

УДК 519.3

Л.П.Фельдман, И.А.Назарова
Донецкий национальный технический университет
feldman@r5.dgtu.donetsk.ua, nazar451@gmail.com

Параллельные экспоненциальные методы решения задачи Коши с оценкой локальной апостериорной погрешности

Экспоненциальные методы решения линейной задачи Коши относятся к специальным методам решения систем линейных ОДУ и обладают высокой степенью параллелизма. Данная статья посвящена разработке и анализу эффективности параллельных экспоненциальных алгоритмов на базе явных одношаговых вложенных и экстраполяционных методов. Получены оценки времени выполнения и обменов, общих накладных расходов на параллелизм, ускорения и эффективности параллельного решения.

Ключевые слова: параллельные алгоритмы, задача Коши, экспоненциальные методы, вложенные и экстраполяционные методы.

Введение

Данная статья посвящена построению и анализу эффективности параллельных алгоритмов решения задачи Коши для систем линейных обыкновенных дифференциальных уравнений (СЛОДУ) со встроенными способами оценки локальной апостериорной погрешности. В частности, в ней предложены специальные методы решения однородных и неоднородных СЛОДУ с постоянными коэффициентами. Как показали исследования, учет специфики задачи позволяет строить более эффективные параллельные вычислительные алгоритмы, чем в случае применения стандартных численных схем.

1. Экспоненциальный метод решения линейной задачи Коши

Экспоненциальный метод относится к частным методам численного интегрирования задачи Коши для СЛОДУ, основан на точном представлении решения в аналитической форме и приближенном вычислении матричной экспоненты [1-3].

Общий вид задачи Коши для однородных СЛОДУ с постоянными коэффициентами:

$$\begin{cases} \bar{y}'(x) = A \cdot \bar{y}(x), \\ \bar{y}(x_0) = \bar{y}_0, \quad A = const, \end{cases} \quad (1)$$

где $\bar{y} = (y_1, y_2, \dots, y_m)^T$ – вектор неизвестных;

$\bar{y}_0 = (y_{10}, y_{20}, \dots, y_{m0})^T$ – вектор начальных условий;

$A = \|a_{ij}\|, i, j = \overline{1, m}$ – матрица коэффициентов линейной системы, элементы матрицы являются константами.

Точное решение задачи Коши для СЛОДУ вида (1) на одном шаге интегрирования h с использованием матричной экспоненты определяется следующим образом:

$$\begin{cases} \bar{y}(x_n + h) = e^{hA} \cdot \bar{y}(x_n), \\ e^{hA} = F(hA) = \sum_{i=0}^{\infty} \frac{(hA)^i}{i!}. \end{cases} \quad (2)$$

Приближенное решение можно построить, аппроксимировав матричную экспоненту отрезком её ряда Тейлора при малом h :

$$\begin{cases} \bar{y}_{n+1} = F_r(hA) \cdot \bar{y}_n, \\ F_r(hA) = \sum_{i=0}^r \frac{(hA)^i}{i!}, \end{cases} \quad (3)$$

а затем, используя некоторый алгоритм умножения матриц, вычислить численное решение (1). Полученная вычислительная схема (3) интегрирования однородных СЛОДУ с постоянными коэффициентами соответствует разностному методу порядка $O(h^r)$ [1].

На основе экспоненциального метода построены три различных параллельных алгоритма с учетом альтернативных способов оценки локальной погрешности:

- 1) экспоненциальный и правило Рунге;
- 2) вложенный - экспоненциальный;
- 3) экспоненциальный с локальной экстраполяцией.

Матричная экспонента $F(hA)$ обладает следующим свойством:

$$F(hA) = F\left(\frac{h}{2}A\right) \cdot F\left(\frac{h}{2}A\right),$$

что позволяет достаточно легко встраивать алгоритмы определения локальной погрешности на основе дублирования шага и локальной

экстраполяции в вычислительные схемы явных численных методов для линейных СОДУ. Применение экспоненциального метода для вложенных форм делает практически ненужными вычисления по формуле высшего порядка. Предлагаемые алгоритмы особенно эффективны при решении задач с большой константой Липшица и требуют меньшего объема вычислений, чем стандартные методы решения линейной задачи Коши [2-3].

2. Вложенные экспоненциальные методы решения СЛОДУ

Применение идеи вложенных форм в сочетании с экспоненциальным методом решения предполагает вычисление двух аппроксимаций решения смежных порядков точности в одной и той же точке интегрирования x_{n+1} : \bar{y}_{n+1} порядка

$O(h^r)$ и $\widehat{\bar{y}}_{n+1}$ порядка $O(h^{r+1})$:

$$\begin{cases} \widehat{\bar{y}}_{n+1}(x_n+h; \bar{y}_n) = F_{r+1}(hA) \cdot \bar{y}_n, \\ \bar{y}_{n+1}(x_n+h; \bar{y}_n) = F_r(hA) \cdot \bar{y}_n, \\ d_n = \|\widehat{\bar{y}}_{n+1} - \bar{y}_{n+1}\|. \end{cases}$$

Как правило, формула более высокого, $(r+1)$ -го, порядка используется только для оценки локальной погрешности, а в качестве аппроксимации решения на n -том шаге интегрирования берется формула r -го порядка. Поэтому вычислительная схема вложенного-экспоненциального метода после элементарных преобразований принимает вид:

$$\begin{cases} \bar{y}_{n+1} = \left(E + hA + \frac{h^2 A^2}{2!} + \dots + \frac{h^r A^r}{r!} \right) \cdot \bar{y}_n; \\ d_n = \left\| \frac{h^{r+1}}{(r+1)!} A^{r+1} \cdot \bar{y}_n \right\|. \end{cases} \quad (4)$$

Такой подход позволяет уменьшить время каждого шага интегрирования за счет создания подготовительного этапа, который будет выполняться до начала основного счета и содержать операции, не связанные непосредственно с номером шага. Для вложенных методов величина шага интегрирования h – не есть постоянная величина. Следовательно, основными операциями подготовительного этапа будут вычисления степеней матрицы коэффициентов однородной системы A и умножение матриц на константы. Заметим, что эти операции являлись наиболее трудоемкими в схеме (4) [4-5].

Время вычисления решения в точке x_{n+1} по преобразованной схеме вложенного-

экспоненциального метода для однопроцессорной ВС можно представить, как сумму двух слагаемых: времени выполнения подготовительного этапа и, собственно, шага интегрирования.

Время подготовительного этапа вычисляется, как:

$$T_{I,0}^1 = (2m^3 + m^2)r \cdot t_{op},$$

где t_{op} – время выполнения операции с плавающей точкой.

Время последовательного выполнения одного шага интегрирования по схеме (4) определяется с учетом того факта, что нет необходимости в вычислении обоих приближений, достаточно знать \bar{y}^{n+1} и d_n :

$$T_I^1 = (2m^2 r + 6m^2) \cdot t_{op}.$$

Параллельная реализация вложенного экспоненциального метода включает подготовительный этап вычисления степеней матрицы коэффициентов и, собственно, N шагов интегрирования и концентрируется на выполнении одного шага интегрирования, состоящего из вычисления матричной формы, аппроксимации решения r -го порядка и величины d_n . До начала интегрирования (на нулевом шаге) ровно один раз каждый процессор p_{ij} вычисляет соответствующий блок $\langle i, j \rangle$ степеней $i = \overline{2, r+1}$ матрицы коэффициентов СЛОДУ на основе параллельного блочного систолического алгоритма умножения квадратных матриц и умножение их на константы:

$$\left(\begin{array}{cccc} \frac{A_{11}^2}{2!}, \dots, \frac{A_{11}^{r+1}}{(r+1)!} & \dots & \frac{A_{1q}^2}{2!}, \dots, \frac{A_{1q}^{r+1}}{(r+1)!} \\ \dots & \dots & \dots & \dots \\ \frac{A_{q1}^2}{2!}, \dots, \frac{A_{q1}^{r+1}}{(r+1)!} & \dots & \frac{A_{qq}^2}{2!}, \dots, \frac{A_{qq}^{r+1}}{(r+1)!} \end{array} \right).$$

На произвольном шаге интегрирования каждый процессор с номером $\langle i, j \rangle$ вычисляет соответствующий блок формы F_r , а каждый процессор строки $\langle i \rangle$ вычисляет подвектора решения: $Y_{n+1} = (Y_{n+1,1}, Y_{n+1,2}, \dots, Y_{n+1,q})$ и величины d_n : $D_n = (D_{n,1}, \dots, D_{n,q})$.

Время вычислений и коммуникационная составляющая подготовительного этапа рассмотренного алгоритма равны:

$$T_{p,comp,0}^1 = (2m^3 / p^2) \cdot r \cdot t_{op},$$

$$T_{p,comm,0}^1 = 2(p-1)r \cdot \left(t_s + \frac{m^2}{p^2} t_w \right),$$

Где t_s – латентность, а t_w – время передачи одного слова.

Время параллельной реализации одного шага интегрирования для вложенного экспоненциального метода определяется, как:

$$T_{p,comp}^1 = \left[(2r+6) \cdot \frac{m^2}{p^2} + \lceil \log_2 p \rceil \cdot \frac{2m}{p} \right] \cdot t_{op},$$

а время обменов составляет:

$$T_{p,comm}^1 = 2 \cdot \left(2^{\lceil \log_2 p \rceil} + p - 2 \right) \cdot \left(t_s + \frac{m}{p} t_w \right).$$

Для сравнения приведем второй алгоритм решения СЛОДУ с постоянными коэффициентами по методу вложенных форм, основанный на использовании стандартной явной одношаговой схемы:

$$\begin{cases} \bar{y}_{n+1} = \bar{y}_n + h \cdot \sum_{i=1}^s b_i \cdot \bar{k}_i, \\ \hat{y}_{n+1} = \bar{y}_n + h \cdot \sum_{i=1}^{s'} \hat{b}_i \cdot \bar{k}_i, \\ \bar{k}_l = A \cdot \bar{y}_n + h \cdot \bar{g}_l, \\ \bar{g}_l = A \cdot \sum_{i=1}^{l-1} c_{li} \cdot \bar{k}_i, l = \overline{1, s'}, \\ d_n = \|\hat{y}_{n+1} - \bar{y}_{n+1}\|. \end{cases} \quad (5)$$

Последовательный алгоритм явного вложенного метода Рунге-Кутты $r(\hat{r})$ в применении к линейной задаче Коши по схеме (5) включает: вычисление коэффициентов \bar{k}_i ; определение двух аппроксимаций решения \bar{y}_{n+1} и \hat{y}_{n+1} ; вычисление величины d_n для определения локальной погрешности.

Для правомерности сравнения вложенных методов решения СЛОДУ одного и того порядка точности $O(h^r)$ необходимо учесть два момента. Как правило, число стадий для стандартных алгоритмов вложенных пар удовлетворяет соотношению: $s' = s + 1$. Более того, число стадий превышает порядок метода, как минимум, на единицу. Так, например, для широко используемого вложенного метода Фельберга 4(5) число стадий равно 6, а уже для метода Дормана-Принса 8(7) $s = 13$. Поэтому далее, число стадий метода будет вычисляться по соотношению: $s := s + c$, где величина c зависит от порядка используемого метода [1-2].

Тогда, время выполнения алгоритма по схеме (5) равно:

$$T_1^2 = [(2s+2)m^2 + (s^2 + 3s + 7)m] \cdot t_{op}.$$

Для параллельного алгоритма стандартной вычислительной схемы явного вложенного метода время арифметических операций и операций обмена для алгоритма вложенного ЯМПК:

$$T_{p,comp}^2 = \left[(2s+2) \frac{m^2}{p^2} + (s^2 + 4s + \lceil \log_2 p \rceil) \frac{m}{p} \right] \cdot t_{op},$$

$$T_{p,comm}^2 = (s+1) \cdot \left(2^{\lceil \log_2 p \rceil} + p - 2 \right) \cdot \left(t_s + \frac{m}{p} t_w \right).$$

Сравнение характеристик двух приведенных вычислительных схем решения линейной задачи Коши с оценкой локальной погрешности на основе идеи вложенных форм позволяет сделать выводы о преимуществах предложенного экспоненциального метода (рис. 1-2). Во-первых, доля обменных операций для традиционной схемы превышает долю обменов для схемы с экспонентой.



Рисунок 1 – Графики зависимости доли обменных операций параллельных вложенных методов решения СЛОДУ

Во-вторых, на основе стандартной и экспоненциальной схем время реализации параллельного алгоритма для экспоненциальной схемы меньше, чем для стандартной и традиционно возрастает с ростом размерности задачи: $m \uparrow \Rightarrow (T_p^1 \uparrow) < (T_p^2 \uparrow)$ и уменьшается с ростом числа процессоров: $p \uparrow \Rightarrow (T_p^2 \downarrow) < (T_p^1 \downarrow)$.

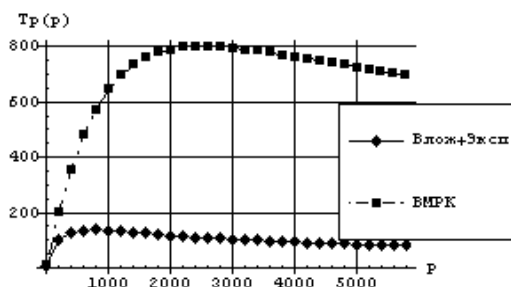


Рисунок 2 – Графики зависимости времени реализации параллельных вложенных методов на основе стандартной и экспоненциальной схем от размерности задачи

Аналогичные результаты можно получить при оценке асимптотической сложности обеих вычислительных схем на основе вложенных форм, ограничимся учетом наиболее ресурсоемкой операции: умножения матрицы на вектор. Тогда, имеем следующее соотношение:

$$T_p^2 / T_p^1 \approx \frac{s+c}{2}, \quad c = 1, 2, \dots,$$

то есть стандартная схема решения обладает большей вычислительной сложностью, чем экспоненциальная.

Таким образом, вложенный метод на основе экспоненты требует меньших накладных расходов по сравнению с традиционной схемой для любых размерностей задачи, числа процессоров, вычислительных схем различных порядков.

3. Параллельные алгоритмы технологии локальной экстраполяции для решения слоду

Ускорение вычислений по технологии локальной экстраполяции при решении линейной однородной задачи Коши с постоянными коэффициентами также может быть получено на основе экспоненциального метода. По результатам предыдущих исследований [4-6] минимизировать вычислительные затраты можно реализацией h^2 -экстраполяции для симметричных опорных методов малого порядка точности.

Использование в качестве формы F частичной суммы из первых r_0 членов разложения матричной экспоненты в ряд Тейлора соответствует применению определенного метода Рунге-Кутты r_0 -го порядка точности:

$$\bar{y}_{n+1} = \left(E + hA + \dots + \frac{h^{r_0} A^{r_0}}{r_0!} \right) \cdot \bar{y}_n = F_{r_0}(hA) \cdot \bar{y}_n.$$

Так, например, классический явный метод Рунге-Кутты четвертого порядка точности соответствует использованию матричной формы $F_4(hA)$ [2].

Если h – базовый шаг интегрирования, то $T_{11} = F_{r_0}(hA) \cdot \bar{y}_n$ – есть аппроксимация решения $\bar{y}(x_n + h)$ порядка $O(h^{r_0})$.

При получении решения на шаге $n+1$ заданного порядка точности r , необходимо провести вычисления для шагов $h/n_2, \dots, h/n_k$ и получить соответствующие аппроксимации решения T_{21}, \dots, T_{k1} :

$$T_{21} = F_{r_0}^{n_2} \left(\frac{hA}{n_2} \right) \cdot \bar{y}_n, \dots,$$

$$T_{k1} = F_{r_0}^{n_k} \left(\frac{hA}{n_k} \right) \cdot \bar{y}_n.$$

Затем, используя соотношение для полиномиальной экстраполяции, вычислить приближения: T_{kk} порядка $O(h^r)$ и $T_{k-1,k}$

порядка $O(h^{r-1})$. Для того, чтобы обеспечить необходимый порядок аппроксимации r решения при симметричном методе в качестве P_i может быть выбрана только четная последовательность чисел. Тогда, необходимое количество строк экстраполяционной таблицы составит: $k = r/2$.

Время вычисления по последовательной схеме для симметричного опорного метода порядка r_0 равно:

$$\begin{aligned} T_1^{31} &= T_{T_{11}} + T_{T_{21}} + \dots + T_{T_{k1}} + T_1^{ext-tab}, \\ T_{1,0}^{r_0,exp} &= (r_0 - 1)(2m^3 + m^2) \cdot t_{op}, \\ T_1^{r_0,exp} &= 2m^2(N(k) + kr_0) \cdot t_{op}. \end{aligned}$$

Для числовой последовательности P_2 и симметричного опорного метода второго порядка имеем: $N(k) = k^2 - k + 1$. Тогда, время вычислений по схеме опорного метода, $T_1^{r_0,exp}$ и общее время вычислений по технологии локальной экстраполяции с учетом определения $r/2$ строк экстраполяционной таблицы, T_1^{31} равны:

$$\begin{aligned} T_1^{r_0,exp} &= \frac{m^2}{2}(r^2 + 6r + 4) \cdot t_{op}, \\ T_1^{31} &= \left[\frac{m^2}{2}(r^2 + 6r + 4) + \frac{5}{8}m(r^2 - 2r) \right] \cdot t_{op}. \end{aligned}$$

При разработке параллельного алгоритма на базе экспоненциального метода с экстраполяцией по Ричардсону необходимо выделить две подзадачи:

а) вычисление k аппроксимаций решения $T_{11}, T_{21}, \dots, T_{k1}$ по опорному методу с шагами h_1, h_2, \dots, h_k ;

б) построение экстраполяционной таблицы по схеме Эйткена-Невилла, вычисление результирующих аппроксимаций решения T_{k-1k}, T_{kk} .

Параллельное выполнение первой макрооперации по аналогии с другими рассмотренными алгоритмами будет содержать подготовительный этап, выполняемый до начала интегрирования и содержащий вычисления степеней матрицы A , умноженных на константы.

Время вычислений и время обменов для подготовительного этапа, соответственно, равны:

$$\begin{cases} T_{p,0,comp}^{r_0,exp} = (r_0 - 1) \left(\frac{2m^3}{p^2} \right) \cdot t_{op}, \\ T_{p,0,comm}^{r_0,exp} = 2(r_0 - 1)(p - 1) \cdot \left(t_s + \frac{m^2}{p^2} t_w \right). \end{cases}$$

Далее k раз вычисляются матричные формы $F_{n_i}^{r_0}$, $i = \overline{1, k}$ и аппроксимации решения, соответствующие первому столбцу экстраполяционной таблицы, затем на основе полиномиальной экстраполяции вычисляются остальные элементы таблицы. Разбиение данных по процессорам соответствует топологии 2D-тор и блочному систолическому алгоритму выполнения матричного умножения, следовательно, время вычислений в t_{op} равно:

$$\begin{aligned} T_{p,comp}^{3l} &= \frac{m^2}{p^2} \left(\frac{r^2 + 6r + 4}{2} \right) \cdot t_{op} + \\ &+ \frac{m}{p} \left(\frac{r^2 - 2r + 4}{4} \lceil \log_2 p \rceil + \frac{5}{8} (r^2 - 2r) \right) \cdot t_{op}. \end{aligned}$$

Коммуникационная составляющая описанного алгоритма определяется обменами при вычислении k аппроксимаций решения на основе опорного метода.

Вычисление элементов экстраполяционной таблицы при данном способе распределения данных не требует обменов и перегруппировки данных для следующего шага интегрирования. Таким образом, имеем:

$$\begin{aligned} T_{p,comm}^{3l} &= \left(\frac{r^2 - 2r + 4}{4} \right) \left(2^{\lceil \log_2 p \rceil} + p - 2 \right) \times \\ &\times \left(t_s + \frac{m}{p} t_w \right). \end{aligned}$$

Разработанный параллельный алгоритм решения линейной задачи Коши с оценкой шаговой погрешности по технологии локальной экстраполяции сравним со стандартной схемой вычислений. Для этого в качестве опорного метода принимается явный одношаговый s_0 -стадийный метод порядка $O(h^{r_0})$:

$$\begin{cases} \bar{y}_{n+1} = \bar{y}_n + h \cdot \sum_{i=1}^{s_0} b_i \cdot \bar{k}_i, \\ \bar{k}_l = A \cdot \left(\bar{y}_n + h \cdot \sum_{i=1}^{l-1} c_{li} \cdot \bar{k}_i \right), l = \overline{1, s_0}. \end{cases}$$

Для сравнения вычислительной сложности двух параллельных алгоритмов решения СЛОДУ на базе локальной экстраполяции ограничимся учетом наиболее ресурсоемкой операции, а именно, умножения матрицы на вектор. Тогда, справедливо следующее соотношение:

$$T_p^{32} / T_p^{31} \approx \frac{s_0 \cdot N(k)}{N(k)} = s_0,$$

то есть стандартная схема решения обладает вычислительной сложностью в s_0 раз большей, чем экспоненциальная.

Таким образом, как и при других способах оценки апостериорной шаговой погрешности, применение экспоненциального метода интегрирования для технологии локальной экстраполяции Ричардсона, сокращает вычислительные и коммуникационные затраты.

Экспоненциальный метод может быть применен для ускорения решения неоднородной системы линейных обыкновенных дифференциальных уравнений с постоянными коэффициентами:

$$\begin{cases} \bar{y}'(x) = A \cdot \bar{y}(x) + b, \\ \bar{y}(x_0) = \bar{y}_0, A, b = const. \end{cases} \quad (6)$$

Численное решение задачи Коши вида (6) на шаге интегрирования можно записать следующим образом:

$$\bar{y}_{n+1} = \bar{y}_n + R_r(hA) \cdot (A\bar{y}_n + b),$$

где $R_r(hA)$ – матричная функция, которая также может быть аппроксимирована отрезком ряда Тейлора:

$$R_r(hA) = \sum_{i=1}^r (A^{i-1} h^i / i!).$$

На основе вычислительной схемы (6) разработаны параллельные алгоритмы решения неоднородных линейных СОДУ со встроенными способами оценки локальной погрешности.

Анализ динамических характеристик полученных алгоритмов позволяет сделать выводы: применение экспоненциального метода для решения неоднородной задачи Коши, как и однородной, позволяет сократить вычислительные и коммуникационные затраты по сравнению со стандартными схемами интегрирования. Более того, этот подход может быть использован и в случае линейных систем общего вида:

$$\bar{y}'(x) = A(x) \cdot \bar{y}(x) + b(x). \quad (7)$$

Так, при заданной гладкости матрицы коэффициентов правой части уравнений (7) возможно на каждом отрезке интегрирования аппроксимировать исходную задачу системой (1) или (6) в зависимости от однородности исходной СОДУ [3-4].

Разработка параллельных алгоритмов со встроенными способами оценки локальной

погрешности, теоретический анализ сложности и проведенный численный эксперимент позволяют сделать вывод, что применение экспоненциального метода для решения неоднородной линейной задачи Коши, как и для однородной, позволяет уменьшить затраты на параллелизм по сравнению со стандартными схемами решения СЛОДУ.

Заключение

Численный эксперимент на базе тестов для СОДУ [7] и проведенный сравнительный анализ динамических характеристик разработанных параллельных алгоритмов показал, что вне зависимости от способа вычисления локальной погрешности, параллельные алгоритмы, основанные на применении экспоненциального метода решения, имеют меньшую вычислительную сложность и лучшие показатели качества параллелизма по сравнению со своими стандартными аналогами. Так, для правила Рунге в s раз, вложенных методов $s/2$ раз, и для локальной экстраполяции в s_0 раз, здесь s – число стадий сравниваемого ЯМРК, s_0 – число стадий опорного метода для явного метода типа Рунге-Кутты.

Исследована эффективность альтернативных способов определения локальной апостериорной погрешности при решении линейной задачи Коши для СОДУ на основе явных схем, выявлены предпочтительные области применения различных алгоритмов вычисления шаговой погрешности. Наименее трудоемкий способ определения локальной апостериорной погрешности решения для управления шагом

интегрирования – это методы вложенных форм. Преимущества локальной экстраполяции Ричардсона проявляются при получении высокоточного решения.

Разработаны вычислительные схемы отображения полученных параллельных алгоритмов на структуры параллельных ВС с распределенной памятью и топологиями решетка/тор, гиперкуб. Для каждого из приведенных алгоритмов рассчитаны трудоемкости получения численного решения, получены характеристики параллелизма, которые достаточно убедительно свидетельствуют о высокой эффективности предложенных методов.

Исследована эффективность полученных вычислительных схем отображения параллельных алгоритмов на структуры ВС в зависимости от размерности процессорных полей, модели вычислений, ориентированной на SIMD или MIMD и кластерные структуры, модели передачи данных в различных топологиях, коммуникационных констант. Апробация параллельных алгоритмов проводилась на моделях ВС SIMD, MIMD, кластерной архитектуры с различными топологиями. В частности, наилучшие характеристики распараллеливания достигаются при использовании топологии гиперкуб.

Повышение эффективности решения линейной задачи Коши для СОДУ также может быть получено за счет применения эффективных алгоритмов выполнения наиболее трудоемких операций экспоненциального метода, в частности операций умножения матриц и умножения матрицы на вектор.

Литература

1. Хайрер Э. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи / Э. Хайрер, С. Нёрсет, Г. Ваннер. - М.: Мир, 1990. -512с.
2. Иванов В.В. Методы вычислений на ЭВМ / В.В. Иванов. – К.: Наукова думка, 1986. – 584с.
3. Арушанян О.Б. Численное решение обыкновенных дифференциальных уравнений на Фортране / О.Б. Арушанян, С.Ф. Залеткин. – М.: МГУ, 1990.–336с.
4. Фельдман Л.П. Параллельные алгоритмы численного решения задачи Коши для систем обыкновенных дифференциальных уравнений / Л.П. Фельдман, И.А. Назарова // Математическое моделирование. – 2006. – Т.18. – № 9. – С. 17-31.
5. Фельдман Л.П. Эффективность параллельных алгоритмов оценки локальной апостериорной погрешности для численного решения задачи Коши / Л.П. Фельдман, И.А. Назарова // Электронное моделирование. – 2007. – Т. 29. – № 3. – С.11-25.
6. Назарова И.А. Эффективность применения технологии локальной экстраполяции в параллельных алгоритмах численного решения задачи Коши / И.А. Назарова // Научно-теоретический журнал ИПИИ НАН Украины «Искусственный интеллект». – 2006. – №3. – С. 192-202.
7. Арушунян О.Б. Тесты для вычислительного практикума по обыкновенным дифференциальным уравнениям / О.Б. Арушунян, С.Ф. Залеткин, Н.Н. Калиткин // Вычислительные методы и программирование. – 2002. – Т. 3. – С. 11-19.

Надійшла до редакції 10.03.2011