

4. Matusevych K. M. Hnoseologiiia intelektualnogo potencialu suspilstva. Ekonomika ta pipryiemnytsva, *Zbirnyk naukovykh pratsmolodykh vchenykh ta aspirantiv*, випуск 22, Kiev, KNEU, 2009, pp. 6–10.
5. Filipenko A.S., Vergun V., Burakivskiy I. Ekonomika zarubizhnykh krain. Kiev, Lybid, 1998, 267 p.
6. Vaniyancz S. V. Informacionny'j resurs v e'konomicheskoy sfere. Moscow, Progress, 2006, pp. 56–59.
7. Gorodov O. A. Informaciya kak ob'ekt grazhdanskikh prav, *Pravovedenie*, №6, 2007, 41 p.
8. Kopy'lov V. A. Informacionnoe pravo. Moscow, Yurist, 1997, 470 p.
9. Slavnova A. O. Informacionnaya e'konomika. Moscow, RRB, 2006, 392 p.
10. Litvak B.G. E'kspertnaya informaciya. Metody' polucheniya i analiza. Moscow, Radio i svyaz', 1982, 184 p.
11. Kharchenko O. G., Yatsyshyn V. V., Bondarchuk I. O. Metody zabezpechennia ta kontroliu yakosti WEB-zastosuvan na stadiiakh zhyttievogo tsyклу. *Vymiriuvalna ta obchysliuvalna tekhnika v obchysliuvalnykh protsesakh*, 2011, № 1, pp. 139–144.
12. Kini R. L., Raifa H. Priniatie reshenij pri mnogih kriteriyah: predpochteniya i zameshheniya. Moskva: Radio i svyaz', 1981, 560 p.
13. Litvak B.G. E'kspertny'e ocenki i prinyatie reshenij. Moskva, Patent, 1996, 560 p.
14. Petrovskij A. L., Pankova L. A., Shnejderman M. V. Organizaciya e'kspertny'h procedur. Moscow, Nauka, 1984, 120 p.
15. Mas-Colell, Andreu, Whinston, Michael, & Green, Jerry. *Microeconomic Theory*. Oxford, Oxford University Press, 1995, 460 p.
16. Varian, Hal R. *Intermediate Microeconomics*. W. W. Norton & Company, 2005, 240 p.

УДК 004.272.2:519.63

Дмитриева О. А.

Канд. техн. наук Донецкого национального технического университета

ГЕНЕРАЦИЯ ОПЕРАТОРОВ ПЕРЕХОДА ДЛЯ ПАРАЛЛЕЛЬНОГО УПРАВЛЕНИЯ ШАГОМ ПРИ МОДЕЛИРОВАНИИ ЛИНЕЙНЫХ ДИНАМИЧЕСКИХ СИСТЕМ

Предложены параллельные алгоритмы моделирования линейных динамических объектов, основанные на построении операторов перехода. Разработанные алгоритмы базируются на вложенных стадийных схемах и позволяют осуществлять автоматический выбор оптимального размера шага в каждой точке сетки. Параллельная реализация ориентирована на кластерные вычислительные системы типа MIMD.

Ключевые слова: параллельное моделирование, стадийный метод, оператор перехода, адаптация шага, кластер.

ВВЕДЕНИЕ

Наибольшая необходимость в эффективных параллельных вычислениях возникает при моделировании динамических процессов, описываемых системами обыкновенных дифференциальных уравнений (ОДУ) большой размерности [1]. Такие задачи возникают как непосредственно в процессе математического моделирования [2], так и при генерации вторичных систем уравнений, например, при решении задач математической физики методом прямых [3]. При этом можно говорить о некоторых особенностях сгенерированных систем. Очень часто они носят линейный характер. Кроме того, матрица коэффициентов, как правило, является разреженной. Системы такого вида возникают при дискретизации уравнений в частных производных, при сведении однородного дифференциального уравнения любого порядка с постоянными коэффициентами к системе уравнений 1-го порядка и т. п.

Учет таких особенностей моделируемого объекта позволяет разрабатывать алгоритмы, которые изначально ориентированы на параллельную архитектуру и, кроме того, дают возможность автоматически генерировать шаг

интегрирования, который обеспечивает требуемую точность. Практически все известные в настоящее время численные методы с автоматическим выбором шага интегрирования основаны на вычислении главного члена локальной ошибки и последующем выборе такого размера для очередного шага, который является максимальным для заданного предела локальной точности [4–5]. Но при этом возникает необходимость повторных вычислений в каждой точке, что приводит к значительному увеличению вычислительных затрат. Кроме того, жестко регламентируются пропорции сокращения шага [6–7].

Основная идея, на которой базируется конструирование операторов перехода для решения линейных систем ОДУ на параллельных компьютерах, заключается в одновременном получении двух приближений с разными порядками. Алгоритмы управления шагом, предлагаемые в работе, базируются на использовании вложенных стадийных методов. Параллельный счет осуществляется в пределах каждого шага с числом стадий s и $s+1$. Две нити вычислений проходят независимо и необходимость в обменах возникает только после получения конечных результатов для расчетных точек.

1. ГЕНЕРАЦИЯ ОПЕРАТОРОВ ПЕРЕХОДА ДЛЯ ПАРАЛЛЕЛЬНОГО МОДЕЛИРОВАНИЯ ОДНОРОДНОЙ ЗАДАЧИ

Пусть математическую модель динамической системы можно представить в виде системы с постоянными коэффициентами и начальными условиями

$$x' = Dx + f(t), \quad x(t_0) = (x_1(t_0), x_2(t_0), \dots, x_N(t_0))^T, \quad (1)$$

где D – матрица размерностью $N \times N$ с постоянными коэффициентами.

Здесь вычисление значения вектора неизвестных на очередном шаге требует предварительного определения значений x_n . Управление шагом в работе осуществляется за счет использования вложенных стадийных методов порядка p , которые кроме численного приближения значений вектора неизвестных x_{n+1} на $n+1$ шаге содержат некоторые выражения \tilde{x}_{n+1} более высокого порядка, как правило, $p+1$.

В частности, если система (1) является однородной, т.е. $f_i(t) = 0, i = 1, 2, \dots, N$, тогда, в зависимости от выбранного метода интегрирования

$$\frac{c}{b^i} \Big| \frac{A}{b^i}, \quad (2)$$

можно прийти к соотношениям

$$x_{n+1} = x_n + \tau_n \sum_{i=1}^s b_i k_i,$$

$$k_i = D \left(x_n + \tau_n \sum_{j=1}^{i-1} a_{ij} k_j \right), \quad i = 1, 2, \dots, s, \quad (3)$$

которые позволяют получить результат с точностью порядка p . Поправочный или вложенный вектор неизвестных

$$\tilde{x}_{n+1} = x_n + \tau_n \sum_{i=1}^s \tilde{b}_i k_i \quad (4)$$

оценивает результат с точностью $p+1$. Иногда могут быть сгенерированы методы, поправочный вектор в которых оценивается с точностью $p-1$. Особый вид векторов $k_i, i = 1, 2, \dots, s$ в (3), который обусловлен линейностью правых частей в (1) позволяет на каждом шаге искать решения в виде

$$x_{n+1} = Q_n x_n, \quad (5)$$

$$\tilde{x}_{n+1} = \tilde{Q}_n x_n, \quad (6)$$

где Q_n, \tilde{Q}_n – операторы (матрицы) переходов.

Добавленный в расчетную схему оператор более высокого порядка \tilde{Q}_n может служить для управления погрешностью и длиной шага. Поскольку в качестве исходных методов для построения оператора перехода Q_n

выбираются известные последовательные стадийные методы, то генерация оператора сводится к сравнению рядов Тейлора для точного и численного решений. Для наглядного представления получаемых при разложении в ряд элементарных дифференциалов используются древовидные графовые структуры, при этом существует взаимно однозначное соответствие между множеством элементарных дифференциалов и множеством деревьев. Для генерации коэффициентов вложенных векторов стадийных методов в работе использовался подход, приведенный в [8] и основанный на формировании набора абстрактных деревьев требуемого порядка. Построение модельных уравнений опирается на следующий факт [8]: для любого дерева можно построить такую систему уравнений, что разложение в ряд Тейлора одной из переменных содержит только один элементарный дифференциал, соответствующий данному дереву.

Что касается оператора перехода \tilde{Q}_n , то для его построения необходимо дополнительно определить коэффициенты $\tilde{b}_i, i = 1, 2, \dots, s$ для (4). Определение коэффициентов метода порядка $p+1$, требует формирования всех абстрактных деревьев, порядок которых укладывается в интервал от 1 до $p+1$. Для построения каждого из таких деревьев tr необходимо и достаточно выполнения равенств

$$\sum_j^i \tilde{b}_j \Psi_j(tr) = \frac{1}{\gamma(tr)}, \quad i = 1, 2, \dots, q, \quad (7)$$

где $q=p+1$ или $q=p-1$, $\Psi_j(tr)$ – некоторая комбинация коэффициентов дерева tr , $\gamma(tr)$ – коэффициент, определяемый как произведение весовых коэффициентов вершин дерева.

Поскольку количество соотношений типа (7) напрямую зависит от порядка точности p и определяется комбинаторными зависимостями, для их генерации и построения операторов перехода привлекался программный интерфейс *Mathematica Wolfram Research*. Формировались операторы перехода как на основе известных стадийных методов, так и для методов, которые были предложены автором в работах [9–12]. Так, например, для пятистадийного метода Зонневельда оператор перехода может быть представлен как

$$Q_n = \left(E + D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{3} D\tau_n \left(E + \frac{1}{4} D\tau_n \right) \right) \right) \right).$$

Вложенный оператор для управления шагом

$$\tilde{Q}_n = \left(E + D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{3} D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{2} D\tau_n \right) \right) \right) \right) \right).$$

Оператор перехода метода Фельберга 4-го порядка

$$Q_n = \left(E + D\tau_n \left(E + \frac{257}{114} D\tau_n \left(E + \frac{2303}{100233} D\tau_n \left(E - \frac{593}{4606} D\tau_n \right) \right) \right) \right).$$

Вложенный оператор метода Фельберга для управления шагом 5-го порядка

$$\tilde{Q}_n = \left(E + D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{3} D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{2} D\tau_n \right) \right) \right) \right) \right)$$

Оператор перехода метода Фельберга 4-го порядка

$$Q_n = \left(E + D\tau_n \left(E + \frac{257}{114} D\tau_n \left(E + \frac{2303}{100233} D\tau_n \left(E - \frac{593}{4606} D\tau_n \right) \right) \right) \right)$$

Вложенный оператор метода Фельберга для управления шагом 5-го порядка

$$\begin{aligned} \tilde{Q}_n = & \left(E + D\tau_n \left(E + \frac{2707}{1254} D\tau_n \left(E + \frac{10283}{40605} D\tau_n \times \right. \right. \right. \\ & \left. \left. \left. \times \left(E - \frac{47077}{534716} D\tau_n \left(E + \frac{6523}{94154} D\tau_n \right) \right) \right) \right) \right). \end{aligned}$$

Полученные операторы перехода, которые необходимо определить один раз до начала вычислений, позволяют находить значения вектора неизвестных параллельно. Такая реструктуризация известных стадийных методов позволяет преобразовать последовательность вычислений таким образом, что доминирующей операцией становится операция *gapsy* (*general Ax plus y*). Приведение вычислений к последовательности *gapsy* обладает значительными преимуществами при параллельной реализации, а алгоритмы выполнения таких операций подробно изложены в [13].

2. ГЕНЕРАЦИЯ ОПЕРАТОРОВ ПЕРЕХОДА ДЛЯ ПАРАЛЛЕЛЬНОГО МОДЕЛИРОВАНИЯ НЕОДНОРОДНОЙ ЗАДАЧИ

Особый вид системы (1) позволяет построить некоторые операторы перехода и для неоднородного случая. Но при этом речь будет идти о линейной комбинации операторов перехода для получения решения основным и вложенным методами. Для получения одного решения на шаге необходимо построить по одному оператору для основного и вложенного методов преобразования линейной части Dx . Количество операторов для преобразования нелинейной части $f(t)$ будет зависеть от числа стадий метода. Т. е. для неоднородного случая можно получить зависимости вида

$$\begin{aligned} x_{n+1} = & Q_n x_n + G_{n,1} f(t_n) + G_{n,2} f(t_n + c_1 \tau_n) + \\ & + \dots + G_{n,s} f(t_n + c_{s-1} \tau_n), \end{aligned} \quad (8)$$

$$\begin{aligned} \tilde{x}_{n+1} = & \tilde{Q}_n x_n + \tilde{G}_{n,1} f(t_n) + \tilde{G}_{n,2} f(t_n + c_1 \tau_n) + \\ & + \dots + \tilde{G}_{n,s+1} f(t_n + c_s \tau_n). \end{aligned} \quad (9)$$

Добавленные в расчетную схему операторы более высокого порядка $\tilde{Q}_n, \tilde{G}_{n,1}, \tilde{G}_{n,2}, \dots, \tilde{G}_{n,s+1}$ так же, как и в разд. 1, будут служить для управления погрешностью и длиной шага. В качестве примера сгенерируем по методу Мерсона 4(5) [8] операторы перехода типа (8)–(9). Введем определения стадийных векторов с учетом неоднородности

$$k_1 = \tau_n * (Dx_n + f(t)),$$

$$k_2 = \tau_n * (D(x_n + 1/3k_1) + f(t + c_1 \tau_n)),$$

$$k_3 = \tau_n * (D(x_n + 1/6k_1 + 1/6k_2) + f(t + c_2 \tau_n)),$$

$$k_4 = \tau_n * (D(x_n + 1/8k_1 + 3/8k_3) + f(t + c_3 \tau_n)),$$

$$k_5 = \tau_n * (D(x_n + 1/2k_1 - 3/2k_3 + 2k_4) + f(t + c_4 \tau_n))$$

и вектора сборки решений основным и вложенным методами

$$\Delta x_n = 1/2k_1 - 3/2k_3 + 2k_4,$$

$$\Delta \tilde{x}_n = 1/6k_1 + 2/3k_4 + 1/6k_5.$$

После построения и сравнения рядов Тейлора для точного и численного решений, получим следующий вид операторов перехода определения x_{n+1} для неоднородной задачи

$$Q_n = D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{3} D\tau_n \left(E + \frac{1}{4} D\tau_n \right) \right) \right),$$

$$G_{n,1} = \frac{\tau_n}{2} \left(E + \frac{1}{12} D^2 \tau_n^2 (E + D\tau_n) \right),$$

$$G_{n,2} = -\frac{1}{4} \tau_n \left(E - \frac{1}{2} D\tau_n \right),$$

$$G_{n,3} = -\frac{3}{2} \tau_n \left(E - \frac{1}{2} D\tau_n \right),$$

$$G_{n,4} = 2\tau_n.$$

Для оценки \tilde{x}_{n+1} операторы перехода будут иметь вид

$$\tilde{Q}_n = D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{3} D\tau_n \left(E + \frac{1}{4} D\tau_n \left(E + \frac{1}{6} D\tau_n \right) \right) \right) \right),$$

$$\tilde{G}_{n,1} = \frac{\tau_n}{6} \left(E + D\tau_n \left(E + \frac{1}{4} D\tau_n \left(E + \frac{1}{2} D\tau_n \left(E + \frac{1}{3} D\tau_n \right) \right) \right) \right),$$

$$\tilde{G}_{n,2} = \frac{1}{48} D^3 \tau_n^4, \quad \tilde{G}_{n,3} = \frac{1}{8} D^2 \tau_n^3,$$

$$\tilde{G}_{n,4} = \frac{2}{3} \tau_n \left(E + \frac{1}{2} D \tau_n \right), \quad \tilde{G}_{n,5} = \frac{1}{6} \tau_n.$$

Здесь вычисление каждого из полученных операторов перехода на шаге может осуществляться автономно, что при реализации в параллельных вычислительных системах позволяет избавиться от обменов на стадиях. Кроме того, вычисление, по сути, сводится к выполнению операций матрично-векторного умножения, для которых разработаны эффективные алгоритмы параллельной реализации [13].

3. УПРАВЛЕНИЕ ШАГОМ ПРИ РЕШЕНИИ ЛИНЕЙНЫХ ЗАДАЧ, ОСНОВАННОЕ НА ОПЕРАТОРАХ ПЕРЕХОДА

Рассмотрим решение однородной задачи (1), реализованное с помощью операторов переходов (5) и (6). Выделим две линейки вычислительных узлов, на которых запустим параллельные процессы. Размерность процессорного поля в идеале должна совпадать с размерностью

системы, хотя это требование не является обязательным. Если размерность системы N очень велика (или просто превышает количество доступных вычислительных узлов r), то за каждым процессорным элементом необходимо закрепить группу уравнений системы так, чтобы обеспечить равномерную загрузку. На первой линейке будет осуществляться формирование решения x_{n+1} , на второй линейке будет формироваться вложенный вектор \tilde{x}_{n+1} (рис. 1). Поскольку получение решений x_{n+1} с порядком p и \tilde{x}_{n+1} с порядком $p+1$ ($p-1$) основано на матрично-векторном умножении, проводится мелкозернистое распараллеливание, основанное на выполнении операций *gapsu* по соответствующим операторам переходов.

После выполнения $s-1$ однотипных матричных операций умножения и суммирования на первой решетке и s на второй решетке, формируются решения x_{n+1} и \tilde{x}_{n+1} . Специальный вид операторов перехода позволит обходиться без обменов на каждом шаге вычислений, если каждый вычислительный узел будет иметь доступ к значениям искомого вектора, полученным на предыдущем шаге, и к соответствующим коэффициентам исходной матрицы D . Схема алгоритма, реализующего параллельное решение неоднородной системы, основанное

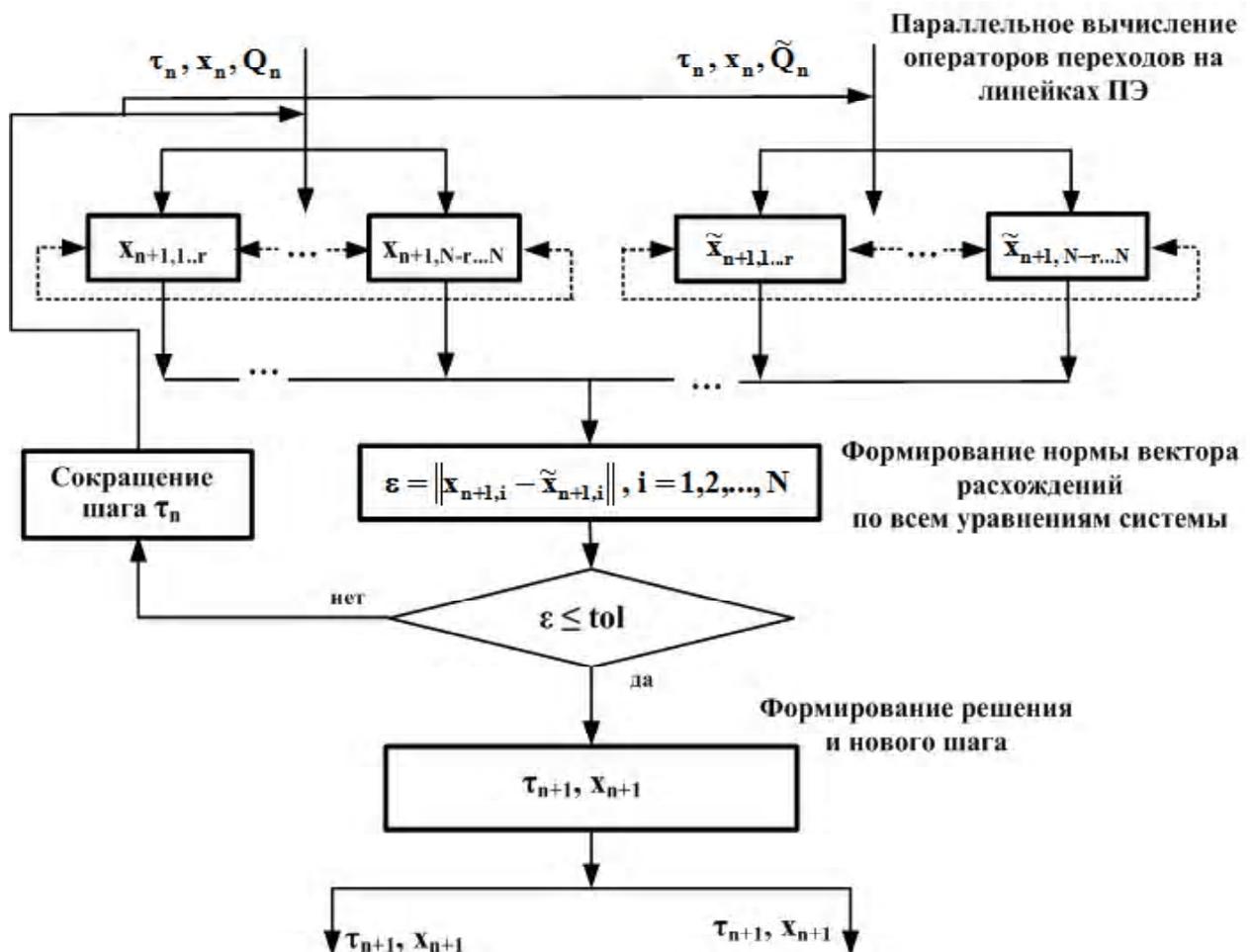


Рис. 1. Алгоритм управления шагом интегрирования при использовании операторов переходов для однородной задачи

на использовании операторов перехода (8) и (9), приведена на рис. 2. Для основного и вложенного методов формируются результаты по произведениям $Q_n x_n$ и $\tilde{Q}_n x_n$ для линейной части (1) и в соответствующих стадийных точках для неоднородных значений $f(t)$.

После формирования решений x_{n+1} и \tilde{x}_{n+1} погрешность на каждом шаге определяется на основе соотношения

$$\varepsilon = \frac{1}{2^p - 1} \max \frac{|x_{n+1} - \tilde{x}_{n+1}|}{\tilde{x}_{n+1}}. \quad (10)$$

Если полученная погрешность на шаге не превышает заданную точность tol , рассчитывается новая длина шага

$$\tau_{n+1} = \tau_n \min \left(fax \max, \max \left(fax \min, \omega \left(\frac{tol}{\varepsilon} \right)^{\frac{1}{p+1}} \right) \right). \quad (11)$$

Параметры $fax \min, fax \max, \omega$ выбираются, исходя из соображений, приведенных в [10]. В противном случае от шага придется отказаться и повторить просчет с шагом

$$\tau_n = \tau_n fax \min.$$

В случае, если используется решетка процессорных элементов (ПЭ), позиция каждого элемента в строке определяет порядковый номер уравнения в системе, для которого ведется расчет, а номер элемента в столбце – номер рассчитываемого элемента вектора неизвестных.

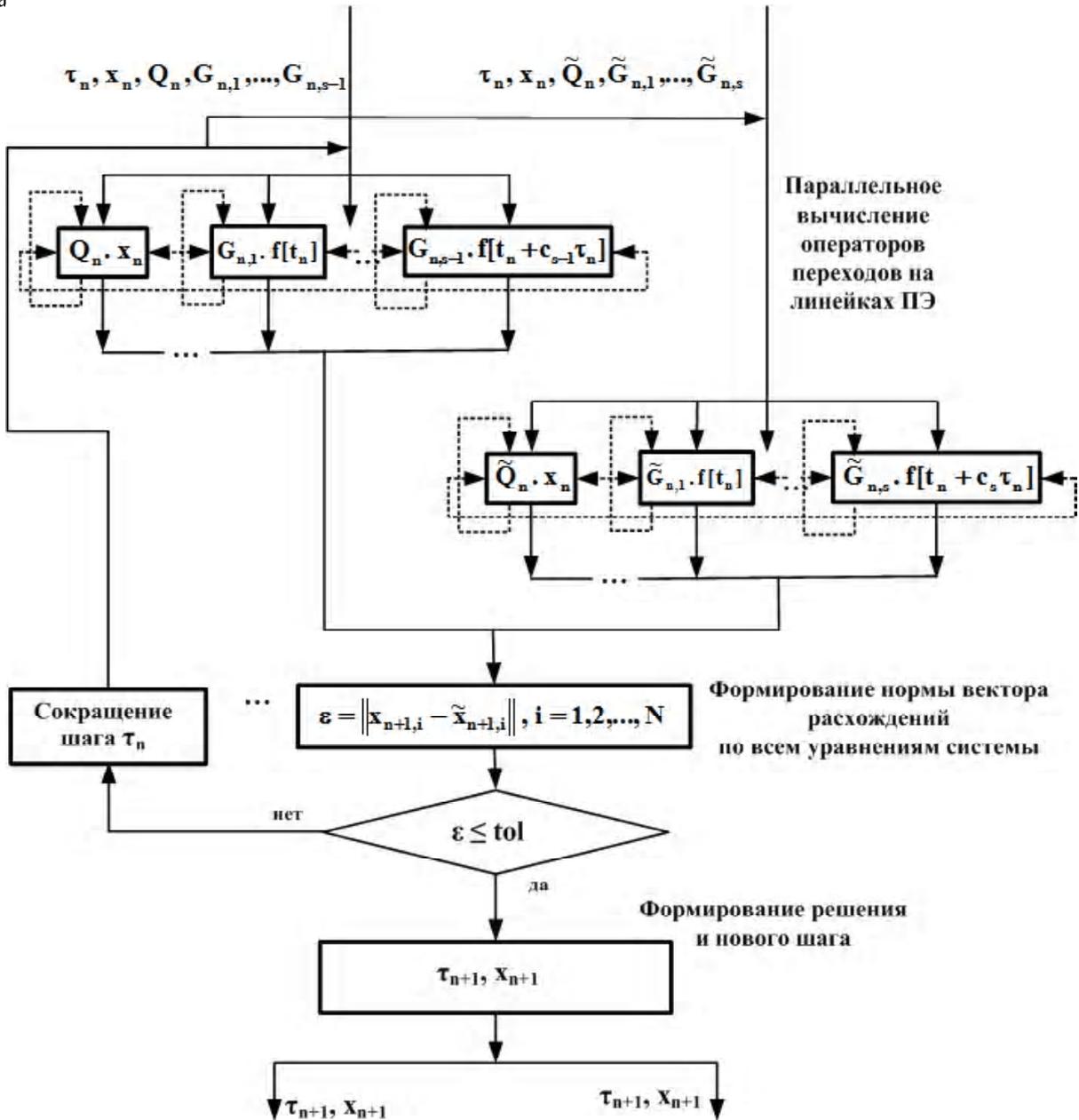


Рис. 2. Алгоритм управления шагом интегрирования для неоднородной системы с операторами перехода

4. ТЕСТОВАЯ РЕАЛИЗАЦИЯ АЛГОРИТМОВ УПРАВЛЕНИЯ ШАГОМ, ОСНОВАННЫХ НА ОПЕРАТОРАХ ПЕРЕХОДА

В качестве тестовой задачи выберем систему однородных линейных ОДУ, возникающих при дискретизации уравнения диффузии методом прямых. Количество уравнений в системе N может быть сколь угодно велико и определяется размерностью пространственной сетки, которая вводится для решения исходного уравнения в частных производных. Решение для системы (1) [14] с правыми частями вида

$$f(t, x) = Dx, \tag{12}$$

где матрица D размерностью $N \times N$ имеет вид

$$D = \begin{bmatrix} -2 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 \end{bmatrix},$$

необходимо найти на интервале $[0, 3]$.

Решение системы (1) с правыми частями вида (10) представляет из себя нежесткий случай. На рис. 3–4 приведены получаемые погрешности и динамика изменения шага

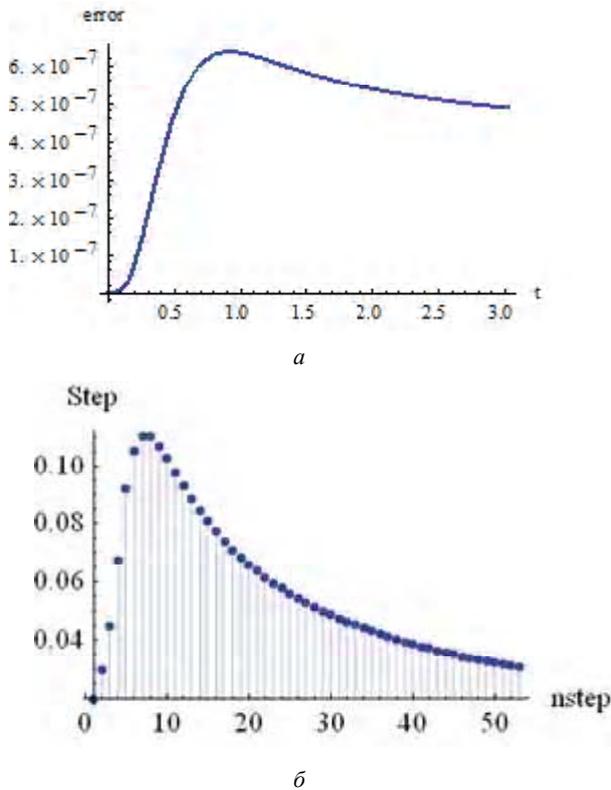


Рис. 3. Погрешность (а) и динамика шага интегрирования (б) для задачи (12) с операторами перехода метода Фельберга 7(8) с точностью 10^{-6}

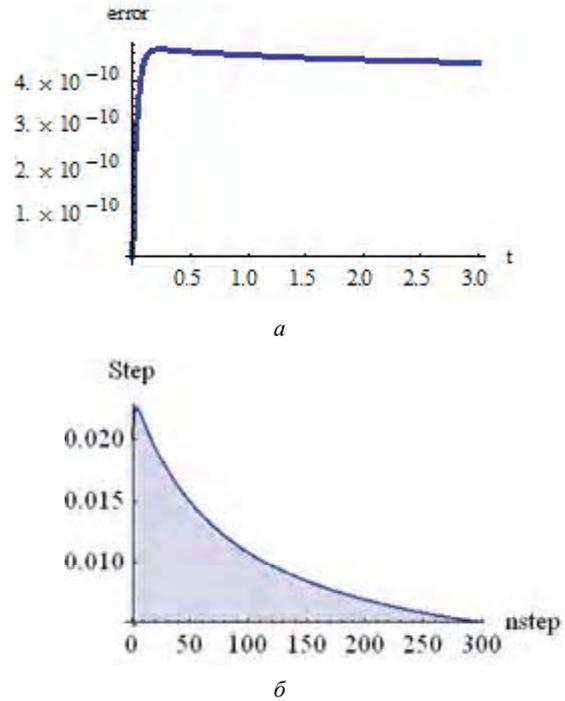


Рис. 4. Погрешность (а) и динамика шага интегрирования (б) для задачи (12) с операторами перехода метода Фельберга 7(8) с точностью 10^{-9}

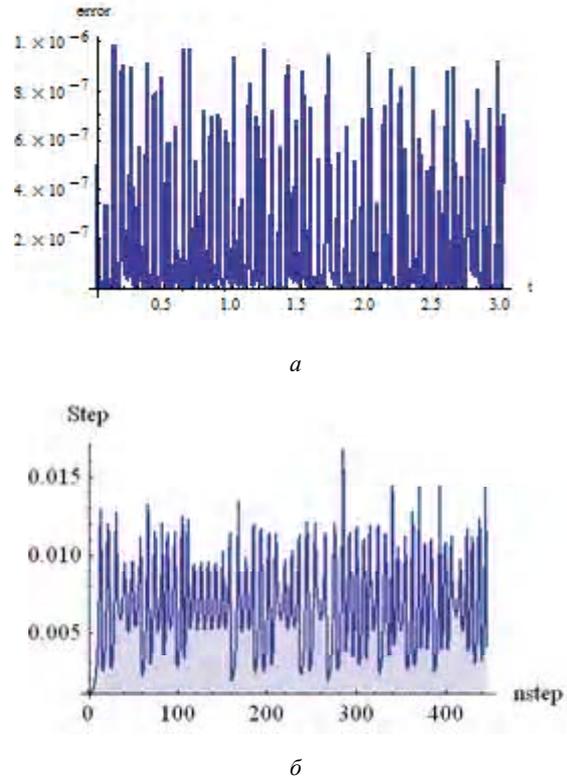


Рис. 5. Погрешность (а) и динамика шага интегрирования (б) для задачи (13) с операторами перехода метода Фельберга 7(8) с точностью 10^{-6}

интегрирования для различных операторов переходов и заданной точности интегрирования, полученные для системы (1) с правыми частями вида (12) и с $N=10000$.

Если видоизменить правую часть в (12), записав ее в виде

$$f(t, x) = (N+1)^2 \begin{bmatrix} -2 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 \end{bmatrix} x, \quad (13)$$

получим жесткий вариант линейной однородной системы с вариантами реализаций, основанных на приведенных операторах переходов. На рис. 5–6 приведены получаемые погрешности и динамика изменения шага интегрирования для различных операторов переходов и заданной точности интегрирования, полученные для системы (1) с правыми частями вида (13) и с $N=10000$.

В качестве тестовой жесткой задачи выберем систему неоднородных линейных ОДУ, возникающих при дискретизации уравнения диффузии методом прямых. Количество уравнений в системе N может быть сколь угодно велико и определяется размерностью пространственной сетки, которая вводится для решения исходного уравне-

ния в частных производных. Система (1) [14] с линейными неоднородными правыми частями вида

$$f(t, x) = (N+1)^2 \begin{bmatrix} -2 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & -2 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & -2 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & -2 & 1 \\ 0 & 0 & 0 & 0 & \dots & 1 & -2 \end{bmatrix} x + (N+1)^2 \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ \varphi(t) \end{bmatrix}, \quad (14)$$

где

$$\varphi(t) = \alpha \sin(\sqrt{2}) \exp(-9t) - \sin(1) \exp(-\mu t),$$

$$\alpha = \cos(\sqrt{2}) / \sqrt{2} \cos\left(\frac{1}{\sqrt{2}}\right),$$

$$9 = 2(N+1)^2 \left(1 - \cos\left(\frac{\sqrt{2}}{N+1}\right)\right),$$

$$\mu = 2(N+1)^2 \left(1 - \cos\left(\frac{1}{N+1}\right)\right),$$

точные решения которой известны и равны

$$x_i(t) = \alpha \sin\left(\frac{\sqrt{2}}{N+1}\right) \exp(-9t) - \sin\left(\frac{1}{N+1}\right) \exp(-\mu t), \quad i = 1, 2, \dots, N.$$

На графиках приведены результаты решения, полученные для системы (14) с $N=10000$.

ВЫВОДЫ

Для моделирования линейных динамических объектов в работе предложены подходы, позволяющие реструктуризировать стадийные вложенные методы и перейти от последовательной реализации к параллельному выполнению вычислений с контролем погрешности на шаге. Такая реструктуризация стадийного метода осуществляется за счет построения оператора перехода, сводящего процедуру получения решения на шаге для однородных систем к выполнению параллельных матрично-векторных операций.

Для линейных неоднородных систем требуется построение нескольких операторов перехода, количество которых напрямую зависит от числа стадий исходного метода. Операторы перехода для каждого s -стадийного ме-

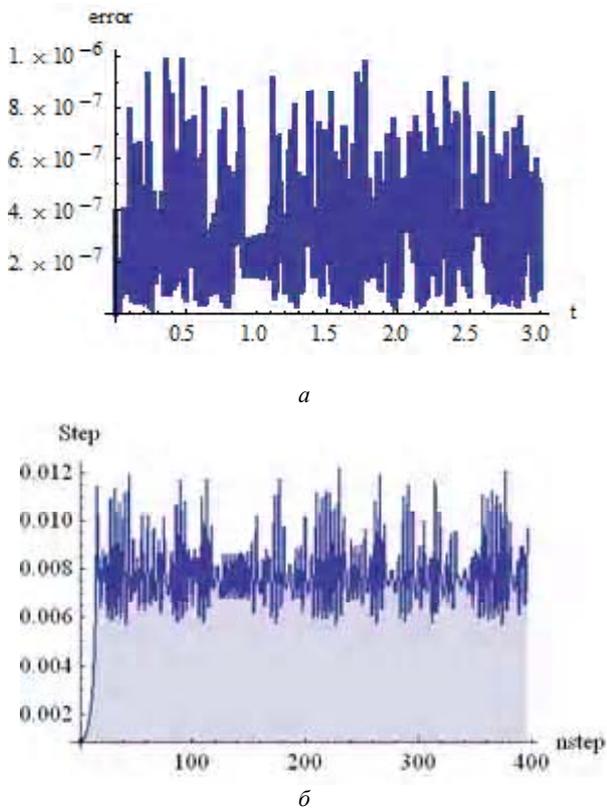


Рис. 6. Погрешность (а) и динамика шага интегрирования (б) для задачи (13) с операторами перехода метода Дормана-Принса 5(4) с точностью 10^{-6}

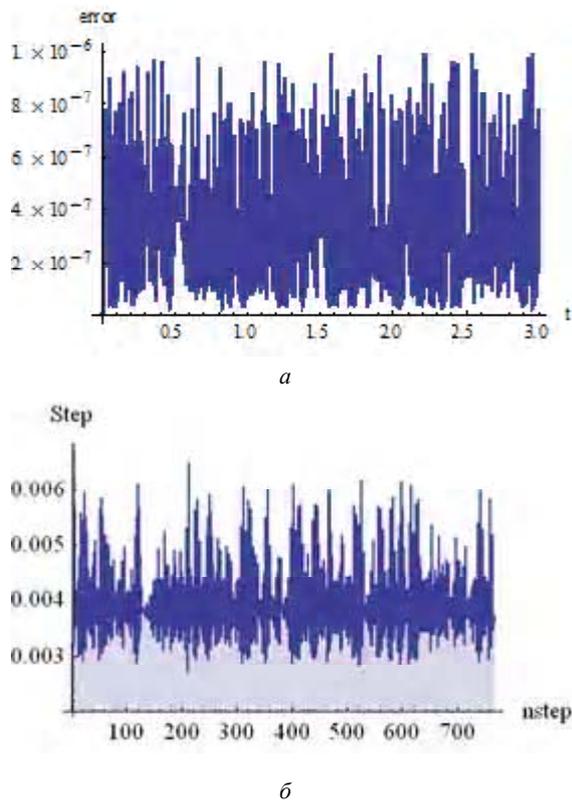


Рис. 7. Погрешность (а) и динамика шага интегрирования (б) для задачи (14) с операторами перехода метода Дормана-Принса 5(4) с точностью 10^{-6}

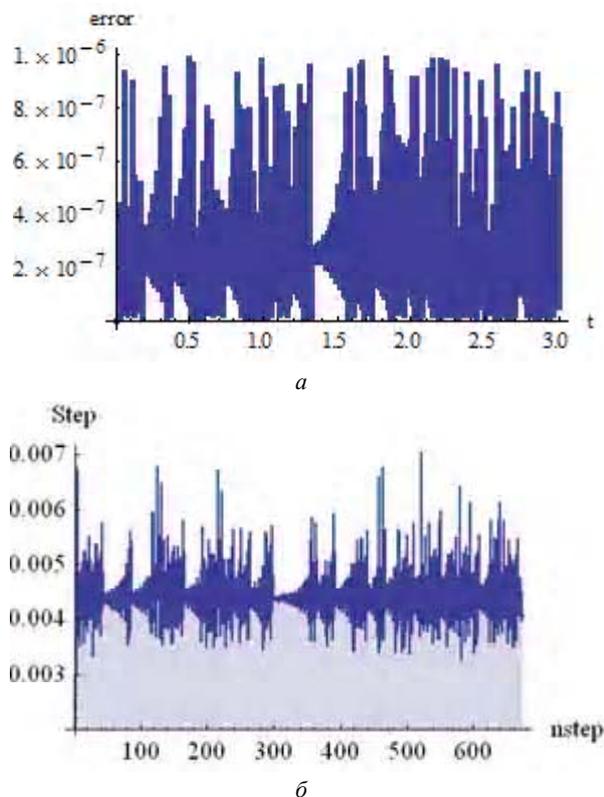


Рис. 8. Погрешность (а) и динамика шага интегрирования (б) для задачи (14) с операторами перехода метода Фельберга 7(8) системы с точностью 10^{-6}

туда строятся один раз, до начала вычислений, и позволяют за счет параллельной реализации в s раз сократить число матричных операций, выполняемых на каждом шаге. При этом для умножения матриц используются модифицированные параллельные алгоритмы, позволяющие в 2 раза сократить число обменов между процессорными элементами, выполняемых на каждом шаге. Сокращение числа обменов достигается за счет изменения порядка начальной загрузки значений коэффициентов матриц в решающее поле микропроцессоров параллельных вычислительных систем.

Для всех типов рассмотренных подходов предложены алгоритмы управления шагом интегрирования с отображением на топологические структуры параллельных вычислительных систем.

СПИСОК ЛИТЕРАТУРЫ

1. *Zanariah, A. M.* Solving Large Systems of Ordinary Differential Equations on Parallel Computer / A. M. Zanariah, M. B. Suleiman // Journ. of Scientific Research. – 2009. – Vol. 29, № 4. – P. 491–501.
2. *Enright, W. H.* Software for ordinary and delay differential equations: Accurate discrete approximate solutions are not enough / W. H. Enright // Applied Numerical Mathematics. – 2006. – Vol. 56. – № 3–4, P. 459–471.
3. *Хайпер, Э.* Решение обыкновенных дифференциальных уравнений. Жесткие задачи / Э. Хайпер, Г. Ваннер. – М.: Мир, 1999. – 685 с.
4. *Soderlind, G.* Digital filters in adaptive time-stepping / G. Soderlind // ACM Trans. Math. Softwar. – 2003. – Vol. 29. – P. 126.
5. *Дмитрієва, О. А.* Параллельное моделирование динамических объектов с выбором шага на основе экстраполяционных методов / О. А. Дмитриева // Радіоелектронні і комп'ютерні системи. – 2012. – № 6 (58). – С. 312–317.
6. *Куликов, Г. Ю.* Об автоматическом управлении размером шага и порядком в явных одношаговых экстраполяционных методах / Г. Ю. Куликов, Е. Ю. Хрусталева // Журнал вычислительной математики и математической физики. – 2008. – Т. 48, № 8. – С. 1392–1405.
7. *Schmitt, B. A.* Parameter optimization for explicit parallel peer two-step methods / B. A. Schmitt, R. Weiner, S. Jebens // Applied Numerical Mathematics. – 2009. – Vol. 59, № 3–4. – P. 769–782.
8. *Хайпер, Э.* Решение обыкновенных дифференциальных уравнений. Нежесткие задачи / Э. Хайпер, С. Нерсетт, Г. Ваннер. – М.: Мир, 1990. – 512 с.
9. *Firsova, A.* Dynamic System Simulation. Robust algorithms of state estimation of dynamic lumped parameters systems / Firsova A, Dmitrieva O. – LAPublishing, 2011. – 92 p.
10. *Дмитрієва, О. А.* Паралельні різницеві методи розв'язання задач Коші / О. А. Дмитрієва. – Донецьк: ДонНТУ, 2011. – 265 с.
11. *Дмитрієва, О. А.* Параллельное моделирование жестких систем на основе диагонализации полной матрицы / О. А. Дмитриева // Искусственный интеллект. – 2011. – № 4. – С. 46–53.
12. *Дмитрієва, О. А.* Генерація стійких блокових методів для розв'язання жорстких диференціальних рівнянь і їх систем / О. А. Дмитрієва // Наукові праці Донецького національного технічного університету. Серія «Інформатика, кібернетика та обчислювальна техніка» (ІКОТ-2011). Випуск 14 (188). – Донецьк: ДонНТУ. – 2011. – С. 36–43.

13. Голуб, Дж. Матричные вычисления / Дж. Голуб, Ч. Ван Лоун. – М. : Мир, 1999. – 548 с.
14. Butcher, J. C.: ARK methods for stiff problems / J. C. Butcher, N. Rattenbury // *Applied Numerical Mathematics*. – 2005. – Vol. 53. – № 1. – P. 165–181.

Стаття надійшла до редакції 27.08.2012.

Дмитрієва О. А.

ГЕНЕРАЦІЯ ОПЕРАТОРІВ ПЕРЕХОДУ ДЛІЯ ПАРАЛЛЕЛЬНОГО УПРАВЛІННЯ КРОКОМ ПРИ МОДЕЛЮВАННІ ЛІНІЙНИХ ДИНАМІЧНИХ СИСТЕМ

Запропоновано паралельні алгоритми моделювання лінійних динамічних об'єктів, засновані на побудові операторів переходу. Розроблені алгоритми базуються на вкладених стадійних схемах і дозволяють здійснювати автоматичний вибір оптимального розміру кроку в кожній точці сітки. Паралельна реалізація орієнтована на кластерні обчислювальні системи типу MIMD.

Ключові слова: паралельне моделювання, стадійний метод, оператор переходу, адаптація кроку, кластер.

Dmitrieva O. A.

GENERATION OF THE TRANSITION OPERATOR FOR PARALLEL STEP CONTROL AT MODELING LINEAR DYNAMIC SYSTEMS

Parallel algorithms of modeling of the linear dynamic objects based on the construction of transition operators are offered. The developed algorithms are based on the nested stages schemes and allow one to fulfill an automatic selection of the optimum step size at each grid point. Parallel realization is focused on the cluster computing systems of type MIMD.

The basic idea, which is based on the construction of the transition operators for solving linear systems of ordinary differential equations on parallel computers, is the simultaneous receipt of two approximations of different orders. Step control algorithms proposed in the article are based on the use of nested stages methods. Parallel processing is carried out within each step with the number of stages s and $s+1$. Two strands of calculations are independent and need to exchange occurs only after the final results for the calculation points.

Transition operators for each s -stage method are built once, before the calculations and allow reducing the number of matrix operations performed at each step by s times due to parallel implementation.

Key words: parallel simulation, stepwise method, transition operator, step adaptation, the cluster.

REFERENCES

1. Zanaiah A. M., Suleiman M. B Solving Large Systems of Ordinary Differential Equations on Parallel Computer, *Journ. of Scientific Research*, 2009, Vol. 29, No. 4, pp. 491–501.
2. Enright W. H. Software for ordinary and delay differential equations: Accurate discrete approximate solutions are not enough, *Applied Numerical Mathematics*, 2006, Vol. 56, No. 3–4, pp. 459–471.
3. Hairer E., Wanner G. Solving ordinary differential equations. Stiff and Differential-Algebraic Problems. Moscow, Mir, 1999, 685 p.
4. Soderlind G. Digital filters in adaptive time-stepping, *ACM Trans. Math. Softwar*, 2003, Vol. 29, pp. 1–26.
5. Dmitrieva O. A. Parallel simulation of dynamic objects with automatic step selection on the basis of extrapolation methods, *Radioelectronic and computer systems*, 2012, No. 6(58), pp. 312–317.
6. Kulikov G. Yu., Khrustaleva E. Y. Automatic step size and order control in explicit one-step extrapolation methods, *Computational Mathematics and Mathematical Physics*, 2008, Vol. 48. No. 8, pp. 1392–1405.
7. Schmitt B. A., Weiner R., Jebens S. Parameter optimization for explicit parallel peer two-step methods, *Applied Numerical Mathematics*, 2009, Vol. 59, No. 3–4. pp. 769–782
8. Hairer E., Norsett S., Wanner G. Solving ordinary differential equations. Nonstiff problems. Moscow, Mir, 1990, 512 p.
9. Firsova A, Dmitrieva O. Dynamic System Simulation. Robust algorithms of state estimation of dynamic lumped parameters systems. LAPublishing, 2011, 92 p.
10. Dmitrieva O. Parallel difference methods for solving the Cauchy problem. Donetsk, DonNTU, 2011, 265 p.
11. Dmitrieva O. Parallel modeling of stiff systems on the basis of diagonalization of a full matrix, *Artificial Intelligence*, 2011, No. 4, pp. 46–53.
12. Dmitrieva O.A. Generation of stable methods for stiff differential equations and their systems solving, *Informatics, Cybernetics and Computer Science*, 2011, No. 14(188), pp. 36–43.
13. Golub G., Loan C. Matrix Computations. Moscow, Mir, 1999, 548 p.
14. Butcher J. C, Rattenbury N. ARK methods for stiff problems, *Applied Numerical Mathematics*, 2005, Vol. 53, No. 1, pp. 165–181.