On Stable Integration of Stiff Ordinary Differential Equations with Global Error Control⋆

Gennady Yur'evich Kulikov and Sergey Konstantinovich Shindin

School of Computational and Applied Mathematics,
University of the Witwatersrand, Private Bag 3,
Wits 2050, Johannesburg, South Africa
{gkulikov, sshindin}@cam.wits.ac.za

Abstract. In the paper we design an adaptive numerical method to solve stiff ordinary differential equations with any reasonable accuracy set by the user. It is a two-step second order method possessing the *A*-stability property on any nonuniform grid [3]. This method is also implemented with the local-global step size control developed earlier in [8] to construct the appropriate grid automatically. It is shown that we are able to extend our technique for computation of higher derivatives of fixed-coefficient multistep methods to variable-coefficient multistep methods. We test the new algorithm on problems with exact solutions and stiff problems as well, in order to confirm its performance.

1   Introduction

The problem of an automatic global error control for the numerical solution of ordinary differential equations (ODEs) of the form

$$x'(t) = g\big(t, x(t)\big), \quad t \in [t_0, t_0 + T], \quad x(t_0) = x^0, \tag{1}$$

where $x(t) \in \mathbf{R}^n$ and $g : D \subset \mathbf{R}^{n+1} \to \mathbf{R}^n$ is a sufficiently smooth function, is one of the challenges of modern computational mathematics. ODE (1) is quite usual in applied research and practical engineering (see, for example, [1], [4], [6], [7]). Often, problem (1) is stiff and requires numerical methods with special properties of stability. *A*-stable methods are desired in such a situation (see, for example, [1], [4], [7]).

   Unfortunately, there are very few algorithms with the property indicated above among linear multistep formulas because of the Dahlquist's second barrier [2]. It says that there exist no *A*-stable multistep methods of any order higher than two (even on uniform grids). The problem becomes more complicated on nonuniform grids. Thus, the only known now family of multistep formulas which

---

are A-stable on any grid is the one-parameter family of two-step methods derived by Dahlquist et al. [3]. Note there is no sense to consider "longer" methods because, anyway, they will be of order two at most. We concentrate on two particular choices of the parameter. One of them was made in the paper mentioned above (see [3]). Another one is our own choice. We compare both algorithms on numerical examples.

The methods presented in [3] are the good choice for a variable step size implementation. They have no step size restriction with a point of view of stability. Our idea is to supply these methods with the local-global step size control [8] aiming to attain any reasonable accuracy for the numerical solution of problem (1) in automatic mode. We also extend the technique of computation of higher derivatives [9] to variable-coefficient multistep methods. It imposes a weaker condition on the right-hand side of problem (1) for the step size selection to be correct, than in [8], where we differentiated interpolating polynomials.

The paper is organized as follows: Sect. 2 presents the family of A-stable two-step methods on uniform and nonuniform grids. Sect. 3 is devoted to the local and global errors estimation technique for the numerical methods mentioned above. The last section in the paper gives us numerical experiments confirming practical importance of the algorithms under consideration.

2 A-Stable Two-Step Methods

Further, we suppose that ODE (1) possesses a unique solution $x(t)$ on the whole interval $[t_0, t_0 + T]$. To solve problem (1) numerically, we introduce a uniform grid w_τ with step size τ on the interval $[t_0, t_0 + T]$ and apply the A-stable linear two-step method of order 2 in the form

$$\sum_{i=0}^{2} a_i x_{k+1-i} = \tau \sum_{i=0}^{2} b_i g(t_{k+1-i}, x_{k+1-i}), \quad k = 1, 2, \ldots, K-1, \qquad (2)$$

where

$$a_0 = \frac{1}{\gamma+1}, \qquad a_1 = \frac{\gamma-1}{\gamma+1}, \qquad a_2 = -\frac{\gamma}{\gamma+1},$$

$$b_0 = \frac{3\gamma+1}{2(\gamma+1)^2}, \quad b_1 = \frac{(\gamma-1)^2}{2(\gamma+1)^2}, \quad b_2 = \frac{\gamma(\gamma+3)}{2(\gamma+1)^2}$$

and the free parameter satisfies the condition $0 < \gamma \leq 1$. Note that we have used a slightly different way to present the family of stable two-step methods from [3]. The starting values x_k, $k = 0, 1$, are considered to be known.

We apply the following idea in order to fix the parameter γ. Let us consider the linear test equation $x' = \lambda x$ where λ is a complex number. We want to provide the best stability at infinity for method (2). This property is close to L-stability of Ehle [5] and useful when integrating very stiff ODEs. It means for multistep methods that we need to minimize the spectral radius of the companion matrix of method (2) (see [7]).

The companion matrix of method (2) when $\mathrm{Re}\,\mu \to -\infty$ has the following form:

$$C_\infty(\gamma) \overset{\text{def}}{=} \lim_{\mathrm{Re}\,\mu \to -\infty} \begin{pmatrix} \dfrac{\mu b_1 - a_1}{a_0 - \mu b_0} & \dfrac{\mu b_2 - a_2}{a_0 - \mu b_0} \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} -\dfrac{b_1}{b_0} & -\dfrac{b_2}{b_0} \\ 1 & 0 \end{pmatrix}$$

where $\mu = \tau\lambda$. Unfortunately, $\rho(C_\infty(\gamma)) > 0$ (i.e., the spectral radius of the matrix $C_\infty(\gamma)$ is greater than zero) for any $0 < \gamma \leq 1$ because both coefficients b_1 and b_2 cannot vanish simultaneously (see (2)). Nevertheless, a simple computation shows that eigenvalues of the matrix $C_\infty(\gamma)$ are

$$\lambda_{1,2} = \frac{-(\gamma - 1)^2 \pm (\gamma + 1)\sqrt{\gamma^2 - 18\gamma + 1}}{6\gamma + 2}.$$

Then, we easily calculate that the minimum of the expression $\max\{|\lambda_1|, |\lambda_2|\}$ will be achieved when $\gamma = \gamma_1 = 9 - 4\sqrt{5} \approx 0.055$. Thus, we conclude that $\rho(C_\infty(\gamma_1)) = |\lambda_1| = |\lambda_2| = |18 - 8\sqrt{5}|/|3\sqrt{5} - 7| \approx 0.381$.

We remark that Dahlquist et al. [3] suggested another choice for γ. They tried to minimize the error constant of method (2) and preserve good stability properties. Their choice was $\gamma_2 = 1/5$.

Formula (2) implies that the step size τ is fixed. Unfortunately, the latter requirement is too restrictive for many practical problems. Therefore we determine continuous extensions to nonuniform grids for both methods (2) with different γ's and come to the following formulas:

$$\begin{aligned}
x_{k+1} + (8 - 4\sqrt{5})x_k + (4\sqrt{5} - 9)x_{k-1} = \tau_k \bigg(& \frac{\theta_k^2 + (2\theta_k + 1)(9 - 4\sqrt{5})}{2\theta_k(\theta_k + 9 - 4\sqrt{5})} \\
\times g(t_{k+1}, x_{k+1}) + \frac{\theta_k^2(2\sqrt{5} - 4) + 76 - 34\sqrt{5}}{\theta_k(\theta_k + 9 - 4\sqrt{5})} & g(t_k, x_k) \\
+ \frac{(9 - 4\sqrt{5})(\theta_k^2 + 2\theta_k + 9 - 4\sqrt{5})}{2\theta_k(\theta_k + 9 - 4\sqrt{5})} & g(t_{k-1}, x_{k-1}) \bigg),
\end{aligned} \tag{3}$$

$$\begin{aligned}
x_{k+1} - \frac{4}{5}x_k - \frac{1}{5}x_{k-1} = \tau_k \bigg(& \frac{5\theta_k^2 + 2\theta_k + 1}{2\theta_k(5\theta_k + 1)} g(t_{k+1}, x_{k+1}) \\
+ \frac{10\theta_k^2 - 2}{5\theta_k(5\theta_k + 1)} g(t_k, x_k) + & \frac{5\theta_k^2 + 10\theta_k + 1}{10\theta_k(5\theta_k + 1)} g(t_{k-1}, x_{k-1}) \bigg)
\end{aligned} \tag{4}$$

where τ_k is a current step size of the nonuniform grid w_τ with a diameter τ (i.e., $\tau \overset{\text{def}}{=} \max_k\{\tau_k\}$) and $\theta_k \overset{\text{def}}{=} \tau_k/\tau_{k-1}$ is a ratio of adjacent step sizes. We have used our choice for γ, i.e. γ_1, in formula (3) and γ_2 to obtain method (4).

3   Local and Global Errors Estimation

We recall that both methods (3) and (4) are A-stable on an arbitrary nonuniform grid. Thus, we control step sizes by the accuracy requirement only. With this idea in mind, we impose the following restriction on the step size change:

$$\tau/\tau_{\min} \leq \Omega < \infty. \tag{5}$$

Formula (5) implies that the ratio of the maximum step size to the minimum one is bounded with the constant Ω. We need the latter formula for the local-global step size control to be correct (see, for example, [10]). On the other hand, any code solving real life problems must be provided with bounds for the maximum step size and for the minimum one, that is equivalent to (5), because of an asymptotic form of the theory of ODE methods and round-off errors. Thus, condition (5) gives us nothing new in practice.

Further, we present the theory of local and global errors computation for methods (3) and (4) together. So, it is convenient to consider the family of numerical methods [3] in the general form

$$\frac{\theta_k}{\theta_k + \gamma}x_{k+1} + \frac{\theta_k(\gamma - 1)}{\theta_k + \gamma}x_k - \frac{\theta_k\gamma}{\theta_k + \gamma}x_{k-1}$$
$$= \tau_k\frac{\theta_k^2 + (2\theta_k + 1)\gamma}{2(\theta_k + \gamma)^2}g(t_{k+1}, x_{k+1}) + \tau_k\frac{(1 - \gamma)(\theta_k^2 - \gamma)}{2(\theta_k + \gamma)^2}g(t_k, x_k) \tag{6}$$
$$+\tau_k\frac{\gamma(\theta_k^2 + 2\theta_k + \gamma)}{2(\theta_k + \gamma)^2}g(t_{k-1}, x_{k-1})$$

where γ is the free parameter and θ_k is the most recent step size ratio.

For method (6), the standard theory in [8] gives

$$\Delta\tilde{x}_{k+1} \approx \frac{-1}{6}\left(a_0(k)I_n - \tau_k b_0(k)\partial_x g(t_{k+1}, \tilde{x}_{k+1})\right)^{-1}\tilde{x}_{k+1}^{(3)}$$
$$\times \tau_k^3 \sum_{i=1}^{2}\left(a_i(k)\psi_i^3(\theta_k) + 3b_i(k)\psi_i^2(\theta_k)\right), \tag{7}$$

$$\Delta x_{k+1} \approx \left(a_0(k)I_n - \tau_k b_0(k)\partial_x g(t_{k+1}, x_{k+1})\right)^{-1}$$
$$\times \sum_{i=1}^{2}\left(\tau_k b_i(k)\partial_x g(t_{k+1-i}, x_{k+1-i}) - a_i(k)I_n\right)\Delta x_{k+1-i} + \Delta\tilde{x}_{k+1}, \tag{8}$$

$k = l - 1, l, \ldots, K - 1$, where $a_i(k)$ and $b_i(k)$, $i = 0, 1, 2$, are the correspondent coefficients of method (6), and functions ψ_i in formula (7) are defined as follows:

$$\psi_i(\theta_k) \overset{\text{def}}{=} 1 + \sum_{m=1}^{i-1}\theta_k^{-1}, \quad i = 1, 2.$$

Here, the corrected numerical solution $\tilde{x}_{k+1} \overset{\text{def}}{=} x_{k+1} + \Delta x_{k+1}$ is of order 3, $\partial_x g(t_{k+1}, x_{k+1})$ denotes a partial derivative of the mapping $g(t_{k+1}, x_{k+1})$ with respect to the second variable, I_n is the identity matrix of dimension n. The starting errors Δx_k, $k = 0, 1, \ldots, l - 1$, are considered to be zero because the starting values are computed accurately enough (see the starting procedure in [10] or [11]).

Note that formulas (7) and (8) have been derived with errors of $O(\tau^4)$ and $O(\tau^3)$, respectively. We also point out that formula (3) has been given in a slightly different form than it was presented in [8]. Here, we have derived the local error of method (6) with respect to the more recent step size τ_k and the necessary step size ratio θ_k rather than with respect to the step sizes τ_k and τ_{k-1}. We have done that for a convenience of presentation of further results concerning derivative computation.

To calculate the approximate derivative $\tilde{x}_{k+1}^{(3)}$ (at most with an error of $O(\tau)$) one can use a Newton (or Hermite) interpolation formula of sufficiently high degree [8]. On the other hand, it imposes an unnecessarily stiff smoothness requirement. Therefore we show how to adapt the method of derivative computation in [9] to variable-coefficient method (6).

First of all we introduce the matrix

$$
V_k(l,s) \stackrel{\text{def}}{=}
\begin{pmatrix}
1 & 0 & 0 & \cdots & 0 \\
1 & \left(-\psi_1(\Theta_k)\right)^1 & \left(-\psi_1(\Theta_k)\right)^2 & \cdots & \left(-\psi_1(\Theta_k)\right)^s \\
1 & \left(-\psi_2(\Theta_k)\right)^1 & \left(-\psi_2(\Theta_k)\right)^2 & \cdots & \left(-\psi_2(\Theta_k)\right)^s \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & \left(-\psi_l(\Theta_k)\right)^1 & \left(-\psi_l(\Theta_k)\right)^2 & \cdots & \left(-\psi_l(\Theta_k)\right)^s \\
0 & 1 & 0 & \cdots & 0 \\
0 & 1 & 2\left(-\psi_1(\Theta_k)\right)^1 & \cdots & s\left(-\psi_1(\Theta_k)\right)^{s-1} \\
0 & 1 & 2\left(-\psi_2(\Theta_k)\right)^1 & \cdots & s\left(-\psi_2(\Theta_k)\right)^{s-1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 1 & 2\left(-\psi_{l-s+1}(\Theta_k)\right)^1 & \cdots & s\left(-\psi_{l-s+1}(\Theta_k)\right)^{s-1}
\end{pmatrix}
\tag{9}
$$

where

$$
\psi_i(\Theta_k) \stackrel{\text{def}}{=} \psi_i(\theta_k, \theta_{k-1}, \ldots, \theta_{k-l+1}) \stackrel{\text{def}}{=} 1 + \sum_{m=1}^{i-1} \prod_{j=1}^{m} \theta_{k+1-j}^{-1}, \quad i = 1, 2, \ldots, l, \tag{10}
$$

for any l-step method of order s, when computing the $(s+1)$-th derivative of a numerical solution. Formula (9) is a generalization of the extended Vandermonde matrices in [9] to nonuniform grids. The principal point for us is that the matrix $V_k(l,s)$ is nonsingular for any grid. The latter follows from Lemma 2 in the paper mentioned above, formula (10) and the fact that all the step size ratios are positive. Then, the way presented in [9] gives a formula for computation of the necessary derivative $\tilde{x}_{k+1}^{(3)}$ with an error of $O(\tau)$. Thus, formula (7) is transformed to the convenient form

$$
\Delta \tilde{x}_{k+1} \approx \left(a_0(k)I_n - \tau_k b_0(k)\partial_x g(t_{k+1}, \tilde{x}_{k+1}) \right)^{-1} \tau_k \sum_{i=1}^{2} c_i(k)g\left(t_{k+1-i}, \tilde{x}_{k+1-i}\right)
$$

where

$$
c_0(k) = \frac{-P(\theta_k)}{6\theta_k^2(\theta_k + 1)(\theta_k + \gamma)}, \quad c_1(k) = \frac{P(\theta_k)}{6\theta_k^2(\theta_k + \gamma)}, \quad c_2(k) = \frac{-P(\theta_k)}{6\theta_k(\theta_k + 1)(\theta_k + \gamma)}
$$

and $P(\theta_k) = \theta_k^4 + 4\gamma\theta_k^3 + 6\gamma\theta_k^2 + 4\gamma\theta_k + \gamma^2$ is a polynomial with respect to the parameter γ, being a fixed number, and the most recent step size ratio θ_k.

We further refer to [10] or [11] for the local-global step size control algorithm and for the starting procedure. The step size selection is based on the error estimates presented above.

4   Numerical Experiments

In this section, we give a number of numerical examples confirming the efficiency of the methods presented above for nonstiff and stiff integrations as well. We start with numerical experiments on problems with known solutions. They are nonstiff, and our goal is to check the capacity of both algorithms with the local-global step size control to attain the set accuracy of computation in automatic mode.

The first test problem is taken from [6] and it has the form

$$x_1'(t) = 2tx_2(t)^{\frac{1}{5}}x_4(t), \quad x_2'(t) = 10t\exp\Big(5\big(x_3(t) - 1\big)\Big)x_4(t), \tag{11a}$$

$$x_3'(t) = 2tx_4(t), \quad x_4'(t) = -2t\ln\big(x_1(t)\big), \quad t \in [0, 3] \tag{11b}$$

with $x(0) = (1, 1, 1, 1)^T$. Problem (11) possesses the exact solution

$$x_1(t) = \exp\big(\sin t^2\big), \ x_2(t) = \exp\big(5\sin t^2\big), \ x_3(t) = \sin t^2 + 1, \ x_4(t) = \cos t^2.$$

Therefore it is convenient to verify how our adaptive methods will reach the required accuracy.

The second problem is quite practical. This is the restricted three body problem (see, for example, [6]):

$$x_1''(t) = x_1(t) + 2x_2'(t) - \mu_1\frac{x_1(t) + \mu_2}{y_1(t)} - \mu_2\frac{x_1(t) - \mu_1}{y_2(t)}, \tag{12a}$$

$$x_2''(t) = x_2(t) - 2x_1'(t) - \mu_1\frac{x_2(t)}{y_1(t)} - \mu_2\frac{x_2(t)}{y_2(t)}, \tag{12b}$$

$$y_1(t) = \Big((x_1(t)+\mu_2)^2+x_2(t)^2\Big)^{3/2}, \quad y_2(t) = \Big((x_1(t)-\mu_1)^2+x_2(t)^2\Big)^{3/2}, \tag{12c}$$

where $t \in [0, T]$, $T = 17.0652165601579625588891$, $\mu_1 = 1 - \mu_2$ and $\mu_2 = 0.012277471$. The initial values of problem (12) are: $x_1(0) = 0.994$, $x_1'(0) = 0$, $x_2(0) = 0$, $x_2'(0) = -2.00158510637908252240$. It has no analytic solution, but its solution-path is periodic. Thus, we are also capable to observe the work of both methods in practice.

Having fixed the global error bounds and computed the local tolerances by the formula $\epsilon_l = \epsilon_g^{3/2}$, we apply methods (3) and (4) with the local-global step size control to problems (11) and (12) and come to the data collected in Tables 1, 2. We see that both choices of the parameter γ in the family of numerical

Table 1. Global errors obtained for variable-coefficient methods (3) and (4) (with the local-global step size control) applied to problem (11)

Method	required accuracy				
	$\epsilon_g = 10^{-01}$	$\epsilon_g = 10^{-02}$	$\epsilon_g = 10^{-03}$	$\epsilon_g = 10^{-04}$	$\epsilon_g = 10^{-05}$
(3)	8.005×10^{-02}	7.305×10^{-03}	7.281×10^{-04}	9.702×10^{-05}	9.823×10^{-06}
(4)	7.191×10^{-02}	9.041×10^{-03}	8.648×10^{-04}	7.758×10^{-05}	7.566×10^{-06}

Table 2. Global errors obtained for variable-coefficient methods (3) and (4) (with the local-global step size control) applied to problem (12)

Method	required accuracy				
	$\epsilon_g = 10^{-01}$	$\epsilon_g = 10^{-02}$	$\epsilon_g = 10^{-03}$	$\epsilon_g = 10^{-04}$	$\epsilon_g = 10^{-05}$
(3)	$2.083 \times 10^{+00}$	9.445×10^{-03}	7.113×10^{-04}	7.077×10^{-05}	7.081×10^{-06}
(4)	9.092×10^{-02}	9.373×10^{-03}	7.704×10^{-04}	7.703×10^{-05}	7.714×10^{-06}

methods (6) lead to quite nice results. Both methods have computed the numerical solutions with the set accuracy. We only want to point out that our choice (method (3)), when $\gamma = 9 - 4\sqrt{5}$, gives the required numerical solutions faster. The average execution time for method (3) is less by a factor of 1.4 for the first test problem and by a factor of 1.3 for the second test problem compared with method (4).

Now we try methods (3) and (4) on the Van der Pol's equation

$$x_1'(t) = x_2(t), \quad x_2'(x) = \mu^2 \Big(\big(1 - x_1(t)^2\big) x_2(t) - x_1(t) \Big), \quad t \in [0, 2] \qquad (13)$$
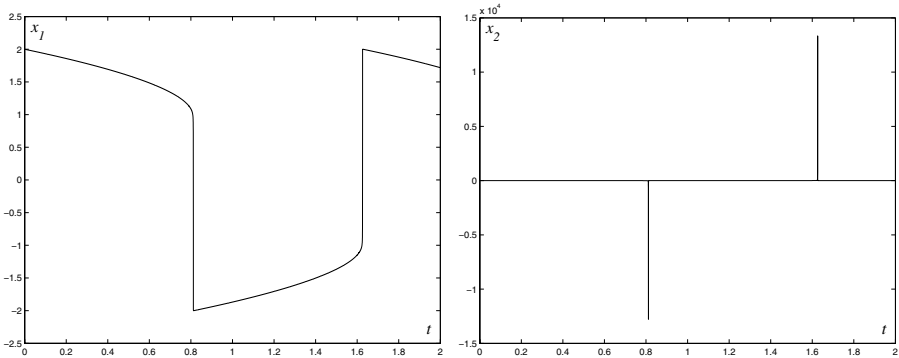

Fig. 1. The components x_1 and x_2 of the Van der Pol's equation calculated by methods (3) and (4) with $\epsilon_g = 10^{-1}$

where $x(0) = (2, 0)^T$, and $\mu = 100$. Problem (13) is considered to be very stiff when the parameter μ is a big number. Despite the small order of the methods under consideration the results obtained are quite promising. The components of the numerical solution of problem (13) are given in Figure 1. Both methods have produced the same result (up to an error of 10^{-1}) which completely corresponds to the picture in [7].

The final point to mention is that our choice of γ (method (3)) has again computed the numerical solution of Van der Pol's equation faster (with a factor of 1.4). This is a good reason to implement it in practice.

References

1. Butcher, J.C.: Numerical methods for ordinary differential equations. John Wiley and Son, Chichester, 2003
2. Dahlquist, G.: A special stability problem for linear multistep methods. BIT. **3** (1963) 27–43
3. Dahlquist, G.G., Liniger W., Nevanlinna, O.: Stability of two-step methods for variable integration steps. SIAM J. Numer. Anal. **20** (1983) 1071–1085
4. Dekker, K., Verwer, J.G. Stability of Runge-Kutta methods for stiff nonlinear differential equations. North-Holland, Amsterdam, 1984
5. Ehle B.L.: On Padé approximations to the exponential function and A-stable methods for the numerical solution of initial value problems. Research report CSRR 2010, Dept. AACS, Univ. of Waterloo, Ontario, Canada, 1969
6. Hairer, E., Nørsett, S.P., Wanner, G.: Solving ordinary differential equations I: Nonstiff problems. Springer-Verlag, Berlin, 1987
7. Hairer, E., Wanner, G.: Solving ordinary differential equations II: Stiff and differential-algebraic problems. Springer-Verlag, Berlin, 1996
8. Kulikov, G.Yu., Shindin, S.K.: A technique for controlling the global error in multistep methods. (*in Russian*) Zh. Vychisl. Mat. Mat. Fiz. **40** (2000) No. 9, 1308–1329; *translation in* Comput. Math. Math. Phys. **40** (2000) No. 9, 1255–1275
9. Kulikov, G.Yu., Shindin, S.K.: On effective computation of asymptotically correct estimates of the local and global errors for multistep methods with fixed coefficients. (*in Russian*) Zh. Vychisl. Mat. Mat. Fiz. **44** (2004) No. 5, 847–868; *translation in* Comput. Math. Math. Phys. **44** (2004) No. 5, 794–814
10. Kulikov, G.Yu., Shindin, S.K.: On interpolation-type multistep methods with automatic global error control. (*in Russian*) Zh. Vychisl. Mat. Mat. Fiz. **44** (2004) No. 8, 1400–1421; *translation in* Comput. Math. Math. Phys. **44** (2004) No. 8, 1314–1333
11. Kulikov, G.Yu., Shindin, S.K.: On automatic global error control in multistep methods with polynomial interpolation of numerical solution. In: Antonio Lagana et al (eds.): Computational Science and Its Applications — ICCSA 2004. International Conference, Assisi, Italy, May 2004. Proceedings, Part III. Lecture Notes in Computer Science. **3045** (2004) 345–354