свой вклад в извлечение квантовых характеристик изображения, квантовую сегментацию изображения, морфологию квантового изображения, сравнение квантовых изображений, фильтрацию квантовых изображений, квантовую классификацию изображений, квантовую стабилизацию изображения.

Квантовое преобразование изображения

Посредством кодирования и обработки информации изображения в квантово-механических системах [11] представлена основа квантовой обработки изображения, где чистое квантовое состояние кодирует информацию изображения: для кодирования значений пикселей в амплитудах вероятности и положений пикселей в базовых состояниях вычислений. Дано изображение $F = (F_{i,j})_{M \times L}$, где $F_{i,j}$ представляет значение пикселя в позиции (i, j) с $i = 1, \dots, M$ и $j = 1, \dots, L$, вектор f с элементами ML можно сформировать, поместив первые элементы M в вектор f будет первым столбцом F , следующие M элементов – вторым столбцом и т. д.

Большой класс операций с изображениями является линейным, например, унитарные преобразования, свертки и линейная фильтрация. В квантовых вычислениях линейное преобразование может быть представлено как $|g\rangle = U|f\rangle$ с состоянием входного изображения $|f\rangle$ и состоянием выходного изображения $|g\rangle$. Унитарное преобразование может быть реализовано как унитарная эволюция. Некоторые основные и обычно используемые преобразования изображений (например, вейвлет-преобразования Фурье, Адамара и Хаара) могут быть выражены в виде выражения $G = P \cdot F \cdot Q$ с результирующим изображением G и матрица преобразования строки (столбца) $P(Q)$. Соответствующий унитарный оператор U может быть

записан как $U = Q^T \cdot P$. Несколько обычно используемых двумерных преобразований изображений, таких как вейвлет Хаара, преобразования Фурье и Адамара, экспериментально продемонстрированы на квантовом компьютере с экспоненциальным ускорением по сравнению с их классическими аналогами. Кроме того, предложен и экспериментально реализован новый высокоэффективный квантовый алгоритм для обнаружения границы между различными областями изображения: он требует только одного однокбитового гейта на этапе обработки, независимо от размера изображения. Квантовые вычисления, которые используют квантовый параллелизм, в принципе быстрее, чем классический компьютер для определенных задач. Квантовое изображение кодирует информацию изображения в квантово-механических системах вместо классических, и замена классической квантовой обработки информации может облегчить некоторые из этих проблем.

Оценка моделей преобразования изображений

Пригодность модели для решения задач исследования характеризуется тем, в какой степени она обладает так называемыми целевыми свойствами. Основными из них являются адекватность, устойчивость и чувствительность.

Оценка адекватности модели

В общем случае под адекватностью понимают степень соответствия модели тому реальному явлению или объекту, для описания которого она строится. Адекватность модели определяется степенью ее соответствия не столько реальному объекту, сколько целям исследования. Один из способов обоснования адекватности разработанной модели – использование методов математической статистики. Суть этих методов заключается в проверке выдвинутой гипотезы на основе статистических критериев. Процедура оценки основана на сравнении измерений на реальной системе и результатов экспериментов на модели и может проводиться по средним значениям откликов модели и системы или по максимальному значению относительных отклонений откликов модели от откликов системы.

В квантовом моделировании любая система (модель) полностью описывается заданием волновой функции $\psi(r, t)$. Она является коэффициентом разложения вектора состояния (содержащего полную информацию о состоянии системы) по базису:

$$|\psi(t)\rangle = \int \psi(x, t) |x\rangle dx,$$

где $|x\rangle = |x_1, x_2, \dots, x_n\rangle$ – координатный базисный вектор.

Оценка устойчивости модели

Устойчивость модели – это ее способность сохранять адекватность при исследовании эффективности системы на всем возможном диапазоне рабочей нагрузки, а также при внесении изменений в конфигурацию системы. Чем ближе структура модели структуре системы и чем выше степень детализации, тем устойчивее модель. Устойчивость результатов моделирования может быть также оценена методами математической статистики.

Оценка чувствительности модели

Достаточно часто возникает задача оценивания чувствительности модели к изменению параметров рабочей нагрузки и внутренних параметров самой системы. При анализе моделей можно рассматривать каждую модель с различных сторон (визуальная часть, архитектура, возможности моделирования), что позволяет так же классифицировать модели. Такой анализ подразумевает иерархическую структуру характеристик. Анализ и классификация моделей позволяют только систематизировать полученные данные и качественно охарактеризовать каждую, выделив имеющиеся преимущества и недостатки, что является субъективной оценкой. Данная оценка не позволяет четко распределить проанализированные модели в виде общей оценки модели. При классификации сред моделирования на основе реализуемых функций были выделены независимые друг от друга группы характеристик – категорий. Если оценивать модель по всем категориям одновременно, то получится оценить модель комплексно, просуммировав оценки по категориям:

$$T_{\text{общ}} = T_1 + T_2 + \dots + T_8,$$

где T_1, T_2 – суммарные оценки для характеристик; $T_{\text{общ}}$ – суммарная оценка модели.

В соответствии со степенью значимости или критичностью данной категории характеристик, ей назначается свой весовой коэффициент, таким образом, общая оценка имеет следующий вид:

$$T_{\text{общ}} = w_1 \cdot T_1 + w_2 \cdot T_2 + \dots + w_8 \cdot T_8,$$

где $w_1, w_2, \dots$ – весовые коэффициенты для каждой категории.

Таким образом, оценка модели будет складываться из суммарных оценок на уровне характеристик, умноженных на соответствующий коэффициент на уровне категории. Такая организация расчета оценок позволит точно для каждой категории характеристик оценить его значимость и произвести численную оценку модели. Примечательно, что количество уровней в такой системе подсчета может быть увеличено.

Реализация среды моделирования квантовых вычислений

На рисунке 3 изображена разработанная модель квантового вычислительного устройства и результат выполнения квантового алгоритма преобразования набора пикселей. Данная программная разработка является офлайн десктопной вычислительной системой с открытой архитектурой, однако в дальнейшем планируется ее трансляция в онлайн-режим. Процесс измерения заключается в разыгрывании числа случайного характера из промежутка $[0, 1]$.

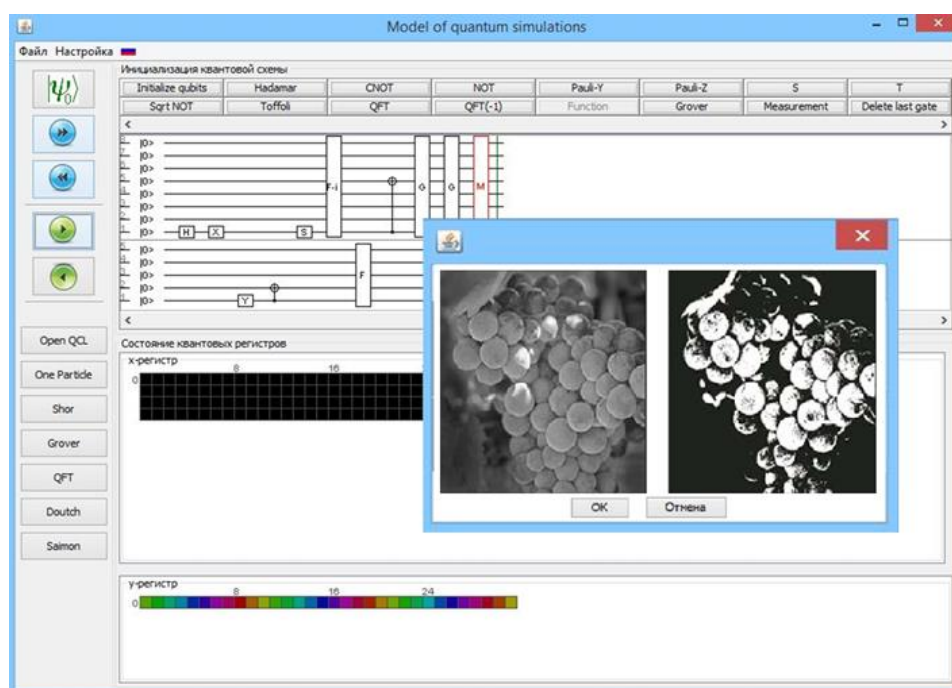


Рис. 3. Модель квантового вычислительного устройства

Fig. 3. Model of a Quantum Computing Device

В разработанном алгоритме используются квантовые принципы, такие как суперпозиция квантовых состояний вычислительной системы, запутанность квантовых состояний и преобразование классического изображения в квантовое состояние путем кодирования цветовой палитры пиксельного набора в рамках комплексных амплитудных квантовых состояний.

Основные параметры модели квантового вычислительного устройства

Разработанная модель квантового вычислителя обладает рядом важных характеристик: модульность, библиотечный набор, исходное количество и возможность добавления гейтов, редуцированная матрица состояний, поддержка платформ Windows/Linux, редактор квантовой схемы, задание входных значений кубитов, числовой вывод вероятностей/амплитуд кубита, таблица цветов вероятностей/амплитуд состояний кубитов, автоматический режим, пошаговый режим, моделирование физических процессов, матричное математическое ядро, нет ограничения на количество кубитов, сохранение/загрузка программы/схемы/алгоритма, сохранение результатов вычислений, загрузка и продолжение вычислений, открытая архитектура, отработанные примеры, документация.

Разработка архитектуры вычислителя подразумевает под собой разработку структуры программы, которая включает разбиение структуры на программные компоненты и разработку схемы взаимодействия между этими компонентами с помощью доступных снаружи свойств и методов этих компонентов. Одним из самых продуктивных методов, направленных на увеличение функциональности модели, является ее дополнение вспомогательными внешними библиотеками и модулями.

Данный алгоритм опишем с помощью сети Петри для более детального понимания работы модели квантового вычислителя (МКВ) в несколько этапов (рисунок 4).

Определение средств разработки модели преобразования изображений

В разработанном списке требований особое место занимает требование независимости отдельных компонентов среды моделирования друг от друга. Главная его цель выдвижения – это предоставление возможности в будущем заменять некоторые компоненты и упрощение разработки среды несколькими программистами, поэтому на стадии проектирования это требование определяет структуру компонентов среды моделирования, основные способы взаимодействия между ними и задает направление развития и ограничения разработчикам компонентов. Дополнительным плюсом данного подхода является облегченная дальнейшая

поддержка, обновление и настройка среды моделирования. Почти все остальные требования в списке задают набор графических компонентов, которые необходимо реализовать, а также возможности, которые стоит предоставить пользователю для более удобного процесса моделирования. Разработку решено вести на основе платформы Microsoft.Net-Framework 4.0 и соответственно на языке C#.

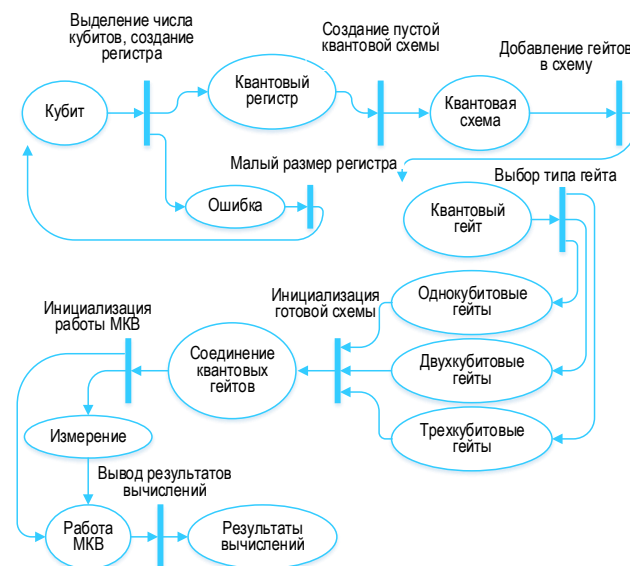


Рис. 4. Сеть Петри. Алгоритм работы модели

Fig. 4. Petri Net. Algorithm of the Model

Реализация общей архитектуры модели преобразования изображений

Разработка архитектуры МКВ подразумевает под собой разработку структуры программы, которая включает разбиение структуры на программные компоненты, и схемы взаимодействия между этими компонентами с помощью доступных снаружи свойств и методов этих компонентов.

Смоделируем и рассмотрим, по мнению авторов, наилучший вариант архитектуры, который предполагают разделение приложения минимум на две части: математическое ядро и пользовательский интерфейс. Данная схема представлена на рисунке 5.

Математическое ядро выносится зачастую в динамическую библиотеку, таким образом, в модели может быть большое количество математических ядер. Функциональный блок Main () анализирует аргументы командной строки, открывает подключение к экземпляру/создает экземпляр симулятора и выполняет основной цикл работы. Симулятор взаимодействует с кодом пользователя через обратные вызовы, как выполняющиеся инструкции. Эти вызовы представляют собой функции, передающиеся в качестве аргумента другой функции. Немаловажным обратным вызовом является «Квантовый регистр». Квантовый регистр – это упорядоченное множество конечного числа кубитов. Он тесно связан в первую очередь с блоком «Операции с памятью». Под памятью понимается

определенное количество кубитов, с которыми работает и которыми оперирует модель квантового вычислителя. Блок «Конструктор» взаимодействует непосредственно с симулятором. Он получает от него данные, инструкции и другую информацию, полученную ранее от блоков обратных вызовов, преобразует ее для работы квантового вычислителя. Блок «Файл трассировки» служит для отчетности, анализа и прогнозирования дальнейшей работы модели.

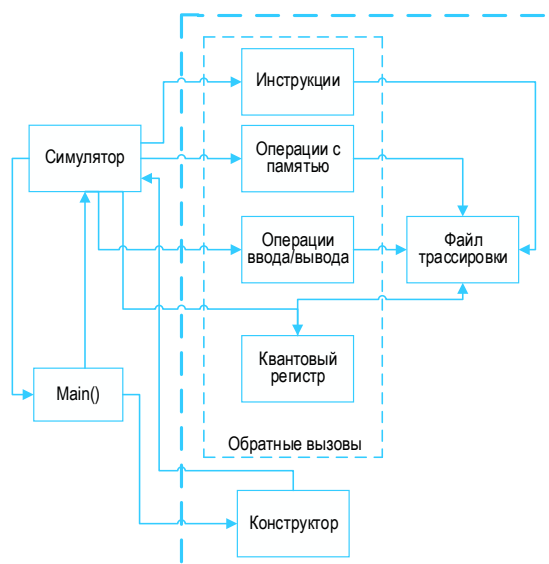


Рис. 5. Схема взаимодействия элементов модели квантового вычислителя

Fig. 5. Scheme of Interaction Some Elements in Model of a Quantum Computer

Для программной реализации модели квантового вычислительного устройства, которая способна модифицировать имеющуюся модель, дополняя ее различными функциями, необходимо придерживаться ряда принципов. Декомпозиция модели на отдельные по своей функциональной составляющей модули ведет не только к улучшению уже существующей модели за счет отбора более необходимых модулей-компонентов и удаления менее нужных для дальнейшей работы, но и разработке перспективно новых моделей за счет открытой архитектуры модели (рисунок 6).

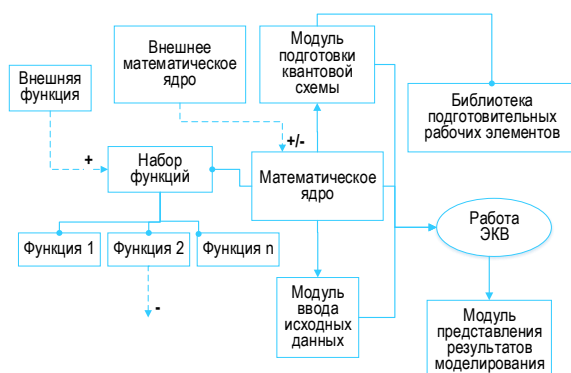


Рис. 6. Подключение/отключение внешнего модуля

Fig. 6. Connecting/Disconnecting an External Module

Необходимо отметить, что каждый из модулей в свою очередь может делиться на отдельные внутренние компоненты, по усмотрению разработчика.

Разработка математического ядра

Математическое ядро – это модуль, который содержит в себе все необходимое для моделирования кубит (квантового регистра) и набора операций (гейтов), которые применяются к данным кубита. Во всех моделях используется ограниченный набор квантовых операторов, на которых можно построить все остальные необходимые преобразования. Предлагается следующий набор функций, которые необходимо реализовать в математическом ядре:

```
// Инициализация диалога для работы модели с
// пользователем
public InitDialog(java.awt.Frame parent, boolean modal, Properties bundle)

// Инициализация начального состояния квантового
// регистра и схемы
public InitialStateDialog(java.awt.Frame parent, boolean modal, Properties bundle, int[] qubits, int xRegisterSize)

// гейт Адамара
private void hadamardButtonActionPerformed(java.awt.event.ActionEvent evt)

// гейт CNOT
private void cNOTButtonActionPerformed(java.awt.event.ActionEvent evt)

// гейт NOT
private void xButtonActionPerformed(java.awt.event.ActionEvent evt)

// Тoffoli гейт
private void toffoliButtonActionPerformed(java.awt.event.ActionEvent evt)

// Обратное квантовое преобразование Фурье
private void invQftButtonActionPerformed(java.awt.event.ActionEvent evt)

// Квантовое преобразование Фурье
private void qftButtonActionPerformed(java.awt.event.ActionEvent evt)

// Оператор вычисления функции
private void U_fButtonActionPerformed(java.awt.event.ActionEvent evt)

// Оператор Гровера
private void groverButtonActionPerformed(java.awt.event.ActionEvent evt)
```

Реализация указанных функций упрощает задачу разработки математического ядра. Но также перед разработчиками встает другая задача: взаимодействие всех этих функций между собой.

Сравнение моделей преобразования изображений и их классификация

Технически эти новаторские усилия с последующими исследованиями, связанными с ними, можно разделить на три основные группы, из которых) первая и вторая считаются вне сферы действия этого обзора:

Группа 1. Квантовая обработка цифровых изображений: эти приложения предназначены для использования некоторых из свойств, ответственных

за эффективность алгоритмов квантовых вычислений, чтобы улучшить некоторые хорошо известные цифровые или классические обработки изображений задачи и приложения

Группа 2. Квантовая визуализация на основе оптики: эти приложения сосредоточены на разработке новых методов оптической визуализации и параллельной обработки информации на квантовом уровне за счет использования квантовой природы света и внутреннего параллелизма оптических сигналов.

Группа 3. QIP в классическом стиле: эти приложения черпают вдохновение из ожидания того, что оборудование квантовых вычислений скоро будет физически реализовано и, следовательно, такие исследования направлены на расширение классических задач обработки изображений и приложения к структуре квантовых вычислений.

С. Венегас-Андраца и Д. Боже предложили представление изображений на квантовых компьютерах, предложив метод решетки кубитов, где каждый пиксель был представлен в своем квантовом состоянии, а затем была создана квантовая матрица с ними. Однако это всего лишь квантовый аналог классического образа, и нет никакого дополнительного преимущества в квантовой форме. Тем не менее, следующим огромным достижением является создание модели гибкого представления квантовых изображений (ГПКИ) для нескольких уровней интенсивности. Представление ГПКИ выражается математически следующим образом:

$$|I(\theta)\rangle = \frac{1}{2^n} \sum_{i=0}^{2^{2n}-1} (\sin(\theta_i)|0\rangle + \cos(\theta_i)|1\rangle)|i\rangle,$$

где θ_i соответствует интенсивности i -го пикселя. Поскольку значения интенсивности кодируются в амплитудах квантового состояния, относительно просто применять различные преобразования, такие, как квантовое преобразование Фурье, так как оно применяется непосредственно к амплитуде изображения. Другой подход описывает представление значения пикселей изображения в базисных состояниях вместо амплитуд, как показано ниже:

$$|I\rangle = \frac{1}{2^n} \sum_{X=0}^{2^{2n}-1} \sum_{Y=0}^{2^{2n}-1} |f(X, Y)\rangle |XY\rangle,$$

где $f(X, Y)$ относится к интенсивности пикселей в (X, Y) . Эти методы довольно всеобъемлющие, но у них есть свои собственные недостатки. Первый главный недостаток заключается в том, что они требуют квадратное изображение $(2^N \cdot 2^N)$ наряду с необходимостью кубитов для кодирования позиций вместе с интенсивностью пикселей. Кроме того, очень сложно получить классическую версию изображения из квантового изображения из-за дополнительной неопределенности. Действительно, изображения из-

влекаются с помощью нормализованного распределения вероятностей. Предложим представление квантового изображения через двумерные квантовые состояния и амплитуду нормализации. Этот подход может использоваться для представления прямоугольных изображений и интенсивности пикселя без использования дополнительных кубитов. Поэтому изображение с размерами $(2^N \cdot 2^M)$ может быть закодировано только в $M + N$ кубитах. Сравнение существующих (представление значения пикселей изображения в базисных состояниях вместо амплитуд ПЗПИБСВА, а также представление квантового изображения через двумерные квантовые состояния и амплитуду нормализации ПКЧДКСАН) и предложенной квантовой модели изображений (гибкое представление квантовых изображений ГПКИ) можно увидеть в таблице 2. Большинство квантовых моделей изображения может быть подготовлено с помощью гейтов Адамара и контролируемого переворота операторов. Для $2^N \cdot 2^N$ первый шаг должен инициализировать n кубитов при $|0\rangle^{\otimes 2n+1}$.

ТАБЛИЦА 2. Сравнение моделей квантовых изображений

TABLE 2. Comparison of Quantum Imaging Models

Модель	ГПКИ	ПЗПИБСВА	ПКЧДКСАН
Форма	$(2^N \cdot 2^N)$	$(2^N \cdot 2^N)$	$(2^N \cdot 2^M)$
Число кубит	$2N + 1$	$2N + 1$	$M + N$
Вычислительная сложность	$O(2^{4m})$	$O(2^{2m})$	чистое состояние

Затем мы применяем гейт Адамара к каждому кубиту, чтобы разместить их в суперпозиции.

$$H(|0\rangle^{\otimes 2n+1}) = \frac{1}{2^n} |0\rangle \otimes \sum_{i=0}^{2^{2n}-1} |i\rangle.$$

Далее применяем операторы управляемого вращения к каждому базовому состоянию, приводящему к совместному состоянию:

$$R \left| H \right\rangle = \left(\prod_{i=0}^{2^{2n}-1} R_i \right) \left| H \right\rangle \geq |I(\theta)\rangle.$$

Каждое из вращений R_i соответствует определенному пиксельному вращению. Следовательно, кодируем интенсивности посредством вращения этих квантовых состояний. Схема реализации может достигаться путем объединения гейтов Адамара, CNOT и сдвига фазы. Внедрим ГПКИ из-за его простой и эффективной функции, которая позволяет сохранять интенсивность изображения как амплитуды вероятности квантового состояния. Доступ к этим амплитудам будет связан с измерением. Однотипные гейты также используются в схеме квантового преобразования Фурье. Реализуем ГПКИ как на классическом компьютере, так и на квантовом симуляторе IBM.

Моделирование квантовых систем в классическом компьютере довольно дорого в вычислительном отношении. Это потому, что нам нужно явно определить суперпозицию (другими словами, хранить суперпозицию в памяти).

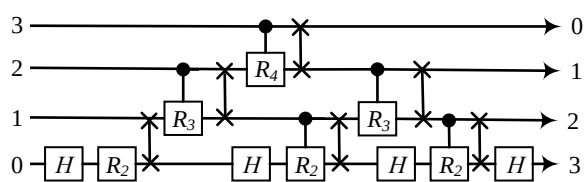


Рис. 7. Схема квантового преобразования Фурье

Fig. 7. Scheme of Quantum Fourier Transform

В реальном квантовом компьютере суперпозиция является собственностью системы, следовательно, не требует его хранения. Например, рассмотрим кодирование N битов информации в $\log_2 N$ битов, суперпозиция всех возможных комбинаций 0 и 1 могут быть сохранены в квантовом регистре, в отличие от квантового регистра, в котором мы можем сохранить только один из всех возможных состояний. Классический компьютер должен был бы хранить все возможные состояния в виде списка (занимая дополнительное место) в отличие от квантового компьютера. Это также означает, что мы не получаем доступ к этим квантовым битам, как в классической системе. Доступ к n битам информации в этих n кубитах возможен только посредством измерения. Состояние, которого мы хотим достичь, представлено в уравнении 3. Для построения схемы используем изображение 2×2 , как показано ниже:

$$\begin{bmatrix} \theta_1 & \theta_2 \\ \theta_3 & \theta_4 \end{bmatrix}.$$

Закодируем информацию как:

$$\frac{1}{2}[(\cos(\theta_1) + \sin(\theta_1))|00\rangle + (\cos(\theta_2) + \sin(\theta_2))|01\rangle + (\cos(\theta_3) + \sin(\theta_3))|10\rangle + (\cos(\theta_4) + \sin(\theta_4))|11\rangle].$$

Такое кодирование возможно при использовании гейта управляемого вращения для поворота отдельных базовых состояний на основании интенсивности. Проверим этот алгоритм на 2×2 изображении. Симулятор IBM позволяет с помощью небольшого числа кубит хранить изображение больше, чем 2×2 . Следовательно, представляем механику схемы, кодируя 2×2 изображение в квантовом регистре IBM. Будем использовать серверную часть симулятора для IBM Q 5 Tenerife. Используем простое 2×2 изображение:

$$I = \begin{bmatrix} 5 & 1 \\ 2 & 3 \end{bmatrix}.$$

Отметим, что эти интенсивности будут нормализованы. Следовательно, мы не сможем точно вос-

становить изображение. Распределение вероятностей изображения показано на рисунке 8 (синий – исходное изображение).

Поскольку манипулирование интенсивностями просто означает контролируемое вращение, можем использовать схему (см. рисунок 7) для определения конкретных гейтов. Например, отрицание изображения будет просто означать вращение амплитуды на π . Операции над квантовым изображением могут быть сделаны вращением квантовых битов вокруг сферы Блоха. Например, чтобы инвертировать квантовое изображение, можно просто вращать кубиты вокруг сферы Блоха на 180° (рисунок 8, где красный – инвертированное изображение).

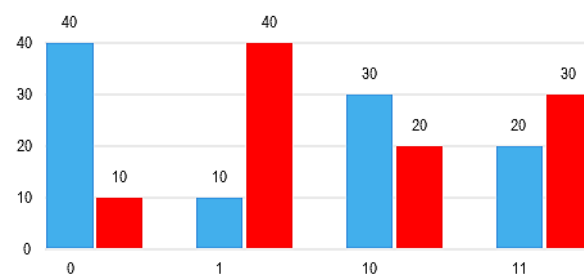


Рис. 8. Полученное изображение как распределение вероятностей

Fig. 8. The Resulting Image as a Probability Distribution

Заключение

Рассмотренная обработка изображений основывается на применении квантовых вычислений и последующей обработке информации для создания и работы с квантовыми изображениями. Исследования в области распознавания образов и объектов, можно разделить на три основные группы:

- квантовая обработка изображений;
- оптическая квантовая визуализация;
- квантовая обработка изображений, основанная на классических методах.

В рамках рассмотренного исследования была разработана модель преобразования различных типов изображений, отличающаяся от известных аналогов более высокой скоростью выполнения операций по распознаванию объектов на изображении. Что касается качества распознавания, то здесь выигрыш алгоритма квантовой природы минимален. Данный выигрыш обусловлен квантовыми преимуществами, а именно – параллелизмом всего вычислительного процесса, позволяющим производить ряд вычислений по распознаванию объектов на изображении в асинхронном режиме в несколько потоков. Также произведено компьютерное моделирование квантового алгоритма на разработанной ранее модели квантового вычислительного устройства для решения задачи преобразования классического изображения с использованием квантовых вычислительных средств и методов для подтверждения теоретических показателей, полученных в работе.

ИСТОЧНИК ФИНАНСИРОВАНИЯ

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта №19-07-01082.

Список используемых источников

1. Potapov V., Gushanskiy S., Guzik V., Polenov M. The Computational Structure of the Quantum Computer Simulator and Its Performance Evaluation // Proceedings of the 7th Computer Science On-line Conference on Software Engineering and Algorithms in Intelligent Systems (CSOC 2018, Vsetin, Czech Republic, 25–28 April 2018). Cham: Springer, 2018. Vol. 1. PP. 198–207. DOI:10.1007/978-3-319-91186-1_21
2. Vlasov A.Y. Quantum computations and images recognition. 1997. URL: <https://arxiv.org/abs/quant-ph/9703010> (Accessed 12th March 2021)
3. Samoylov A., Gushanskiy S., Polenov M., Potapov V. The Quantum Computer Model Structure and Estimation of the Quantum Algorithms Complexity // Proceedings of the Computational Methods in Systems and Software on Computational and Statistical Methods in Intelligent Systems (CoMeSySo 2018, Szczecin, Poland, 12–14 September 2018). Cham: Springer, 2019. PP. 307–315. DOI:10.1007/978-3-030-00211-4_27
4. Beach G., Lomont C., Cohen C. Quantum image processing (QuIP) // Proceedings of the 32nd Applied Imagery Pattern Recognition Workshop (Washington, USA, 15–17 October 2003). IEEE, 2003. DOI:10.1109/AIPR.2003.1284246
5. Валиев К.А., Кокин К.К. Квантовые компьютеры: надежды и реальность. М.: Регулярная и хаотическая динамика, 2001. 352 с.
6. Yan F., Ilyasu A.M., Venegas-Andraca S.E. A survey of quantum image representations // Quantum Information Processing. 2016. Iss. 15. PP. 1–35. DOI:10.1007/s11128-015-1195-6
7. Yan F., Venegas-Andraca S.E. Quantum Image Processing. Singapore: Springer, 2020. DOI:10.1007/978-981-32-9331-1
8. Boneh D., Zhandry M. Quantum-Secure Message Authentication Codes // Proceedings of the 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2013, Athens, Greece, 26–30 May 2013). Berlin, Heidelberg: Springer, 2013. PP. 592–608. DOI:10.1007/978-3-642-38348-9_35
9. Potapov V., Gushanskiy S., Polenov M. Optimization of Models of Quantum Computers Using Low-Level Quantum Schemes and Variability of Cores and Nodes // Proceedings of 8th Computer Science On-line Conference on Cybernetics and Automation Control Theory Methods in Intelligent Algorithms (SCOS 2019, Zlin, Czech Republic, 24–27 April 2019). Cham: Springer, 2019. PP. 264–273. DOI:10.1007/978-3-030-19813-8_27
10. Kleppner D., Kolenkow R. An Introduction to Mechanics. Cambridge: Cambridge University Press, 2014.
11. Raedt K.D., Michielsen K., De Raedt H., Trieu B., Arnold G., Richter M., et al. Massively parallel quantum computer simulator // Computer Physics Communications. 2007. Vol. 176. Iss. 2. PP. 121–136. DOI:10.1016/j.cpc.2006.08.007

* * *

Development and Research of Quantum Transformation Models

V. Guzik¹, S. Gushansky¹, V. Potapov¹

¹Southern Federal University,
Taganrog, 347939, Russian Federation

Article info

DOI:10.31854/1813-324X-2021-7-1-20-30

Received 18th August 2020

Accepted 26th January 2021

For citation: Guzik V., Gushansky S., Potapov V. Development and Research of Quantum Transformation Models. *Proc. of Telecom. Universities*. 2021;7(1):20–30. (in Russ.) DOI:10.31854/1813-324X-2021-7-1-20-30.

Abstract: The article discusses the theory of quantum image processing and quantum image manipulation (fast geometric transformations, including the flip, orthogonal rotations and limited geometric transformations). In this work we have investigated the area of quantum image processing and have tested the FPQI model. Models (developed in the work and others) were investigated by explicitly modeling quantum behavior on a large quantum system. A basic scheme for encoding 2×2 image has been created to gain a clearer understanding of the mechanics of quantum image models.

Keywords: quantum simulation, quantum algorithm, quantum bit, quantum computing model, quantum entanglement, superposition, quantum parallelism.

FUNDING

This research was funded by RFBR according to the research project №19-07-01082.

References

1. Potapov V., Gushanskiy S., Guzik V., Polenov M. The Computational Structure of the Quantum Computer Simulator and Its Performance Evaluation. *Proceedings of the 7th Computer Science On-line Conference on Software Engineering and Algorithms in Intelligent Systems, CSOC 2018, 25–28 April 2018, Vsetin, Czech Republic*. Cham: Springer; 2018. vol.1. p.198–207. DOI:10.1007/978-3-319-91186-1_21
2. Vlasov A.Y. *Quantum computations and images recognition*. 1997. Available from: <https://arxiv.org/abs/quant-ph/9703010> [Accessed 12th March 2021]
3. Samoylov A., Gushanskiy S., Polenov M., Potapov V. The Quantum Computer Model Structure and Estimation of the Quantum Algorithms Complexity. *Proceedings of the Computational Methods in Systems and Software on Computational and Statistical Methods in Intelligent Systems, CoMeSySo 2018, Szczecin, Poland, 12–14 September 2018*. Cham: Springer; 2019. p. 307–315. DOI:10.1007/978-3-030-00211-4_27
4. Beach G., Lomont C., Cohen C. Quantum image processing (QulP). *Proceedings of the 32nd Applied Imagery Pattern Recognition Workshop, 15–17 October 2003, Washington, USA*. IEEE; 2003. DOI:10.1109/AIPR.2003.1284246
5. Valiev K.A., Kokin K.K. *Quantum Computers: Hopes and Reality*. Moscow: Reguliarnaia i khaoticheskaia dinamika Publ.; 2001. 352 p. (in Russ.)
6. Yan F., Iliyasu A.M., Venegas-Andraca S.E. A survey of quantum image representations. *Quantum Information Processing*. 2016;15:1–35. DOI:10.1007/s11128-015-1195-6
7. Yan F., Venegas-Andraca S.E. *Quantum Image Processing*. Singapore: Springer; 2020. DOI:10.1007/978-981-32-9331-1
8. Boneh D., Zhandry M. Quantum-Secure Message Authentication Codes. *Proceedings of the 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, EUROCRYPT 2013, 26–30 May 2013, Athens, Greece*. Berlin, Heidelberg: Springer; 2013. p.592–608. DOI:10.1007/978-3-642-38348-9_35
9. Potapov V., Gushanskiy S., Polenov M. Optimization of Models of Quantum Computers Using Low-Level Quantum Schemes and Variability of Cores and Nodes. *Proceedings of 8th Computer Science On-line Conference on Cybernetics and Automation Control Theory Methods in Intelligent Algorithms, SCOS 2019, 24–27 April 2019, Zlin, Czech Republic*. Cham: Springer; 2019. p.264–273. DOI:10.1007/978-3-030-19813-8_27
10. Kleppner D., Kolenkow R. *An Introduction to Mechanics*. Cambridge: Cambridge University Press; 2014.
11. Raedt K.D., Michielsen K., De Raedt H., Trieu B., Arnold G., Richter M., et al. Massively parallel quantum computer simulator. *Computer Physics Communications*. 2007;176(2):121–136. DOI:10.1016/j.cpc.2006.08.007

Сведения об авторах:

ГУЗИК
Вячеслав Филиппович

доктор технических наук, профессор кафедры вычислительной техники Инженерной технологической академии Южного федерального университета, yfguzik@sfedu.ru
 <https://orcid.org/0000-0001-9169-8787>

ГУШАНСКИЙ
Сергей Михайлович

кандидат технических наук, доцент кафедры вычислительной техники Инженерной технологической академии Южного федерального университета, smgushansky@sfedu.ru
 <https://orcid.org/0000-0002-4549-3334>

ПОТАПОВ
Виктор Сергеевич

ассистент кафедры вычислительной техники Инженерной технологической академии Южного федерального университета, vpotapov@sfedu.ru
 <https://orcid.org/0000-0003-0661-6810>