

УДК 519.6

ПРИМЕНЕНИЕ МНОГОПРОЦЕССОРНЫХ СИСТЕМ ДЛЯ РЕШЕНИЯ ДВУМЕРНЫХ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ ФРЕДГОЛЬМА I РОДА ТИПА СВЕРТКИ ДЛЯ ВЕКТОРНЫХ ФУНКЦИЙ

Н. А. Евдокимова¹, Д. В. Лукьяненко², А. Г. Ягола²

Рассматриваются особенности численной реализации решения двумерных интегральных уравнений Фредгольма I рода типа свертки для векторных функций с применением многопроцессорных систем. Для решения этой некорректной задачи применяется алгоритм решения интегрального уравнения типа свертки с использованием метода регуляризации, основанного на минимизации функционала А. Н. Тихонова с регуляризатором — квадратом нормы в пространстве $W_2^2 [(-\infty, +\infty) \times (-\infty, +\infty)]$. Для отыскания экстремали функционала А. Н. Тихонова применяется двумерное дискретное преобразование Фурье. Выбор параметра регуляризации осуществляется в соответствии с принципом обобщенной невязки. Предлагаются схемы распараллеливания задачи, показывается эффективность данного подхода. Работа выполнена при финансовой поддержке РФФИ (коды проектов 08-01-00160 и 07-01-92103-ГФЕН_а).

Ключевые слова: обратная задача, уравнение типа свертки, векторная функция, математическое моделирование, регуляризация Тихонова, параллельные алгоритмы.

1. Введение. Многие физические задачи сводятся к решению двумерных интегральных уравнений Фредгольма I рода типа свертки для векторных функций. До сих пор были широко известны алгоритмы решения интегральных уравнений типа свертки (как одномерных, так и двумерных) для обычных скалярных функций [1]. Поэтому в том случае, если прикладная физическая задача сводилась к операторному уравнению, содержащему ядро типа свертки, а искомые физические величины представляли собой векторные функции, приходилось использовать менее эффективные вычислительные алгоритмы, не использующие преимущества специфического вида уравнений типа свертки и соответствующих интегральных преобразований.

В настоящей статье рассмотрено обобщение метода решения двумерного интегрального уравнения типа свертки на случай, когда в качестве входной и искомой функции рассматриваются векторные функции. Необходимость решать уравнение данного типа в такой постановке возникла при решении задачи восстановления магнитного поля на плоской поверхности по данным измерений на сенсорной плоскости. Кроме того, предлагаемые методы могут эффективно применяться при решении достаточно широкого класса физических задач. Такие задачи при большом объеме входных данных неприступны для обычных последовательных компьютеров, и для их решения лучше всего использовать мощные многопроцессорные кластеры.

2. Постановка задачи и метод решения. Метод решения двумерного интегрального уравнения типа свертки подробно описан в работе [1]. Здесь предлагается его обобщение на случай, когда в качестве входной и искомой функций рассматриваются векторные функции. Пусть задано уравнение

$$Az = \iint_{-\infty}^{+\infty} K(x-s, y-t) z(s, t) ds dt = u(x, y), \tag{1}$$

где $u(x, y)$ и $z(s, t)$ — векторные функции, а ядро $K(x-s, y-t)$ — матричная функция с элементами, зависящими от соответствующей разности аргументов:

$$u(x, y) = \begin{pmatrix} u^1(x, y) \\ u^2(x, y) \\ u^3(x, y) \end{pmatrix}, \quad z(s, t) = \begin{pmatrix} z^1(s, t) \\ z^2(s, t) \\ z^3(s, t) \end{pmatrix}, \quad K = \begin{pmatrix} K^{11} & K^{12} & K^{13} \\ K^{21} & K^{22} & K^{23} \\ K^{31} & K^{32} & K^{33} \end{pmatrix}.$$

¹ Южно-Уральский государственный университет, механико-математический факультет, просп. Ленина, 76, 454080, Челябинск; старший преподаватель, e-mail: naevdok@math.susu.ac.ru

² Московский государственный университет им. М. В. Ломоносова, физический факультет, Ленинские горы, 119992, Москва; Д. В. Лукьяненко, аспирант, e-mail: lukyanenko-dmitry@yandex.ru; А. Г. Ягола, профессор, e-mail: yagola@yahoo.com

Пусть элементы матричной функции $K(u, w)$ и правая часть $\mathbf{u}(x, y)$, имеющая компоненты $u^1(x, y)$, $u^2(x, y)$ и $u^3(x, y)$, принадлежат пространству $L_2[(-\infty, +\infty) \times (-\infty, +\infty)]$, точное решение $\mathbf{z}(s, t)$, имеющее компоненты $z^1(s, t)$, $z^2(s, t)$ и $z^3(s, t)$, принадлежит пространству $W_2^2[(-\infty, +\infty) \times (-\infty, +\infty)]$, а оператор A типа свертки с ядром $K(u, w)$ взаимно-однозначен. Нормы правой части и решения вводятся следующим образом: $\|\mathbf{u}\|_{L_2} = \sqrt{\|u^1\|_{L_2}^2 + \|u^2\|_{L_2}^2 + \|u^3\|_{L_2}^2}$, $\|\mathbf{z}\|_{W_2^2} = \sqrt{\|z^1\|_{W_2^2}^2 + \|z^2\|_{W_2^2}^2 + \|z^3\|_{W_2^2}^2}$. Пусть вместо точно известных $\bar{\mathbf{u}}$ и оператора A известны их приближенные значения \mathbf{u}_δ и A_h , такие, что $\|\mathbf{u}_\delta - \bar{\mathbf{u}}\|_{L_2} \leq \delta$, $\|A - A_h\|_{W_2^2 \rightarrow L_2} \leq h$. Рассмотрим далее функционал Тихонова

$$M^\alpha[\mathbf{z}] = \|A_h \mathbf{z} - \mathbf{u}_\delta\|_{L_2}^2 + \alpha \|\mathbf{z}\|_{W_2^2}^2. \quad (2)$$

Для любого $\alpha > 0$ существует единственная экстремаль функционала Тихонова \mathbf{z}_η^α , $\eta = \{\delta, h\}$, реализующая минимум $M^\alpha[\mathbf{z}]$. При выборе параметра $\alpha = \alpha(\eta)$ по обобщенному принципу невязки

$$\rho(\alpha) = \|A_h \mathbf{z}_\eta^\alpha - \mathbf{u}_\delta\|_{L_2}^2 - (\delta + h \|\mathbf{z}_\eta^\alpha\|_{W_2^2})^2 = 0 \quad (3)$$

\mathbf{z}_η^α стремится при $\eta \rightarrow 0$ к точному решению задачи в норме W_2^2 , а следовательно, и равномерно на каждом прямоугольнике $[a, A] \times [b, B]$.

Пусть известно, что носитель ядра $\text{supp } K(u, w) \subset [l_1, L_1] \times [l_2, L_2]$ и $\text{supp } \mathbf{z}(s, t) \subset [a, A] \times [b, B]$, тогда $\text{supp } \mathbf{u}(x, y) \subset [c, C] \times [d, D]$. Доопределив $\mathbf{z}(s, t)$ нулем вне $[a, A] \times [b, B]$, получим уравнение

$$A\mathbf{z} = \int_{c-(l_1+L_1)/2}^{c-(l_1+L_1)/2} \int_{d-(l_2+L_2)/2}^{d-(l_2+L_2)/2} K(x-s, y-t) \mathbf{z}(s, t) ds dt = \mathbf{u}(x, y),$$

поскольку $[a, A] \times [b, B] \subset \left[c - \frac{l_1 + L_1}{2}, c - \frac{l_1 + L_1}{2} \right] \times \left[d - \frac{l_2 + L_2}{2}, d - \frac{l_2 + L_2}{2} \right]$.

Продолжая ядро с его локального носителя $[l_1, L_1] \times [l_2, L_2]$ с периодом $T_1 = C - c$ по первому аргументу и $T_2 = D - d$ по второму, мы получаем возможность применить результаты, описанные в [1], для решения двумерных интегральных уравнений типа свертки для векторной функции с помощью быстрого дискретного преобразования Фурье.

3. Структура алгоритма и возможности для распараллеливания. Для численного решения задачи переходим к конечномерным пространствам.

Мы будем рассматривать случай, когда локальные носители $\mathbf{z}(s, t)$ и $\mathbf{u}(x, y)$ лежат внутри прямоугольника $[0, 2r] \times [0, 2R]$, а сами \mathbf{z} и \mathbf{u} доопределены нулями вне их носителей на этом прямоугольнике. Прделав аналогичную процедуру с ядром, будем считать все функции периодически продолженными (с периодом $2r$ по первому аргументу и $2R$ — по второму) и рассмотрим уравнение (1) на всей плоскости.

Введя равномерные сетки по (x, y) и (s, t) : $x_k = s_k = k\Delta x$, $\Delta x = \frac{2r}{n_1}$, $k = 0, 1, \dots, n_1 - 1$; $y_l = t_l = l\Delta y$,

$\Delta y = \frac{2R}{n_2}$, $l = 0, 1, \dots, n_2 - 1$ (n_1 и n_2 будем считать четными) и аппроксимируя уравнение по формуле прямоугольников, получим

$$\sum_{m=0}^{n_1-1} \sum_{j=0}^{n_2-1} \sum_{p=0}^3 K_{k-m, l-j}^{qp} z_{mj}^p \Delta x \Delta y = u_{kl}^q, \quad (4)$$

где z_{mj}^p и u_{kl}^q ($q, p = 1, 2, 3$) — компоненты векторных функций \mathbf{z}_{mj} и \mathbf{u}_{kl} , $u_{kl}^q = u^q(x_k, y_l)$, $z_{mj}^p = z^p(s_m, t_j)$ и $K_{k-m, l-j}^{qp} = K^{qp}(x_k - s_m, y_l - t_j)$. Все переменные будем считать вещественными.

Пусть задана матрица f_{mn} , где $m = 0, 1, \dots, n_1 - 1$ и $n = 0, 1, \dots, n_2 - 1$. Тогда дискретное преобразование Фурье определим как

$$\tilde{f}_{mn} = \sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} f_{kl} e^{-i\omega_m x_k - i\Omega_n y_l}, \quad (5)$$

где $\omega_m = m\Delta\omega$, $\Delta\omega = \frac{\pi}{r}$, $\Omega_n = n\Delta\Omega$, $\Delta\Omega = \frac{\pi}{R}$. Обратное преобразование имеет вид

$$f_{kl} = \frac{1}{n_1 n_2} \sum_{m=0}^{n_1-1} \sum_{n=0}^{n_2-1} \tilde{f}_{mn} e^{i\omega_m x_k + i\Omega_n y_l}.$$

Запишем двумерный аналог равенства Планшереля: $\sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} f_{kl}^2 = \frac{1}{n_1 n_2} \sum_{m=0}^{n_1-1} \sum_{n=0}^{n_2-1} \tilde{f}_{mn}^2$ и теоремы о свертке: $\sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} \left\{ \sum_{p=0}^{n_1-1} \sum_{j=0}^{n_2-1} H_{k-p, l-j} z_{pj} \Delta x \Delta y \right\} e^{-i\omega_m x_k - i\Omega_n y_l} = \tilde{H}_{mn} \tilde{z}_{mn} \Delta x \Delta y$.

Конечно-разностная аппроксимация функционала $M^\alpha [z]$ для уравнения (2) имеет вид

$$M^\alpha [z] = \sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} \sum_{q=1}^3 \left(\sum_{p=0}^{n_1-1} \sum_{j=0}^{n_2-1} \sum_{m=0}^3 K_{k-p, l-j}^{qm} z_{pj}^m \Delta x \Delta y - u_{kl}^q \right)^2 \Delta x \Delta y + \alpha \sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} \sum_{q=1}^3 \left\{ z_{kl}^{q2} + \left[\frac{\partial^2 z^q(s_k, t_l)}{\partial s^2} \right]^2 + 2 \left[\frac{\partial^2 z^q(s_k, t_l)}{\partial s \partial t} \right]^2 + \left[\frac{\partial^2 z^q(s_k, t_l)}{\partial t^2} \right]^2 \right\} \Delta x \Delta y,$$

где для коэффициентов дискретного преобразования Фурье векторов, являющихся аппроксимацией производных, будем иметь

$$\left[\frac{\partial^2 z^q(s_k, t_l)}{\partial s^2} \right] = -\omega_m^2 \tilde{z}_{mn}^q, \quad \left[\frac{\partial^2 z^q(s_k, t_l)}{\partial t^2} \right] = -\Omega_n^2 \tilde{z}_{mn}^q, \quad \left[\frac{\partial^2 z^q(s_k, t_l)}{\partial s \partial t} \right] = -\omega_m \Omega_n \tilde{z}_{mn}^q.$$

Воспользовавшись равенством Планшереля и теоремой о свертке, конечно-разностную аппроксимацию функционала $M^\alpha [z]$ представим в форме

$$M^\alpha [z] = \frac{\Delta x \Delta y}{n_1 n_2} \sum_{m=0}^{n_1-1} \sum_{n=0}^{n_2-1} \sum_{q=1}^3 \left\{ \left| \sum_{p=1}^3 \tilde{K}_{mn}^{qp} \tilde{z}_{mn}^p \Delta x \Delta y - \tilde{u}_{mn}^q \right|^2 + \alpha \left[1 + (\omega_m^2 + \Omega_n^2) \right] |\tilde{z}_{mn}^q|^2 \right\}. \quad (6)$$

В результате минимизации этого функционала можно найти $(\tilde{z}^\alpha)^q$. Решение $(z_\eta^\alpha)^q(s, t)$ на сетке (s_k, t_l) вычисляется обратным дискретным преобразованием Фурье:

$$(z_\eta^\alpha)^q(s_k, t_l) = \frac{1}{n_1 n_2} \sum_{m=0}^{n_1-1} \sum_{n=0}^{n_2-1} (\tilde{z}_{mn}^\alpha)^q e^{i\omega_m s_k + i\Omega_n t_l}. \quad (7)$$

Выражение для $\|A_h z^\alpha - u_\delta\|^2$ при выборе параметра регуляризации по обобщенному принципу невязки (3) может быть получено без вычисления экстремали z^α , что допускает относительно простую и быстродействующую программную реализацию, поскольку существуют методы быстрого преобразования Фурье, для которых созданы стандартные программы.

Основные вычисления приходятся на прямое (5) и обратное (7) преобразования Фурье. Как можно видеть, эти формулы состоят из не связанных между собой слагаемых, что дает возможность применения многопроцессорной системы. Задачу можно распараллелить, т.е. переписать программу таким образом, чтобы независимые части программы выполнялись независимо друг от друга на разных компьютерах.

4. Распараллеливание задачи. При решении модельных задач использовались вычислительные кластеры Научно-исследовательского вычислительного центра МГУ и Южно-Уральского государственного университета. При составлении программ использовалась библиотека MPI (Message Passing Interface). В качестве языка программирования был выбран Fortran 90.

Исходная задача решалась с использованием N параллельных процессов $0, 1, \dots, N - 1$, выполняющихся на отдельных процессорах и при необходимости взаимодействующих друг с другом [2, 3]. Нулевой процесс выполнял нераспараллеливаемые действия: чтение исходных данных из файла, запись результатов, сбор результатов вычислений остальных процессов и перераспределение между ними новых задач. Распараллеливание происходило только при вычислении фурье-образов ядра и правой части уравнения, при выборе параметра регуляризации по обобщенному принципу невязки, а также при финальном вычислении решения с помощью обратного преобразования Фурье.

Вычисление фурье-образов ядра и правой части уравнения (4) давало следующие возможности для распараллеливания. Применяя формулу (5) к ядру и правой части исходного конечно-разностного уравнения, можно разбить вычисления на независимые процессы, которые вычисляют независимые группы слагаемых по формуле дискретного преобразования Фурье:

$$\tilde{f}_{mn} = \sum_{k=0}^{n_1-1} \sum_{l=0}^{n_2-1} f_{kl} e^{-i\omega_m x_k - i\Omega_n y_l} = f_{11} e^{-im\pi/r - in\pi/R} + \dots + f_{n_1-1, n_2-1} e^{-im\pi/r - in\pi/R}.$$

В этом случае есть два подхода к распараллеливанию рассмотренных вычислений. Первый подход заключается в том, что нулевой процесс распределяет группы вычисляемых слагаемых между ненулевыми процессами поровну. Однако при этом возможно, что часть процессов закончат обработку своих данных быстрее остальных и будут простаивать. В связи с этим был применен второй подход: нулевой процесс загружает остальные процессы по мере их освобождения. Как наиболее эффективный, последний подход применялся и в остальных частях программы.

Здесь надо заметить, что нулевой процесс суммирует данные в том порядке, в котором они поступают. Это является одной из характерных особенностей распараллеливания, в результате которой при повторных запусках программы результаты вычислений могут отличаться в последних знаках после запятой. Это связано с тем, что в каждом случае последовательность поступающих на суммирование данных различна, и соответственно итогом округления являются различные числа. На итоговую ошибку влияет также и количество процессов — чем их больше, тем больше и накапливаемая ошибка.

В качестве параллельного варианта быстрого преобразования Фурье была использована одна из стандартных параллельных реализаций данного метода [4], дающая высокую степень эффективности распараллеливания при проведении вычислений на большом числе процессов.

Отметим еще раз, что при решении двумерных интегральных уравнений Фредгольма I рода типа свертки для векторных функций распараллеливание осуществлялось только для вычисления фурье-образов входных функций, при выборе параметра регуляризации посредством метода обобщенной невязки и для вычисления решения обратным преобразованием Фурье. Все остальные вычисления производились последовательно. Поэтому время работы программы на N процессорах не уменьшилось в $N - 1$ раз по сравнению с нераспараллеленной программой (нулевой процесс согласно принятым схемам распараллеливания не принимает участия в основных вычислениях, а лишь принимает и передает данные) в соответствии с эмпирическим законом Амдала. Однако при больших размерностях входных данных и большом числе задействованных в вычислениях узлов кластера время работы нераспараллеленного кода пренебрежимо мало, в связи с чем можно считать, что на больших сетках время работы на N процессорах меньше в $N - 1$ раз.

5. Модельные расчеты. Для модельных вычислений были выбраны точные решения [5, 6]

$$\begin{aligned} z^1(s, t) &= \left(\frac{e^{-(s-0,3)^2/6,7} + e^{-(s-3,7)^2/1,1}}{0,955} - 0,052 \right) e^{-(t-0,5)^2/1,9}, \\ z^2(s, t) &= \left(\frac{e^{-(s+0,3)^2/3,8} + e^{-(s-2,5)^2/3,8}}{0,955} - 0,052 \right) e^{-(t-0,5)^2/1,5}, \\ z^3(s, t) &= \left(\frac{e^{-(s+4,3)^2/5,1} + e^{-(s-7,2)^2/0,4}}{0,955} - 0,052 \right) e^{-(t-0,5)^2/0,9}, \end{aligned}$$

соответствующие распределению магнитных диполей на какой-либо плоскости, на которой задана сетка

(s, t) , и ядро вида $K = \begin{pmatrix} K^{11} & K^{12} & K^{13} \\ K^{21} & K^{22} & K^{23} \\ K^{31} & K^{32} & K^{33} \end{pmatrix}$ с элементами

$$\begin{aligned} K^{11}(x-s, y-t) &= \frac{\mu_0}{4\pi r_{x-s, y-t}^5} (2(x-s)^2 - (y-t)^2 - h^2), \\ K^{12}(x-s, y-t) &= K^{21}(x-s, y-t) = \frac{\mu_0}{4\pi r_{x-s, y-t}^5} 3(x-s)(y-t), \\ K^{13}(x-s, y-t) &= K^{31}(x-s, y-t) = \frac{\mu_0}{4\pi r_{x-s, y-t}^5} 3(x-s)h, \\ K^{22}(x-s, y-t) &= \frac{\mu_0}{4\pi r_{x-s, y-t}^5} (2(y-t)^2 - (x-s)^2 - h^2), \\ K^{23}(x-s, y-t) &= K^{32}(x-s, y-t) = \frac{\mu_0}{4\pi r_{x-s, y-t}^5} 3(y-t)h, \\ K^{33}(x-s, y-t) &= \frac{\mu_0}{4\pi r_{x-s, y-t}^5} (2h^2 - (x-s)^2 - (y-t)^2), \end{aligned}$$

где $r_{x-s, y-t} = \{(x-s)^2 + (y-t)^2 + h^2\}^{1/2}$, h — расстояние между сенсорной плоскостью и плоскостью восстановления магнитного поля, μ_0 — магнитная постоянная.

Выписанное ядро соответствует практической задаче восстановления распределения магнитных диполей на какой-либо плоской поверхности (решение задано на сетке (s, t)) по измеренному магнитному полю в плоскости измерительных датчиков, которые расположены в узлах сетки (x, y) .

Оператор A с ядром K считался точно заданным. Уровень погрешности правой части был задан равным $\delta^2 = 10^{-19}$. Расчет проводился в квадратной вычислительной области. В приведенной таблице представлены результаты эффективности восстановления решения на многопроцессорных вычислительных системах при разном числе процессоров и разной густоте сеток, на которых задавались входные данные и искомое точное решение.

Размерность сетки	Число процессоров						
	1	3	4	5	16	32	64
16 × 16	100%	97%	89%	81%	75%	61%	53%
32 × 32	100%	95%	84%	87%	85%	75%	65%
64 × 64	100%	91%	85%	85%	81%	82%	73%
128 × 128	100%	86%	85%	92%	83%	87%	82%
256 × 256	100%	92%	93%	94%	89%	88%	87%
512 × 512	100%	95%	96%	95%	95%	92%	93%

Под эффективностью подразумевается отношение реального времени вычислений к “идеальному” в соответствии с эмпирическим законом Амдала (время вычисления на q процессорах задается формулой $t_q = \frac{t_2}{q - 1}$, где t_2 — время вычисления на двух процессорах). Напомним, что реально в параллельных вычислениях используется $N - 1$ процесс, так как нулевой процесс только собирает значения и перераспределяет данные между остальными процессами.

Следует заметить, что на двух процессорах время вычислений больше времени вычислений на обычной машине в связи с тем, что и в том, и в другом случае непосредственно в вычислениях участвует только один процесс, но на кластере еще тратится дополнительное время на обмен сообщениями между процессами. В связи с этим эффективность работы параллельной версии программы уменьшается с увеличением числа процессов, участвующих в вычислениях. Однако с другой стороны при увеличении размерности задачи уменьшается и доля времени на межпроцессорные обмены информацией по сравнению с общим временем вычислений. Поэтому наиболее эффективно параллельные алгоритмы могут быть использованы для решения задач на сетках большой размерности (т.е. для наиболее трудоемких задач), что и видно из приведенной выше таблицы.

6. Заключение. В статье описан эффективный метод решения физических задач, приводящих к необходимости решать двумерные интегральные уравнения Фредгольма I рода типа свертки для векторных функций. Для данного типа задач продемонстрирована эффективность использования многопроцессорных систем.

СПИСОК ЛИТЕРАТУРЫ

1. Тихонов А.Н., Гончарский А.В., Степанов В.В., Ягола А.Г. Численные методы решения некорректных задач. М.: Наука, 1990.
2. Воеводин В.В., Воеводин В.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002.
3. Ягола А.Г., Васильев М.П. Применение многопроцессорных систем для решения двумерных интегральных уравнений Фредгольма 1-го рода // Вычислительные методы и программирование. 2003. 4, № 2. 156–159.
4. Вычислительный кластер НИВЦ МГУ (<http://parallel.ru/cluster>).
5. Lukyanenko D.V., Pei Y.H., Yagola A.G., Gui-Rong L., Evdokimova N.A. Numerical methods for solving ill-posed problems with constraints and applications to inversion of the magnetic field // Международная конференция “Обратные и некорректные задачи математической физики”, посвященная 75-летию академика М. М. Лаврентьева, 20–25 августа 2007 г., Новосибирск. Тезисы докладов секции № 3. 1–2 (<http://www.math.nsc.ru/conference/ipmp07/section3.htm>).
6. Pei Y.H., Yagola A.G. Constraint magnetization parameter inversion by iterative Tikhonov regularization // Международная конференция “Обратные и некорректные задачи математической физики”, посвященная 75-летию академика М. М. Лаврентьева, 20–25 августа 2007 г., Новосибирск. Тезисы докладов секции № 3. 1–2 (<http://www.math.nsc.ru/conference/ipmp07/section3.htm>).

Поступила в редакцию
10.11.2008