UDC 519.688

A PARALLEL ALGORITHM FOR SYMBOLIC SOLVING PARTIAL DIFFERENTIAL EQUATIONS

© **Natalia Aleksandrovna Malaschonok**
Tambov State University named after G.R. Derzhavin, Internatsionalnaya, 33, Tambov, 392000, Russia, Candidate of Physics and Mathematics, Associate Professor of Mathematical Analysis Department, e-mail: namalaschonok@gmail.com

Key words: parallel algorithms; computer algebra; partial differential equations; Laplace–Carson transform; compatibility conditions.
A parallel algorithm for symbolic solving partial differential equations by means of Laplace–Carson transform is produced. The problem is reduced to solving linear algebraic systems with polynomial coefficients, for which efficient parallel algorithms exist. It permits to construct a fast parallel algorithm for systems of partial differential equations. An algorithm includes a procedure to obtain compatibility conditions for initial data.

1 Introduction

An application of Laplace and Laplace–Carson transform is useful in many problems of solving differential equations (for example [1, 2, 3, 4]) It reduces a system of partial differential equations to an algebraic linear system with polynomial coefficients. Parallel algorithms for solving such systems are being developed actively (for example, [5, 6]). It enables to construct parallel algorithms for solving linear partial differential equations with constant coefficients and systems of equations of various order, size and types. The application of Laplace–Carson transform permits to obtain compatibility conditions in symbolic way for many types of PDE equations and systems of PDE equations.

The steps, at which parallel calculations are possible and reasonable we denote by term **Block**. If indexes are contained, the ways of parallelization are pointed by them.

2 Input data

Denote $\widetilde{m} = (m_1, \ldots, m_n)$. Consider a system

$$\sum_{k=1}^{K} \sum_{m=0}^{M} \sum_{\widetilde{m}} a^j_{\widetilde{m}k} \frac{\partial^m}{\partial^{m_1} x_1 \ldots \partial^{m_n} x_n} u_k(x) = f_j(x), \tag{1}$$

where $j = 1, \ldots, K,\ u_k(x)\,,\ k = 1, \ldots, K,$ — are unknown functions of $x = (x_1, \ldots, x_n) \in \mathbf{R}^n_+$, $f_j \in S$, $a^j_{\widetilde{m}k}$ are real numbers, m is the order of a derivative, and k –the number of an unknown function. Here and further summing by $\widetilde{m} = (m_1, \ldots, m_n)$ is executed for $m_1 + \ldots + m_n = m$.

We consider all input functions reducible to the form;

$f_j(t) = f^i_j(x),\ x^i_j < t < t^{i+1}_j,\ i = 1, \ldots, I_j, x^1_l = 0, t^{I_j+1}_j = \infty,$

where

$$f_j^i(t) = \sum_{s=1}^{S_j^i} P_{js}^i(t) e^{b_{js}^i t}, \quad i = 1, \dots, I_j, \quad j = 1, \dots, k, \tag{2}$$

and $P_{js}^i(x) = \sum_{l=0}^{L_{js}^i} c_{sl}^{ji} x^l$.

Denote by $\mathbf{A}$ a class of functions which are reducible to the form (2).

We solve a problem with initial conditions for each variable. Introduce notations for them. Denote by Γ^ν a set of vectors $\gamma = (\gamma_1, \dots, \gamma_n)$ such that $\gamma_\nu = 1$, $\gamma_i = 0$, if $i < \nu$, and γ_i equals 0 or 1 in all possible combinations for $i > \nu$. The number of elements in Γ^ν equals $2^{\nu-1}$.

Denote $\beta = (\beta_1, \dots, \beta_n)$, $\beta_i = 0, \dots, m_i$, a set of indexes such that the derivative of $u^k(x)$ of the order β_i with respect to the variables with numbers i equals $u_{\beta,\gamma}^k(x^{(\gamma)})$ at the point $x = x^\gamma$ with zeros at the positions μ for which the coordinates γ_μ of γ equal 1 . For example, if zeros stand only at the places with the numbers $1, 2, 3$, then $\gamma = (1, 1, 1, 0, \dots, 0)$. Functions $u_{\beta,\gamma}^k(x^{(\gamma)})$ must also belong to $\mathbf{A}$. To be short we shall not write down the expressions for $u_{\beta,\Gamma}^k(x^{(\gamma)})$.

The algorithm component is the definition of compatible initial conditions. The system (1) is to be solved under such conditions.

Data file contains the coefficients, the initial conditions and the right-hand members f_j , $l = 1, \dots, K$.

The data for functions f_j consists of the polynomial coefficients, parameters of exponents, the bounds of smoothness intervals.

3 Laplace–Carson transform

Consider the space S of functions $f(x)$, $x = (x_1, \dots, x_n) \in \mathbf{R}_+^n$, $\mathbf{R}_+^n = \{x : x_i \geqslant 0, i = 1, \dots, n\}$, for which $\mathcal{M} > 0, a = (a_1, \dots, a_n) \in \mathbf{R}^n$, $a_i > 0$, $i = 1, \dots, n$, exist such that for all $x \in \mathbf{R}_+^n$ the following is true: $|f(x)| \leqslant \mathcal{M} e^{ax}$, $ax = \sum\limits_{i=1}^n a_i x_i$.

On the space S the Laplace–Carson transform (**LC**) is defined as follows:

$$LC : f(x) \mapsto F(p) = p^1 \int_0^\infty e^{-px} f(x) dx,$$

$$p = (p_1, \dots, p_n), \quad p^1 = p_1 \dots p_n,$$

$$px = \sum_{i=1}^n p_i x_i, \quad dx = dx_1 \dots dx_n.$$

LC is performed symbolically at the class $\mathbf{A}$.

4 Parallel LC algorithm

4.1 LC of a system

Let $LC : u^k \mapsto U^k, u_{\beta,\gamma}^k(x^{(\gamma)}) \mapsto U_{\beta,\gamma}^k(p^{(\gamma)}), f_j \mapsto F_j$, the notation $p^{(\gamma)}$ is correspondent to the notation $x^{(\gamma)}$. Denote by $\|\gamma\|$ the "length" of γ — the number of units in γ , $p^{\widetilde{m}} = p_1^{m_1} \dots p_n^{m_n}$.

Block 10

The LC of the left-hand side of the system (1) excluding images of initial conditions is written formally.

Block 1r

r runs trough the set of multiindexes of $u^k_{\beta,\Gamma}(x^\Gamma)$.

Then

$$LC : \frac{\partial^m}{\partial^{m_1} x_1 \ldots \partial^{m_n} x_n} u_k(x) \ \mapsto$$

$$p^{\widetilde{m}} U^k(p) \ + \sum_{\nu=1}^{n} \sum_{\beta_\nu=0}^{m_\nu} \ \sum_{\gamma\in\Gamma^\nu} (-1)^{\|\gamma\|} p_1^{m_1-\beta_1-\gamma_1} \ldots p_n^{m_n-\beta_n-\gamma_n} U^k_{\beta,\gamma}(p^{(\gamma)}).$$

Denote

$$\Phi^j_{mk} = \sum_{\widetilde{m}} a^j_{\widetilde{m}k} \sum_{\nu=1}^{n} \sum_{\beta_\nu=0}^{m_\nu} \sum_{\gamma\in\Gamma^\nu} (-1)^{\|\gamma\|} p_1^{m_1-\beta_1-\gamma_1} \ldots p_n^{m_n-\beta_n-\gamma_n} U^k_{\beta,\gamma}(p^{(\gamma)}).$$

As a result of Laplace–Carson transform of the system (1) according to initial conditions we obtain an algebraic system relative to U^k

$$\sum_{k=1}^{K} \sum_{m=0}^{M} \sum_{\widetilde{m}} a^j_{\widetilde{m}k} p^{\widetilde{m}} U^k(p) = F_j - \sum_{k=1}^{K} \sum_{m=0}^{M} \Phi^j_{mk}, j = 1, \ldots, K. \tag{3}$$

Block 2k

k runs from 1 to K.

These blocks performs LC of the right-hand parts of (1). **A** allows a further parallelization of calculations.

4.2 Solution of algebraic system

Block 3

As a result of Laplace–Carson transform of the system (1) according to initial conditions we obtain the algebraic system (3) relative to U^k.

Efficient methods of parallel solving such systems are developed (for example [5, 6]).

At this stage the problem of definition of compatibility conditions arises (see blocks 4s,5). With respect to compatible conditions we use the inverse Laplace–Carson transform and obtain the correct solution of PDE system.

4.3 Compatibility conditions

Call a rational fraction *"a proper fraction"* if the degree of each variable (over **C**) in numerator is less then its degree in denominator.

Call a set of equations, defined by conditions

• the solutions of algebraic system may be represented as sums of proper fractions with exponential coefficients;

• the denominators of these proper fractions may be reduced to a product of linear functions.
the class **B**.

(Note that the class **B** does not exhaust all cases that admit pure symbolic computations.)

Denote by D the determinant of the system (3), D_i the maximal order minors of the extended matrix of (3). A case when there is a set $\mathcal{Q}$ of zeros of D with infinite limit point at $\operatorname{Re} p_k > 0$, $k = 1, \ldots, n$, is of most interest. Solving the system (1) we obtain U^k as fractions with D in the denominators. The inverse Laplace–Carson transform is possible if α_k, $k = 1, \ldots, n$, exist such that these functions are holomorphic in the domain $\operatorname{Re} p_k > \alpha_k$. So we make a demand: D_i has zeros at $\mathcal{Q}$ of multiplicity not less than multiplicity of corresponding zeros of D. This demand produces requirements to the LC images of initial conditions functions, and after LC $^{-1}$ transform – to initial conditions. They turn to be dependent. We obtain the so-called compatibility conditions.

Block 4s

s depends upon the number of relations, from which the compatibility conditions arise.

The blocks calculate the values of numerators at zeros of denominators.

Block 5

The block implements parallel solving of the system of equations, produced by relations for compatibility conditions.

Block 6k

The blocks perform the LC $^{-1}$ of U^k. Note, that the steps of calculation of multivariate LC $^{-1}$ are produced sequentially.

5 Example

We take a simple example to demonstrate the method and the places where parallelization is possible.

It is convenient here to change notations for unknown functions, their Laplace transform, variables, initial conditions.

Example 1

Take a system of two equations with two unknown functions on $\mathbf{R}_+^2$.

$$\begin{cases} \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} & = & x, \\ \frac{\partial f}{\partial y} + \frac{\partial g}{\partial x} & = & y, \end{cases}$$

$f = f(x, y); \quad g = g(x, y)$.

Initial conditions: $f(0, y) = a(y); \quad f(x, 0) = b(x); \quad g(0, y) = c(y); \quad g(x, 0) = d(x),$

Block 1r , r=1,2.

$$a(y) \mapsto \alpha(q), \quad b(x) \mapsto \beta(p),$$
$$c(y) \mapsto \delta(q), \quad d(x) \mapsto \gamma(p).$$

Block 2k , k=1,2.
LC:

$$f(x,y) \mapsto u(p,q), \quad g(x,y) \mapsto v(p,q).$$

As a result of LC we obtain the algebraic system:

$$pu - p\alpha(q) + qv - q\gamma(p) = 1/p, \quad qu - q\beta(p) + pv - p\delta(q) = 1/q.$$

Block 3
Then

$$u = -\frac{-\alpha p^2 + \beta q^2 + (\delta - \gamma)pq}{p^2 - q^2}, \quad v = -\frac{-p^2 + q^2 + (\alpha - \beta)p^2 q^2 - (\delta p^2 - \gamma q^2)pq}{pq(p^2 - q^2)}.$$

The denominator D: $D(p,q) = pq(p^2 - q^2)$.

Block 4s , s=1.

The set of zeros of D with infinite limit points at the right half-plane is $q = p$.
Substituting $q = p$ into the nominator of u and v we obtain the compatibility condition: $\alpha - \beta + \gamma - \delta = 0$.

Block 5

For example we may take $\beta = 0; \quad \gamma = \frac{2}{p}; \quad \delta = \frac{2}{q}; \quad \alpha = 0.$
Then

$$u = -\frac{2}{p+q}, \quad v = -\frac{p + 2p^2 + q + 2q^2 + 2pq}{pq(p+q)}.$$

Block 6s , s=1,2.
LC^{-1} :

$$f = -\begin{cases} 2y, & y < x, \\ 2x, & y \geqslant x, \end{cases}$$

$$g = \begin{cases} (2+y)x, & y < x, \\ y(2+x), & y \geqslant x. \end{cases}$$

References

1. *Dahiya R.S., Jabar Saberi-Nadjafi.* Theorems on n-dimensional Laplace transforms and their applications // 15th Annual Conf. of Applied Math., Univ. of Central Oklahoma, Electr. Journ. of Differential Equations, Conf.02. 1999. P. 61-74.

2. *Dimovski I., Spiridonova M.* Computational approach to nonlocal boundary value problems by multivariate operational calculus // Mathem. Sciences Research Journal. Dec. 2005. V. 9. N. 12. P. 315-329.

3. *Malaschonok N.* An algorithm for symbolic solving of differential equations and estimation of accuracy // Computer Algebra in Scientific Computing. CASC 2009, Springer-Verlag Berlin Heidelberg, 2009. P. 213-225.

4. *Picone M.* Nuovi metodi risolutivi per i problemi d'integrazione delle equazioni lineari a derivate parziali e nuova applicazionne della trasformata multipla di Laplace nel caso delle equazioni a coefficienti costanti // Atti Accad.Sci.Torino,75. 1940. P. 1-14.

5. *Watt S.M.* Pivot-Free Block Matrix Inversion, Proc 8th International Symposium on Symbolic and Numeric Algorithms in Symbolic Computation (SYNASC), IEEE Computer Society, 2006. P. 151-155. URL: http://www.csd.uwo.ca/ watt/pub/reprints/2006-synasc-bminv.pdf.

6. *Malaschonok G.I.* Parallel Algorithms of Computer Algebra // Materials of the conference dedicated for the 75 years of the Mathematical and Physical Dep. of Tambov State University. (November 22-24, 2005). Tambov: TSU, 2005. P. 44-56.

GRATITUDES: Supported by the Sci. Program Devel. Sci. Potent. High. School, RNP 2.1.1.1853.

Accepted for publication 7.06.2010.

ПАРАЛЛЕЛЬНЫЙ АЛГОРИТМ СИМВОЛЬНОГО РЕШЕНИЯ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ С ЧАСТНЫМИ ПРОИЗВОДНЫМИ

© **Наталия Александровна Малашонок**
Тамбовский государственный университет им. Г.Р. Державина, Интернациональная, 33, Тамбов, 392000, Россия, кандидат физико-математических наук, доцент кафедры математического анализа, e-mail: malaschonok@ya.ru

Ключевые слова: параллельные алгоритмы, компьютерная алгебра, уравнения в частных производных, преобразование Лапласа–Карсона, условия согласованности.

Представлен параллельный алгоритм символьного решения системы уравнений с частными производными с помощью преобразования Лапласа–Карсона. Задача сводится к решению линейной алгебраической системы с полиномиальными коэффициентами, для которой существуют быстрые параллельные алгоритмы. это позволяет сконструировать быстрый параллельный алгоритм для систем дифференциальных уравнений с частными производными. Составной частью алгоритма является процедура получения условий согласованности для начальных условий.