

P11-2015-19

А. В. Волохова, Е. В. Земляная, В. С. Рихвицкий

ПАРАЛЛЕЛЬНАЯ ОПТИМИЗАЦИЯ МЕТОДА
РЕШЕНИЯ СИСТЕМЫ УРАВНЕНИЙ ПОЛЯРОНА
С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМА РАЗБИЕНИЙ

Направлено в журнал «Вычислительные методы и программирование»

Параллельная оптимизация метода решения системы уравнений полярона с использованием алгоритма разбиений

Разработанный ранее метод численного моделирования процесса формирования поляронных состояний в конденсированных средах модифицирован с применением алгоритма разбиений, что обеспечивает существенное ускорение вычислений при расчетах в параллельном режиме на многопроцессорных системах. Программная реализация выполнена на основе технологии параллельного программирования MPI. Проведенные методические расчеты на многопроцессорном кластере ЦИВК ЛИТ ОИЯИ (Дубна) с различным количеством задействованных процессоров при выборе различных параметров вычислительной схемы подтверждают эффективность разработанного подхода для численного решения системы нелинейных дифференциальных уравнений в частных производных, описывающих динамическую модель полярона.

Работа выполнена в Лаборатории информационных технологий ОИЯИ.

Parallel Optimization Method for Numerical Solving of the Polaron Equations System Using the Partitions Algorithm

Previously developed method for the numerical simulation of the polaron states formation in condensed matter has been modified using the partitions algorithm that provides a significant speedup in the parallel calculations on multiprocessor systems. Software implementation is based on the MPI technology. Methodical calculations are carried out on a multiprocessor cluster CICC LIT JINR (Dubna) with a different processors number and a different choice of computational parameters. The calculations confirm the effectiveness of the developed approach for numerical solution of the system of nonlinear differential equations describing the polaron dynamical model.

The investigation has been performed at the Laboratory of Information Technologies, JINR.

ВВЕДЕНИЕ

Сольватированные (гидратированные) состояния электронов образуются, согласно современным представлениям, в результате радиолиза конденсированных сред. При образовании такого состояния происходит захват избыточного свободного электрона поляризационной потенциальной ямой, образованной индуцированной им поляризацией окружающих его молекул среды. Таким образом, под действием облучения вода (или иная среда) переходит в особое состояние, характеризующееся специальными физическими и химическими свойствами. Изучение этих свойств имеет огромное теоретическое и прикладное значение. Важную информацию о гидратированном электроне и кинетике химических реакций с его участием дает динамика спектров поглощения при его образовании в облученной воде, получаемых на основе современных фемтосекундных спектроскопических методов. В работе [1] предложена поляронная модель гидратированного электрона и показано, что теоретические оценки динамики поглощения лазерного излучения в ультрафиолетовом диапазоне качественно согласуются с экспериментальными данными из [2].

В работах [3–5] разработан метод и комплекс программ для численного решения системы нелинейных уравнений, описывающей динамическую модель полярона. Проведенное на этой основе численное моделирование процесса образования гидратированного электрона подтвердило, что используемый подход обеспечивает согласие как с экспериментальными данными поглощения света поляризованной водой, так и с теоретическими оценками энергии связи и среднего радиуса гидратированного электрона. С вычислительной точки зрения расчеты сводятся к многократному численному решению упомянутой системы уравнений с различными начальными условиями с тем, чтобы найти начальное состояние, обеспечивающее согласующуюся с теорией и экспериментом динамику вычисляемых на каждом шаге по времени физических характеристик. Представленный в работе [4] комплекс программ оптимизирован для двухпроцессорной системы, что позволяет почти в два раза ускорить расчеты по сравнению с последовательным однопроцессорным

режимом. Однако и в двухпроцессорном режиме каждый расчет с заданным начальным условием занимает примерно 2 ч. Поскольку, как уже сказано, для полного численного исследования процессов формирования сольватированных электронов в конденсированных средах требуются массовые расчеты с разными начальными условиями, задача эффективной параллельной оптимизации остается актуальной.

В работе описана параллельная вычислительная схема с использованием на каждом временном слое алгоритма разбиений [8] и представлены результаты методических расчетов для различного количества параллельных процессоров и узлов дискретной сетки по пространственной и временной координатам, подтверждающие эффективность разработанного параллельного алгоритма. В разд. 1 дана математическая постановка задачи для системы нелинейных дифференциальных уравнений, описывающей динамику поляронных состояний. В разд. 2 представлена схема численного решения системы уравнений полярона. Разд. 3 посвящен алгоритму разбиений. В разд. 4 обсуждаются полученные численные результаты.

1. СИСТЕМА УРАВНЕНИЙ ПОЛЯРОНА

На основе функционала полной энергии оптического полярона Пекара в работе [1] сформулированы соответствующие уравнения движения, описывающие динамику волновой функции полярона, плотности распределения поляризационного заряда, индуцированного электроном, и потенциала поляризации с учетом релаксации. С использованием разложения указанных характеристик по сферическим гармоникам, переходя к безразмерным величинам и ограничиваясь рассмотрением сферически-симметричного случая, мы получаем систему пространственно-одномерных нелинейных дифференциальных уравнений в частных производных, описывающих эволюцию заданного в начальный момент времени поляронного состояния.

Динамическая модель полярона подробно описана в [5]. Здесь мы приводим только окончательную форму уравнений, описывающих эволюцию заданного в нулевой момент времени поляронного состояния в сферически-симметричном приближении:

$$\begin{cases} \left[i2\bar{m}\frac{\partial}{\partial t} + \frac{\partial^2}{\partial x^2} + 2\bar{m}\frac{r_{00}}{\bar{\epsilon}}\frac{\phi(x,t)}{x} \right] \psi(x,t) = 0, \\ \frac{\partial^2}{\partial x^2} \phi(x,t) = \Theta(x,t), \\ \left[\frac{\partial^2}{\partial t^2} + \gamma\frac{\partial}{\partial t} + \omega^2 \right] \Theta(x,t) = -\omega^2 \frac{|\psi(x,t)|^2}{x}. \end{cases} \quad (1)$$

Граничные условия имеют вид

$$\phi(0, t) = \phi'(\infty, t) = 0, \quad \psi(0, t) = \psi(\infty, t) = 0, \quad \Theta(0, t) = \Theta(\infty, t) = 0. \quad (2)$$

Здесь $\psi(x, t)$ — безразмерная радиальная составляющая волновой функции электрона; $\phi(x, t)$ — безразмерная радиальная часть потенциала поляризации, индуцируемого электроном; безразмерная функция $\Theta(x, t)$ имеет физический смысл радиальной части плотности распределения поляризационного заряда; $\tilde{\epsilon} = 1,81$ — коэффициент диэлектрической проницаемости; $\bar{m} = 2,692$, $\gamma = 2,145$, $\omega = 1$ — безразмерные параметры эффективной массы гидратированного электрона, релаксации (трения) и частоты; r_{00} — масштабирующий множитель, введенный для того, чтобы «уравновесить» отличающиеся более чем на 10 порядков исходные значения физических параметров.

2. КОНЕЧНО-РАЗНОСТНАЯ СХЕМА

Для численного решения система дифференциальных уравнений (1) заменяется системой разностных уравнений на равномерной дискретной сетке с узлами $x_m = mh_x$ и $t_n = nh_t$, где h_x и h_t — шаги по пространственной и временной переменным. В результате подстановок известных конечно-разностных формул [6] мы получаем систему сеточных уравнений относительно значений функций $\psi_m^n = \psi(x_m, t_n)$, $\phi_m^n = \phi(x_m, t_n)$, $\Theta_m^n = \Theta(x_m, t_n)$ в узлах дискретной сетки:

$$\left\{ \begin{array}{l} \frac{\psi_m^{n+1} - \psi_m^n}{h_t} = i \left\{ \sigma \left[\frac{\psi_{m+1}^{n+1} - 2\psi_m^{n+1} + \psi_{m-1}^{n+1}}{2\bar{m}h_x^2} + \frac{r_{00}}{\tilde{\epsilon}} \frac{\varphi_m^{n+1}}{mh_x} \psi_m^{n+1} \right] + \right. \\ \left. + (1 - \sigma) \left[\frac{\psi_{m+1}^n - 2\psi_m^n + \psi_{m-1}^n}{2\bar{m}h_x^2} + \frac{r_{00}}{\tilde{\epsilon}} \frac{\varphi_m^n}{mh_x} \psi_m^n \right] \right\}, \\ \frac{\varphi_{m+1}^{n+1} - 2\varphi_m^{n+1} + \varphi_{m-1}^{n+1}}{h_x^2} = \Theta_m^{n+1}, \\ \frac{\Theta_m^{n+1} - 2\Theta_m^n + \Theta_m^{n-1}}{h_t^2} + \gamma \frac{\Theta_m^{n+1} - \Theta_m^n}{h_t} + \omega^2 \Theta_m^{n+1} = -\omega^2 \frac{|\psi_m^n|^2}{mh_x}, \\ \sigma = 0,5, \quad \Theta_m^{-1} = -\frac{|\psi_m^0|^2}{mh_x}; \quad \Theta_m^0 = \Theta_m^{-1}; \end{array} \right. \quad (3)$$

$$\varphi_0^n = 0; \quad \varphi_l^n = \varphi_{l-1}^n; \quad m = 1, 2, \dots, l; \quad n = 0, 1, 2, \dots$$

В работе [7] была предложена схема численного исследования динамической модели полярона, построенная так, чтобы обеспечить возможность на каждом n -м временном слое последовательно вычислять значения Θ^{n+1} , φ^{n+1} и ψ^{n+1} , избегая тем самым численного решения для каждого n системы трех нелинейных уравнений. Корректность схемы применительно к численному решению задачи (3) проверена путем тестовых расчетов в [3]. Итак, на каждом слое с номером n выполняется следующая последовательность действий.

На **первом шаге** из третьего уравнения системы (3) при известных ψ^n явно вычисляем Θ^{n+1} для всех m по формуле

$$\Theta_m^{n+1} = \left[-\frac{-2\Theta_m^n + \Theta_m^{n-1}}{h_t^2} - \gamma \frac{-\Theta_m^n}{h_t} - \omega^2 \frac{|\psi_m^n|^2}{mh_x} \right] \cdot \left[\frac{1}{h_t^2} + \frac{\gamma}{h_t} + \omega^2 \right]^{-1}. \quad (4)$$

На **втором шаге** найденные Θ_m^{n+1} подставляются во второе уравнение. Полученная система $N = l - 1$ алгебраических уравнений

$$\hat{A}\mathbf{X} = \mathbf{D}; \quad \hat{A} = \begin{bmatrix} b_1 & c_1 & 0 & \dots & 0 \\ a_2 & b_2 & c_2 & \dots & 0 \\ 0 & a_3 & b_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & b_N \end{bmatrix}; \quad \mathbf{D} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \dots \\ d_N \end{bmatrix}; \quad \mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \dots \\ x_N \end{bmatrix} \quad (5)$$

с коэффициентами

$$\begin{aligned} a_m &= 1, & b_m &= -2, & c_m &= 1, & d_m &= \Theta_m h_x^2, & m &= 2, \dots, N-1; \\ a_1 &= 0, & b_1 &= 1, & c_1 &= 0, & d_1 &= 0, \\ a_N &= -1, & b_N &= 1, & c_N &= 0, & d_N &= 0 \end{aligned} \quad (6)$$

решается тем или иным методом для вычисления φ_m^{n+1} .

Наконец, на **третьем шаге** полученные значения φ_m^{n+1} подставляются в первое уравнение системы (3). В результате получается алгебраическая система вида (5) с коэффициентами (для $m = 2, \dots, N-1$)

$$\begin{aligned} a_m &= \frac{i\sigma}{2\bar{m}h_x^2}, & c_m &= \frac{i\sigma}{2\bar{m}h_x^2}, & b_m &= \frac{1}{h_t} + \frac{i\sigma \left(2 - 2h_x \frac{\bar{m}r_{00}\varphi_m^{n+1}}{\bar{\epsilon}m} \right)}{2\bar{m}h_x^2}, \\ d_m &= \frac{\psi_m^n}{h_t} + \frac{i(1-\sigma) \left(2h_x \frac{\bar{m}r_{00}\varphi_m^n \psi_m^n}{\bar{\epsilon}m} + \psi_{m+1}^n - 2\psi_m^n + \psi_{m-1}^n \right)}{2\bar{m}h_x^2}, \\ a_1 &= 0, & b_1 &= 1, & c_1 &= 0, & d_1 &= 0, & a_N &= 0, & b_N &= 1, & c_N &= 0, & d_N &= 0. \end{aligned} \quad (7)$$

Эта система решается относительно ψ_m^{n+1} , после чего процедура повторяется для следующего значения n .

Для начала расчетов необходимо задать значения φ_m^0 . В работах [3–5] начальное условие варьировалось с тем, чтобы воспроизвести теоретически предсказанное и экспериментально измеренное поведение физических характеристик гидратированного электрона в зависимости от времени. Кроме того, расчеты в методических целях проводились также с начальными условиями в форме стационарных решений задачи (1).

Численное решение трехдиагональных алгебраических систем уравнений (5) на втором и третьем шагах вышеописанного алгоритма может осуществляться разными способами. В начальном варианте компьютерной реализации использовался классический метод прогонки. Применение метода встречных прогонок обеспечивает возможность двукратного ускорения расчетов путем распределения вычислений на два параллельно работающих процессора. В настоящей реализации для решения алгебраических систем на 2-м и 3-м шагах нашего алгоритма используется метод разбиений, позволяющий распределить вычисления в принципе на произвольное число процессоров [8]. Подробное описание алгоритма разбиений дано в следующем разделе. Естественно, расчеты на первом шаге алгоритма по формуле (4) также распараллеливаются по пространственной координате между задействованными параллельными процессорами.

3. МЕТОД РАЗБИЕНИЙ

Идея метода состоит в следующем. Исходная система (5) N линейных алгебраических уравнений с трехдиагональной структурой матрицы \hat{A} разбивается на группы по числу задействованных параллельных процессоров (MPI-процессов). Пусть P — число параллельных MPI-процессов, имеющих номера $p = 0, 1, \dots, P - 1$. Число N не обязано делиться на P нацело. Пусть $\Delta = \lfloor N/P \rfloor$. Каждому процессору с номером $p < (N \bmod P)$ назначается для дальнейшей обработки группа строк с номерами от $i_{\min} = p(\Delta + 1) + 1$ до $i_{\max} = (p + 1)(\Delta + 1)$, каждому процессору с номером $p \geq (N \bmod P)$ назначается группа строк с номерами от $i_{\min} = p\Delta + (N \bmod P) + 1$ до $i_{\max} = (N \bmod P) + (p + 1)\Delta$. В каждой группе ненулевые элементы сосредоточены в блоке размером $\Delta \times \Delta$ либо $(\Delta + 1) \times (\Delta + 1)$, расположенном в области главной диагонали исходной матрицы \hat{A} .

Над уравнениями внутри каждой группы проводятся преобразования, в результате которых ненулевыми оказываются только элементы, расположенные на главной диагонали блока и в вертикальных столбцах на левой и правой границах блока. Соответствующие арифметические операции осуществляются и над элементами вектора правой части \mathbf{D} . После преобразований матрица \hat{A}

в случае $\Delta = 5$ принимает следующий вид:

$$\left[\begin{array}{ccccc|ccccc|c} b'_1 & 0 & 0 & 0 & c'_1 & 0 & 0 & 0 & 0 & 0 & \dots \\ a'_2 & b'_2 & 0 & 0 & c'_2 & 0 & 0 & 0 & 0 & 0 & \dots \\ a'_3 & 0 & b'_3 & 0 & c'_3 & 0 & 0 & 0 & 0 & 0 & \dots \\ a'_4 & 0 & 0 & b'_4 & c'_4 & 0 & 0 & 0 & 0 & 0 & \dots \\ a'_5 & 0 & 0 & 0 & b'_5 & c'_5 & 0 & 0 & 0 & 0 & \dots \\ \hline 0 & 0 & 0 & 0 & a'_6 & b'_6 & 0 & 0 & 0 & c'_6 & \dots \\ 0 & 0 & 0 & 0 & 0 & a'_7 & b'_7 & 0 & 0 & c'_7 & \dots \\ 0 & 0 & 0 & 0 & 0 & a'_8 & 0 & b'_8 & 0 & c'_8 & \dots \\ 0 & 0 & 0 & 0 & 0 & a'_9 & 0 & 0 & b'_9 & c'_9 & \dots \\ 0 & 0 & 0 & 0 & 0 & a'_{10} & 0 & 0 & 0 & b'_{10} & \dots \\ \hline \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{array} \right]. \quad (8)$$

Далее из первой и последней строк каждой группы формируется вспомогательная матрица малой размерности, которая также имеет трехдиагональную структуру. С помощью метода прогонки вычисляются соответствующие элементы x_i ($i = i_{\min}$ и $i = i_{\max}$ для каждой группы), на основе которых рекурсивно рассчитываются остальные элементы вектора \mathbf{X} внутри каждой группы. Все расчеты по преобразованию матрицы и получению решений внутри каждой группы осуществляются назначенным ей процессором (MPI-процессом) в параллельном режиме. Нераспараллеливаемая часть алгоритма включает расчеты, связанные с формированием вспомогательной матрицы и вычислением соответствующих ей решений. Эти вычисления проводятся на процессоре с номером $p = 0$. Взаимодействие между процессами необходимо на этапе сборки вспомогательной матрицы и на этапе рассылки полученных решений. Используются процедуры MPI: Gather, Scatter, Reduce, Send, Recv.

Для реализации алгоритма разбиений удобно записать три диагонали матрицы (5) и вектор правой части в виде четырех столбцов — a , b , c и d . Преобразование матрицы \hat{A} в форму \hat{A}' осуществляется «на месте», т. е. новые значения замещают соответствующие старые, а те элементы, для которых замещение не указано, сохраняют прежние значения. Преобразование осуществляется независимо каждым вычислительным узлом в два этапа.

На первом этапе преобразование двух смежных уравнений с номерами i и $i + 1$ в каждом фрагменте происходит по формулам

$$a_i^{\text{new}} = a_i/b_i, \quad c_i^{\text{new}} = c_i/b_i, \quad d_i^{\text{new}} = d_i/b_i, \quad (9)$$

$$b_{i+1}^{\text{new}} = b_{i+1} - c_i a_{i+1}, \quad d_{i+1}^{\text{new}} = d_{i+1} - d_i a_{i+1}, \quad a_{i+1}^{\text{new}} = -a_i a_{i+1}, \quad (10)$$

где i последовательно принимает значения от $i_{\min} + 1$ до $i_{\max} - 1$.

На втором этапе осуществляется преобразование строк в обратном порядке:

$$a_i^{\text{new}} = a_i - a_{i+1}c_i, \quad d_i^{\text{new}} = d_i - d_{i+1}c_i, \quad c_i^{\text{new}} = -c_{i+1}c_i. \quad (11)$$

Здесь i последовательно «пробегают» значения от $i_{\max} - 2$ до $i_{\min} + 1$.

Вычисление коэффициентов первого уравнения в каждом фрагменте производится по формулам (здесь $i = i_{\min}$)

$$b_i^{\text{new}} = b_i - a_{i+1}c_i, \quad d_i^{\text{new}} = d_i - d_{i+1}c_i, \quad c_i^{\text{new}} = -c_{i+1}c_i. \quad (12)$$

Процессору с номером $p = 0$ передаются первые и последние ($i = i_{\min}$ и $i = i_{\max}$) строки каждого блока преобразованной системы (8). Получаемая в результате вспомогательная система уравнений размерности $2P$ имеет трехдиагональную структуру. Для $\Delta = 5$ она выглядит следующим образом:

$$\begin{bmatrix} b'_1 & c'_1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ a'_5 & b'_5 & c'_5 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & a'_6 & b'_6 & c'_6 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & a'_{10} & b'_{10} & c'_{10} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & \dots & a'_{N-4} & b'_{N-4} & c'_{N-4} \\ 0 & 0 & 0 & 0 & \dots & \dots & 0 & b'_N & c'_N \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_5 \\ x_6 \\ x_{10} \\ \dots \\ x_{N-4} \\ x_N \end{bmatrix} = \begin{bmatrix} d'_1 \\ d'_5 \\ d'_6 \\ d'_{10} \\ \dots \\ d'_{N-4} \\ d'_N \end{bmatrix}. \quad (13)$$

Решение системы (13) осуществляется методом прогонки. Вычисленные значения вектора \mathbf{X} 0-й процессор рассылает остальным вычислительным узлам. Используя полученные значения, все процессоры рекурсивно вычисляют остальные значения вектора \mathbf{X} в назначенных им блоках.

4. РЕЗУЛЬТАТЫ РАСЧЕТОВ

Программная реализация представленной вычислительной схемы выполнена на языке C++ с использованием технологии параллельного программирования MPI. Затраты времени в методе разбиений при решении системы l алгебраических уравнений (5) с использованием P параллельных MPI-процессов можно приближенно оценить, как

$$T(N, P) \sim A_{\Delta}N/P + B_0P + C_t(P, N). \quad (14)$$

Первое слагаемое отражает затраты каждого MPI-процесса на обработку на каждом шаге назначенного ему фрагмента на первом и заключительном этапах метода разбиений. Второе слагаемое отражает затраты на решение процессом $p = 0$ малой системы (13) размера $2P$, которые растут пропорционально P , а также пропорциональные P затраты на межпроцессорный обмен,

объем которого в данном алгоритме имеет для каждого процесса фиксированный размер. При больших N и относительно небольших P можно ожидать уменьшения времени счета пропорционально $1/P$. Третье слагаемое в оценке (14) «отвечает» за нелинейную часть временных затрат на межпроцессорный обмен, зависящий, вообще говоря, от архитектуры конкретной вычислительной системы. Представленные в конце этого раздела расчеты подтверждают соответствие реальных временных затрат оценке (14) и незначительный вклад третьего слагаемого при $P \leq 12$.

В ходе вычислительных экспериментов на многопроцессорном кластере ЦИВК (ЛИТ ОИЯИ, Дубна) выявлены закономерности зависимостей времени счета от количества параллельных вычислительных узлов при различном выборе параметров конечно-разностной схемы.

Проведенные расчеты с числом процессоров от 1 до 40 при различных значениях l (число узлов дискретной сетки по x) представлены на рис. 1. Количество шагов по t здесь равно $n = 2\,000\,000$.

При изменении количества процессоров от 1 до 12 имеет место уменьшение времени счета с ростом числа параллельных вычислительных узлов по гиперболическому закону (рис. 1, *а*). Далее уменьшение времени счета замедляется, а при $P = 13$ и $P = 24$ наблюдается скачкообразное увеличение времени счета с последующим плавным снижением. Эта закономерность прослеживается при различных значениях l . При этом для $l = 5000$ минимальное время счета достигается на 12 процессорах, в то время как для большего l минимум временных затрат происходит при $P = 23$.

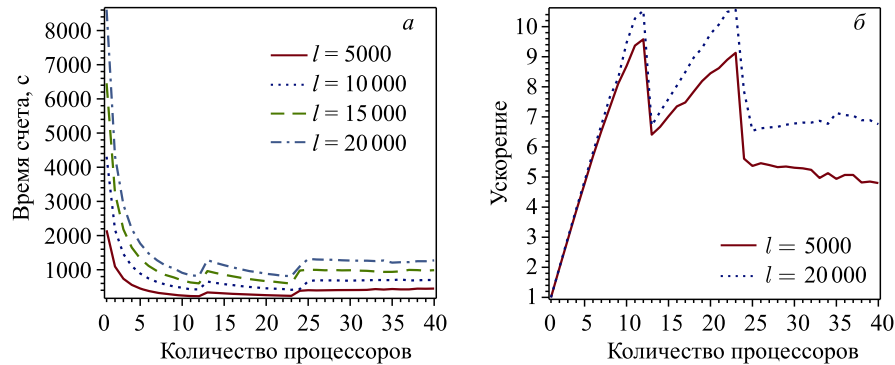


Рис. 1. *а*) Зависимость длительности вычислений от количества параллельных процессоров (MPI-процессоров) при различном числе узлов дискретной сетки по пространственной координате x . *б*) Ускорение вычислений при увеличении числа процессоров при различном количестве узлов дискретной сетки по x . Количество шагов по временной координате t равно $2\,000\,000$

Аналогичные скачки, по-видимому, связанные с особенностями архитектуры используемого многопроцессорного кластера, видны при указанных значениях P и на графике ускорения вычислений в зависимости от количества параллельных вычислительных узлов (рис. 1, *a*). На этом же рисунке показано, что максимальное ускорение вычислений по сравнению с расчетами в однопроцессорном режиме составляет примерно 10.

На рис. 2 приведены результаты расчетов с $l = 5000$ (*a*) и $l = 20000$ (*b*) для различного числа шагов по t в зависимости от количества параллельных вычислительных узлов P (от 1 до 25). В случае $l = 5000$ и $n = 2000000$

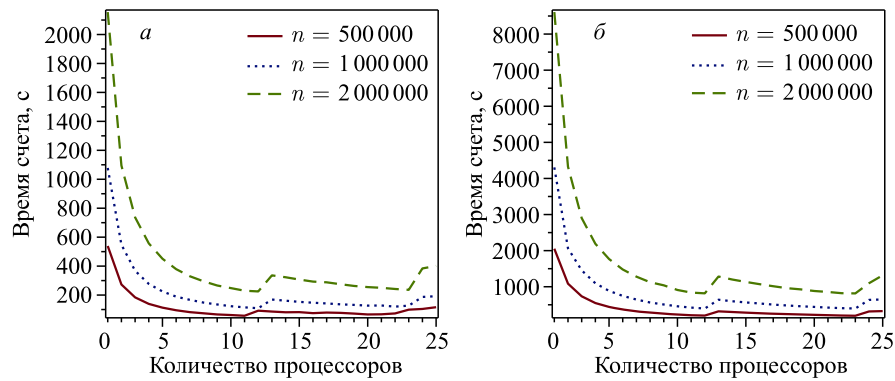


Рис. 2. Зависимость длительности вычислений от числа процессоров при различном количестве шагов по временной координате t . Количество шагов по пространственной координате $l = 5000$ (*a*) и $l = 20000$ (*b*)

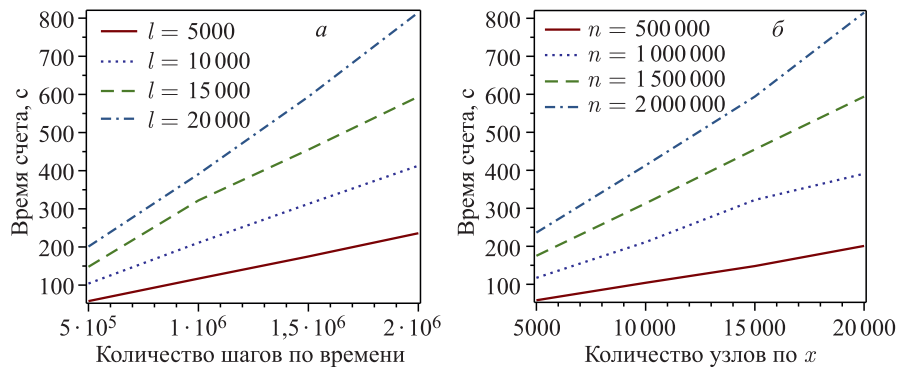


Рис. 3. Зависимость длительности вычислений от количества шагов по t при разных значениях l (*a*) и от количества узлов дискретной сетки по x при разных n (*b*). Расчеты проведены для фиксированного числа параллельных процессоров $P = 23$

минимальное значение времени счета достигается при $P = 12$. С уменьшением n до 500 000 наблюдается смещение минимума к $P = 11$. При $l = 20\,000$ наблюдается два минимума: на 12 и 23 процессорах, глубина которых незначительно варьируется при изменении n .

На рис. 3 представлена зависимость времени счета от n для различных значений l и зависимость времени счета от l для различных n при фиксированном значении $P = 23$. Видно, что при увеличении количества шагов по t время расчета растет быстрее с возрастанием количества узлов по x , причем рост временных затрат более интенсивный при $l = 20\,000$. При больших l временные затраты растут быстрее с увеличением n .

Подгонка параметров A_Δ и B_0 в формуле (5) в предположении $C_t(P, l) = 0$ ($l = N + 1$) по совокупности $T_{l,P}$ результатов вычислительных экспериментов при $n = 2\,000\,000$ и $P \leq 12$ путем минимизации по A_Δ и B_0 величины χ^2

$$\chi^2 = \sum \frac{(T_{l,P} - T(l, P))^2}{T_{l,P}} \quad (15)$$

дала значения $A_\Delta = 0,4331$ и $B_0 = 3,9995$ при доверительной вероятности $CL = 0,999$.

ЗАКЛЮЧЕНИЕ

Можно констатировать, что применение метода разбиений в рассмотренной здесь задаче оказалось вполне эффективным. При этом разработанная ранее и хорошо себя зарекомендовавшая вычислительная схема, описанная в разд. 3, осталась неизменной. Достигнутое ускорение в целом соответствует результатам, обычно получаемым при параллельной реализации типовых разностных методов для пространственно-одномерных дифференциальных уравнений в частных производных (см., например, [9, 10]). Для сравнения приведем также работу [11], где метод разбиений реализован в рамках численного исследования известной в астрофизике системы уравнений Янга–Миллса с дилатонным полем. С точки зрения математической постановки задача представляет собой систему двух нелинейных пространственно-одномерных волновых уравнений с соответствующими граничными и начальными условиями и решается на основе консервативной конечно-разностной схемы с применением итерационного метода на каждом временном слое. Решение системы линейных алгебраических уравнений на каждой итерации осуществляется с использованием алгоритма, аналогичного описанному в разд. 4. При расчетах на кластере ЦИВК при использовании 7 MPI-процессов в упомянутой работе [11] достигалось приблизительно четырехкратное ускорение. В наших расчетах в этом же случае ускорение составляет 6,7 раз.

Вычисления показывают, что при количестве узлов дискретной сетки по пространственной координате $l \leq 5000$ и $l > 5000$ минимальное время счета достигается соответственно при $P = 12$ и $P = 23$. При этом, поскольку разница между временами счета для этих случаев невелика, оптимальным для данной задачи можно считать использование 12 процессоров. По результатам представленных расчетов можно сделать вывод, что использование алгоритма разбиений для параллельной оптимизации метода численного решения системы нелинейных дифференциальных уравнений в частных производных, описывающей динамическую модель полярона, позволяет фактически на порядок сократить затраты компьютерного времени и тем самым повысить эффективность дальнейшего численного исследования механизмов формирования сольватированных электронов в различных конденсированных средах.

Авторы признательны О. И. Стрельцовой и М. И. Зуеву за предоставленные материалы по алгоритму разбиений. Авторы благодарят И. В. Пузынина и Т. П. Пузынину за внимание к работе и полезные замечания. Работа выполнена при частичной финансовой поддержке РФФИ (грант 13-01-00060а).

ЛИТЕРАТУРА

1. *Lakhno V. D.* Dynamical Polaron Theory of the Hydrated Electron // *Chem. Phys. Lett.* 2007. V. 437. P. 198–202.
2. *Long F. H., Hong L., Kenneth B. E.* Femtosecond Studies of the Presolvated Electron: An Excited State of the Solvated Electron // *Phys. Rev. Lett.* 1990. V. 64, No. 12. P. 1469–1472.
3. *Волохова А. В. и др.* Компьютерное моделирование формирования состояний гидратированного электрона // *Вестник РУДН. Серия: «Математика. Информатика. Физика»*. М.: Изд-во Российского университета дружбы народов, 2014. № 2. С. 244–247.
4. *Волохова А. В. и др.* Численное исследование фотовозбужденных поляронных состояний в воде // *Компьютерные исследования и моделирование*. 2014. Т. 6, № 2. С. 253–261.
5. *Лакно В. Д. и др.* Поляронная модель формирования состояний гидратированного электрона // *Поверхность. Рентгеновские, синхротронные и нейтронные исследования*. 2015. № 1. С. 82–87.
6. *Березин И. С., Жидков Н. П.* Методы вычислений. Т. 2. М.: Наука, 1959.
7. *Амирханов И. В. и др.* Математическое моделирование эволюции поляронных состояний // *Поверхность. Рентгеновские, синхротронные и нейтронные исследования*. 2011. № 1. С. 66–70.

8. *Wang H. H.* A Parallel Method for Tridiagonal Equations // ACM Trans. Math. Software. 1981. No. 7. P. 170–183.
9. *Баркалов К. А.* Методы параллельных вычислений. Н. Новгород: Изд-во Нижегородского государственного университета им. Н. И. Лобачевского, 2011.
10. *Старченко А. В. и др.* Практикум по методам параллельных вычислений. Серия: «Суперкомпьютерное образование». М.: МГУ, 2010.
11. *Айрян Э. А. и др.* Численное исследование распада возмущенных стационарных решений системы уравнений Янга–Миллса с дилатоном с использованием технологии MPI // Математическое моделирование. 2005. Т. 17, № 6. С. 103–121.

Получено 24 марта 2015 г.

Редактор *А. И. Петровская*

Подписано в печать 19.05.2015.

Формат 60 × 90/16. Бумага офсетная. Печать офсетная.

Усл. печ. л. 0,94. Уч.-изд. л. 1,17. Тираж 245 экз. Заказ № 58542.

Издательский отдел Объединенного института ядерных исследований
141980, г. Дубна, Московская обл., ул. Жолио-Кюри, 6.

E-mail: publish@jinr.ru

www.jinr.ru/publish/