

Раздел V. Моделирование, алгоритмы, защита информации

УДК 681.3.06:530.145.001.57

О.К. Евсеев, С.М. Гушанский, В.Ф. Гузик

МЕТОД МОДЕЛИРОВАНИЯ КВАНТОВЫХ ВЫЧИСЛЕНИЙ НА ОСНОВЕ QUIDD-ГРАФОВ

Для моделирования квантовых вычислений характерно использование матриц большого размера, содержащих малое количество различных элементов. Наиболее эффективный способ снижения вычислительных затрат при моделировании квантовых вычислений был предложен университетом Мичигана и получил название QuIDD. В данной статье приводятся его ключевые аспекты и предлагаются уровнево-рекурсивные методы редукции QuIDD-графов и поэлементных операций над ними, способные повысить эффективность базовой методики. Изложенные в статье методы могут быть использованы при построении графовой математической модели универсального квантового вычислителя, позволяющей моделировать вычисления на квантовом регистре с предельной размерностью порядка 50 q-бит.

Квантовые вычисления; моделирование; q-бит; QuIDD-граф; матрицы; вектор состояния; тензорное произведение; поуровневые операции.

О.К. Evseev, S.M. Gushanskiy, V.F. Guzik

QUIDD-BASED QUANTUM COMPUTER MODELING METHOD

Quantum computer simulation requires processing of huge-size matrixes containing several distinct elements. The most effective method of decreasing calculation costs for simulation was developed in the University of Michigan and was called Quantum Information Decision Diagrams. In order to improve the efficiency of the base QuIDD methodic this article covers its key aspects and advises level-wise recursive methods of QuIDD graph reduction and matrix member-wise operations. Methods set out in this article can be used for developing a graph-based mathematical model of a universal quantum computer, operating at the maximal dimension of the quantum register of about 50 q-bits.

Quantum computation; simulation; q-bit; QuIDD graph; matrix; state vector; tensor product; level-wise operations.

В настоящее время, когда уже разработан набор квантовых алгоритмов, позволяющих решать некоторые NP-задачи [1, 2], и продолжаются попытки создания аппаратного квантового вычислителя, актуальна задача программного моделирования квантовых вычислений. Существующие математические модели квантового вычислителя, по критерию эффективности используемой в них реализации математического аппарата, можно разделить на 3 класса: матричные, матричные с архивацией и графовые.

Для моделирования квантовых вычислений характерно использование матриц большого размера, но содержащих малое количество различных элементов. Таким образом, модели матричного класса [3, 4] имеют экспоненциальное снижение эффективности с ростом моделируемой системы. Матричные модели с архивацией [5] позволяют работать с системами значительно большей размерности, од-

нако каждое обращение к состоянию системы или оператору требует частичной либо полной разархивации матриц, что приводит к увеличению и без того экспоненциальных затрат времени. Более эффективными являются графовые модели. В их контексте матрица хранится в виде набора уникальных элементов и графа (диаграммы) их расположения. Их отличие от моделей с архивацией заключается в наличии операций, позволяющих моделировать квантовые вычисления на основе графов без восстановления исходной матрицы. В случае, если количество уникальных элементов невелико и, тем паче, наиболее распространённым элементом является «0», вычислительные затраты таких моделей могут выражаться полиномом от размерности моделируемой системы. Наиболее эффективным образцом графовых моделей является QuIDD Pro [6], использующий метод представления квантовых вычислений, называемый «Quantum Information Decision Diagrams».

Моделирование квантовых вычислений на основе QuIDD-графов включает метод представления матриц и векторов состояния, метод редукции графов, метод применения поэлементных операций, методы матричного [6, 7] и тензорного перемножения графов [6].

Процесс моделирования состоит из следующих этапов: создание моделей q-бит, сборка из них модели квантового регистра через тензорное произведение, сборка и применение с помощью матричного умножения вычислительных операторов, моделирование измерения терминального состояния квантового регистра. QuIDD-графы, используемые для представления матриц и векторов состояния, строятся из 2 типов узлов: промежуточных (представляют двоичный разряд индекса элемента матрицы или вектора) и терминальных (представляют элемент матрицы). На рис. 1,а представлен вид графа вектора состояния квантового бита, на рис. 1,б – матрицы оператора на 1 квантовый бит. Для каждого графа задаётся тип (V – вектор, M – матрица) и размерность (здесь 1). Промежуточные узлы, представляющие двоичные разряды индекса столбца, обозначаются символом «C», индекса строки – символом «R». Узел R_0 представляет нулевой разряд индекса строки.

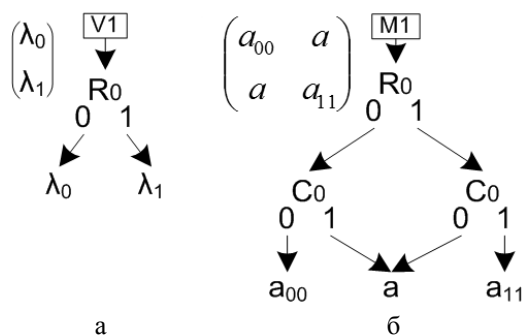


Рис. 1. Общий вид QuIDD-графов

Граф на рис. 1,б интерпретируется следующим образом: в матрице оператора на 1 q-бит строка 0 содержит элемент a_{00} в столбце 0 и элемент a в столбце 1. Строка 1 – a в столбце 0 и a_{11} в столбце 1. Для сборки модели квантового регистра из моделей квантовых бит используется тензорное перемножение векторов их состояний. В терминах QuIDD-графов это действие выглядит как увеличение индексов промежуточных вершин графа-множимого на количество q-бит графа-множителя и замена терминалов графа-множимого на граф-множитель, чьи терминалы умножаются на заменяемый терминал графа-множимого (рис. 2).

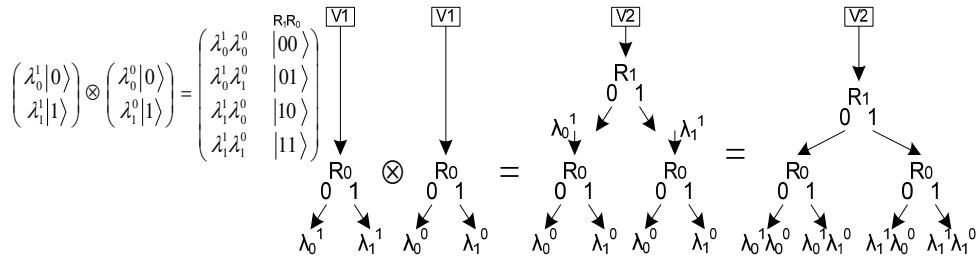


Рис. 2. Сборка квантового регистра

QuIDD-графы имеют выигрыш в сравнении с матрицами благодаря правилам редукции, а именно: исключения изоморфных подграфов и исключения замкнутых вершин. Вместо метода рекурсивной редукции, представленного в работе [8], здесь предлагается метод поуровневой редукции, существенное улучшение которого состоит в отсутствии необходимости поиска изоморфных подграфов, состоящих более чем из одной вершины. Метод состоит в обходе графа по уровням, начиная с уровня терминальных вершин и выше, заканчивая корневой вершиной. На каждом уровне из вершин с одинаковыми потомками оставляется только 1. На неё замыкаются восходящие дуги исключаемых копий. Помимо этого удаляются вершины, обе нисходящие дуги которых входят в общую вершину.

Результирующий граф на рис. 2 демонстрирует чётко выраженную блочную структуру, в которой узел R_1 делит вектор состояния пополам. Данное наблюдение характерно и для графов матриц. На нём основывается операция матричного перемножения, которая является основным и наиболее ресурсоёмким действием в моделировании квантовых вычислений. В общем случае матричное перемножение требует 2 рекурсий операции Apply [6].

Операция Apply является рекурсивной операцией над парой узлов графов. Она определяется по правилам, изложенным в [7] и применяется для поэлементных действий над графами. В общем случае она рекурсивно вызывается с корневых вершин и использует кэш для предотвращения повторных действий над уже обработанными узлами.

В данной работе предлагается уровнево-рекурсивная модификация метода Apply, при которой рекурсия осуществляется путём предварительного размещения вершин вызова в упорядоченном по уровням списке. На каждом шаге операция производится над самым старшим уровнем списка, причём пары вершин предварительно анализируются и дублирующиеся исключаются с замыканием связей на оставляемую уникальную пару. Такой подход позволяет повысить эффективность операции на 10–15 %.

Изложенные методы могут быть использованы при построении графовой математической модели универсального квантового вычислителя, позволяющей моделировать вычисления на квантовом регистре с предельной размерностью порядка 50 q-бит, что значительно превосходит возможности матричных моделей.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Shor P.W.* Scheme for reducing decoherence in quantum memory // *Phys. Rev.* – 1995. – Vol. A52, № 4. – P. R2493-R2496.
2. *Grover L.K.* A fast quantum mechanical algorithm for database search // *Proc. of 28th Annual ACM Symposium on the Theory of Computing.* – 1996. – P. 212-232.
3. *Chris McCubbin.* Модель квантового вычислителя с открытым исходным кодом на Maple / URL: <http://web.archive.org/web/20060116174553/http://userpages.umbc.edu/~cmccub1/quacs/quacs.html> (дата обращения: 27.03.2012).

4. Lib Quantum. Моделирование квантовой механики / URL: <http://www.enyo.de/libquantum> (дата обращения: 27.03.2012).
5. Black P.E., Lane A.W. Modeling Quantum Information Systems // Proc. of the International Society for Optical Engineering. – 2004. – № 5436. – P. 340-347.
6. Viamontes G.F., Markov I.L., Hayes J.P. Quantum circuit simulation – Quantum Information Processing, Springer. – 2009. – 194 p.
7. Bahar R.I., Frohm E.A., Gaona C.M. Algebraic decision diagrams and their applications // ICCAD '93, Santa Clara, CA, USA, November 07-11 – Los Alamitos, CA, USA: IEEE Computer Society Press. – 1993. – P. 188-191.
8. Sanner S., McAllester D. Affine algebraic decision diagrams and their Application to Structured Probabilistic Inference // Proc. of IJCAI-05, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. 2005. – 7 p.

Статью рекомендовал к опубликованию д.т.н., профессор Н.И. Витиска.

Гузик Вячеслав Филиппович – Федеральное государственное автономное образовательное учреждение высшего профессионального образования «Южный федеральный университет»; e-mail: gvf@tsure.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371656; кафедра вычислительной техники; профессор.

Гушанский Сергей Михайлович – e-mail: kron@pbox.ttn.ru; кафедра вычислительной техники; доцент.

Евсеев Олег Константинович – e-mail: aleg.a33@rambler.ru; кафедра вычислительной техники; аспирант.

Guzik Vyacheslav Filippovich – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education “Southern Federal University”; e-mail: gvf@tsure.ru; 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: +78634371656; the department of computer engineering; professor.

Gushansky Sergei Mikhailovich – e-mail: kron@pbox.ttn.ru; the department of computer engineering; associate professor.

Evseev Oleg Konstantinovich – e-mail: aleg.a33@rambler.ru; the department of computer engineering; postgraduate student.

УДК 532.5.031

А.В. Никитина, И.С. Семенов

ПАРАЛЛЕЛЬНАЯ РЕАЛИЗАЦИЯ МОДЕЛИ ДИНАМИКИ ТОКСИЧНОЙ ВОДОРΟΣЛИ В АЗОВСКОМ МОРЕ С ПРИМЕНЕНИЕМ МНОГОПОТОЧНОСТИ В ОПЕРАЦИОННОЙ СИСТЕМЕ WINDOWS

*Работа посвящена применению функций winapi для создания параллельного алгоритма с общей памятью, реализующего задачу распространения токсичной водоросли *Skeletonema costatum* в мелководном водоеме. Задача является актуальной, поскольку при ее решении производятся расчеты концентраций вредных веществ и токсичных водорослей в Азовском море.*

Предложенный способ реализации параллельных вычислений позволяет сократить время работы программы более чем в 2 раза. За основу был взят механизм создания дополнительных потоков, который обеспечивает лучшее распределение ресурсов ЭВМ и повышает эффективность алгоритма.

Математическая модель; многопоточность; функции winapi ; токсичные водоросли; Азовское море.