

Суперкомпьютерные методы моделирования связанной квантовой системы: кубит-измерительный осциллятор*

В.П. Гергель, М.В. Денисенко, А.В. Линев, Д.С. Пашин, А.М. Сатанин
Нижегородский государственный университет им. Н.И. Лобачевского

Разработан алгоритм моделирования диссипативной динамики многоуровневой квантовой системы на примере кубита, связанного с квантовым нелинейным осциллятором, основанный на стохастическом методе квантовых траекторий (квантовом методе Монте-Карло). Квантовый метод Монте-Карло позволяет осуществить эффективное распараллеливание и программную реализацию на гетерогенном высокопроизводительном вычислительном кластере при расчете процессов релаксации в многоуровневых квантовых системах. Продемонстрирована эффективность работы графических ускорителей, масштабируемость метода и описывается возможность взаимодействия нескольких графических ускорителей с применением технологии MPI.

1. Введение

Двухуровневая система - кубит - является базовым элементом квантового компьютера, который в настоящее время активно разрабатывается во многих лабораториях мира [1-4]. Кубит описывается двухкомпонентной волновой функцией, подчиняющейся уравнению Шредингера. Решение такого уравнения не представляет никаких проблем и может быть выполнено на персональном компьютере. Принципиальные трудности возникают при моделировании работы n связанных кубитов (квантового регистра). Во-первых, это обусловлено тем, что число компонентов волновой функции - размерность гильбертова пространства состояний - растет как $N = 2^n$. Во-вторых, в реальных условиях эксперимента кубиты подвержены воздействию шума, что естественно разрушает квантовую когерентность, лежащую в основе квантовых алгоритмов [1]. Шумовое воздействие на кубиты можно описать как взаимодействие с термостатом, следовательно, систему необходимо описывать уравнением для матрицы плотности, число элементов матричного уравнения растет как $N \times N$. Во многих экспериментах детектирующее устройство можно представить как осциллятор (линейный или нелинейный), который добавляет дополнительные степени свободы и шумовое влияние, что необходимо принимать во внимание при моделировании. Поставленную проблему можно переформулировать следующим образом: можно ли промоделировать работу квантового компьютера на привычном «классическом» ЭВМ? [3].

Современные компьютерные технологии, предполагающие использование распределенных вычислительных систем, включающих графические процессорные устройства, позволяют моделировать квантовую динамику и процессы релаксации систем с огромным числом уровней N ($N > 1000$). В данной работе рассматривается простейшая модель многоуровневой квантовой системы: кубит, связанный с квантовым нелинейным осциллятором, который используется для проективных измерений состояний кубита [5, 6]. Влияние окружения на кубит и осциллятор имитируется их взаимодействием с бозонным термостатом [7]. При условии слабого обратного воздействия на термостат и локального по времени возмущения (марковское приближение), динамика системы может быть описана уравнением для матрицы плотности [7]. Число состояний связанной системы равно $N + 2$. Для выполнения измерений важно, чтобы осциллятор, производящий детерминирование состояний кубита, работал в квазиклассическом режиме, т.е. число активных состояний N было велико. Однако, вместо того чтобы решать связанную систему дифференциальных уравнений для $(N + 2) \times (N + 2)$ матричных урав-

* Работа поддержана грантом РФФИ (проект № 14-07-00582), программой СИ7 02.В.49.21.0003 МОН РФ и ННГУ.

нений, мы используем квантовый метод Монте-Карло, который позволяет решать $N + 2$ уравнений для стохастической волновой функции [8]. Это означает, что необходимо многократно решать уравнение Шредингера со случайными параметрами, а наблюдаемые величины определять путем усреднения билинейной комбинации монте-карловской волновой функции по большому числу реализаций (число определяется необходимой точностью). Решение независимых дифференциальных уравнений хорошо параллелизуется, поэтому можно ожидать значительного ускорения процесса вычисления усредненных величин, аналогичным тем, что получают при решении матрицы плотности. В работе описаны принципы распараллеливания и численного моделирования стохастического уравнения Шредингера, а также программный комплекс, позволяющий осуществлять высокопроизводительные параллельные расчеты на GPU.

2. Модель системы "кубит+осциллятор"

В качестве примера измерительного устройства выбран джозефсоновский осциллятор, взаимодействующий с зарядовым кубитом специального вида (так называемый "квантроний кубит" - *quantronium qubit*) [5, 6, 9]. Такой кубит работает на основе двух хорошо известных явлений, которые подробно описаны в литературе [10]. Во-первых, это эффект Джозефсона, имеющий место при протекании сверхпроводящего тока через тонкую диэлектрическую прослойку [10]. Во-вторых, используется эффект кулоновской блокады, возникающий вследствие существования конечной энергии, которую нужно затратить, чтобы перенести заряд между контактом и маленьким сверхпроводящим островком [5, 6].

Предполагается, что два джозефсоновских перехода соединены со сверхпроводящим островком, функционирующим в режиме кулоновской блокады, имеют полную емкость C_J , и каждый из них обладает джозефсоновской энергией $E_J/2$ [5, 6]. Для управления зарядом на островок подается постоянный потенциал V_g (см. рис. 1).

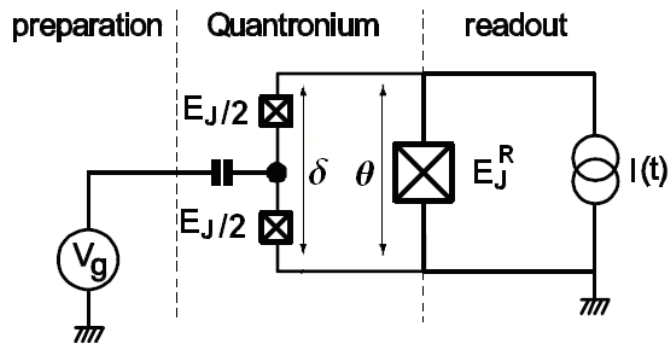


Рис. 1. Схема кубита (*quantronium qubit*) (центр рисунка), связанного с устройством записи (левая часть рисунка) и взаимодействующего со слабодиссипативным измерительным джозефсоновским осциллятором (правая часть рисунка). Перечеркнутыми квадратами обозначены джозефсоновские переходы. Островок (кружок, разделяющий перечеркнутые квадраты) является в то же время вторым контактом переходов (см. [5,6]).

Полный гамильтониан системы включает гамильтониан расщепленного куперовского островка

$$H_{\text{QO}} = 4E_c \left(\hat{N} - \frac{1}{2} + \frac{C_g V(t)}{2e} \right)^2 - E_J \cos \frac{\delta}{2} \cos \theta, \quad (1)$$

а гамильтониан измерительного джозефсоновского осциллятора имеет вид:

$$H_J = \frac{\hat{Q}^2}{2C} - E_J^R \cos \theta - \frac{\hbar^2}{2e} I(t) \theta, \quad (2)$$

где \hat{N} и $\hat{Q}/2e$ обобщенные импульсы, сопряженные к фазам δ и θ , соответственно купе-

ровского и джозефсоновского переходов. Управление состояниями кубита осуществляется переменным напряжением $V(t) = V_g + V_{rf}(t) \cos \omega t$, а переменный ток $I(t)$ служит для управления измерительным джозефсоновским осциллятором.

Вблизи точки кулоновского вырождения можно использовать базисные состояния $|N\rangle$ и $|N+1\rangle$ ($\hat{N}|N\rangle = N|N\rangle$) для состояний островка. В этом подпространстве: $|N\rangle = (1,0)$, $|N+1\rangle = (0,1)$, а гамильтониан аппроксимируется выражением

$$H \cong 2E_C \frac{C_g V_{rf}(t)}{e} \sigma_x - E_J \cos \frac{\theta}{2} \sigma_z + \frac{Q^2}{2C} - E_J^R \cos \theta - \frac{\hbar^2}{2e} I(t) \theta, \quad (3)$$

где σ_x и σ_z - спиновые матрицы Паули.

При относительно малом токе $I(t)$ джозефсоновский осциллятор работает в слабо нелинейном режиме. Тогда потенциал в выражение (3) можно разложить по степеням θ :

$$H \cong 2E_C \frac{C_g V_{rf}(t)}{e} \sigma_x - E_J \sigma_z + \frac{Q^2}{2C} + E_J (1 + \lambda \sigma_z) \frac{\theta^2}{2} - E_J \left(1 + \frac{\lambda}{4} \sigma_z\right) \frac{\theta^4}{4!} - \frac{\hbar^2}{2e} I(t) \theta, \quad (4)$$

где $\lambda = E_J / 4E_J^R$.

Вводя операторы рождения a^+ и уничтожения a для параметризации координаты $\theta = (a + a^+) (2E_C^R / E_J)^{1/4}$ и импульса осциллятора $\hat{N} = -i(a - a^+) (E_C^R / 2E_J)^{-1/4}$ ($\hat{Q} = 2e\hat{N}$, $\hat{N} = -i\partial/\partial\theta$) с учетом $[\theta, \hat{N}] = i$, можно гамильтониан (4) связанной системы "кубит+нелинейный осциллятор" представить в виде составляющих:

$$H = H_q + H_{osc} + H_{int}. \quad (5)$$

В выражении (5) первое слагаемое - гамильтониан кубита, который запишем в стандартном виде:

$$H_q = \frac{\hbar}{2} (\omega_q \sigma_z + \varepsilon(t) \sigma_x). \quad (6)$$

Можно видеть, что расстояние между уровнями кубита ($\hbar\omega_q = -2E_J$) определяется частотой

$\omega_q / 2\pi$ (~10 ГГц); а управляющая функция $\varepsilon(t) = 4E_C \frac{C_g V_{rf}(t)}{e}$ будет индуцировать пе-

реходы между состояниями кубита $|\sigma\rangle$ ($\sigma_z |\sigma\rangle = \sigma |\sigma\rangle$, $\sigma = \pm 1$). Предполагается, что управление кубитом будет осуществляется Раби-импульсами переменного поля $\varepsilon(t) = A \cos \omega t$ определенной длительности, позволяющими приготовить наперед заданную суперпозицию состояний кубита.

Таким образом, измерение состояний кубита будет осуществляться нелинейным осциллятором,

$$H_{osc} = \hbar\omega_0 a^+ a - \mu (a^+ + a)^4 + f(t) (a + a^+), \quad (7)$$

где собственная частота осциллятора $\omega_0 = \sqrt{\frac{2eI_c}{\hbar C}}$, параметр нелинейности $\mu = \frac{E_c^R}{12}$,

$f(t) = f_0 \cos \Omega t$, где амплитуда $f_0 = \frac{\hbar}{2e} \left(\frac{2E_c^R}{E_J}\right)^{1/4}$ и частота Ω внешнего возмущения.

Наконец, последнее слагаемое в (5) отвечает за взаимодействие кубита с измерительным осциллятором:

$$H_{\text{int}} = \lambda \left(\hbar \omega_0 a^+ a - \frac{1}{4} \mu (a + a^+)^4 \right) \sigma_z, \quad (8)$$

где λ - константа связи кубита и осциллятора. Отметим, если управляющее поле отсутствует, $\mathcal{E}(t) = 0$, то оператор взаимодействия (8) коммутирует с гамильтонианом кубита (6), поэтому осциллятор производит так называемое «неразрушающее измерение» [5].

Для системы "кубит+осциллятор" уравнение для оператора плотности ρ в форме Линдблада, в борн-марковском приближении [7] записывается в виде:

$$\begin{aligned} \frac{\partial \rho}{\partial t} = & \frac{1}{i\hbar} [H, \rho] + \gamma_\varphi (\sigma_z \rho \sigma_z - \rho) + \frac{\gamma_e}{2} (2\sigma_- \rho \sigma_+ - \sigma_+ \sigma_- \rho - \rho \sigma_+ \sigma_-) \\ & + \frac{\gamma}{2} (2a \rho a^+ - a^+ a \rho - \rho a^+ a) \end{aligned}, \quad (9)$$

где $\sigma_\pm = (\sigma_x \pm i\sigma_y)/2$, а γ_φ и γ_e - соответственно фазовая и энергетическая скорости релаксации кубита, γ - параметр релаксации осциллятора. Отметим, что данные параметры выражаются стандартным образом через корреляционными функциями бозонного термостата ($\langle F_z(t) F_z^+ \rangle$, $\langle R(t) R^+ \rangle$ и т.д.) (см. вывод в [7]).

3. Метод квантовых траекторий (квантовый метод Монте-Карло) для системы "кубит+осциллятор"

Уравнение для матрицы плотности (9) дает усредненную по ансамблю динамику системы. Чтобы проследить динамику кубита и осциллятора в каждом акте измерения (аналог "однократных" измерений) обратимся к эквивалентному языку описания динамики - методу квантовых траекторий или, как его ещё называют, квантовому методу Монте-Карло [8]. Согласно данному подходу эволюция системы, может быть переформулирована как динамика состояний «квантовых траекторий».

3.1. Алгоритм расчетов квантовым методом Монте-Карло для системы "кубит+осциллятор"

Применяя конечно-разностную аппроксимацию для производной по времени в уравнении (9), перепишем его в виде:

$$\rho(t + \Delta t) = U \rho(t) U^+ + \Delta t \left(\gamma_\varphi \sigma_z \rho \sigma_z + \gamma_e \sigma_- \rho \sigma_+ + \gamma a \rho a^+ \right), \quad (10)$$

где первое слагаемое в (2) определяется оператором эволюции $U = e^{-iH_{\text{dis}} \Delta t / \hbar}$ и отвечает за диссипативную динамику системы с гамильтонианом Вигнера-Вайскопфа [7]

$$H_{\text{dis}} = H_q - i\hbar \frac{\gamma_\varphi}{4} \sigma_z^+ \sigma_z - i\hbar \frac{\gamma_e}{2} \sigma_+ \sigma_- - i\hbar \frac{\gamma}{2} a^+ a. \quad (11)$$

Второе же слагаемое в уравнении (10) можно связать со спонтанными переходами (скачками) в системе за счет взаимодействия с бозонным термостатом. Рассмотрим некоторую реализацию чистого состояния $\rho(t) = |\psi(t)\rangle \langle \psi(t)|$. Тогда из (10) можно увидеть, что изменение матрицы плотности $\rho(t + \Delta t) = |\psi(t + \Delta t)\rangle \langle \psi(t + \Delta t)|$ обусловлено двумя возможными вкладками:

- 1) либо с вероятностью $1 - \Delta P$ согласно

$$|\psi^d(t)\rangle = \frac{e^{-iH_{dis}\Delta t/\hbar} |\psi(t)\rangle}{\sqrt{1 - \Delta P}}, \quad (12)$$

происходит изменение волновой функции (“траектории”) за счет диссипативной динамики (здесь введены следующие обозначения $\Delta P = \Delta t \bar{\gamma}$),

$$\bar{\gamma} = \gamma_\varphi \langle \psi | \sigma_z \sigma_z | \psi \rangle + \gamma_e \langle \psi | \sigma_+ \sigma_- | \psi \rangle + \gamma \langle \psi | a^+ a | \psi \rangle;$$

2) либо с вероятностью ΔP изменение “траектории” сопровождается одним из скачков:

$$|\psi_\varphi^s(t)\rangle = \frac{\sigma_z |\psi(t)\rangle}{\sqrt{\langle \psi | \sigma_z \sigma_z | \psi \rangle}}, \quad |\psi_e^s(t)\rangle = \frac{\sigma_- |\psi(t)\rangle}{\sqrt{\langle \psi | \sigma_+ \sigma_- | \psi \rangle}} \quad \text{и} \quad |\psi_\gamma^s(t)\rangle = \frac{a |\psi(t)\rangle}{\sqrt{\langle \psi | a^+ a | \psi \rangle}}, \quad (13)$$

взятыми соответственно с весами

$$p_\varphi = \frac{\gamma_\varphi}{\bar{\gamma}} \langle \psi | \sigma_z^+ \sigma_z | \psi \rangle, \quad p_e = \frac{\gamma_e}{\bar{\gamma}} \langle \psi | \sigma_+ \sigma_- | \psi \rangle \quad \text{и} \quad p = \frac{\gamma}{\bar{\gamma}} \langle \psi | a^+ a | \psi \rangle.$$

Для нахождения усредненной динамики (аналог того, что получаем в результате решения уравнения (11)) следует получить набор M единичных реализаций, а далее усреднить величину по ним

$$\rho = \frac{1}{M} \sum_{i=1}^M |\psi_i(t)\rangle \langle \psi_i(t)|. \quad (13)$$

После того, как матрица плотности найдена, можно вычислить ожидаемое значение любой наблюдаемой квантовой системы. Например, среднее число квантов осциллятора (средняя энергия):

$$\bar{n} = \frac{\sum_{i=1}^M \sum_{n=0}^N n |C_n^i(t)|^2}{M}, \quad (14)$$

где $C_n^i(t)$ - коэффициенты разложения “монте-карловской” функции по базису, образуемому собственными функциями оператора чисел заполнения $\hat{n}|n\rangle = n|n\rangle, \hat{n} = a^+ a$:

$$|\psi^i(t)\rangle = \sum_{n=0}^{N-1} C_n^i(t) |n\rangle.$$

3.2. Схема распараллеливания квантового метода Монте-Карло

Расчет физических величин может быть организован с использованием GPU ускорителей с применением технологии CUDA [11-12]. Поскольку реализации статистически независимы, отсутствует необходимость в обмене данными между блоками и потоками внутри отдельных блоков, создаваемых на GPU, то есть на каждом процессоре выполняется одна и та же подпрограмма, что позволяет обеспечить параллелизм на уровне данных (SIMT). Однако так как для достижения точности метода МК в 1% необходимо произвести расчет эволюции системы с учетом усреднения по 10000 квантовым траекториям. Поэтому для большей производительности программы эффективно использовать в расчетах кластерную систему, узлы которой содержат GPU. В частности, вычислительные эксперименты проводились на кластере “Лобачевский” Нижегородского государственного университета им. Н.И.Лобачевского [13]. Мы использовали до 64 узлов следующей конфигурации: 2x Intel Xeon E5-2660 CPU (8 cores, 2.2 GHz), 64 GB RAM, NVIDIA Kepler K20X (2 или 3), OS Linux (CentOS 6.4), Intel C/C++ Compiler, Intel MPI, CUDA 6.4 Toolkit. На каждом узле использовались 2 графических сопроцессора. Распределение вычислительной нагрузки между узлами кластера выполнено с использованием технологии MPI. На каждом вычислительном узле кластера для расчета траекторий используются возмож-

ности графических ускорителей. Поскольку расчет отдельных квантовых траекторий является независимым, каждая из них рассчитывается отдельным потоком GPU, при этом один поток GPU может последовательно рассчитать несколько траекторий. Незначительная доля последовательного кода не оказывает существенного влияния на время выполнения.

Для использования всех имеющихся графических ускорителей работа с данными организована следующим образом:

- на управляющем хосте (нулевой процесс) запускается основной модуль программы, который инициализирует данные о параметрах системы, вычисляет начальное состояние, выполняет инициализацию MPI, а также создает структуры необходимых библиотек;
- считываются данные о количестве реализаций (квантовых траекториях) и создаются несколько копий структуры с информацией о гамильтониане, которые будут использоваться на отдельных параллельных потоках, исполняемых на каждом из используемых GPU, причем данные разбиваются на пачки (для каждого GPU) таким образом, чтобы на один блок приходилось по 256 потоков, а количество блоков CUDA определяет самостоятельно запрашивая необходимые данные о встроенных GPU-ускорителях;
- с помощью технологии MPI, с исполняемого хоста, создаются параллельные потоки на каждом вычислительном узле, в соответствии с числом GPU устройств, каждый из которых имеет свой уникальный идентификационный номер. Количество исполняемых заданий на каждой видеокарте соответствует числу квантовых траекторий, по которым производится усреднение. Для этого общее число траекторий делится на части: m_1, m_2, \dots, m_M , каждая из которых исполняется на своем GPU.
- инициализируется ядро вычислений на каждом из GPU и запускается алгоритм программы, промежуточные результаты накапливаются в памяти каждого из используемых графических адаптеров;
- после завершения расчетов всех траекторий на отдельной GPU происходит усреднение данных (т.е. соответственно по m_1, m_2, \dots, m_M);
- на управляющем хосте производится усреднение данных по всем GPU устройствам, запись в файл необходимых сведений о расчетах, а также освобождение выделенной памяти и деинициализация MPI.

Одним из важных условий эффективности работы программы на GPU является оптимальное использование имеющихся типов памяти: глобальной, разделяемой и памяти констант. В частности для решения данной задачи необходимы следующие наборы данных: информация о гамильтониане (хранятся в регистрах разделяемой памяти), константы (хранятся в кешируемой памяти констант), текущие населённости уровней (хранятся в регистрах разделяемой памяти) и результирующие данные (имеют большой объем, хранятся в глобальной памяти). Регистры являются быстрым типом памяти, но имеют небольшой объем, поэтому использование их для работы с результирующими данными не представляется возможным. Глобальная память, напротив, позволяет хранить большие объемы данных, но является относительно медленной. Однако в данной задаче обращения к результирующим данным массивов, находящимся в глобальной памяти, немного, поэтому хранение в глобальной памяти не оказывает существенного влияния на производительность программы. А также отметим, что если при усовершенствовании архитектуры GPU объем разделяемой памяти будет увеличен, то существует возможность хранить результирующие данные в данной памяти GPU, что позволит ещё более эффективно использовать обращение к памяти и ускорить расчет.

Заметим, что точность квантового метода Монте-Карло зависит от числа реализаций $\sim 1/\sqrt{M}$, где M – число реализаций. В силу наличия стохастических процессов двойная точность для расчётов не требуется, а использование операций с одинарной точностью на GPU, существенно ускоряет работу приложения, т.к. вычисления с двойной точностью на несколько порядков медленнее. Для обеспечения сходимости результатов моделирования диссипативной динамики необходимо наличие большого числа реализаций ($M > 10^3$) и генерирования последовательностей случайных чисел непосредственно для каждой точки временной траектории. Формирование массива случайных чисел, необходимого для реализации скачков, выполнялось посредством алгоритма MCG59.

4. Результаты расчетов и полученное ускорение

Несмотря на идейную простоту измерений состояний кубита с помощью нелинейного осциллятора, полная теория процесса проектирования в системе «кубит+осциллятор» далека от завершения. В частности, в такой системе возможен режим сильного взаимодействия, когда осуществляется «перепутывание» (entanglement) подсистем. При этом ранее обсуждалось только поведение связанной системы в среднем, когда она описывалась на языке матрицы плотности. В данном разделе мы исследуем динамику нелинейного осциллятора, обращая особое внимание на его потенциальную возможность являться устройством для редукиции (измерения) квантового состояния кубита. Как известно, проективные измерения кубита следует выполнять осциллятором, который переведен внешней силой в квазиклассический режим.

В результате работы над данным вопросом об измерениях было разработано приложение, реализованное на языке CUDA C с использованием пакета CUDA Toolkit 6.5 и технологии MPI. При расчете диссипативной динамики системы с гамильтонианом Вигнера-Вайскопфа (10) использовалась классическая схема Рунге-Кутты четвертого порядка, хорошо подходящая для систем с диссипацией, а в случае происхождения квантовых скачков производилась перенормировка волновой функции по описанным ранее уравнениям (13).

Для начала были проведены численные эксперименты по вычислению средней энергии нелинейного джозефсоновского осциллятора. Рассмотрим, как ведет себя среднее число фотонов в каждом акте измерения и в среднем по реализациям. Пусть осциллятор в начальный момент времени был заселен на уровень $n_0 = 28$ тогда, если внешнего возбуждения нет, т.е. $f_0 = 0$, среднее число фотонов (средняя энергия) с течением времени уменьшается экспоненциально, что видно на рис. 2, где кривая 1, полученная прямым решением уравнения для матрицы плотности (9). Однако, данная зависимость описывает лишь поведение системы в среднем, в каждой реализации моменты испускания кванта энергии (фотона) случайны, что было промоделировано на основе квантового метода Монте-Карло (кривые 2, 3 и 4 на рис. 2). При числе реализаций равном $M = 100000$ наблюдается точное совпадение зависимостей, посчитанных решением уравнения (9) и квантовым методом Монте-Карло (кривая 1 на рис. 2). При расчетных параметрах ($\gamma = 0,005$) система должна затухнуть на времени $t = 200T = 1256$ (ед), что и подтверждается численным расчетом. Кривая 2 для $M = 10000$ и кривая 3 для $M = 1000$ квантовых траекторий показывает существенное возрастание погрешности статистического метода, а кривая 4 демонстрирует квантовые скачки в уникальной единичной реализации.

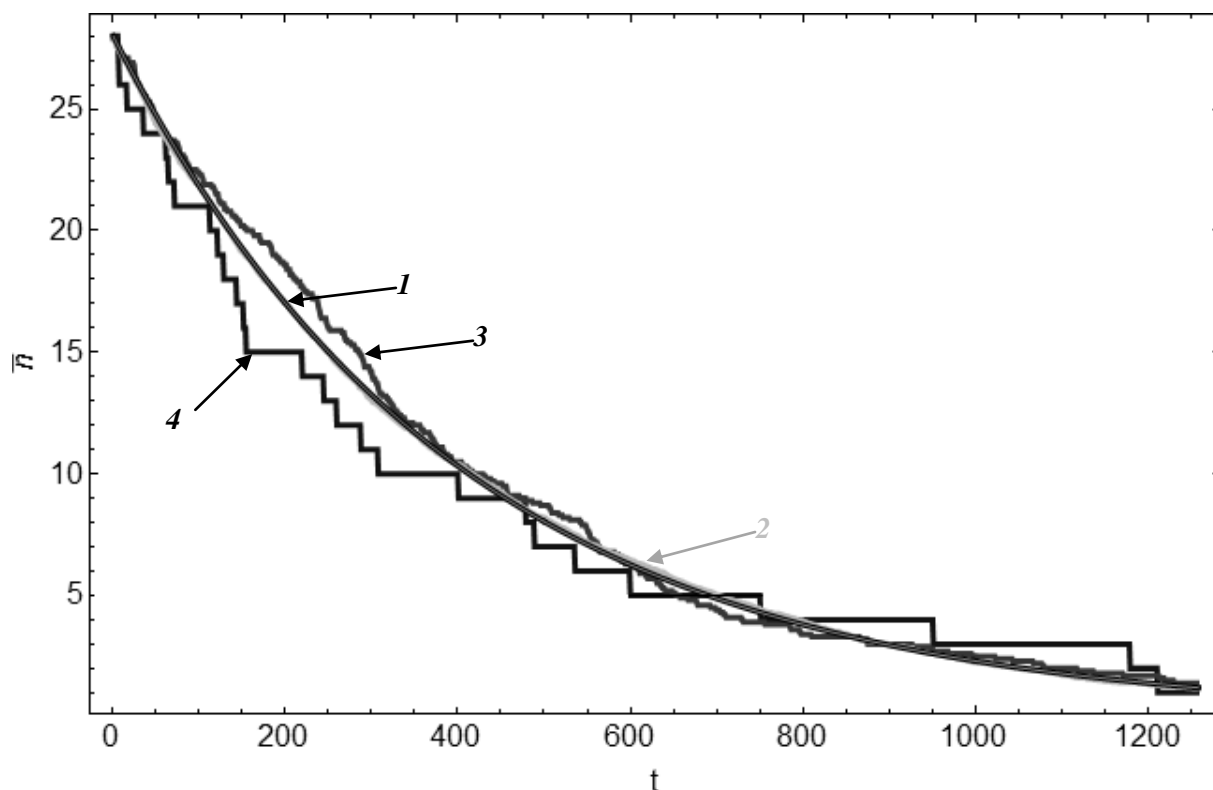


Рис. 2. Зависимость среднего числа фотонов \bar{n} нелинейного осциллятора от времени для разного числа реализаций: кривая 1 для $M = 100000$, кривая 2 - $M = 10000$, кривая 3 - $M = 1000$, кривая 4 - $M = 1$. Параметры системы: $\omega_0 = 1$, $f_0 = 0$, $\mu = 0.001$, $\gamma = 0.005$.

Далее производилось моделирование с возбуждающей силой на интервале времени $0 \leq t \leq 50T$, шаг по времени при расчете скачков $dt = 0.02$, шаг по времени при расчете методом Рунге-Кутты $ddt = 0.000667$. Параметры системы для гамильтониана (5): $\omega_0 = 1.02$, $\mu = 0.001$, $f_0 = 0.2$, $\gamma = 0.001$. Результаты измерения ускорения работы приложения при 50 уровнях осциллятора, связанных с двухуровневой системой кубитом (т.е. эффективно матрица плотности связанной системы имеет размерность 100×100) при усреднении по различному числу реализаций квантовой системы приведены на рис. 3. Описание параметров запуска приведено в таблице 1.

Таблица 1. Параметры запуска

Кол-во GPU	Число квантовых траекторий (M)			
	$M = 1$	$M = 1000$	$M = 10000$	$M = 100000$
1	94.50 сек	120,8 сек	150,91 сек	786,156 сек
2	94,625 сек	110,707 сек	113,758 сек	480,123 сек
4	94,565 сек	108,065 сек	109,466 сек	360,70 сек
6	94,559 сек	107,478 сек	108,95 сек	243,613 сек
8	94,692 сек	106,293 сек	108,39 сек	123,625 сек
10	95,13 сек	106,50 сек	107,86 сек	126,965 сек

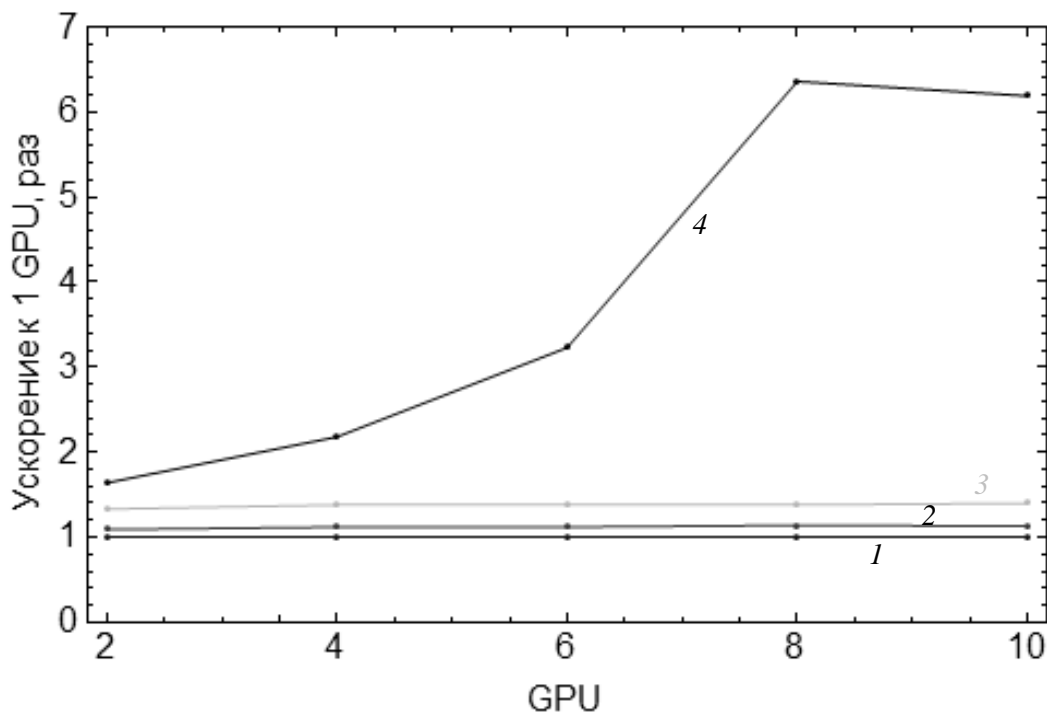


Рис. 3. Ускорение работы алгоритма в зависимости от числа использованных GPU по сравнению с одним GPU. Кривая 1 для $M = 1$, кривая 2 - $M = 100$, кривая 3 - $M = 4000$, кривая 4 - $M = 100000$.

Из графика, представленного на рис. 3, видно, что при увеличении числа реализаций (траекторий M в квантовом методе Монте-Карло) наблюдается существенный выигрыш в ускорении при использовании GPU устройств. Для малого числа реализаций, т.е. если число квантовых траекторий меньше числа потоков на видеокарте, ускорения при использовании одновременно нескольких GPU ускорителей получить не удастся (см. на рис. 3 кривые 1, 2, 3). Это связано с тем, что рассылаемое число заданий для вычислений меньше, чем количество потоков даже на одной видеокарте, не говоря уже о нескольких. Для эффективной работы на каждый GPU должно приходиться не менее 1024 путей (и желательно, чтобы количество путей было кратно 1024), т.е. например при $M = 4000$ достаточно использовать 4 процесса (GPU), ускорения от использования большего количества процессов не будет. Это обусловлено тем, что время расчета одной траектории мало, а при «перегрузке» графического ускорителя наступает эффективная загруженность и достигается оптимальная работа. Прирост производительности вычислений в зависимости от количества используемых графических ускорителей для квантового метода Монте-Карло при условии загруженности (число квантовых траекторий больше, чем число доступных потоков на видеокартах, см. на рис. 3 кривая 4), это говорит о хорошем масштабировании задачи при переходе к множеству GPU. Например, прирост производительности для 100000 квантовых траекторий при использовании 10 GPU увеличился в 6 раз по сравнению с одной видеокартой.

Если сравнивать производительность разработанных параллельных версий программ с использованием технологии CUDA и последовательной версии кода на языке C++ на CPU, то выигрыш в расчетах по времени оказывается очень существенным. Тестирование CPU версии кода производилось на (Intel Core i7 3770K @ 3.50 GHz, использовалось 1 ядро), данные по времени собраны в таблице 2. Видно, что даже для малого числа реализаций ($M \sim 100$), наблюдается ускорение во времени счета в 5 раз относительно CPU за счет использования графического ускорителя. С ростом числа реализаций GPU версия имеет неоспоримое преимущество, что видно из нижней строки таблицы 2.

Таблица 2. Время расчета усредненных M траекторий на GPU устройствах и последовательной версии на CPU. Параметры системы аналогичны тем, которые были взяты для моделирования в таблице 1.

Число реализаций, M	1 CPU	1 GPU	4 GPU	10 GPU

Тип устройства				
$M = 1$	63,73 сек	94.50 сек	94,565 сек	95,13 сек
$M = 1000$	639,10 сек	120,8 сек	108,065 сек	106,50 сек
$M = 10000$	64590,26 сек	150,91 сек	109,466 сек	107,86 сек
$M = 100000$	653247,14 сек	786,156 сек	360,70 сек	126,965 сек

5. Обсуждение результатов

В работе представлена математическая модель связанной системы "кубит+измерительный" осциллятор. Разработан алгоритм, основанный на стохастическом методе квантовых траекторий (квантовом методе Монте-Карло), для моделирования диссипативной динамики многоуровневой квантовой системы на примере кубита, связанного с квантовым нелинейным осциллятором. Квантовый метод Монте-Карло позволяет осуществить эффективное распараллеливание и программную реализацию на гетерогенном высокопроизводительном вычислительном кластере при расчете процессов релаксации в многоуровневых квантовых системах. Создан программный комплекс, который реализует параллельные вычисления на многопроцессорном кластере (с использованием протокола MPI) и графических процессорных устройств (с использованием технологии CUDA). Продемонстрирована эффективность работы графических ускорителей, масштабируемость и описывается возможность взаимодействия нескольких графических ускорителей с применением технологии MPI. Показано, что применимость GPU ускорителей позволило существенно ускорить вычисления. Развита в работе техника расчета диссипативной динамики многоуровневых систем естественным образом может быть распространена на более сложные физические системы.

Литература

1. Нильсен М., Чанг И., Квантовые вычисления и квантовая информация. М.: Мир, 2006, 824 с.
2. Ladd T. D., Jelezko F., Laflamme R., Nakamura Y., Monroe C & O'Brien J. L., Quantum computers // Nature, 2010. V 464, P. 45.
3. Zagoskin A. M., Quantum Engineering – Theory Design of Quantum Coherent Structures, (Cambridge: Cambridge University Press), 2011, 346 p.
4. You, J.Q., Nori, F.: Atomic physics and quantum optics using superconducting circuits. Nature, 2011. V. 474, P. 589.
5. Siddiqi I, Vijay R., Pierre F., Wilson C. M., Metcalfe M., Rigetti C., Frunzio L., and Devoret M. H., RF-Driven Josephson Bifurcation Amplifier for Quantum Measurement//Phys. Rev. Lett., 2004. V. 93. P. 207002.
6. Siddiqi I. et. al., An RF-Driven Josephson Bifurcation Amplifier for Quantum Measurements//Phys. Rev. B, 2006. V. 73. P. 054510.
7. М.О. Скалли, М.С. Зубайри Квантовая оптика. Под ред. В.В. Самарцева. – М.: ФИЗМАТ-ЛИТ, 2003. 512 с.
8. Plenio M. B. and Knight P. L. The Quantum Jump Approach to Dissipative Dynamics in Quantum Optics// Rev. Mod. Phys., 1998. V. 70. P. 101.
9. Vijay R., Slichter D. H., Siddiqi I., Observation of Quantum Jumps in a Superconducting Artificial Atom// Phys. Rev. Lett., 2011. V. 106. P. 110502.

10. Лихарев К.К., Введение в динамику джозефсоновских переходов. М.: Наука, 1985. 320 с.
11. Сандерс Д., Кэндрот Э., Технология CUDA в примерах. Введение в программирование графических процессов: Пер. с англ. Слинкина А.А., научный редактор Боресков А.В. М.: ДМК Пресс, 2011. 232 с.
12. А.В. Боресков, А.А. Харламов, Основы работы с технологией CUDA. Издательство "ДМК Пресс", 2010. 234 с.
13. URL: http://hpc-education.unn.ru/files/seminars/_lobachevsky/01_Lobachevsky_general_info.pdf (дата обращения 27.07.2015).
14. Гергель В.П., Стронгин Р.Г., Основы параллельных вычислений для многопроцессорных вычислительных систем. - Н.Новгород, ННГУ, 2 изд, 2003.
15. Гергель В. П. Теория и практика параллельных вычислений. Бинوم. Лаборатория знаний, 2007. 424 с.