Parallel Laplace Method with Assured Accuracy for Solutions of Differential Equations by Symbolic Computations

Natasha Malaschonok

Tambov State University,
Internatsionalnaya 33, 392622 Tambov, Russia
malaschonok@narod.ru

Abstract. We produce a parallel algorithm realizing the Laplace transform method for symbolic solution of differential equations. In this paper we consider systems of ordinary linear differential equations with constant coefficients, nonzero initial conditions, and the right-hand sides reduced to the sums of exponents with the polynomial coefficients.

1 Introduction

We produce a parallel algorithm applying the Laplace transform method to symbolic solution of differential equations.

An application of Laplace transform in differential equations theory in spite of its long history is topical. It has been very useful in classical or modified forms for solving ordinary or partial differential equations [2, 3, 8, 19, 21]. It is frequently applied for problems of fractional order equations [6, 20].

We consider systems of ordinary linear differential equations with constant coefficients, nonzero initial conditions, and right-hand sides as composite functions reducible to the sums of exponents with the polynomial coefficients. We place an emphasis on the symbolic character of computations. The efficient algorithmizataton of symbolic solution is achieved at several stages.

At the first stage, the preparation of data functions for the formal Laplace transform is performed (section 3). It is achieved by applying the Heaviside function and moving the obtained functions into the bounds of smoothness intervals. The parallelization of computations is realized as the multilevel tree, in the paper it is evident from the numeration of algorithm blocks.

The second stage is the parallel solution of the algebraic system with polynomial coefficients and the right-hand side obtained after the Laplace transform of the data system (section 4). There are parallel algorithms that are very efficient for solving this type of equations, and are different for various types of such systems.

At the third stage, the obtained solution of algebraic system is prepared to the inverse Laplace transform. It is reduced to the sum of partial fractions with exponential coefficients. One of the problems is calculation of roots of a polynomial. In [17, 18] the algorithm to determine the error of the roots sufficient for

V.G. Ganzha, E.W. Mayr, and E.V. Vorozhtsov (Eds.): CASC 2006, LNCS 4194, pp. 251–260, 2006.

the required accuracy of the data system solution is obtained. The solution of the algebraic system for reducing into the sum of partial fractions is performed by means of parallel algorithms cited in the paper.

At the last stage, the solution of the data system is produced (section 5). It is obtained as the real part of the inverse Laplace transform image of the algebraic system solution prepared previously.

In the last section, an example is considered.

2 Input Data

Denote by $x_j, j = 1, \ldots, n$, unknown functions of argument $t, t \geq 0$, x_j^k is the derivative of order k of the function x_j, $k = 0, \ldots, N$. As the right-hand sides of equations we consider here composite functions $f_l, l = 1, \ldots, n$ whose components are represented as finite sums of exponents with polynomial coefficients. So we have to solve the system

$$\sum_{j=1}^{n} \sum_{k=0}^{N} a_{kj}^l x_j^k = f_l, \quad l = 1, \ldots, n, \quad a_{kj}^l \in \mathbb{R}, \tag{1}$$

of n differential equations of order N with initial conditions $x_j^k(0) = x_{0j}^k, k = 0, \ldots, N-1$ with functions f_l reduced to the form

$$f_l(t) = f_l^i(t),\ t_l^i < t < t_l^{i+1},\ i = 1, \ldots, I_l, t_l^1 = 0, t_l^{I_l+1} = \infty, \tag{2}$$

where

$$f_l^i(t) = \sum_{s=1}^{S_l^i} P_{ls}^i(t) e^{b_{ls}^i t}, \quad i = 1, \ldots, I_l, \quad l = 1, \ldots, n,$$

and $P_{ls}^i(t) = \sum_{m=0}^{M_{ls}^i} c_{sm}^{li} t^m$.

Remark. The algorithm tree is exposed by multi-index numbering of blocks. For example, **Block** $42jk$, $k = 1, \ldots, K_\Theta$, denotes the vertex $42jk$, which is an entrance of the kth tree-edge, outgoing from the vertex $42j$ – **Block** $42j$, and there are K_Θ such edges. As k is the fourth index in the multi-index $42jk$, **Block** $42jk$ is the vertex of the fourth level. All the blocks **Block** $42jk$, $k = 1, \ldots, K_\Theta$ are performed independently and parallel.

Block1: Block10, Block1l, $l = 1, \ldots, n$. *Data file.*

Data file contains the coefficients a_{kj}^l, the initial conditions x_{0j}^k, $k = 0, \ldots, N-1$, $j = 1, \ldots, n$, and the right-hand sides $f_l, l = 1, \ldots, n$.

The data for functions f_l consists of the polynomial coefficients c_{sm}^{li}, parameters b_{ls}^i of exponents, the bounds t_i of smoothness intervals. Here $m = 0, \ldots, M_{ls}^i$, $s = 1, \ldots, S_l^i$, $i = 1, \ldots, I_l$. The numbers M_{ls}^i are degrees of corresponding polynomials, S_l^i are amounts of exponents in the expressions for f_l.

3 Laplace Transform

Denote the Laplace images of the functions $x_j(t)$ and $f_l(t)$ by $X_j(p)$ and $F_l(p)$, respectively.

The Laplace transform of the left-hand side of system (1) with respect to the initial conditions is performed by formal writing of the expression

$$\sum_{j=1}^{n}\sum_{k=0}^{N} a_{kj}^{l} p^{k} X_j(p) - \sum_{j=1}^{n}\sum_{k=0}^{N-1} x_{0j}^{k} p^{N-1-k},$$

starting directly from input data.

Block21*l*, $\boldsymbol{l = 1, \ldots, n}$. *Preparation of right-hand functions $f_l(t)$ to the Laplace transform .*

The functions $f_l(t), l = 1, \ldots, n$ are composite and reduced to form (2).

We use the Heaviside function $\eta(t)$ and present $f_l(t)$ as a sum

$$f_l(t) = \sum_{i=2}^{I_l-1} [f_l^i(t) - f_l^{i-1}(t)]\eta(t - t_l^i) + f_l^1(t)\eta(t).$$

Block21*li*, $\boldsymbol{i = 1, \ldots, I_l}$. *Transform into the function of $t - t_l^i$.*

Transform $f_l^i(t) - f_l^{i-1}(t)$ into the function of $t - t_l^i$:

$$f_l^i(t) - f_l^{i-1}(t) = \phi_l^i(t - t_l^i).$$

Generally, the functions $f_l^i(t) - f_l^{i-1}(t)$ are decomposed into power series at point t_l^i.

In our case the function $\phi_l^i(t - t_l^i)$ is represented as a finite sum

$$\phi_l^i(t - t_l^i) = \sum_{s=1}^{S_l^i} \psi_{ls}^i(t - t_l^i) e^{b_{ls}^i t_l^i} e^{b_{ls}^i (t - t_l^i)} - \sum_{s=1}^{S_l^{i-1}} \psi_{ls}^{i-1}(t - t_l^i) e^{b_{ls}^{i-1} t_l^i} e^{b_{ls}^{i-1}(t - t_l^i)}.$$

Here $\psi_{ls}^k(t - t_l^i) = P_{ls}^k(t)$ and $\psi_{ls}^k(t - t_l^i) = \sum_{m=0}^{M_{ls}^k} \gamma_{lsm}^{ki}(t - t_l^i)^m$. Coefficients γ_{lsm}^{ki} are calculated by the formula

$$\gamma_{lsm}^{ki} = \sum_{j=0}^{M_{ls}^k - m} c_{s,m+j}^{lk} \binom{m+j}{j} (t_l^i)^j.$$

Finally the function $f_l(t)$ is reduced to the form

$$f_l(t) = \sum_{i=2}^{I_l-1} \phi_l^i(t - t_l^i)\eta(t - t_l^i) + \sum_{s=1}^{S_l^1} P_{ls}^1(t) e^{b_{ls}^1 t}\eta(t).$$

Block22l, $l = 1, \ldots, n$. *The parallel Laplace transform of the functions $f_l(t)$.*

Since the Laplace image of $(t - t^*)^n e^{\alpha(t-t^*)}\eta(t - t^*)$ is $\frac{n!}{(p-\alpha)^{n+1}} e^{-t^* p}$ the Laplace transform of $\phi_l^i(t - t_l^i)\eta(t - t_l^i)$ equals

$$\Phi_l^i(p) = \left[\sum_{s=1}^{S_l^i} \sum_{m=0}^{M_{ls}^i} \gamma_{lsm}^{ii} e^{b_{ls}^i t_l^i} \frac{m!}{(p - b_{ls}^i)^{m+1}} \right.$$
$$\left. - \sum_{s=1}^{S^{i-1} i_l} \sum_{m=0}^{M_{ls}^{i-1}} \gamma_{lsm}^{i,\ i-1} e^{b_{ls}^{i-1} t_l^i} \frac{m!}{(p - b_{ls}^{i-1})^{m+1}} \right] e^{-t_l^i p}.$$

Finally, the Laplace transform of $f_l(t)$ is the following:

$$F_l(p) = \sum_{i=2}^{I_l - 1} \Phi_l^i(p) + \sum_{s=1}^{S_l^1} \sum_{m=0}^{M_{s_l}^1} c_{sm}^{l1} \frac{m!}{(p - b_{ls}^1)^{m+1}}. \tag{3}$$

In the case when the right-hand side of the given system is exposed in the form

$$f_l(t) = \sum_{s=1}^{S_l} \sum_{m=0}^{M_{s_l}} c_{sm}^l t^m e^{b_{ls} t}, \quad l = 1, \ldots, n,$$

the Laplace transform is performed formally – according to input data we present the expression for $F_l(p)$:

$$F_l(p) = \sum_{s=1}^{S_l} \sum_{m=0}^{M_{s_l}} c_{sm}^l \frac{m!}{(p - b_{ls})^{m+1}}, l = 1, \ldots, n. \tag{4}$$

For each $l = 1, \ldots, n$ we reduce (3)(or (4)) to the common denominator. The common denominator is left factorized. At that the nominator is the sum of exponents with polynomial coefficients.

In the case of a periodic function $f_l(t)$ with the period T, the respective denominator contains the expression $1 - e^{-pT}$. Then such fraction is expanded into power series.

4 Parallel Solution of Algebraic System

Block 31. *The construction of the algebraic system.*

As a result of the Laplace transform of system (1) we obtain an algebraic system relative to $X_j, j = 1, \ldots, n$:

$$\sum_{j=1}^{n} \sum_{k=0}^{N} a_{kj}^l p^k X_j(p) = \sum_{j=1}^{n} \sum_{k=0}^{N-1} x_{0j}^k p^{N-1-k} + F_l(p), \quad l = 1, \ldots, n. \tag{5}$$

For each $l = 1, \ldots, n$ the expressions on the right-hand side of (5) are reduced to the common denominator. The calculations are carried out in parallel. Denote

$$\Phi_l(p) = \sum_{j=1}^{n} \sum_{k=0}^{N-1} x_{0j}^k p^{N-1-k} + F_l(p).$$

We obtain the system

$$\sum_{j=1}^{n} \sum_{k=0}^{N} a_{kj}^l p^k X_j(p) = \Phi_l(p),\ l = 1, \ldots, n. \tag{6}$$

Block 32. *The parallel solution of the algebraic system.*

The system (6) may be solved by any possible classical method, for example, Cramer's method. But now there are new effective procedures for parallel computations, for example, p-adic method ([5], [9], [10]), modula method ([10] - [14]), the method based on determinant identities ([10] – [16]). The fastest method for solving such systems is p-adic method. But its code parallelization is not very effective. The best one for parallelization is the modula method based on Chinese Remainder Theorem.

5 Inverse Laplace Transform

Block41j, $j = 1, \ldots, n$. *Preparation of $X_j(p)$ to the inverse Laplace transform.*

Finally the solution of (6), i.e., each desired function $X_j(p),\ j = 1, \ldots, n$, is represented as a fraction with polynomial denominator $D_j(p)$. Note that $D_j(p)$ is partially factorized – it contains the multipliers of $F_l(p)$ denominators and the determinant $D(p)$ of system (6). The nominator is the sum of exponents with polynomial coefficients.

We reduce the function $X_j(p),\ j = 1, \ldots, n$, to the sum of exponents with fractional coefficients. The denominator of each fraction is $D_j(p)$, and the numerators are polynomials.

The next step is the decomposition of each fraction in the $X_j(p)$ expansion into the sum of partial fractions $A/(p - p^*)^v,\ p^* \in \mathbb{C}$. The first action here is the determination of the $D_j(p)$ roots.

Block42j, $j = 1, \ldots, n$. *Computation of the denominator roots.*

As it was pointed out the denominator $D_j(p)$ is already represented as a product of partial multipliers and the polynomial $D(p)$. So we have to find the roots of $D(p)$.

The accuracy of these calculations is determined first of all. Its value must be sufficient for the preassigned precision of system solution. An algorithm to compute such accuracy is written about in §5.

Block42jk, $k = 1, \ldots, K_\Theta$. *Decomposition into a sum of partial fractions.*

We decompose rational fractions or fractional coefficients of exponents into the sums of partial fractions $A/(p-p^*)^v,\ p^* \in \mathbb{C}$. The calculations for all fractions are in parallel, the number of blocks is formally denoted by K_Θ. It depends upon the parameters that we do not describe in detail here.

One step of the algorithm is the solution of a system of linear equations with constant coefficients. Depending on the size of system matrix we use one or another fast parallel algorithm, for example, modular method ([4] - [7], [10] - [16]).

If the roots of $D(p)$ have been found exactly, then we obtain the exact solution of the system (6) – the functions $X_j(p)$. Each of them is represented as a sum

$$X_j(p) = \sum_m \sum_k \frac{A_{mk}}{(p-p_{ik})^{\beta_{mk}}} e^{-\alpha_m p}. \tag{7}$$

Denote by $\Xi_j(p)$ the expression that represents $X_j(p)$ after its reduction to the partial fractions form in the case when the roots of $D(p)$ are calculated not exactly. Each $\Xi_j(p)$ is also written in the form (7).

Block43j, $j = 1, \ldots, n$. *Inverse Laplace transform.*

The Laplace originals of functions $X_j(p)$ are obtained formally – by writing the expressions

$$x_j(t) = \sum_m \sum_k \frac{A_{mk}}{(\beta_{mk}-1)!}(t-\alpha_m)^{\beta_{mk}-1} e^{p_{ik}(t-\alpha_m)} \eta(t-\alpha_m), j = 1, \ldots, n. \tag{6}$$

In the case when the roots of $D(p)$ are calculated not exactly denote by $\xi_j(t)$ the Laplace original of $\Xi_j(p)$. It is also written in the form (7). In general, the functions $\xi_j(t)$ are complex valued. We take the real part of $\xi_j(t)$ for each $j = 1, \ldots, n$. The functions $\mathrm{Re}\xi_j(t)$ may be taken as the solution of system (1), i.e., the required functions $x_j(t)$. It is easy to show that the error would not exceed the established precision assured by the calculated accuracy of roots of $D(p)$.

6 On Accuracy Estimation

We shall consider all functions and make calculations on the interval $[0, T]$, where T is sufficiently high for the input problem. Denote by $\tilde{x}_i(t)$ the approximate value of the solution $x_i(t)$. We require the following accuracy for the solutions on the interval T:

$$\max_{t\in[0,T]} |x_i(t) - \tilde{x}_i(t)| < \varepsilon,\ i = 1, \ldots, n.$$

We must determine an error Δ of the $D(p)$ roots sufficient for the required accuracy ε for $x_i(t)$. An algorithm for computation of Δ is produced in ([17] -

[18]). Notice only that Δ depends on the input parameters of the problem, T, the numbers $\mathcal{M}(f_i) = \max_{t\in[0,T]}|f_i(t)|, i = 1, \dots, n$, the appreciation δ of minimal distance between the roots of $D(p)$, the number σ such that the functions $X_j(p)$ are analytic in the domain $\mathrm{Re}p > \sigma$ for all $j = 1, \dots, n$.

7 Example

Block 10 $(a^1_{kj}) = \begin{pmatrix} 1\ 0\ -1\ -2 \\ 3\ 1\ -2\ 0 \end{pmatrix}; \quad (a^2_{kj}) = \begin{pmatrix} -1\ 0\ 0\ 1 \\ 1\ \ \ 0\ 0\ 1 \end{pmatrix};$
$x^0_{01} = 5, \quad x^1_{01} = 10, x^2_{01} = 30, \quad x^0_{02} = 4, \quad x^1_{02} = 14, \quad x^2_{02} = 20;$

Block 11 $f^1_1 = e^t, \quad f^2_1 = t^2e^{2t}, \quad t^1_1 = 0, \quad t^2_1 = 1;$

Block 12 $f^1_2 = te^t, \quad f^2_2 = e^{2t}, \quad t^1_2 = 0, \quad t^2_2 = 1;$

Block 211 $f_1 = (f^2_1 - f^1_1)\eta(t - t^2_1) + f^1_1\eta(t);$

Block 212 $f_2 = (f^2_2 - f^1_2)\eta(t - t^2_2) + f^1_2\eta(t);$

Block 221 $F_1 = \dfrac{1}{-1+p} - \dfrac{e^{1-p}}{-1+p} + \dfrac{e^{2-p}(p^2-2p+2)}{(-2+p)^3};$

Block 222 $F_2 = \dfrac{e^{2-p}}{-2+p} + \dfrac{1}{(-1+p)^2} - \dfrac{e^{1-p}p}{(-1+p)^2};$

Block 31

$$\begin{cases} -2X_1 - pX_1 + p^3X_1 + X_2 - p^3 = 10 - 4p - 4p^2 + 5(-1+p^2) \\ \qquad + \dfrac{1}{-1+p} - \dfrac{e^{1-p}}{-1+p} + \dfrac{e^{2-p}(p^2-2p+2)}{(-2+p)^3} \\ -2pX_1 + p^2X_1 + 3p^3X_1 + X_2 + p^3X_2 = 110 + 14p + 4p^2 + 10(1+3p) + \\ \qquad + (-2+p+3p^2) \\ \qquad + \dfrac{e^{2-p}}{-2+p} + \dfrac{1}{(-1+p)^2} - \dfrac{e^{1-p}p}{(-1+p)^2}; \end{cases}$$

$D(p) = -2 + p - p^2 - 4p^3 - 3p^4 + p^5 + 4p^6;$

Block 411

$$X_1(p) = e^{-p}\Big(\frac{8e+2e^2-4ep-4e^2p+2ep^2+2e^2p^2+9ep^3-6e^2p^3-19ep^4}{(-2+p)^3(-1+p)(-2+p-p^2-4p^3-3p^4+p^5+4p^6)}$$
$$+ \frac{12e^2p^4+11ep^5-8e^2p^5-2ep^6+2e^2p^6}{(-2+p)^3(-1+p)(-2+p-p^2-4p^3-3p^4+p^5+4p^6)}\Big)$$
$$+ \frac{-856+1692p-982p^2+1061p^3-1991p^4+1398p^5-412p^6+160p^7-95p^8+20p^9}{(-2+p)^3(-1+p)(-2+p-p^2-4p^3-3p^4+p^5+4p^6)};$$

Block 412

$$\begin{aligned} X_2(p) = e^{-p}\Big(& \frac{-8e^2-32ep+24e^2p+64ep^2-28e^2p^2-32ep^3+17e^2p^3-24ep^4-5e^2p^4}{(-2+p)^3(-1+p)^2(-2+p-p^2-4p^3-3p^4+p^5+4p^6)}+ \\ &+ \frac{34ep^5-3e^2p^5-14ep^6+5e^2p^6+2ep^7-2e^2p^7}{(-2+p)^3(-1+p)^2(-2+p-p^2-4p^3-3p^4+p^5+4p^6)}\Big)+ \\ &+ \frac{1776-4576p+3568p^2-1404p^3+2465p^4-2751p^5+841p^6+133p^7+2p^8-68p^9+16p^{10}}{(-2+p)^3(-1+p)^2(-2+p-p^2-4p^3-3p^4+p^5+4p^6)}; \end{aligned}$$

Block 421

$p_{x_1}{}^1 = 1, p_{x_1}{}^2 = -0.5949378 - 0.830714i,$
$p_{x_1}{}^3 = -0.5949378 + 0.830714i,$
$p_{x_1}{}^4 = 0.355937 - 0.513128i, p_{x_1}{}^5 = 0.355937 + 0.513128i, p_{x_1}{}^6 = 1,$
$p_{x_1}{}^7 = 1.228, p_{x_1}{}^8 = 2, p_{x_1}{}^9 = 2, p_{x_1}{}^{10} = 2,$

Block 422

$p_{x_2}{}^1 = -1, p_{x_2}{}^2 = -0.5949378 - 0.830714i, p_{x_2}{}^3 = -0.5949378 + 0.830714i,$
$p_{x_2}{}^4 = 0.355937 - 0.513128i, p_{x_2}{}^5 = 0.355937 + 0.513128i, p_{x_2}{}^6 = 1,$
$p_{x_2}{}^7 = 1, p_{x_2}{}^8 = 1.228, p_{x_2}{}^9 = 2, p_{x_2}{}^{10} = 2, p_{x_2}{}^{11} = 2;$

Block 431

$$x_1(t) = \begin{cases} \quad 10.031249e^{-t} - 1.25e^{t} + 5.538602e^{1.228001t}+ \\ + 2e^{0.355937t}(-3.735568\text{Cos}[0.513128t] + 15.529795\text{Sin}[0.513128t])+ \\ + 2e^{-0.594937t}(-0.924357\text{Cos}[0.830713t] + 0.061193\text{Sin}[0.830713t]), \\ \quad 0 < t < 1; \\ \quad 9.425260e^{-t} + 4.378584e^{1.228001t}+ \\ + 0.322878e^{2t} - 0.185554e^{2t}t + 0.0441176e^{2t}t^2+ \\ + 2e^{0.355937t}(-3.708886\text{Cos}[0.513128t] + 15.104078\text{Sin}[0.513128t])+ \\ + 2e^{-0.594938t}(-0.953417\text{Cos}[0.830713t] + 0.057591\text{Sin}[0.830713t]), \\ \quad t > 1; \end{cases}$$

Block 432

$$x_2(t) = \begin{cases} \quad 10.03125e^{-t} + 0.5e^{t} - 8.948223e^{1.228001t} + 0.5e^{t}t+ \\ + 2e^{0.355937t}(-0.493116\text{Cos}[0.513128t] + 33.959275\text{Sin}[0.513128t])+ \\ + 2e^{-0.594938t}(1.701602\text{Cos}[0.830713t] + 0.929609\text{Sin}[0.830713t]), \\ \quad 0 < t < 1; \\ \quad 9.42526e^{-t} - 7.074087e^{1.228001t}- \\ - 0.577176e^{2t} + 0.435986e^{2t}t - 0.117647e^{2t}t^2+ \\ + 2e^{0.355937t}(-0.636666\text{Cos}[0.513128t] + 33.06373\text{Sin}[0.513128t])+ \\ + 2e^{-0.594938t}(1.748891\text{Cos}[0.830714t] + 0.968596\text{Sin}[0.830714t]), \\ \quad t > 1; \end{cases}$$

The table gives the values of Δ for three values of ε and three values of T.

$T\backslash\varepsilon$	$\varepsilon = 0.1$	$\varepsilon = 0.01$	$\varepsilon = 0.001$
$T = 2$	$1.37 \cdot 10^{-10}$	$1.37 \cdot 10^{-11}$	$1.37 \cdot 10^{-12}$
$T = 3$	$1.25 \cdot 10^{-12}$	$1.25 \cdot 10^{-13}$	$1.25 \cdot 10^{-14}$
$T = 4$	$5.93 \cdot 10^{-15}$	$6.10 \cdot 10^{-16}$	$5.55 \cdot 10^{-17}$

The work is partially supported by the grant RFBR 04-07-90268b.

References

1. Akritas, A.G.: Elements of Computer Algebra with Applications. J. Wiley Interscience, New York, 1989
2. Burghelea, D., Haller, S.: Laplace transform, dynamics and spectral geometry. arXiv:math.DG/0405037v2, 17 Jan. (2005)
3. Dahiya, R.S., Jabar Saberi-Nadjafi: Theorems on n-dimensional Laplace transforms and their applications. In: 15th Annual Conf. of Applied Math., Univ. of Central Oklahoma, Electr. Journ. of Differential Equations, Conf.02 (1999) 61–74
4. Davenport, J., Siret, Y., Tournier, E.: Calcul formel. Systèmes et algorithmes de manipulations algébriques. MASSON, Paris, 1987
5. Dixon, J.: Exact solution of linear equations using p-adic expansions. Numer. Math. **40** (1982) 137–141
6. Felber, F.S.: New exact solutions of differential equations derived by fractional calculus. arXiv:math/0508157v1, 9 Aug. (2005)
7. Von zur Gathen, J., Gerhard, J.: Modern Computer Algebra. Cambridge University Press, 1999
8. Goursat E.: Sur les équations linéaires et la méthode de Laplace. Amer. J. Math. **18** (1896) 347–385
9. Malaschonok, G.I.: Solution of systems of linear equations by the p-adic method. Programming and Computer Software **29** (2) (2003) 59–71
10. Malaschonok, G.I.: Matrix Computational Methods in Commutative Rings. Tambov, TSU, 2002
11. Malaschonok, G.I.: Effective Matrix Methods in Commutative Domains. In: Formal Power Series and Algebraic Combinatorics, Springer, Berlin (2000) 506–517
12. Malaschonok, G.I.: Recursive method for the solution of systems of linear equations. In: Computational Mathematics (15th IMACS World Congress Vol. I, Berlin, August 1997), Wissenschaft und Technik Verlag, Berlin (1997) 475–480
13. Malaschonok, G.I.: Algorithms for computing determinants in commutative rings. Discrete Math. Appl. **5** (6) (1996) 557–566
14. Malaschonok, G.I.: Algorithms for the solution of systems of linear equations in commutative rings. In: Effective Methods in Algebraic Geometry. Progr. Math. V. 94, Birkhauser, Boston (1991) 289–298
15. Malaschonok, G.I.: Solution of a system of linear equations in an integral domain. USSR J. Comput. Math. and Math. Phys. **23** (6) (1983) 497–1500
16. Malaschonok, G.I.: The solution of a system of linear equations over a commutative ring. Math. Notes **42**, Nos. 3–4 (1987) 801–804
17. Malaschonok, N.: An algorithm to settle the necessary exactness in Laplace transform method. In: Computer Science and Information Technologies. Yerevan (2005) 453–456

18. Malaschonok, N.: Estimations in Laplace transform method for solutions of differential equations in symbolic computations. In: Differential Equations and Computer Algebra Systems. Minsk (2005) 195–199
19. Mizutani, N., Yamada H.: Modified Laplace transformation method and its applications to anharmonic oscillator. 12 Feb. (1997)
20. Podlubny, I.: The Laplace transform method for linear differential equations of the fractional order. arXiv:funct-an/9710005v1, 30 Oct. (1997)
21. Roux, J. Le.: Extensions de la méthodes de Laplace aux équatons linéaires aux deriveées partielles d'ordre supérieur du second. Bull. Soc. Math. de France **27** (1899) 237–262