# О ЧИСЛЕННОМ РЕШЕНИИ ДВУМЕРНОГО НЕЛИНЕЙНОГО УРАВНЕНИЯ ТЕПЛОПРОВОДНОСТИ


Нефедова О.А.[a)], Спевак Л.Ф.[b)], Казаков А.Л.[c)]

*ИМАШ УрО РАН, д. 34, ул. Комсомольская, г. Екатеринбург, 620034, Российская Федерация*
*e-mail: [a)]nefedova@imach.uran.ru, [b)]lfs@imach.uran.ru, [c)]kazakov@icc.ru*



**Аннотация** ***Работа посвящена численному решению задачи о движении тепловой волны для вырождающегося нелинейного уравнения второго порядка параболического типа с источником. Нелинейность уравнения обусловлена степенной зависимостью коэффициента теплопроводности от температуры. Рассматривается задача при краевом условии, задающем закон движения фронта тепловой волны. Предложен алгоритм решения, основанный на комбинации методов граничных элементов и коллокаций. Алгоритм реализован в виде программы, написанной на языке программирования C++. Организация параллельных вычислений построена с использованием открытого стандарта OpenCL.***


**Постановка краевой задачи** В работе рассмотрено вырождающееся нелинейное уравнение параболического типа с источником, заданным функцией $\varphi(u)$,

$$u_{x_1x_1} + u_{x_2x_2} = \frac{1}{u}\left(u_t - \frac{u_{x_1}^2 + u_{x_2}^2}{\sigma} - \varphi(u)\right), \tag{1}$$

при краевом режиме: $$u\,\big|_{\mathbf{x}\in S^{(t_k)}} = 0\,,\quad q\,\big|_{\mathbf{x}\in S^{(t_k)}} = \frac{\sigma b_t}{\sqrt{b_{x_1}^2 + b_{x_2}^2}}\,. \tag{2}$$

Уравнение $b(t,x_1,x_2)=0$ в каждый момент времени определяет нулевой фронт тепловой волны $S^{(t)}$ – замкнутую гладкую линию, ограничивающую область $V^{(t)}$.
Ранее, в других работах, авторами был предложен алгоритм решения А1 задачи (1), (2) на основе методов граничных элементов и двойственной взаимности.

**Алгоритм решения А2** Решение задачи выполняется по шагам по времени. В каждый момент времени $t_k$ решение представляется в виде суммы двух функции

$$u(t_k, x_1, x_2) = v(x_1, x_2) + w(x_1, x_2), \tag{3}$$

где $v(x_1,x_2)$ – частное решение уравнения (1), а $w(x_1,x_2)$ – решение краевой задачи для уравнения Лапласа: $$w_{x_1x_1} + w_{x_2x_2} = 0\,, \tag{4}$$

с граничными условиями: $$w\,\big|_{\mathbf{x}\in S^{(t_k)}} = -v\,\big|_{\mathbf{x}\in S^{(t_k)}}\,,\quad q_w^{(n)}\Big|_{\mathbf{x}\in S^{(t_k)}} = \frac{\sigma b_t}{\sqrt{b_{x_1}^2 + b_{x_2}^2}} - \frac{\partial v^{(n)}}{\partial n} \tag{5}$$

Задача решена итерационно. Задача (4), (5) на каждой итерации решается методом граничных элементов. Частное решение $v(x_1,x_2)$ ищется с помощью разложения неоднородностей уравнения (1) по системе РБФ. Итерационный процесс останавливается на $n$-ой итерации, когда значения $u^{(n-1)}$ и $u^{(n)}$ достаточно близки.

**Программная реализация** Была разработана программа, реализующая представленный алгоритм А2. Работа программы была протестирована сравнением результатов расчетов с известными точными решениями и с данными, полученными авторами ранее.

**Пример** Исследовано уравнение параболического типа с источником вида $\varphi(u)=u$. Результаты расчетов сравнивались с точным решением для нулевого фронта вида: $b(t,x_1,x_2)=x_1{}^2+x_2{}^2-r^2(t)$, $r(t)=e^t$. В таблице 1 приведены относительные погрешности численных решений, рассчитанных с помощью разработанного ранее алгоритма А1 и нового алгоритма А2 на внутренней границе для $t=1$ при 400 граничных элементах и 100 внутренних точках коллокации. В таблице 2 представлено время счета для трех реализаций параллельных вычислений при различных количествах граничных элементов. Расчеты проводились на двухъядерном процессоре, на графическом процессоре, имеющем 720 ядер, и на вычислительном модуле NVIDIA Tesla K40m суперкомпьютера «УРАН» ИММ УрО РАН.

*Таблица 1. Относительные погрешности полученных по алгоритмам А1 и А2 численных решений задачи, рассчитанные для момента времени $t=1$*

| алг-тм | относительные погрешности | | | | | |
|---|---|---|---|---|---|---|
| | $h$ | $f_i = r_i$ | $f_i = r_i^3$ | $f_i = 1 + r_i$ | $f_i = e^{-(\varepsilon r_i)^2}$ | $f_i = \sqrt{1+(\varepsilon r_i)^2}$ |
| А1 | 0.1 | 0.000953 | 0.000817 | 0.000750 | 0.000709 | 0.000702 |
| А2 | 0.1 | 0.000624 | 0.000551 | 0.000516 | 0.000335 | 0.000321 |
| А1 | 0.05 | 0.000849 | 0.000709 | 0.000673 | 0.000614 | 0.000609 |
| А2 | 0.05 | 0.000601 | 0.000524 | 0.000488 | 0.000297 | 0.000291 |

*Таблица 2. Время решения задачи для различных реализаций алгоритмов*

| алг-тм | кол-во гр.эл-тов | реализация OpenCL на CPU с SSE, с | реализация OpenCL на ATI Radeon HD 5750 GPU, с | реализация OpenCL на NVIDIA Tesla K40m, с |
|---|---|---|---|---|
| А1 | 200 | 62 | 46 | 10 |
| А2 | 200 | 21 | 17 | 9 |
| А1 | 300 | 126 | 61 | 14 |
| А2 | 300 | 59 | 38 | 13 |
| А1 | 400 | 269 | 130 | 20 |
| А2 | 400 | 135 | 79 | 17 |

**Заключение** **Предложен численный алгоритм А2 решения вырождающегося нелинейного уравнения параболического типа с источником и выполнена его программная реализация. Результаты расчетов показали стабильную сходимость и корректную работу нового метода. Сравнение показывает более высокую точность численных решений по алгоритму А2 по сравнению с разработанным ранее алгоритмом А1 и высокую эффективность распараллеливания А2.**