2017
Volume 6
issue 4

36

MODERN TAYLOR SERIES METHOD IN NUMERICAL INTEGRATION: PART 1

36.1 INTRODUCTION

The paper deals with the solution of technical initial value problems (IVPs) representing the problems which arise from common technical practice (especially from electrical and mechanical engineering). To solve technical IVPs means to find the numerical solution of the system of ordinary differential equations (ODEs).

The best-known and most accurate method of calculating a new value of the numerical solution of ODE [7]:

$$y' = f(t, y), \qquad y(t_0) = y_0\,, \tag{36.1}$$

is to construct the Taylor series in the form

$$y_{n+1} = y_n + h \cdot f(t_n, y_n) + \frac{h^2}{2!} \cdot f'(t_n, y_n) + \ldots + \frac{h^p}{p!} \cdot f^{(p-1)}(t_n, y_n), \tag{36.2}$$

where h is the integration step.

The Taylor series can be very effictively implemented as the variable step/variable order numerical method (see TKSL software [14]). The Modern Taylor series method (MTSM) [10] is based on a recurrent calculation of the Taylor series terms for each time step (h). Therefore, the complicated calculation of higher order derivatives (much criticised in the literature) does not need to be performed, but rather the value of each Taylor series term is numerically calculated [10]. Equation (36.2) can then be rewritten in the form (36.3).

$$y_{n+1} = DY0 + DY1 + DY2 + \cdots + DYp \tag{36.3}$$

Theoretically, it is possible to compute the solution of homogeneous linear differential equations with constant coefficients with arbitrary order (use the arbitrary number of

Taylor series terms) with arbitrary accuracy. The resulting system of linear equations can be effectively solved either sequentially or in parallel.

An important part of the method is an automatic integration order setting ($ORD = p$), i.e. using as many Taylor series terms as the defined accuracy requires. Thus it is common that the computation uses different numbers of Taylor series terms for different time steps of constant length.

The following paper is divided into several sections, which consider concrete technical IVPs and effective usage of MTSM. In Section 36.2, the numerical solution of a partial differential equation (wave equation) using higher order space approximations schemes and combination of higher order MTSM is shown. The next section is focused on the multiple integral computation using transformation to ODEs, and then MTSM is effectively used for high-accurate computation.

Several papers focus on computer implementations of the Taylor series method in a variable order and variable step context (see, for instance TIDES software [3], implemented in [16], TAYLOR [8] including detailed description of a variable step size version, ATOMF [6], COSY INFINITY [4], DAETS [13]. The variable-stepsize variable-order scheme is also described in [2] and [1], where simulations on a parallel computer are shown.

36.2 WAVE EQUATION

The wave equation (36.4) is an important hyperbolic partial differential equation (PDE), and it is widely used in many technical problems (vibration of the string, AC circuits, electromagnetism, etc.).

$$\frac{\partial^2 y(x,t)}{\partial x^2} = \frac{\partial^2 y(x,t)}{\partial t^2} \tag{36.4}$$

Let Ω be a domain of PDE $(x,t) = \langle 0,\pi\rangle \times \langle 0, t_{max}\rangle$ and the homogeneous Dirichlet boundary conditions on $\partial\Omega$ are defined as follows:

$$y(0,t) = 0\,, \quad y(\pi,t) = 0\,. \tag{36.5}$$

where t_{max} is the maximum simulation time. The Cauchy initial values follow:

$$y(x,0) = \sin(x)\,, \quad \frac{\partial y(x,0)}{\partial t} = 0\,. \tag{36.6}$$

The wave equation describes a vibration of an ideal string of specified length. The string is fixed at both ends to x-axis (see boundary values (36.5)). The string is released at time $t = 0$, and the initial velocity of the string is zero. The sine wave is used as a model of the string. The analytical solution is in the form:

$$y = \cos(\pi t)\sin(x)\,. \tag{36.7}$$

The numerical methods for solving PDEs are based on approximations of the derivatives by differences. The method of lines is the technique which uses finite differences for the approximation of derivatives with respect to all the independent variables except one.

Using this method, it is possible to transform PDE into the system of ODEs, which can be solved using MTSM with high accuracy [9, 15]. The following subsections of the paper show the construction of a higher order difference formulae directly using Taylor series terms.

36.2.1 Types of Difference Formulae

Central difference formula uses the same number of points to the left and to the right of the current point of the calculation. To calculate the derivatives at point y_k, points y_{k-2}, y_{k-1}, y_k, y_{k+1} and y_{k+2} are used, and the steps sizes $-2h, -h$, $h, 2h$ are used (see Fig. 36.1). In this case $y_k = y_2$. Let l be a number of points to the left of the current point of the calculation and r be a number of points to the right. Hence, the respective setting is $l = 2$ and $r = 2$.

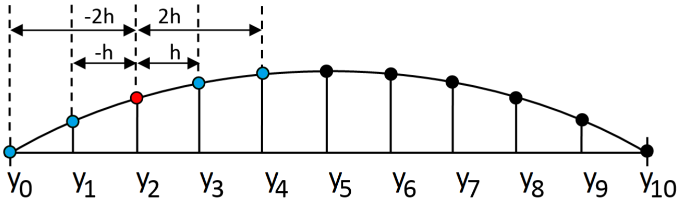

Fig. 36.1 Central difference formula for $y_k = y_2$

Source: own elaboration

Taylor series (36.8), (36.9), (36.10) and (36.11) for the neighbouring point y_k are as follows:

$$\begin{aligned}
y_{k-2} &= y_k + (-2h)y'_k + \frac{(-2h)^2}{2!}y''_k + \frac{(-2h)^3}{3!}y'''_k + \frac{(-2h)^4}{4!}y''''_k, && (36.8)\\
y_{k-1} &= y_k + (-h)y'_k + \frac{(-h)^2}{2!}y''_k + \frac{(-h)^3}{3!}y'''_k + \frac{(-h)^4}{4!}y''''_k, && (36.9)\\
y_{k+1} &= y_k + hy'_k + \frac{h^2}{2!}y''_k + \frac{h^3}{3!}y'''_k + \frac{h^4}{4!}y''''_k, && (36.10)\\
y_{k+2} &= y_k + 2hy'_k + \frac{(2h)^2}{2!}y''_k + \frac{(2h)^3}{3!}y'''_k + \frac{(2h)^4}{4!}y''''_k. && (36.11)
\end{aligned}$$

The system of equations can be expressed in matrix form (36.12). Subsequently, we obtain the first four terms of the Taylor series for point y_k (denoted as $DY1, DY2, DY3$ and $DY4$).

$$\begin{pmatrix} DY1 \\ DY2 \\ DY3 \\ DY4 \end{pmatrix} = \begin{pmatrix} -2 & (-2)^2 & (-2)^3 & (-2)^4 \\ -1 & (-1)^2 & (-1)^3 & (-1)^4 \\ 1 & 1^2 & 1^3 & 1^4 \\ 2 & 2^2 & 2^3 & 2^4 \end{pmatrix}^{-1} \cdot \begin{pmatrix} y_{k-2} - y_k \\ y_{k-1} - y_k \\ y_{k+1} - y_k \\ y_{k+2} - y_k \end{pmatrix} \qquad (36.12)$$

Forward difference formula uses only the points to the right of the current point. Hence, for the calculation of the derivatives at point y_k, we take points $y_{k+1}, y_{k+2}, y_{k+3}$ and y_{k+4}. Only positive step sizes $h, 2h, 3h, 4h$ are used in this case. The setting for five-point approximation is $l = 0$, $r = 4$. *Backward difference formula* can be constructed similarly, with the setting being $l = 4$, $r = 0$. More information can be found in [9] and [11].

36.2.2 Accuracy of the Calculation

Firstly, we discuss the accuracy of the calculation in the space domain and then in the time domain. Let us denote n-point approximation as N and the number of segments of the string as S.

There are two parameters affecting the *space domain* accuracy. The first parameter is the space-step $h(S)$. The second parameter is the order of the difference formulae N.

The absolute error between numerical and analytical solution for the symmetric formulae in space domain is shown in Fig. 36.2 (left). The upper red function shows the absolute error for $N = 5$ segments and $S = 32$ ($h = \pi/S$), where h is a step in space domain. The lower blue function shows this error for $N = 9$ and $S = 315$ ($h = \pi/S$). Note that these functions mostly remain at the same level. Higher deviation between the numerical and analytical solution at the boundary of space domain is caused by using the asymmetric difference formulae.

The absolute error for the forward difference formulae is shown in Fig. 36.2 (right). The settings are the same as the settings for the central difference formulae.

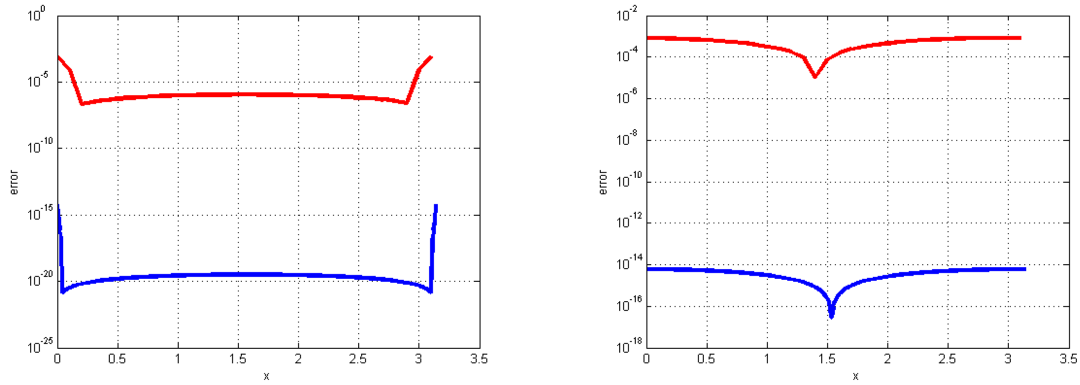

Fig. 36.2 Central (left) and forward (right) difference formulae (space domain)

Source: own elaboration

The absolute error in the *time domain* is shown in Fig. 36.3. The upper red function shows the absolute error for $N = 3$ and $S = 10$, the lower red function shows the absolute error for $N = 3$ and $S = 100$. The error can be effectively decreased by an increase in the order of the difference formula. The upper blue function shows the error for $N = 5$ and $S = 12$. Furthermore, the lower blue function shows the error for $N = 5$ and $S = 100$.

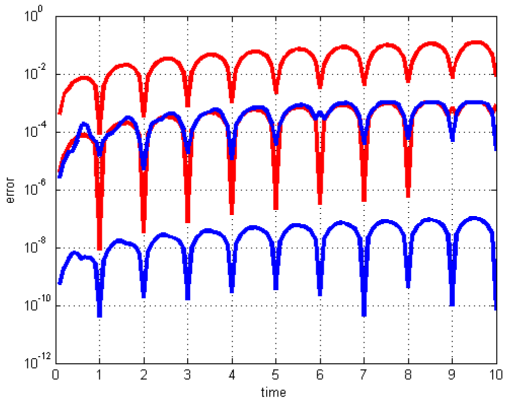

Fig. 36.3 The absolute error between numerical and analytical solution (time domain)

Source: own elaboration

36.3 MULTIPLE INTEGRAL COMPUTATION

In this chapter, we transform the solution of definite integral $F(x) = \int_A^B f(x)\,\mathrm{d}x$ into the ordinary differential equation

$$F'(x) = f(x)\,, \tag{36.13}$$

with initial condition $F(A) = 0$.

The IVP (36.13) can be solved using MTSM very effectively [5, 12].

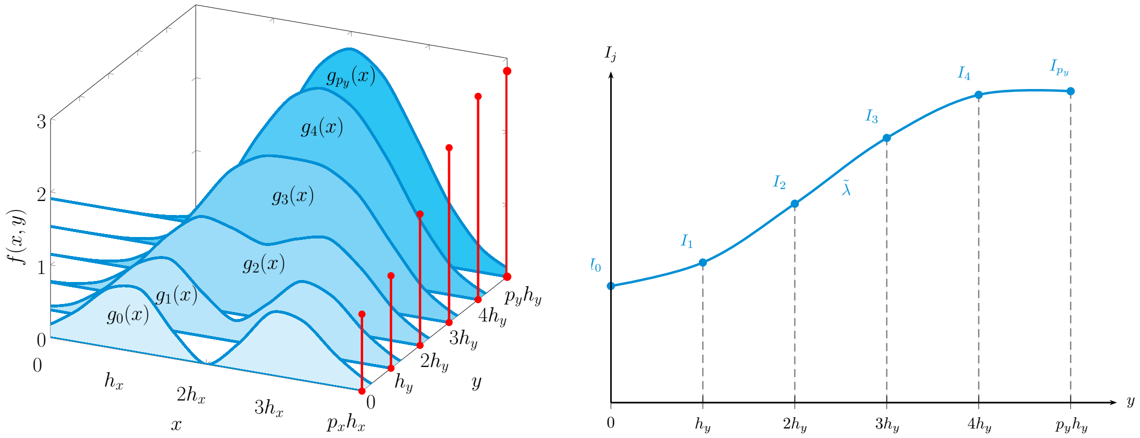

Fig. 36.4 Sampled integrals as a new function $I_j = \Psi(jh_y)$

Source: own elaboration

36.3.1 Sampled Integrals

Given $I_j(x)$ is a numerical solution of $\int_0^a f(x, jh_y)dx$ (all definite integral can be transformed into equivalent integrals with zero lower bound) and p_y a number of samples in y axe, we can approximate a double integral with a sequence of approximated values of a single definite integral computed precisely by MTSM (see Fig. 36.4).

Using ψ (approximated by values $I_0, I_1, \ldots, I_{p_y}$), we can approximate *double integral*

$$\begin{aligned} e(h_y) &= e(0) + \sum_{m=1}^{n} \frac{\psi^{(m-1)}(0)}{m!} h_y^m \\ &\vdots \\ e(jh_y) &= e((j-1)h_y) + \sum_{m=1}^{n} \frac{\psi^{(m-1)}((j-1)h_y)}{m!} h_y^m \\ &\vdots \\ e(p_y h_y) &= e((p_y-1)h_y) + \sum_{m=1}^{n} \frac{\psi^{(m-1)}((p_y-1)h_y)}{m!} h_y^m \,. \end{aligned} \tag{36.14}$$

However, in this case we do not know the derivatives of ψ, so we must take a different approach. We construct the following system:

$$\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}\,, \tag{36.15}$$

$$\text{where } \boldsymbol{A} = \begin{pmatrix} 1 & 1^2 & \ldots & 1^n \\ & \vdots & & \\ j & j^2 & \ldots & j^n \\ & \vdots & & \\ n & n^2 & \ldots & n^n \end{pmatrix}, \ \boldsymbol{b} = \begin{pmatrix} I_1 - I_0 \\ \vdots \\ I_j - I_0 \\ \vdots \\ I_n - I_0 \end{pmatrix}, \text{ and } \quad \boldsymbol{x} = \begin{pmatrix} D\psi(1,0) \\ D\psi(2,0) \\ \ldots \\ D\psi(j,0) \\ \ldots \\ D\psi(n,0) \end{pmatrix}.$$

Solving a vector $\boldsymbol{x}$, we get $D\psi(j,0)$, which can now be used to solve $e(h_y)$ in equations (36.14). We can calculate the remaining $e(jh_y)$ analogically, and the matrix $\boldsymbol{A}$ stays the same, we only need to recalculate the vector $\boldsymbol{b}$.

36.3.2 Multiple Integral

Generally, a multiple integral in the form

$$\int_0^{a_n} \cdots \int_0^{a_1} g(x_1, \ldots, x_n)\, \mathrm{d}x_1 \ldots \mathrm{d}x_n \tag{36.16}$$

can be solved numerically the same way as in the case of the double integral by iteratively reducing integral's dimension. Numerical derivates (of a function f in general) are computed analogically from sampled values $f_{-l}, \ldots, f_{-1}, f_0, f_1, \ldots, f_r$ at given times $-lh, \ldots, -h, 0, h, \ldots, rh$, $n = r + l + 1$. Using Taylor's polynomials, we can construct the following set of equations::

$$\forall i \in \{-l, -l+1, \ldots, -1, 0, 1, \ldots, r-1, r\} :$$

$$f_i = f_0 + \sum_{m=1}^{n} \frac{i^m h^m}{m!} f^{(m)}(0) \tag{36.17}$$

Expressing the system in the matrix form, we can compute function's derivatives and use them to reduce integral's dimension.

36.3.3 Integration of an Exponential Function

The method was tested on the computation of the double integral of an exponential function, i.e. $\int_0^2 \int_0^2 e^{x+y} dydx$. The tests were run for different lengths of an integration step, number of Taylor's terms, and arithmetic precision. The precision is determined by a number of bits used for a mantissa of each number. The numerical solution was compared with the analytical solution of the double integral. The error of computation for each test instance is shown in Tab. 36.1. For the fixed integration step and precision, only a number of Taylor's term with the smallest error is shown.

Tab. 36.1 Error of computation of $\int_0^2 \int_0^2 e^{x+y} dydx$

step	**no. of terms (ORD)**	**precision [bits]**	**error**
0.2	11	400	1.13e-06
0.1	19	400	8.95e-20
0.05	39	400	6.88e-52
0.04	49	400	9.14e-70
0.02	99	800	8.02e-170
0.01	104	900	8.09e-208
0.005	94	900	4.59e-217
0.004	91	900	4.13e-220
0.002	85	900	3.49e-227

Source: own elaboration

CONCLUSION

MTSM as is implemented in TKSL software provides very accurate and stable solution to the wide class of technical IVPs The next paper (Part 2) will be focused on the effective parallel implementation and stiff systems solution using MTSM. The research will be focused on the concrete engineering problems such as modelling of electric circuits, closed-loop systems, fluid engineering etc.

ACKNOWLEDGEMENTS

This research was financially supported by the the Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science - LQ1602". The paper also includes the results of the international AKTION research project Number 76p11 and the internal BUT FIT project FIT-S-17-4014.

REFERENCES

1. R. Barrio. Performance of the Taylor series method for ODEs/DAEs. In *Applied Mathematics and Computation*, volume 163, pages 525–545, 2005. ISSN 00963003.
2. R. Barrio, F. Blesa, and M. Lara. VSVO Formulation of the Taylor Method for the Numerical Solution of ODEs. In *Computers and Mathematics with Applications*, volume 50, pages 93–111, 2005.
3. R. Barrio, M. Rodríguez, A. Abad, and F. Blesa. TIDES: A free software based on the Taylor series method. *Monografías de la Real Academia de Ciencias de Zaragoza*, 35:83–95, 2011.
4. M. Berz. COSY INFINITY version 8 reference manual. *Technical Report MSUCL–1088, National Superconducting Cyclotron Lab., Michigan State University, East Lansing, Mich.*, 1997.
5. J. Chaloupka, F. Kocina, P. Veigend, G. Nečasová, V. Šátek, and J. Kunovský. Multiple integral computations. In *14th International Conference of Numerical Analysis and Applied Mathematics*, 2016.
6. Y. F. Chang and G. Corliss. Atomf: solving odesand daes using Taylor series. *Computers Math. Applic.*, 28:209–233, 1994.
7. E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I.* vol. Nonstiff Problems. Springer-Verlag Berlin Heidelberg, 1987. ISBN 3-540-56670-8.
8. A. Jorba and M. Zou. A software package for the numerical integration of ODE by means of high-order Taylor methods. In *Exp. Math.*, volume 14, pages 99–117, 2005.
9. Filip Kocina, Jiří Kunovský, Martin Marek, Gabriela Nečasová, Alexander Schirrer, and Václav Šátek. New trends in Taylor series based computations. In *12th International Conference of Numerical Analysis and Applied Mathematics*, number 1648. American Institute of Physics, 2014.
10. J. Kunovský. *Modern Taylor Series Method.* FEI-VUT Brno, 1994. Habilitation work.
11. Jiří Kunovský, Václav Šátek, Gabriela Nečasová, Petr Veigend, and Filip Kocina. The Positive Properties of Taylor Series Method. In *Proceedings of the 13th International*

Conference Informatics' 2015, pages 156–160. Institute of Electrical and Electronics Engineers, 2015.

12. A. Martinkovičová, J. Chaloupka, J. Kunovský, V. Šátek, and P. Veigend. Numerical integration of multiple integrals using Taylor's polynomial. In *Proceedings of the 5th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, pages 163–171. SciTePress - Science and Technology Publications, 2015.
13. Nedialko S. Nedialkov and J. Pryce. Solving differential algebraic equations by Taylor series III. the DAETS code. *JNAIAM J. Numer. Anal. Ind. Appl. Math.*, 3:61–80, 2008.
14. TKSL software. *High Performance Computing*, [cit. 21-3-2017]. URL: http://www.fit.vutbr.cz/ kunovsky/TKSL/index.html.en [online].
15. Václav Valenta, Gabriela Nečasová, Jiří Kunovský, Václav Šátek, and Filip Kocina. Adaptive solution of the wave equation. In *Proceedings of the 5th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, pages 154–162. SciTePress - Science and Technology Publications, 2015.
16. Wolfram. *MATHEMATICA software*, [cit. 21-3-2017].

MODERN TAYLOR SERIES METHOD IN NUMERICAL INTEGRATION

Abstract: *The paper deals with extremely exact, stable, and fast numerical solutions of systems of differential equations. It also involves solutions of problems that can be reduced to solving a system of differential equations. The approach is based on an original mathematical method, which uses the Taylor series method for solving differential equations in a non-traditional way. Even though this method is not much preferred in the literature, experimental calculations have verified that the accuracy and stability of the Taylor series method exceed the currently used algorithms for numerically solving differential equations. The Modern Taylor Series Method (MTSM) is based on a recurrent calculation of the Taylor series terms for each time interval. Thus, the complicated calculation of higher order derivatives (much criticised in the literature) need not be performed but rather the value of each Taylor series term is numerically calculated. An important part of the method is an automatic integration order setting, i.e. using as many Taylor series terms as the defined accuracy requires. The aim of our research is to propose the extremely exact, stable, and fast numerical solver for modelling technical initial value problems that offers wide applications in many engineering areas including modelling of electrical circuits, mechanics of rigid bodies, control loop feedback (controllers), etc.*

Keywords: *Taylor series method; ordinary differential equations; technical initial value problems.*

MODERNÍ METODA TAYLOROVY ŘADY V NUMERICKÉ INTEGRACI

Abstrakt: *Článek se zabývá přesným, stabilním a rychlým řešením soustav diferenciálních rovnic. Soustavou diferenciálních rovnic lze reprezentovat velké množství reálných problémů. Numerické řešení je založeno na unikátní numerické metodě, která netradičně využívá Taylorovu řadu. I přesto, že tato metoda není v literatuře příliš preferována, experimentální výpočty potvrdily, že přesnost a stabilita této metody přesahuje aktuálně používané numerické algoritmy pro numerické řešení diferenciálních rovnic. Moderní metoda Taylorovy řady je založena na rekurentním výpočtu členů Taylorovy řady v každém časovém intervalu. Derivace vyšších řádů nejsou pro výpočet přímo využity, derivace jsou zahrnuty do členů Taylorovy řady, které se počítají rekurentně numericky. Důležitou vlastností metody je automatická volba řádu metody v závislosti na velikosti integračního kroku, tzn. je využito tolik členů Taylorovy řady, kolik vyžaduje zadaná přesnost výpočtu. Cílem výzkumu je navrhnout velmi přesný, stabilní a rychlý nástroj pro modelování technických počátečních problémů využitých v praxi při modelování elektrických obvodů, mechaniky tuhých těles, problematiky zpětnovazebního řízení a další..*

Klíčová slova: *metoda Taylorovy řady; obyčejné diferenciální rovnice; technické počáteční úlohy.*

Date of submission of the article to the Editor: 04.2017
Date of acceptance of the article by the Editor: 05.2017

Ing. Jan Chaloupka(*); Ing. Gabriela Nečasová(*); Ing. Petr Veigend(*);
Doc. Ing. Jiří Kunovský, CSc.(*); Ing. Václav Šátek, Ph.D.(*),(**),
(*)Brno University of Technology, Faculty of Information Technology
Božetěchova 2, 612 66
Brno, Czech Republic
email: ichaloupka@fit.vut.cz; inecasova@fit.vut.cz; iveigend@fit.vut.cz;
kunovsky@fit.vut.cz; satek@fit.vut.cz
(**)IT4Innovations, VŠB Technical University of Ostrava
17. listopadu 15/2172, 708 33
Ostrava-Poruba, Czech Republic