Stability and Convergence of Numerical Computations

Pavla Sehnalová*
Department of Intelligent Systems
Faculty of Information Technology
Brno University of Technology
Božetěchova 2, 621 66 Brno, Czech Republic
isehnala@fit.vutbr.cz

Abstract

The stability and convergence of fundamental numerical methods for solving ordinary differential equations are presented. These include one-step methods such as the classical Euler method, Runge–Kutta methods and the less well known but fast and accurate Taylor series method. We also consider the generalization to multistep methods such as Adams methods and their implementation as predictor–corrector pairs. Furthermore we consider the generalization to multiderivative methods such as Obreshkov method. There is always a choice in predictor-corrector pairs of the so-called mode of the method and in this thesis both PEC and PECE modes are considered.

The aim of the paper is the use of a special fourth order method consisting of a two-step predictor followed by an one-step corrector, each using second derivative formulae and the convergence and stability analysis for the new method with constant stepsize for various problems as well as to investigate and to compare the convergence and stability analysis for selected numerical methods. Experiments for linear and non-linear problems and the comparison with classical methods are presented.

Categories and Subject Descriptors

G.1.7 [**Mathematics of Computing**]: Numerical Analysis—*Ordinary Differential Equations, Convergence and stability*

Keywords

Ordinary differential equations, stability, convergence, one-step methods, Runge–Kutta methods, Taylor series method, linear multistep methods, Adams methods, predictor-corrector pairs, Obreshkov method.

*Recommended by thesis supervisor: Assoc. Prof. Jiří Kunovský. Defended at Faculty of Information Technology, Brno University of Technology on September 27, 2011.

Sehnalová, P. Stability and Convergence of Numerical Computations. Information Sciences and Technologies Bulletin of the ACM Slovakia, Vol. 3, No. 3 (2011) 26-35

1. Introduction

Universal computational systems and equipments solve these kinds of special algorithms and problems in less shorter time that in former centuries. One of these problems is the numerical solution of differential equations. Each simulation system includes different type of numerical computations. To summarize numerical methods is very demanding task in terms of extensiveness. Therefore the thesis is focused on non-stiff problems described by ordinary differential equations and their solutions using numerical methods.

Classic application of differential equations is found in many areas of science and technology. They can be used for modelling of physical, technical or biological processes such as in the study of an electric circuit consisting of a resistor, an inductor and a capacitor driven by an electromotive force, in gravitational equilibrium of a star, chemical reactions kinetic, in the psychology, in models of the learning of a task involves the equation, in vibrating strings and propagation of waves, etc. [15, 21]. Main questions of modern technology are how to increase the accuracy of calculations considering short computational time, how to decrease necessary mathematical operations and all these questions have many aspects and criterion, which we need to explore to get the suitable answer.

2. Ordinary differential equations and one-step methods

Ordinary differential equation (ODE) of first order obtains a single independent variable and one or more its derivatives with respect to that variable [3]. The equation is given in the form

$$y'(x) = f(x, y(x)), \tag{1}$$
$$y(x_0) = y_0, \tag{2}$$

where $y'(x) = \frac{dy}{dx}$, x is independent variable, y is dependent variable. A function $y(x)$ is called a solution of equation (1) and the initial value (2) is given.

A second order ODE for y is, under mild assumptions for (1) together with (2), given in the form

$$y'' = f(x, y, y'), \tag{3}$$

with two free parameters which represent two uniquely determined initial values

$$y(x_0) = y_0, \qquad y'(x_0) = y'_0.$$

Generally, an order n ODE in x with $y^{(n)}$ has the explicit form

$$y^{(n)} = f(x, y, y', \cdots, y^{(n-1)}), \tag{4}$$

there is a unique solution with n initial values

$$y(x_0) = y_0,\ y'(x_0) = y'_0,\ \cdots,\ y^{(n-1)}(x_0) = y_0^{(n-1)}. \tag{5}$$

To solve the ordinary differential equations we need to ask how we can solve them. We are also interested in a question if a differential equation has more than one solution. Here we talk about the uniqueness of the solution. If it has at least one solution we need to find a solution which satisfies particular conditions. The answer testifies about the existence of the solution. And we try to discover which method should we use for solving the differential equation to get the accurate result in a suitable time. There are other fundamentals which need to be presented. But only in a way to understand the described methods and generalizations. Generally, the mathematical background is very extensive and described in many other books.

The convergence is the point of the interest together with the stability. The attribute of convergence guaranties the solution reaches the exact solution after few steps of calculation. The stability and convergence determine the consistency of the method [9].

2.1 Analytical solution and the example

Many of ordinary differential equations of arbitrary order can be solved *analytically*. In the most of cases it is very complicated and time-consuming problem.

Description of circuits using differential equations is very convenient for the electrical circuits' behavior analysis [20]. Electrical circuits are described by differential equations for time-dependent elements (capacitors, inductances) together with equations for linear and non-linear time-independent elements (resistors, diodes and transistors). Well-known Ohm's law and Kirchhoff's laws are part of the electronic circuit description.

Assume the differential equation of second order (6) describing the electrical circuit in the figure 1. We assume $y' = dy/dt$ for this example.

$$LCu''_C + RCu'_C + u_C = u, \qquad u_C(0) = 0,\ u'_C(0) = 0 \tag{6}$$

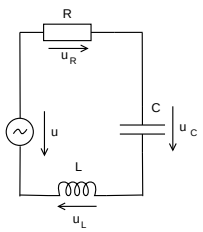

Figure 1: Electrical circuit with serial resistor, capacitor and inductor

The homogeneous equation is transferred to the characteristic equation and solved as a quadratic equation in the first step

$$LC\lambda^2 + RC\lambda + 1 = 0$$
$$\lambda_{1,2} = -\frac{RC \mp \sqrt{(RC)^2 - 4LC}}{2LC}.$$

There are three possible choices of the expected eigenvalues according to the value of the determinant $D = (RC)^2 - 4LC$

1. $D > 0 \longrightarrow \lambda_1 \neq \lambda_2 \in \mathrm{Re}$,
2. $D = 0 \longrightarrow \lambda_1 = \lambda_2 \in \mathrm{Re}$,
3. $D < 0 \longrightarrow \lambda_{1,2} = a \pm ib \in \mathrm{Im}$

and due to three possible homogeneous solutions $y_h = u_{Ch}$ are expected

1. $u_{Ch} = C_1 e^{\lambda_1 t} + C_2 e^{\lambda_2 t}$,
2. $u_{Ch} = e^{\lambda t}\Big(C_1 t + C_2\Big)$,
3. $u_{Ch} = e^{at}\Big(C_1 \cos(bt) + C_2 \sin(bt)\Big)$.

In this example we assume the multiple root ($\lambda_1 = \lambda_2 = -R/2L$), so the expected solution for the circuit is

$$u_{Ch} = e^{\lambda t}(C_1 t + C_2), \tag{7}$$

where C_1 and C_2 are unknown values.

As a second step it is necessary to determine the effect of the right-hand side in the differential equation (6). Let us say the electrical circuit has the alternating voltage source and the corresponding equation is $u = U_0 \sin(\omega t)$. We simplify the example for $U_0 = 1$ V and the expected particular equation $y_p = u_{Cp}$ looks like

$$u_{Cp} = A\sin(\omega t) + B\cos(\omega t). \tag{8}$$

To determine the unknown values A and B we derive the particular solution (8) up to the order the given differential equation

$$u'_{Cp} = A\omega\cos(\omega t) - B\omega\sin(\omega t)$$
$$u''_{Cp} = -A\omega^2\sin(\omega t) - B\omega^2\cos(\omega t)$$

and replace u_{Cp}, u'_{Cp} and u''_{Cp} into the given differential equation (6)

$$LC\omega^2(-A\sin(\omega t) - B\cos(\omega t)) + RC\omega(A\cos(\omega t) - B\sin(\omega t)) + A\sin(\omega t) + B\cos(\omega t) = \sin(\omega t) \tag{9}$$

Comparing functions $\sin(\omega t)$, $\cos(\omega t)$ on both sides of the equation (9) we get

$$-ALC\omega^2 - BRC\omega + A = 1$$
$$-BLC\omega^2 + ARC\omega + B = 0$$
$$A = \frac{1 - LC\omega^2}{(LC\omega^2)^2 + (RC\omega)^2}$$
$$B = -\frac{RC\omega}{(LC\omega^2)^2 + (RC\omega)^2}$$

In the third step we add both homogeneous and particular parts together

$$u_C = u_{Ch} + u_{Cp}$$
$$u_C = e^{-\frac{R}{2L}t}(C_1 t + C_2) + \frac{(1 - LC\omega^2)\sin(\omega t) - RC\omega\cos(\omega t)}{(LC\omega^2)^2 + (RC\omega)^2} \tag{10}$$

To determine the unknown C_1 and C_2 we insert the initial value $u_C(0) = 0$ into (10)

$$C_2 = \frac{RC\omega}{(LC\omega^2)^2 + (RC\omega)^2}$$

for inserting second initial value $u_c'(0) = 0$ we calculate the derivative of the equation (10)

$$u_C' = -\frac{R}{2L}e^{-\frac{R}{2L}t}(C_1 t + C_2) + C_1 e^{-\frac{R}{2L}t} + \frac{\omega(1 - LC\omega^2)\cos(\omega t) + RC\omega^2 \sin(\omega t)}{(LC\omega^2)^2 + (RC\omega)^2} \tag{11}$$

and the initial value $u_C'(0) = 0$ is now inserted in (11)

$$C_1 = \frac{R^2 C\omega - 2L\omega(1 - LC\omega^2)}{2L\Big((LC\omega^2)^2 + (RC\omega)^2\Big)} \tag{12}$$

The analytical solution u_C of the differential equation of second order (6) with multiple root for RLC circuit is given by

$$u_C = e^{-\frac{R}{2L}t}\left(\frac{R^2 C\omega - 2L\omega(1 - LC\omega^2)}{2L\Big((LC\omega^2)^2 + (RC\omega)^2\Big)}t\right) \tag{13}$$

$$+ e^{-\frac{R}{2L}t}\left(\frac{RC\omega}{(LC\omega^2)^2 + (RC\omega)^2}\right) + \frac{(1 - LC\omega^2)\sin(\omega t) - RC\omega\cos(\omega t)}{(LC\omega^2)^2 + (RC\omega)^2} \tag{14}$$

We set the special values of the circuit as

$$R = 20\ \Omega,\ L = 2.5 \cdot 10^{-2}\ \text{H},\ C = 5 \cdot 10^{-5}\ \text{F},$$

$$\omega = 1000\ \text{rad/s},\ u = sin(\omega t)\ \text{V}$$

we solve the equation (14) and we graphically represent the analytical solution of u_C in figure 2.

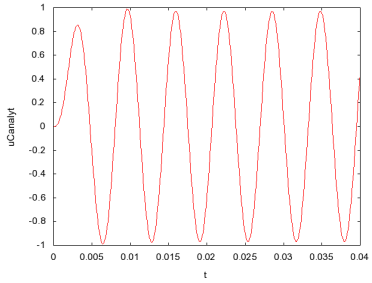

Figure 2: Voltage u_C in RLC circuit - computed from the analytical solution

2.2 Numerical solution

The second way to solve differential equations is the *numerical* solution. The numerical solving is based on approximations and it includes many other aspects of chosen numerical method such as initial conditions, generation and propagation errors, stability and convergence of the method, a variable stepsize etc. By numerical solution of differential equation we mean a sequence of values $y(t_0),\ y(t_1), \cdots,\ y(t_i)$ for $i = 0,\ 1, \cdots,\ n$.

From this part of the work numerical methods for the solution of the *initial value problem* in ordinary differential equations are evaluated and compared. An initial value problem is specified as follows

$$y'(x) = f(y(x)), \quad y(x_0) = y_0. \tag{15}$$

There exist two main types of numerical methods, the first types use for the next approximation y_n only the current already known approximation y_{n-1}, we call them *one-step methods*. The other ones called multistep methods solve the next approximation using current and previous approximations $y_n,\ y_{n-1},\ y_{n-2}, ...$

We proceed from introduction of chosen one-step methods such as the simplest Euler method through generalizations to chosen multistep methods. These generalizations are based on more computations in a step, use of more previous values or higher derivatives.

2.3 Euler method

The simplest and the most analyzed numerical method for solving ordinary differential equations is *Euler method*. It is the simple recursion formula which studies the solution for only certain values $x = 0,\ h,\ 2h, \cdots$, where h is called an integration step or a stepsize and assumes that dy/dx is constant between points. The recursion formula is given by

$$y_n = y_{n-1} + hf(y_{n-1}), \quad y(0) = y_0. \tag{16}$$

The sequence of values starting from the initial value x_0 is used for computation and stepsizes between each values of sequence $x_1 - x_0,\ x_2 - x_1, ...$ are denoted as $h_1,\ h_2, ...$, the highest is denoted by h. For each value of n, each approximation of y_n is computed using a previous value y_{n-1} which is exactly equal to $y(x_{n-1})$. We see that the quality of approximations of y_{n-1} depend on the magnitude of h.

The Euler method is based on a truncated Taylor series expansion which implies the *local truncation error* l_n (or *discretization* error) of the method as a $O(h^2)$. The local truncation error is an error committed by the method in a single step when the values at the beginning of that step are assumed to be exact. From this fact we can say, that the Euler method is first order technique, generally a method with local truncation error equals to $O(h^{p+1})$ is said to be of p-th order. At the n-step the error is defined by

$$\begin{aligned} l_n =& y(x_{n-1} + h) - y(x_{n-1}) - hf(x_{n-1}, y(x_{n-1})) \\ =& y(x_{n-1}) + hy'(x_{n-1}) + \frac{h^2}{2}y''(x_{n-1} + \theta h) \\ & - y(x_{n-1}) - hy'(x_{n-1}) \\ =& \frac{h^2}{2}y''(x_{n-1} + \theta h), \qquad 0 < \theta < 1. \end{aligned}$$

The truncation error is different from *the global error* ϵ_n [9], which is defined as

$$\begin{aligned} \epsilon_n =& y(x_{n-1} + h) - y_n \\ =& y(x_{n-1}) - hf(x_{n-1}, y(x_{n-1})) + l_n \\ & - y_{n-1} - hf(x_{n-1}, y_{n-1}) \\ =& \epsilon_{n-1} - hf(x_{n-1}, y(x_{n-1})) - hf(x_{n-1}, y_{n-1}) + l_n. \end{aligned} \tag{17}$$

In the most cases, the exact solution is unknown and hence the global error cannot be evaluated. Evaluations

of errors are closely linked to a variable stepsize determination, but we will discuss it later. The magnitude of stepsize is important for the convergence of the method. A convergent numerical method is the one where the numerically computed solution approaches the exact solution as the stepsize approaches 0. For problems with unknown exact solution, we choose the solution obtained with a sufficiently small time step as the "exact" solution to study the convergence characteristics. So taking the norm of global error in (17) and applying the triangle inequality, the Lipschitz condition and the bound on the local error, we get the first-order inequality

$$||\epsilon_n|| = (1 + hL)||\epsilon_{n-1}|| + \frac{Mh^2}{2}.$$

Since $\epsilon_0 = 0$, the inequality has solution given by

$$||\epsilon_n|| \leq \frac{Mh}{2L}(1 + hL)^n$$

where as $n \to \infty$ and $h \to 0$, we have $\epsilon_n \to 0$ and $y_n \to y(x_n)$ for some $M < \infty$ that is the numerical solution converges to the exact solution. Then we can say that methods of order higher than one are also convergent [10].

For the Euler methods there are stepsize limitations such as to ensure numerical stability, reasonable required accuracy, also fast convergence behaviour. A bit of improvement is given by *implicit Euler method*

$$y_n = y_{n-1} + hf(y_n), \quad y(0) = y_0 \tag{18}$$

For better understanding of stepsize and convergence of the method, have a look to a simple example also called Dahlquist problem with known exact solution [11].

Consider a Dahlquist problem

$$y' = qy, \quad y(0) = 1 \tag{19}$$

with known analytical solution given by $y(x) = \exp(qx)$. In this case we choose constant $q = 1$.

To check the order of Euler method with the fixed stepsize, we determine the error each time for n steps and set the stepsize such as $h = (t_{max} - t_{min})/n$ for different n values as $n = 10,\ 20,\ 40,\ \cdots\ 10240$, see table 1. We plot the order graph with the log of stepsizes on the x-axis and the log of absolute values of errors on the y-axis.

From now we will use the notation 1e−02, 1e−03, 1e−04, $\cdots$, 1e+03, 1e+04, $\cdots$ respectively for numbers $1 \cdot 10^{-2}$, $1 \cdot 10^{-3}$, $1 \cdot 10^{-4}, \cdots, 1 \cdot 10^{3}$, $1 \cdot 10^{4}, \cdots$ respectively.

Errors give us an order illustrates the rate at which the numerical error decreases with stepsize, see picture 3. Note that the order plot is from now always a log-log plot because the size of the error spans orders of magnitude. The slope of the error curve on a log-log plot gives the order of accuracy of the method. If the slope is unity, the error scales linearly with the stepsize. If the slope is two, then the error scales as the square of the stepsize.

Checking the slope of lines through the points we can say that the order of Euler method is 1. This means that results are consistent with order 1. Generally holds, that if the method has order p, the error for small h approximately satisfies an equation

$$E \approx Ch^p \tag{20}$$

assuming that E is the norm of the error and C is some constant so that everything is scalar.

Table 1: Errors and the order of Euler method for different fixed stepsizes

h	err	ratio
0.2	0.2299618	
0.1	0.1245394	1.846
$0.1{\cdot}2^{-1}$	6.498412e-02	1.916
$0.1{\cdot}2^{-2}$	3.321799e-02	1.956
$0.1{\cdot}2^{-3}$	1.679689e-02	1.978
$0.1{\cdot}2^{-4}$	8.446252e-03	1.989
$0.1{\cdot}2^{-5}$	4.235185e-03	1.994
$0.1{\cdot}2^{-6}$	2.120621e-03	1.997
$0.1{\cdot}2^{-7}$	1.061069e-03	1.999

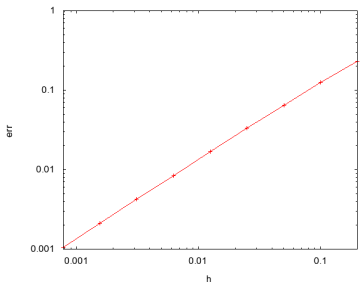

Figure 3: Order of Euler method for Dahlquist problem

Knowing that the Euler method converges and the error increases for increasing time over the tolerable limit, let us study the behaviour of the method over extended interval [11]. Assume the linear system of equations of constant coefficients

$$y'(x) = My(x), \tag{21}$$

where M is the constant matrix. This problem can be transformed using a few assumptions according to [7] to the simpler form

$$y'(x) = q(x) \tag{22}$$

where $z = hq$ with the exact solution $y_{n+1} = \exp(zn)y_0$ and z is a complex number. Using fixed stepsize it was said that $(1 + hq)^n$ is an acceptable approximation to $\exp(nhq)$, where both expressions as $n \to \infty$ are bounded. That also means that if the stability function defined as

$$\mathrm{R}(z) = \frac{y_{n+1}}{y_n} \tag{23}$$

meet the condition $\mathrm{R}(z) \leq 1$, then $|1 + hq|$ is bounded by 1. The set of values for the exact solution is bounded in the non-positive half-plane $z \in C : \mathrm{R}(z) \leq 0$. For this condition is the set of points for Euler method equals to $|1 + z| \leq 1$ (set of points is the closed disc in the complex plane with the centre in -1 and radius of 1). This property is also called boundedness. Property of converging is less strict then the unify, the exact solution lays in the negative left-plane $z \in C : \mathrm{R}(z) < 0$, so the set of points for Euler method lays in the open disc with the centre in -1 and radius of 1. For Euler method and implicit Euler method have been derived stability regions as follows

$$R(z) = \begin{cases} 1 + z, & \text{(Euler method)} \\ \frac{1}{1-z}. & \text{(implicit Euler method)} \end{cases}$$

Stability regions of both methods are plotted and colored in figure 4.

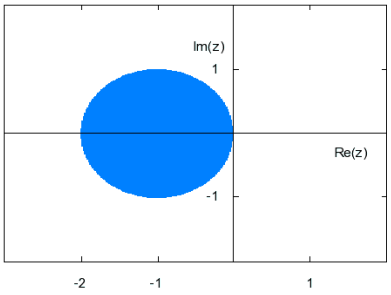

Figure 4: Stability regions for Euler method

We say that the stability region is defined as a set of points in the complex plane, z should stay in the disc for other problems. It can be achieved only by reducing h. This causes many limitations. For example to solve stiff problems with very negative eigenvalues it means to decrease h so much that it makes explicit method unusable. If the stability function has no poles in the left half-plane, this means the stability region includes all zeros of the left half-plane and the method is said to be A-stable. It also holds that the magnitude $|\mathrm{R}(z)|$ must be bounded by 1 for z on the imaginary axes. A-stability is a very desirable property for any numerical algorithm, particularly if initial value problems were to be stiff or stiff oscillatory [16].

Another interesting way how to study the stability region is using *order stars* technique [15], see colored regions in figure 5. This property of multiplying the stability function by $\exp(-z)$ should make no difference in the characteristic of the method stability. Notice the behaviour near $z = 0$ and $z = -1$. For $|\mathrm{Re}(z)|$ large, the behaviour is effected by the exponential function, the behaviour around zero is the same as for the absolute stability region and the behaviour at $z = -1$ is determined by a pole. The regions intersect with zero and $\mathrm{Re}(R(z)\exp(-z))$ positive are called *fingers*. Regions with negative $\mathrm{Re}(R(z)\exp(-z))$ are known as *dual fingers*. Similar technique as order stars is the *order arrows*. The technique of order arrows is about to plot the paths in the complex plane where $\omega(z) = \exp(-z)R(z)$ is real and positive.

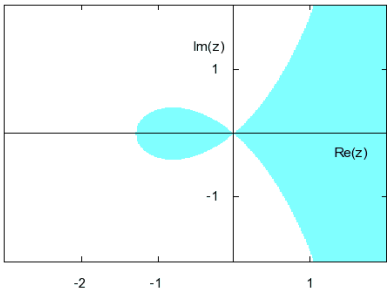

Figure 5: Order stars for explicit Euler method

The next generalization of the Euler method assumes instead of computing f once in a step that the method computes f two or more times with different arguments. This approach defines an important class of one-step method known as *Runge-Kutta methods*.

2.4 Runge–Kutta methods

Suppose we know $y(x_n)$ and we want to determine an approximation y_{n+1} to $y(x_n + h)$. The idea behind the Runge-Kutta methods is to compute the value of $f(x, y)$ at several conveniently chosen points near to the solution in the interval $(x_n, x_n + h)$ and to combine these values in such a way as to get good accuracy in the computed increment.

Generally, this important section of numerical methods can be written in the form of equations such as

$$\begin{aligned} y_n &= y_{n-1} + h\sum_{j=1}^{s} b_j F_j, \\ Y_i &= y_{n-1} + h\sum_{j<i}^{s} a_{ij} F_j, \end{aligned} \tag{24}$$

where $F_i = f(Y_i)$ is evaluated by approximations y_n to $y(x_n)$ for $i = 1, 2, \cdots, s$, constants b_j, a_{ij} can be written into table 2. Types of methods could be specified by different values of those coefficients. The tableau was defined by J. C. Butcher [6].

Table 2: Butcher tableau of Runge–Kutta methods

0					
c_2	a_{21}				
c_3	a_{31}	a_{32}			
$\vdots$	$\vdots$	$\vdots$	$\ddots$		
c_s	a_{s1}	a_{s2}	$\cdots$	$a_{s,s-1}$	
	b_1	b_2	$\cdots$	b_{s-1}	b_s

To specify some types of the method, one needs to provide the constant number s, which determines the number of internal stages, and constants a_{ij} (for $1 \le j < i \le s$), constants b_i (for $i = 1,\ 2, \cdots,\ s$) and constants c_i (for $i = 2,\ 3, \cdots,\ s$) [3].

The local truncation error of Runge–Kutta methods cannot be worse than the Euler method from the view of the consistency condition and it is $\mathrm{O}(h^2)$. The consistency condition guarantees that at least one independent variable is computed correctly. Due to the dependency of the local truncation error on constants a_{ij} and b_i the conditions for a given order accuracy are determined.

The main idea behind the order of each method is the number of stages s required to achieve this order and the number of computed free parameters for given number of stages. The relationship between those numbers is given by conditions, so-called order conditions. We can use the approach of the rooted trees and apply it for the order condition description for all classes of Runge–Kutta algorithms [6].

To illustrate the analysis of the grown of numerical errors in a computed solution to a differential equation, we consider the equation (22) again as in Euler method stability analysis. As we write $hq = z$ the analysis generalizes in the case of explicit Runge-Kutta methods to give a result y_n computed after n steps from $y(0) = 1$. The result is given by $y_n = r(z)^n$. The r is a particular polynomial determined by the coefficients in the method. In the case of implicit Runge-Kutta methods, r is not in general a polynomial but a rational function.

A Runge-Kutta method is said to be A-stable if its stability region contains C^-, the non-positive half-plane. This definition has been redefined in different ways during the time. More requirements on the qualitative behaviour of numerical solutions were proposed. Let us introduce some of the requirements. One of them is that a method to be such that $|r(z)| \leq 1$ for all C^- and in addition that $\lim_{|z|\to\infty} |r(z)| = 0$ and it is known as a L-stability. Quite standard requirement of A-stability is that the stability region include the set $C(\alpha) = z \in C : |arg(-z)| \leq \alpha$ and the stability region contains some left half-plane together with the intersection of the negative half-plane with some open set containing the real axis. This properties was named $A(\alpha)$-stability (Widlund, 1967) and later named as stiff stability (Gear, 1969) [14, 25].

The requirements which refers to the qualitative behaviour of numerical solutions to certain non-linear problems are given by B-stability (Butcher, 1975). The property says that for two particular solutions to such a problem the difference between them is non-increasing and could be applied to numerical solution. This property can be considered also for non-autonomous differential equations and the method preserves it is called BN-stable (Burrage and Butcher, 1979) [4, 8].

Consider a Runge–Kutta method given by

$$
\begin{aligned}
&Y_1 = y_{n-1},\\
&Y_2 = y_{n-1} + ha_{21}f(Y_1),\\
&\cdots\\
&Y_s = y_{n-1} + h(a_{s1}f(Y_1) + a_{s2}f(Y_2) + \cdots + a_{s,s-1}f(Y_{s-1})),\\
&y_n = y_{n-1} + h(b_1f(Y_1) + b_2f(Y_2) + \cdots + b_sf(Y_s))
\end{aligned}
$$

using the Dahlquist problem (19), $z = hq$ and s the number of stages. We rewrite it as

$$
\begin{aligned}
Y &= y_{n-1}e + zAY,\\
y_n &= y_{n-1} + zb^TY,
\end{aligned}
$$

where $e = [1, 1, \cdots, 1]^T$, $Y = [Y_1, Y_2, \cdots, Y_s]^T$ and $b^T = [b_1, b_2, \cdots, b_s]$.

The polynomial r which determines the stability of the method is given by

$$R(z) = \frac{y_n}{y_{n-1}} = 1 + zb^T(y_{n-1}^{-1}Y).$$

Due to some assumptions [5], we find

$$R(z) = 1 + z + \frac{z^2}{2!} + \cdots + \frac{z^p}{p!} + c_{p+1}z^{p+1}, \qquad (25)$$

A method is said to be A-stable if its stability function is bounded by 1 in the left half-plane. It is said to be L-stable if it is A-stable and $R(\infty) = 0$. A method of order p has a stability function with a series that agrees with e^z up to terms in h^p [7]. Hence we obtain the stability regions described in the table 3.

Table 3: Stability functions for Runge–Kutta methods up to order 4

order	$R(z)$
1	$1 + z$
2	$1 + z + \frac{1}{2}z^2$
3	$1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3$
4	$1 + z + \frac{1}{2}z^2 + \frac{1}{6}z^3 + \frac{1}{24}z^4$

2.5 Taylor series method

Substituting derivations and initial values into the formula for the Taylor polynomial, we than obtain a representation of the solution as a power series about the initial point x_0. This procedure, called the Taylor series method, is illustrated of power series (26). The mathematical background was widely described in the history [1, 2, 19].

$$y_n = y_{n-1} + hy'_{n-1} + \frac{h^2}{2!}y''_{n-1} + \cdots + \frac{h^p}{p!}y^{(p)}_{n-1} + O(h^{p+1}) \qquad (26)$$

The method has been implemented in simulators TKSL386, TKSL/C (Kunovsky, 1991, 1998) with different approach than the approach brought by Barton, Willers and Zahar [1]. It uses so-called forming differential equations which implement higher orders more effectively. The Taylor series method can be used for solving a large number of various problems and it has an automatic integration method using Taylor series. It could be used for variable order; the order p is set automatically using as many Taylor series terms for computing as needed to achieve the required accuracy.

The absolute value of the relative error of the computation is the main criterion to chosen the order. Maximum order of this method is computed up to 63 of Taylor series terms. The advantage is in the speed of computation, that is functions are generated by adding, multiplying and superposing elementary functions. The disadvantage of the method is the need to generate higher derivatives.

We again present the example of RLC electrical circuit (14) as a first test problem to show the power of Taylor series method. We focus on the numerical solution of the circuit and we compare it with the analytical solution. We have same constants and we solve the circuit numerically using differential equations. When we compare the numerical solution u_C to an analytical solution $u_Canalyt$ we get very small numbers of the error around values 10^{-17}. Hence, the Taylor series method proves high accuracy of calculation. For those interested in specific equations TKSL/C source code is given in the Appendix A.

It has been shown that generally the method is A-stable [24]. Stability regions of Taylor series methods up to order 4 are identical as stability regions of Runge–Kutta methods up to order 4.

The next chapter provides the generalization in such a way to bring more previous values such as the value y_n depends not only on y_{n-1} and $f(y_{n-1})$ but also on y_{n-2} and $f(y_{n-2})$, y_{n-3} and $f(y_{n-3})$,...

3. Linear multistep methods

The linear multistep method is essentially a polynomial interpolation procedure whereby the solution or its derivative is replaced by a polynomial of appropriate degree in the independent variable x, whose derivative or integral is readily computed. The linear multistep method for the initial value problem is given by

$$y_n = \sum_{i=1}^{k} \alpha_i y_{n-i} + h\sum_{i=0}^{k} \beta_i f(y_{n-i}, y_{n-i}). \qquad (27)$$

According to the coefficient b_0 one separates methods into *Gear methods* and *Adams methods*: explicit *Adams–Bashforth* ($b_0 = 0$) and implicit *Adams–Moulton* ($b_0 \neq 0$).

Adams–Bashforth method is an explicit multistep method whence

$$p = k - 1, \qquad a_1 = a_2 = \cdots = a_{k-1} = 0, \qquad b_{-1} = 0$$

defined by

$$y_n = y_{n-1} + h(a_1 f_{n-1} + a_2 f_{n-2} + \cdots + b_k f_{n-k}).$$

The coefficients a_k (see Tab. **??**) can be determined and rewritten also by

$$j \sum_{i=0}^{p} (-i)^{j-1} b_i = 1, \qquad j = 1, \cdots, k \tag{28}$$

The Adams-Bashforth formula of order 1 for $k = 1$ yields the (explicit) Euler method.

Adams–Moulton method is an implicit multistep method whence

$$p = k - 2, \qquad a_1 = a_2 = \ldots = a_{k-2} = 0$$

defined by

$$y_n = y_{n-1} + h(b_0 f_n + b_1 f_{n-1} + \cdots + b_{k-1} f_{n-k+1}).$$

Similarly, coefficients are obtained for the highest order possible. And however, the Adams-Moulton are implicit methods, thus reach order $p+1$. The Adams-Moulton formula of order 1 yields the (implicit) backward Euler integration method and the formula of order 2 yields method known as the trapezoidal rule.

A comparison of coefficients of both methods reveals that the coefficients of the implicit formula are smaller than those of the corresponding explicit formulas. The smaller coefficients lead to smaller local truncation errors and, hence, to improved accuracy over the explicit Adams-Bashforth methods [12].

A linear multistep method $[\alpha, \beta]$ is stable if the difference equation (29) has only bounded solution. The difference equation represents an one-dimensional problem to equation (27) with $f(x, y) = 0$ gives

$$y_n = \alpha_1 y_{n-1} + \alpha_2 y_{n-2} + \cdots + \alpha_k y_{n-k}. \tag{29}$$

A linear multistep method is said to be stable if all solution of the difference equation (29) are bounded as $n \to \infty$. Let $p(\lambda)$ be the corresponding characteristic polynomial

$$p(\lambda) = \lambda^k - \alpha_1 \lambda^{k-1} - \alpha_2 \lambda^{k-2} - \cdots - a_k.$$

A method is said to satisfy the root condition if $|\lambda_j| \leq 1$ for all j, and if $|\lambda_i|$ is a repeated root then $|\lambda_j| < 1$. That is, all roots must lie within the unit circle and those on the boundary must be simple [7].

3.1 Predictor–corrector methods

Predictor–corrector methods constitute an important algorithm in implementation of linear multistep methods and the most successful codes for the solution of initial value problems of ordinary differential equations. Briefly, these methods are successful because they occur in naturally arising families covering a range of orders, they have reasonable stability properties, and they allow an easy control via suitable stepsize or order changing policies and techniques. The major advantage of the multistep methods is that fewer functional evaluations are usually required per integration step [13].

We obtain different types by combinations of explicit and implicit methods. Usually the predictor is an Adams-Bashforth formula and it predicts first approximation value of the solution. The derivative evaluated from this approximation is used in Adams-Moulton corrector formula in the next step. Apart from the better stability of the predictor-corrector formulae over the explicit formulae, the predictor-corrector formulae are generally more accurate and provide reasonable and adequate error estimators [12].

In the calculation of predictor-corrector pairs are three stages:

1. Predict the starting value for the dependent variable y_{n+k} as y^*_{n+k}.
2. Evaluate the derivative at (x_{n+k}, y^*_{n+k}).
3. Correct the evaluated predicted value.

A combination of three stages is called PEC (predict–evaluate–correct) mode. It is often more desirable in terms of stability considerations to incorporate one additional function evaluation per integration step, thus calculate the PECE (predict–evaluate–correct–evaluate) mode [17]. Other options of repeating stages are possible but we have in mind that it is generally considered that functional evaluations are the most expensive part of the predictor–corrector procedure.

The stability improvement is given by PECE mode. One more evaluation on the end of each computational step makes the stability region more wide [18]. Notice the difference in stability region for Adams-Bashforth method order 2 (AB2) and Adams–Moulton method order 2 (AM2) in PEC mode presented in picture 6 and stability region of same methods orders 3 (AB3, AM3) in PECE mode in picture 7.

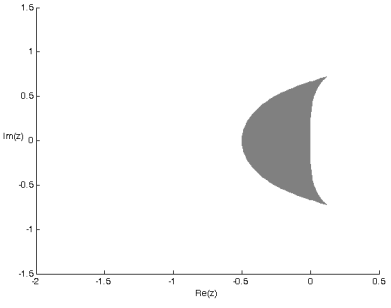

Figure 6: Region of absolute stability - PEC

To obtain the *General linear methods* we have two options. We generalize Runge–Kutta methods in case of using more previous values, or we generalize Linear multistep methods in case of using more stages in the calculation per step. So we have a range of possibilities from 1 input quantity, as in Runge–Kutta methods, to a large number as in multistep methods, more in [5].

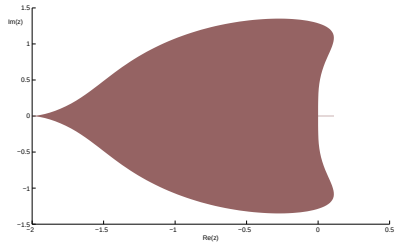

Figure 7: Region of absolute stability - PECE

4. Predictor–corrector Obreshkov method

The main contribution of this thesis is to extend Adams methods to higher derivative methods by using Obreshkov quadrature formulae. We consider a two-step predictor followed by a one-step corrector, in each case using second derivative formulae. As a choice for predictor–corrector pairs we consider both PEC and PECE methods.

We start with a generalization of Adams methods to second derivative methods. We consider problems for which it is efficient to calculate first and second derivatives at any solution point. We denote first and second derivatives by

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

At the start of the step we assume that we already have computed values in previous points

$$y_{n-1},\ y_{n-2},\ \cdots,\ y_{n-k},$$

obtained from the starting method. The question of starting method will be discussed later. We also know from the given problem first and second derivative values,

$$f_{n-1},\ f_{n-2},\ \ldots,\ f_{n-k},\ g_{n-1},\ g_{n-2},\ \ldots,\ g_{n-k},$$

which are given by $f_i = f(x_i, y_i),\ g_i = g(x_i, y_i)$.

To implement the new method in predictor–corrector pairs we consider using an explicit method for a predicted part and using an implicit formula for a corrected part.

Formulae for $f(x, y)$ and $g(x, y)$ are available, hence the Obreshkov method becomes available and we calculate the coefficients for predictor and corrector equations using the Lagrange interpolation formulae. With another restriction of two-steps formulae, we replace the Lagrange integration polynomial by the Lagrange-Hermite integration polynomial. Hence, we determine formulae of predictor equation

$$y(x_n) = y_{n-1} - \frac{1}{2}hf_{n-1} + \frac{3}{2}hf_{n-2} + \frac{17}{12}h^2 g_{n-1} + \frac{7}{12}h^2 g_{n-2} \tag{30}$$

and of corrector equation

$$y(x_n) = y_{n-1} + \frac{1}{2}hf_n + \frac{1}{2}hf_{n-1} - \frac{1}{12}h^2 g_n + \frac{1}{12}h^2 g_{n-1}. \tag{31}$$

We assume the use of the variable stepsize for the new method, thus we implement the new method in Nordsieck representation. By procedure described in [23], we obtain the algorithm of the new method such as

$$Y_n = PY_{n-1} + \delta \begin{bmatrix} \frac{1}{2} \\ 1 \\ 0 \\ -1 \\ -\frac{1}{2} \end{bmatrix} + \epsilon \begin{bmatrix} -\frac{1}{12} \\ 0 \\ 1 \\ \frac{4}{3} \\ \frac{1}{2} \end{bmatrix}, \tag{32}$$

where it holds

$$\delta = hf(Y^*_{n_1}) - [0\ 1\ 2\ 3\ 4]Y^*_{n-1},$$
$$\epsilon = h^2 g(Y^*_{n_1}) - [0\ 0\ 1\ 3\ 6]Y^*_{n-1},$$

where Y_n is an output vector in the Nordsieck representation, Y_{n-1} is an input vector in the Nordsieck representation, P is the Pascal matrix, the term $f(Y^*_n)$ is the f-function evaluation of predicted value Y^*_n and $Y^*_{n_1}$ means the first component of a predicted vector.

As a starting method can be used classical one-step methods such as Runge–Kutta method of order 4 or Adams-Bashforth method of order 4, but it seems reasonable to use the predictor–corrector Obreshkov method itself.

4.1 Order of the method

To check the order of the new method we show the simplest Dahlquist problem (19) with constant $q = 1$. We proceed uniformly as described previously in the section 2.3. We calculate the error each time for n steps with $h = (t_{max} - t_{min})/n$.

Checking the slope of line through points we say that the order of new method (called *vlgm* and represented by violet line in figure 8) is 4. For comparison there are also displayed results for the same problem computed by Euler method (red line), which is the method of order 1, and Taylor series method of orders 2 (green line) and Taylor series method of order 4 (blue line). The corresponding errors for Euler method, Taylor series method of order 2 and of order 4 and for our method are determined in table 4. The interesting fact is that Taylor series method of order 4 is more precise than the classical Runge–Kutta of order 4.

Table 4: Errors for Dahlquist problem of Taylor series method and new method

h	err_{Ts4}	err_{vlgm}
0.1	2.084324e-06	1.147407e-05
$0.1{\cdot}2^{-1}$	1.358027e-07	7.462822e-07
$0.1{\cdot}2^{-2}$	8.666185e-09	4.757726e-08
$0.1{\cdot}2^{-3}$	5.473053e-10	3.003144e-09
$0.1{\cdot}2^{-4}$	3.438494e-11	1.886535e-10
$0.1{\cdot}2^{-5}$	2.155165e-12	1.187850e-11
$0.1{\cdot}2^{-6}$	1.350031e-13	6.847856e-13
$0.1{\cdot}2^{-7}$	4.884981e-15	4.279039e-14

Other two multistep methods were implemented and results for the problem were calculated for the comparison. The new method is motivated by Adams methods so to compare the new method with Adams method in PEC mode is natural. Hence, we implemented and calculate with Adams-Bashforth Adams-Moulton formulae of order 4 used in PEC mode (called ABAM4PEC and illustrated only in picture). And the other method is Adams-Bashforth method of order 4 (AdamBash4). Errors are displayed in table 5 and corresponding slopes of orders are plotted in the figure 9 for comparing those methods via

Table 5: Errors for Dahlquist problem of RK4, AB4 and new method *vlgm*

h	err_{RK4}	$err_{AdamBash4}$	err_{vlgm}
0.1	1.133156e-05	3.767292e-04	1.147407e-05
$0.1{\cdot}2^{-1}$	7.383001e-07	2.761708e-05	7.462822e-07
$0.1{\cdot}2^{-2}$	4.711429e-08	1.865007e-06	4.757726e-08
$0.1{\cdot}2^{-3}$	2.975458e-09	1.211015e-07	3.003144e-09
$0.1{\cdot}2^{-4}$	1.869447e-10	7.713870e-09	1.886535e-10
$0.1{\cdot}2^{-5}$	1.170353e-11	4.866960e-10	1.187850e-11
$0.1{\cdot}2^{-6}$	7.265299e-13	3.055334e-11	6.847856e-13
$0.1{\cdot}2^{-7}$	6.394885e-14	1.900702e-12	4.279039e-14

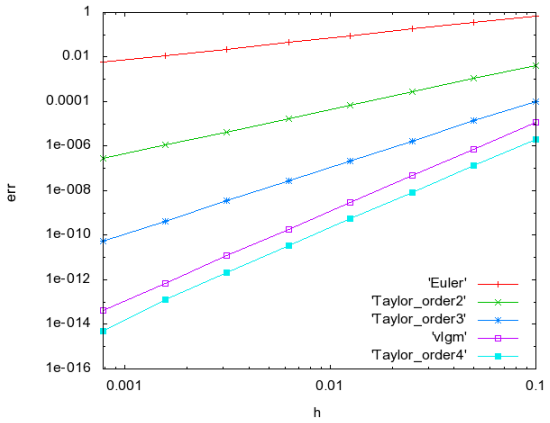

Figure 8: Errors and orders of different methods for Dahlquist problem

positions of lines. Satisfying fact is that our new method has comparable results with Runge–Kutta method of order four and it is more accurate than Adams-Bashforth Adams-Moulton formulae of order 4 used in PEC mode.

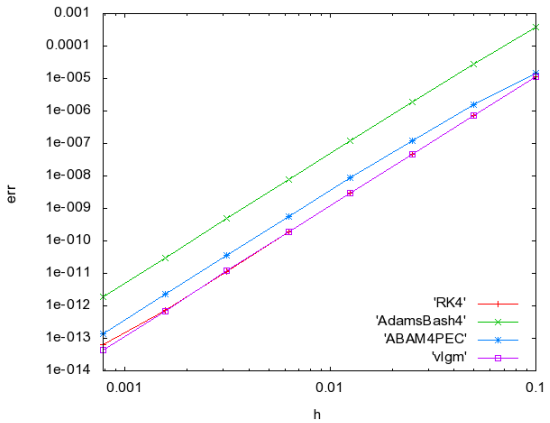

Figure 9: Errors and orders of four methods for Dahlquist problem

4.2 PEC and PECE modes

We are concerned about the different modes of the method. Our method is represented in a PEC mode. We discovered that as we repeat the evaluate–correct step one more time after one cycle of predict–evaluate–correct procedure, results will be improved in the accuracy point of view according to experiments.

As we repeat steps 2. and 3. one more time and we call it PECECE (or PE(CE)^2) mode, results are still improved according to a PEC mode, but they are slightly less accurate than errors for a PECE mode. Those result were also expected, this behaviour occurs in some problems even for classical Adams methods in corresponding modes.

The Prothero–Robinson problem [22] was chosen for the demonstration. Errors of PECE mode are smaller than errors of PEC mode, see the table 6. Ratio numbers represent 2^p with the order p of the new method in corresponding mode.

Table 6: Errors for Prothero–Robinson problem of our method in different modes

h	error$_{PEC}$	ratio	error$_{PECE}$	ratio
0.1	5.197512e-07		3.928242e-07	
$0.1{\cdot}2^{-1}$	2.717284e-08	19.128	2.338723e-08	16.780
$0.1{\cdot}2^{-2}$	1.528272e-09	17.780	1.413994e-09	16.540
$0.1{\cdot}2^{-3}$	9.020651e-11	16.942	8.671031e-11	16.307
$0.1{\cdot}2^{-4}$	5.472844e-12	16.483	5.365375e-12	16.161
$0.1{\cdot}2^{-5}$	3.359535e-13	16.290	3.330669e-13	16.109
$0.1{\cdot}2^{-6}$	1.942890e-14	17.291	1.931788e-14	17.241
$0.1{\cdot}2^{-7}$	6.328271e-15	3.070	6.328271e-15	3.053

4.3 Stability

The stability analysis for two-derivative multistep method is presented in this section. For plotting the stability region we use predictor and corrector equations of our method.

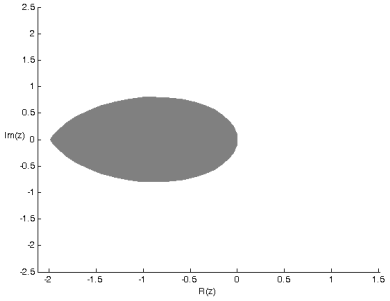

Figure 10: Stability regions of the new method in PEC mode

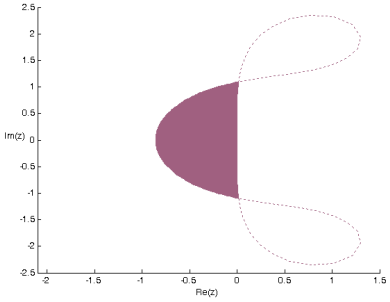

Figure 11: Stability regions of the new method in PECE mode

It has been discovered that the size of stability region is bigger for our method than the stability region of Adams-Bashforth Adams-Moulton method of order 4. The same characteristics holds for the PECE mode.

5. Conclusions

Convergence and stability analysis for the predictor–corrector method in Obreshkov quadrature formulae with constant stepsize for various problems have been shown as well as the comparison between the new method and other selected numerical methods. The method turned out to be just as reliable as the traditional methods. The cost of new method decreases with the complexity of the problem and the accuracy is preserved. The higher order of new method will be more accurate than the classical Adams method.

The size of the stability region for the resulting algorithm is still small, but the stability region is larger than commonly used methods such as Adams–Bashforth Adams–Moulton method of order four in PEC mode or Adams–Bashforth method of order 4. Hence, the new algorithm may be of interest of applications where stability is a strong limitation.

Acknowledgements. The work presented in this thesis was supported by the Czech Ministry of Education, project no. MSM 0021630528.

Appendix

A. TKSL/C code

The code for RLC circuit for TKSL/C is presented here.

```
omega=1e+3;
R=20;
L=2.5e-2;
C=5e-5;
u=sin(omega*t);
% numerical solution
uC'=1/C*i &0;
i' =1/L*uL &0;
uL =u-R*i-uC;
% analytical solution
uCanalyt=exp(-400*t)*(16/17*cos(800*t)
         +13/17*sin(800*t))
         -4/17*sin(omega*t)-16/17*cos(omega*t);
% error between solutions
err=uC-uCanalyt;
```

The program TKSL/C is available on

http://www.fit.vutbr.cz/ kunovsky/TKSL/download.html.

To run the computation copy the code above to the text file named input by the command in the terminal

```
cltksl -t 0.1 -s 1e-4 input > output
```

References

[1] D. Barton, I. M. Willers, and R. V. M. Zahar. Taylor series methods for ordinary differential equations–an evaluation. *Mathematical Software Symposium*, 14(3):243–248, 1970.

[2] D. Barton, I. M. Willers, and R. V. M. Zahar. The automatic solution of ordinary differential equations by the method of Taylor series. *Computing*, 14(3):243–248, 1971.

[3] R. L. Burden and J. D. Faires. *Numerical analysis*. Brooks Cole, 2004.

[4] J. C. Butcher. A stability property of implicit Runge-Kutta methods. *BIT Numerical Mathematics*, 15(4):358–361, 1975.

[5] J. C. Butcher. *The numerical analysis of ordinary differential equations: Runge–Kutta and general linear methods*. John Wiley & Sons, 1987.

[6] J. C. Butcher. A history of Runge–Kutta methods. *Applied Numerical Mathematics*, 20(1):247–260, 1996.

[7] J. C. Butcher. *Numerical methods for ordinary differential equations*. John Wiley & Sons, 2008.

[8] J. C. Butcher and K. Burrage. Stability criteria for implicit Runge–Kutta methods. *SIAM Journal on Numerical Analysis*, 16(1):46–57, 1979.

[9] J. C. Butcher and P. B. Johnson. Estimating local errors for Runge–Kutta methods. *Computational & Applied Mathematics*, 45(1-2):203–212, 1993.

[10] G. G. Dahlquist. Convergence and stability in the integration of ordinary differential equations. *Mathematica Scandinavica*, 4:33–50, 1956.

[11] G. G. Dahlquist. *Numerical methods*. Prentice-Hall Inc., 1974.

[12] S. O. Fatunla. *Numerical methods for initial value problems in ordinary differential equations*. Academic Press, Inc., 1988.

[13] M. Fujii. An extension of Milne's device for the Adams predictor-corrector methods. *Japan Journal of Industrial and Applied Mathematics*, 8(1):1–18, 1991.

[14] C. W. Gear. The automatic integration of ordinary differential equations. *Communications of the ACM*, 14(3):176–179, 1971.

[15] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving ordinary differential equations I: Nonstiff problems*. Springers Verlag, 1993.

[16] T. E. Hull, W. H. Enright, B. M. Fellen, and A. E. Sedgwick. Comparing numerical methods for ordinary differential equations. *SIAM Journal on Numerical Analysis*, 9(4):603 – 637, 1972.

[17] R. W. Klopfenstein and C. B. Davis. PECE algorithms for the solution of stiff systems of ordinary differential equations. *Mathematics of Computation*, 25(115):457–473, 1971.

[18] F. T. Krogh. Predictor–corrector methods of high order with improved stability characteristics. *Journal of the ACM*, 13(3):374–385, 1966.

[19] J. Kunovský. *Modern Taylor series method*. FEI VUT Brno, 1995. Habilitation work.

[20] J. Kunovský and et al. Using differential equations in electrical circuits simulation. *Journal of Autonomic Computing*, 1(2):192–201, 2009.

[21] R. K. Nagle, E. B. Saff, and A. D. Snider. *Fundamentals of differential equations*. Addison-Wesley, 1996.

[22] A. Nordsieck. On numerical integration of ordinary differential equations. *Mathematics of Computation*, 16:22–49, 1962.

[23] P. Sehnalová, J. Butcher, and J. Kunovský. Predictor–corrector obreshkov pairs. *International Conference of Scientific Computation And Differential Equations*, July 2011.

[24] D. Řezáč. *Stiff systems of differential equations and Modern Taylor series method*. FIT VUT Brno, 2004. Ph.D. thesis.

[25] O. B. Widlund. A note on unconditionally stable linear multistep methods. *BIT Numerical Mathematics*, 1(1):65–70, 1967.

Selected Papers by the Author

J. Kunovský, P. Sehnalová, V. Šátek. Stability and Convergence of the Modern Taylor Series Method. In *7th EUROSIM Congress on Modelling and Simulation*, Praha, CZ, VCVUT, 2010.

J. Kunovský, P. Sehnalová, V. Šátek. Explicit and Implicit Taylor Series Based Computations. In *8th International Conference of Numerical Analysis and Applied Mathematics*, Tripolis, GR, AIP, 2010, pp. 587-590.

J. Kunovský, V. Kaluž, J. Kopřiva, P. Sehnalová. Using Differential Equations in Electrical Circuits Simulation. In *International Journal of Autonomic Computing*, vol. 1, no. 2, 2009, London, GB, pp. 192-201.