Martin–Luther–Universität
Halle–Wittenberg
Institut für Mathematik

Global Error Control in Implicit Parallel Peer Methods

G. Yu. Kulikov, R. Weiner

Report No. 16 (2008)

Editors:
Professors of the Institute for Mathematics, Martin-Luther-University Halle-Wittenberg.

Electronic version: see `http://www2.mathematik.uni-halle.de/institut/reports/`

Global Error Control in Implicit Parallel Peer Methods

G. Yu. Kulikov, R. Weiner

Report No. 16 (2008)

Rüdiger Weiner
Martin-Luther-Universität Halle-Wittenberg
Naturwissenschaftliche Fakultät III
Institut für Mathematik
Theodor-Lieser-Str. 5
D-06120 Halle/Saale, Germany
Email: ruediger.weiner@mathematik.uni-halle.de

Gennady Kulikov
School of Computational and Applied Mathematics
University of the Witwatersrand
Private Bag 3
Wits 2050
Johannisburg, South Africa
Email: Gennady.Kulikov@wits.ac.za

Global Error Control in Implicit Parallel Peer Methods

G. Yu. Kulikov[*]and R. Weiner[†]

Abstract

Recently, Schmitt, Weiner and Erdmann proposed an efficient family of numerical methods termed Implicit Parallel Peer (IPP) methods. They are a subclass of s-stage general linear methods of order $s-1$. Most importantly, all stage values of those methods possess the same properties in terms of stability and accuracy of numerical integration. This property results in the fact that no order reduction occurs when they are applied to very stiff differential equations. The special construction of IPP methods allows for a parallel implementation, which is advantageous in modern high-performance computations.

In this paper, we add one more useful functionality to IPP methods, i.e. an automatic global error control. We show that the global error estimation developed by Kulikov and Shindin in multistep formulas is suitable for the methods of Schmitt, Weiner and Erdmann. Moreover, that global error estimation can be done in parallel. An algorithm of efficient stepsize selection is also discussed here.

Keywords: ordinary differential equations, implicit parallel peer methods, local and global error estimations, automatic global error control.
AMS Subject Classification: 65L05, 65L06, 65L50, 65L70.

1 Introduction

Over last three years, Weiner, Schmitt, Podhaisky and Erdmann [13], [14], [15], [16], [17] developed a number of numerical techniques to solve ordinary differential equations (ODE's) of the form

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

where $x(t) \in \mathbb{R}^m$ and $g : D \subset \mathbb{R}^{m+1} \to \mathbb{R}^m$ is a sufficiently smooth function. Problem (1) is assumed to have a unique solution $x(t)$ on the interval $[t_0, t_{end}]$. All those methods are attributed as "peer" methods because they produce a set of numerical solutions which

[*]School of Computational and Applied Mathematics, University of the Witwatersrand, Private Bag 3, Wits 2050, Johannesburg, South Africa. E-mail: Gennady.Kulikov@wits.ac.za

[†]Institut für Mathematik, Martin-Luther-Universität Halle-Wittenberg, Postfach, D-06099 Halle, Germany. Email: weiner@mathematik.uni-halle.de

share the same stability and accuracy properties in each step. Here, we deal with a fully implicit version of those methods.

First of all we denote a variable grid with a diameter τ on the integration interval $[t_0, t_{end}]$ by

$$w_\tau = \{t_{k+1} = t_k + \tau_k,\ k = 0, 1, \ldots, K-1,\ t_K = t_{end}\}$$

where $\tau = \max_{0\le k\le K-1}\{\tau_k\}$. Then, an s-stage IPP method reads

$$x_{ki} - \tau_k \gamma_i(k) g(t_{ki}, x_{ki}) = \sum_{j=1}^{s} b_{ij}(k) x_{k-1,j} + \tau_k \sum_{j=1}^{s} a_{ij}(k) g(t_{k-1,j}, x_{k-1,j}) \qquad (2)$$

where $t_{ki} := t_k + c_i\tau_k$, $i = 1, 2, \ldots, s$. The extra nodes t_{ki} of the grid w_τ are fixed by the constants c_i. Without loss of generality, we consider further that these fixed coefficients are distinct and ordered as follows: $c_1 < c_2 < \ldots < c_s$. Other coefficients $\gamma_i(k)$, $a_{ij}(k)$ and $b_{ij}(k)$, $i, j = 1, 2, \ldots, s$, are variable and depend on the stepsize ratios $\theta_k := \tau_k/\tau_{k-1}$. Formula (2) gives us a way to calculate the stage value x_{ki} independently of other information from the local interval $[t_k, t_{k+1}]$. That is why the computation of each stage value x_{ki} can be done in parallel.

Schmitt et al. [16] show how to choose the free parameters $\gamma_i(k)$, $a_{ij}(k)$, $b_{ij}(k)$ and c_i in method (2) to provide zero-stable numerical integration for stepsize sequences restricted only as follows:

$$0 < \theta_k \le \omega < \infty, \quad k = 1, 2, \ldots, K-1, \qquad (3)$$

where the constant ω satisfies the condition $\omega > 1$. They emphasize that all the numerical solutions x_{ki}, $i = 1, 2, \ldots, s$, obtained by the peer method (2) in the local interval $[t_k, t_{k+1}]$ are of the same order $s-1$. Thus, the IPP methods have no difficulty with dense output.

In our further consideration we deal with the subclass of zero-stable IPP methods satisfying the conditions described above. Thus, we restrict all grids w_τ in this paper as indicated in formula (3) and denote the set of such grids by $\mathbb{W}^\infty_\omega(t_0, t_{end})$. The symbol "$\infty$" means that the ratio of the maximum step size to the minimum step size may be unlimited for grids belonging to the set $\mathbb{W}^\infty_\omega(t_0, t_{end})$, as the diameter $\tau \to 0$.

IPP method (2) can be interpreted as a general linear method (see [1] for the definition of general linear methods and [16] for the interpretation). For the purpose of the present paper, it is convenient to treat method (2) as a set of zero-stable variable-stepsize linear multistep formulas applied to ODE's (1) on the local interval $[t_{k-1}, t_{k+1}]$. The non-equidistant mesh used in each subinterval $[t_{k-1}, t_{k+1}]$ is fixed and determined completely by the constant coefficients c_i. We stress also that all multistep formulas in (2) are implemented independently. So we merely apply the global error estimation developed by Kulikov and Shindin [3] in zero-stable variable-stepsize multistep formulas to IPP scheme (2) also in parallel.

Additionally, we want to mention that automatic global error control in numerical schemes of different sorts is an important issue of numerical analysis starting with 1970's. There is available a great variety of results published on this topic. On the other hand, the problem of global error evaluation and control is still under serious consideration in modern research because of theoretical interest and practical importance as well. Here, we apply the global error estimation and control technique developed by Kulikov and Shindin because of two reasons. First, it fits nicely to the IPP methods and its extension to scheme (2) is straightforward. Second, Kulikov and Shindin's error estimation is powerful and works efficiently in different numerical methods such that fixed-coefficient and variable-coefficient multistep formulas (even weakly stable) applied to both ordinary differential equations and index 1 semi-explicit differential-algebraic systems (see [3], [4], [5], [7], [8]) as well as in one-leg variable-stepsize formulas [9]. Recently, that technique was extended successfully to Nordsieck methods (see [10], [11]).

In Sect. 2, we start our discussion with the error estimation in the zero-stable method (2) of the same order $s-1$ in all the stage values x_{ki}, $i=1,2,\ldots,s$. There, we present the theory of local and global error estimation in the numerical schemes under consideration. Then, we give technical particulars of practical implementation of the error evaluation algorithms as well as stepsize selection based on joint control of the local and global errors in Sect. 3. In Sect. 4, the quality of our error estimation and the capacity to attain the required accuracy of computation in automatic mode are checked on numerical examples with known exact solutions. The last section of this paper summarizes the results obtained and outlines our future plans.

2 Local and global error estimations

Once again, we deal further with the IPP methods (2) satisfying the following conditions: First, the methods are zero-stable on any grid $w_\tau \in \mathbb{W}^\infty_\omega(t_0,t_{end})$. Second, they have s stages and all the stage values calculated in the course of integration are of the same order $s-1$. In other words, the defect of the i-th formula in the scheme (2) is given by

$$\begin{aligned} L_i(t_{ki},x(t),\tau_k) &:= x(t_{ki}) - \tau_k\gamma_i(k)g\big(t_{ki},x(t_{ki})\big) - \sum_{j=1}^{s}\Big(b_{ij}(k)x(t_{k-1,j}) \\ &+ \tau_k a_{ij}(k)g\big(t_{k-1,j},x(t_{k-1,j})\big)\Big) = \Bigg(c_i^s - s\gamma_i(k)c_i^{s-1} \\ &- \sum_{j=1}^{s}\Big(b_{ij}(k)\Big(\frac{c_j-1}{\theta_k}\Big)^s + s\,a_{ij}(k)\Big(\frac{c_j-1}{\theta_k}\Big)^{s-1}\Big)\Bigg) \\ &\times \frac{\tau_k^s}{s!}x^{(s)}(t_k) + \mathcal{O}(\tau_k^{s+1}), \end{aligned} \tag{4}$$

as shown in [16]. We recall that all the stage values in the IPP methods under discussion are computed independently. Therefore it is sufficient to show how to evaluate the error of the i-th component of the numerical solution from the method (2).

In addition, we require all grids used below to satisfy the extra condition

$$\tau/\tau_k \le \Omega < \infty, \quad k=0,1,\ldots,K-1, \tag{5}$$

where Ω is a constant and τ is the diameter of grid w_τ. The set of grids satisfying (2) and (5) is denoted by $\mathbb{W}^{\Omega}_{\omega}(t_0, t_{end})$. The necessity of the extra condition (5) is explained clearly in [7], [8] or [11]. The grids in $\mathbb{W}^{\Omega}_{\omega}(t_0, t_{end})$ are referred further to as admissible grids.

In what follows, we apply the technique developed in [3] to derive formulas for error estimation in the IPP methods (2). We start with the definition of the defect in the i-th stage value given by formula (4) to yield

$$\begin{aligned} x(t_{ki}) &- x_{ki} - \tau_k \gamma_i(k)\Big(g\big(t_{ki}, x(t_{ki})\big) - g\big(t_{ki}, x_{ki}\big)\Big) \\ &= \sum_{j=1}^{s} b_{ij}(k)\big(x(t_{k-1,j}) - x_{k-1,j}\big) + \tau_k \sum_{j=1}^{s} a_{ij}(k)\Big(g\big(t_{k-1,j}, x(t_{k-1,j})\big) \\ &- g\big(t_{k-1,j}, x_{k-1,j}\big)\Big) + L_i(t_{ki}, x(t), \tau_k). \end{aligned}$$

This is simply the difference of (2) and (4). Expanding the terms $g\big(t_{ki}, x(t_{ki})\big)$ and $g\big(t_{k-1,j}, x(t_{k-1,j})\big)$ of the last formula in the Taylor series around the points (t_{ki}, x_{ki}) and $(t_{k-1,j}, x_{k-1,j})$, respectively, we obtain the following asymptotic result:

$$\begin{aligned} \Delta x_{ki} &- \tau_k \gamma_i(k) \partial_x g(t_{ki}, x_{ki}) \Delta x_{ki} = \sum_{j=1}^{s} b_{ij}(k) \Delta x_{k-1,j} \\ &+ \tau_k \sum_{j=1}^{s} a_{ij}(k) \partial_x g(t_{k-1,j}, x_{k-1,j}) \Delta x_{k-1,j} + L_i(t_{ki}, x(t), \tau_k) + \mathcal{O}(\tau^{2s-1}) \end{aligned} \tag{6}$$

where $\Delta x_{ki} := x(t_{ki}) - x_{ki}$ and $\partial_x g(t_{ki}, x_{ki})$ denotes the partial derivative of the mapping $g(t, x)$ with respect to the second argument evaluated at the point (t_{ki}, x_{ki}). In formula (6), we have taken into account that the methods under discussion are of order $s-1$ and the grids used satisfy condition (5). Here and below, we consider the diameter τ of any grid in the set $\mathbb{W}^{\Omega}_{\omega}(t_0, t_{end})$ to be sufficiently small.

Omitting the remaining term in (6) gives us a formula for the error evaluation in the form of

$$\begin{aligned} \Delta \bar{x}_{ki} &- \tau_k \gamma_i(k) \partial_x g(t_{ki}, x_{ki}) \Delta \bar{x}_{ki} = \sum_{j=1}^{s} b_{ij}(k) \Delta \bar{x}_{k-1,j} \\ &+ \tau_k \sum_{j=1}^{s} a_{ij}(k) \partial_x g(t_{k-1,j}, x_{k-1,j}) \Delta \bar{x}_{k-1,j} + L_i(t_{ki}, x(t), \tau_k). \end{aligned} \tag{7}$$

The "bar" is used to distinguish the approximate value of the error calculated by (7) from the exact value mentioned in formula (6). If we assume now that the errors at grid points $t_{k-1,j}$, $j = 1, 2, \ldots, s$, are known precisely (i.e. $\Delta \bar{x}_{k-1,j} = \Delta x_{k-1,j}$) then the error equation derived will calculate the estimate to the error of the method (2) at the next time point t_{ki} accurate to $\mathcal{O}(\tau^{2s-1})$. It will be an approximation to the error of the IPP method committed in one step. The general convergence result is presented by

Theorem 1 *Let ODE* (1) *be sufficiently smooth and the coefficients $\gamma_i(k)$, $a_{ij}(k)$ and $b_{ij}(k)$, $i,j = 1,2,\dots,s$, of a zero-stable s-stage IPP method* (2) *of order $s-1$ be bounded on the set $\mathbb{W}^{\Omega}_{\omega}(t_0, t_{end})$ of admissible grids. Suppose that the starting values x_{0i}, $i = 1,2,\dots,s$, are given with errors of $O(\tau^s)$, and the errors $\Delta\bar{x}_{0i} := x(t_{0i}) - x_{0i}$, $i = 1,2,\dots,s$, in the starting values are known accurate to $O(\tau^r)$ where $r \geq s+1$. Then formula* (7) *converges with order* $\min\{r, 2s-2\}$ *to the error of method* (2) *on grids $w_\tau \in \mathbb{W}^{\Omega}_{\omega}(t_0, t_{end})$ as $\tau \to 0$.*

Proof. The principal idea of the proof is to represent the numerical scheme (2) as well as our error evaluation formula (7) in the form of a one-step method. Let us introduce the following vectors and matrices:

$$T_k := (t_{ki})_{i=1}^s, \quad X_k := (x_{ki})_{i=1}^s, \quad g(T_k, X_k) := g(t_{ki}, x_{ki})_{i=1}^s,$$

$$G(k) := \mathrm{diag}(\gamma_1(k), \gamma_2(k)\dots,\gamma_s(k)), A(k) := \big(a_{ij}(k)\big)_{i,j=1}^s, B(k) := \big(b_{ij}(k)\big)_{i,j=1}^s.$$

Then, scheme (2) can be written as follows:

$$\begin{aligned} X_k - \tau_k\big(G(k)\otimes I_m\big)g(T_k, X_k) &= \big(B(k)\otimes I_m\big)X_{k-1} \\ &+ \tau_k\big(A(k)\otimes I_m\big)g(T_{k-1}, X_{k-1}) \end{aligned} \tag{8}$$

where I_m is the identity matrix of dimension m and $\otimes$ denotes the Kronecker tensor product (see, for example, [12]). With use of the additional notation

$$\Delta\bar{X}_k := (\Delta\bar{x}_{ki})_{i=1}^s, \quad L(T_k, x(t), \tau_k) := \big(L_i(t_{ki}, x(t), \tau_k)\big)_{i=1}^s,$$

$$\partial_x g(T_k, X_k) := \mathrm{diag}\big(\partial_x g(t_{k1}, x_{k1}), \partial_x g(t_{k2}, x_{k2}), \dots, \partial_x g(t_{ks}, x_{ks})\big),$$

formula (7) takes the form

$$\begin{aligned} \Delta\bar{X}_k &- \tau_k\big(G(k)\otimes I_m\big)\partial_x g(T_k, X_k)\Delta\bar{X}_k = \big(B(k)\otimes I_m\big)\Delta\bar{X}_{k-1} \\ &+ \tau_k\big(A(k)\otimes I_m\big)\partial_x g(T_{k-1}, X_{k-1})\Delta\bar{X}_{k-1} + L(T_k, x(t), \tau_k). \end{aligned} \tag{9}$$

It is quite clear that both schemes (8) and (9) belong to the same class of numerical methods. This means that they are stable and convergent under the same conditions. We recall that the following condition of zero-stability:

$$\Bigg\|\prod_{j=0}^{m} B(k-j)\Bigg\| \leq R, \quad m = 0,1,\dots,k-1, \quad k = 1,2,\dots,K-1, \tag{10}$$

where R is a finite constant, must be provided on the admissible grids. Condition (10) imposes an upper bound ω on the rate θ_k of stepsize change, as shown in (3). Thus, convergence of formula (9) follows from convergence of the underlying method (2). Then, we apply the technique developed earlier to prove Theorem 3 in [3] with corrections made in [7] and [8] and complete the proof of Theorem 1. ■

Theorem 1 says that the error evaluation scheme will produce good estimates of the error in convergent IPP methods of the form (2) if starting values x_{0i}, $i = 1,2,\dots,s$, are found with sufficient accuracy. The only difficulty is the correct evaluation of the defect $L_i(t_{ki}, x(t), \tau_k)$ in formula (7). We stress that, by definition, the precise defect evaluation requires the exact solution $x(t)$ to be known. This means that we are not in a position to compute the exact value of $L_i(t_{ki}, x(t), \tau_k)$, but, hopefully, we can replace it with a sufficiently accurate approximation. We follow the technique developed in [3] to do this in practice.

First, we remark that if the defect $L_i(t_{ki}, x(t), \tau_k)$ is computed with an error of $\mathcal{O}(\tau_k^{s+1})$ it will be sufficient for the correct evaluation of the principal term of the local and global errors of method (2) (see [3], [7] or [8] for more details). Second, to calculate such an estimate, we use the Taylor expansion of the defect $L_i(t_{ki}, x(t), \tau_k)$ that is similar to formula (4) but done around the time point t_{ki}. Having neglected all the terms of order $s+1$ or higher (with respect to the step size τ_k) we arrive at the following formula for the defect evaluation:

$$\begin{aligned} \tilde{L}_i(t_{ki}, x(t), \tau_k) \quad &:= \quad \frac{(-1)^{s+1}\tau_k^s x^{(s)}(t_{ki})}{s!} \sum_{j=1}^{s} \left(b_{ij}(k) \left(c_i + \frac{1-c_j}{\theta_k} \right) - a_{ij}(k)\, s \right) \\ &\times \quad \left(c_i + \frac{1-c_j}{\theta_k} \right)^{s-1} \end{aligned} \tag{11}$$

We emphasize that $\tilde{L}_i(t_{ki}, x(t), \tau_k) = L_i(t_{ki}, x(t), \tau_k) + \mathcal{O}(\tau_k^{s+1})$. In other words, formula (11) calculates correctly the principal term of $L_i(t_{ki}, x(t), \tau_k)$ only. Third, the last problem is to evaluate approximately the s-th derivative of the exact solution $x(t)$ at the point t_{ki}. It is obvious from (11) that it is sufficient to calculate this derivative with an error of $\mathcal{O}(\tau_k)$.

It is proved in [3] that such an estimate of the required derivative can be obtained by differentiation of the interpolating polynomial fitted to the points $g(t_{ki}, \tilde{x}_{ki})$ and $g(t_{k-1,j}, \tilde{x}_{k-1,j})$, $j = 2, 3, \ldots, s$. Here, the "tilde" means a numerical solution of problem (1) computed with the local error of $\mathcal{O}(\tau_k^{s+1})$ (or, in other words, by a numerical method of one order higher than the scheme (2)). In practice, we merely use the local extrapolation by the error estimates calculated to derive such a solution in the previous step, i.e. $\tilde{x}_{k-1,j} := x_{k-1,j} + \Delta\bar{x}_{k-1,j}$. The value $\tilde{x}_{ki}$ means further the numerical solution derived in one step of the method (2) on the basis of the extrapolated solution $\tilde{x}_{k-1,j}$, $j = 1, 2, \ldots, s$. Therefore, it is also found with an error of $\mathcal{O}(\tau_k^{s+1})$. All of these imply that we have to store the numerical solutions $x_{k-1,j}$ and their errors $\Delta\bar{x}_{k-1,j}$, $j = 1, 2, \ldots, s$, in order to advance a step of the IPP method (2) with error estimation (7).

Theorem 1 in [3] states that the $(s-1)$-th derivative of the above-mentioned interpolating polynomial (denoted further as $\tilde{x}_{ki}^{(s)}$) represents the s-th derivative of the solution with sufficient accuracy. Moreover, we have excluded one stage value from the consideration (namely $g(t_{k-1,1}, \tilde{x}_{k-1,1})$) in order to simplify the derivative estimation because $\tilde{x}_{ki}^{(s)}$ is just the $(s-1)$-th divided difference calculated for the above-mentioned values of the function $g(t,x)$ and multiplied by $(s-1)!$. So, having substituted the numerical derivative in (11) we arrive at the final formula for the approximate defect evaluation

$$\begin{aligned}\tilde{L}_i(t_{ki}, x(t), \tau_k) \quad &:= \quad \frac{(-1)^{s+1}\tau_k^s \tilde{x}_{ki}^{(s)}}{s!} \sum_{j=1}^{s} \Big(b_{ij}(k) \Big(c_i + \frac{1-c_j}{\theta_k} \Big) - a_{ij}(k)\, s \Big) \\ &\times \quad \Big(c_i + \frac{1-c_j}{\theta_k} \Big)^{s-1}\end{aligned} \tag{12}$$

Theoretically, we can apply expansion (4) to estimate the defect. It will reduce slightly CPU time in a sequential implementation because we will calculate the single divided difference per step of the method. On the other hand, formula (4) might calculate a poor approximation to the defect when the solution path varies rapidly or, might impose too strong a restriction on the step size τ_k. Moreover, the idea to replace the derivatives $x^{(s)}(t_{ki})$ with the single derivative $x^{(s)}(t_k)$ does not improve the parallel implementation of the IPP methods where each processor is assigned to compute one stage value and its error. Thus, we use further the defect evaluation formula (12).

Having substituted formula (12) in the error equation (7) we arrive at the formula computing the principal term of the global error:

$$\begin{aligned}\Delta_1 \bar{x}_{ki} \quad &- \quad \tau_k \gamma_i(k) \partial_x g(t_{ki}, x_{ki}) \Delta_1 \bar{x}_{ki} = \sum_{j=1}^{s} b_{ij}(k) \Delta_1 \bar{x}_{k-1,j} \\ &+ \quad \tau_k \sum_{j=1}^{s} a_{ij}(k) \partial_x g(t_{k-1,j}, x_{k-1,j}) \Delta_1 \bar{x}_{k-1,j} + \tilde{L}_i(t_{ki}, x(t), \tau_k).\end{aligned} \tag{13}$$

Subscript 1 is used to distinguish the principal term of the global error calculated by scheme (13) from the more accurate global error estimate mentioned in formula (7). For local error evaluation in the IPP methods, we assume that the errors in all previous time points are equal to zero. In other words, we simply substitute $\Delta_1 \bar{x}_{k-1,j} \equiv 0$, $j = 1, 2, \ldots, s$, in formula (13) to yield

$$\Delta_1 \tilde{x}_{ki} - \tau_k \gamma_i(k) \partial_x g(t_{ki}, \tilde{x}_{ki}) \Delta_1 \tilde{x}_{ki} = \tilde{L}_i(t_{ki}, x(t), \tau_k). \tag{14}$$

Formula (14) computes the principal term of the local error denoted further as $\Delta_1 \tilde{x}_{ki}$. The complete theoretical result for the global and local error estimations (13) and (14) is given by

Theorem 2 *Let ODE* (1) *be sufficiently smooth and the coefficients $\gamma_i(k)$, $a_{ij}(k)$ and $b_{ij}(k)$, $i, j = 1, 2, \ldots, s$, of a zero-stable s-stage IPP method* (2) *of order $s-1$ be bounded on the set $\mathbb{W}_\omega^\Omega(t_0, t_{end})$ of admissible grids. Suppose that the starting values x_{0i}, $i = 1, 2, \ldots, s$, are given with errors of $O(\tau^s)$, and the errors $\Delta_1 \bar{x}_{0i} := x(t_{0i}) - x_{0i}$, $i = 1, 2, \ldots, s$, in the starting values are known accurate to $O(\tau^{s+1})$. Then formulas* (13) *and* (14) *will calculate the principal terms of the global and local errors of method* (2)*, respectively, on any grid $w_\tau \in \mathbb{W}_\omega^\Omega(t_0, t_{end})$ with a sufficiently small diameter τ if the improved numerical solution*

$$\tilde{x}_{ki} := x_{ki} + \Delta_1 \bar{x}_{ki}, \quad k = 0, 1, \ldots, K-1, \quad i = 1, 2, \ldots, s, \tag{15}$$

is used to evaluate the defect of the method (2) *by formula* (12).

Theorem 2 can be proved in the same way as Theorems 2 and 4 in [3]. We want only to mention here that, theoretically, the local error estimate $\Delta_1\tilde{x}_{ki}$ is also suitable for the local extrapolation (15). However, we recommend the use of the real error estimate $\Delta_1\bar{x}_{ki}$ in practice.

3 Practical implementation and stepsize selection algorithm

In this section we discuss practical aspects of implementation of the local and global error evaluations developed above and concern an algorithm of automatic global error control. In addition, a recommendation for the correct solution of nonlinear problems arising from implementation of the IPP methods (2) is given. It is necessary for the adaptive numerical schemes with the global error control facility presented in this paper to work properly. We also explain how to compute the starting values x_{0i}, $i = 1, 2, \dots, s$, with an error of $\mathcal{O}(\tau^{s+1})$ in practice, that is required in Theorem 2 of the paper.

At first, we replace the Jacobian $\partial_x g(t_{ki}, \tilde{x}_{ki})$ in formula (14) with the Jacobian of the right-hand side of ODE (1) evaluated at the point (t_{ki}, x_{ki}), as in formula (13). It does not influence the accuracy of the local error estimation (14) but simplifies the computation, significantly. After the replacement, formulas (13) and (14) are two linear systems with the same coefficient matrix. So this coefficient matrix demands one *LU* decomposition for evaluation of both errors. Moreover, we recommend to apply the modified Newton iteration in practice. It means that the Jacobian of nonlinear problem (2) is calculated once per time point, i.e. at the point $(t_{ki}, \tilde{x}^0_{ki})$ where the initial guess $\tilde{x}^0_{ki}$ denotes the value of the numerical solution at the point t_{ki} predicted by polynomial interpolation based on the improved values $\tilde{x}_{k-1,j}$, $j = 1, 2, \dots, s$ (see Sect. 7 in [16] for more details). Then, one additional *LU* decomposition will be done once per time point to calculate the numerical solutions x_{ki} and $\tilde{x}^*_{ki}$. We recall that the local and global error estimators presented here use the defect of method (2). The defect calculation requires the higher-order numerical solution $\tilde{x}^*_{ki}$ to be known, and its computation is also done with the same *LU* decomposition as for x_{ki}. Note that we have used the "star" here to distinguish the higher-order numerical solution obtained in one step of the IPP method (2) on the basis of the improved values $\tilde{x}_{k-1,j}$, $j = 1, 2, \dots, s$, from the extrapolated numerical solution $\tilde{x}_{ki}$ calculated by formula (15).

We stress that when calculation of the error $\Delta_1\bar{x}_{ki}$ is completed and this error satisfies the accuracy requirement the vector $\partial_x g(t_{ki}, x_{ki})\Delta_1\bar{x}_{ki}$ will be stored to advance the next step of global error estimation (13). As output, we take the numerical solution $\tilde{x}_{ki}$ (i.e improved by local extrapolation (15)). Thus, our adaptive numerical schemes with the global error control admit effective parallel implementation.

Next, we have to discuss practical details of the local and global error evaluations in the implicit numerical methods considered in this paper. In general, they demand an additional (iterative) scheme to treat nonlinear discrete problems (2). Thus, we have to understand how to control the error in the complex numerical methods consisting of a stable IPP method and an iteration because the stepsize selection algorithm below is aimed only to monitor and control the error of the IPP method (2), but not the error of the iteration used. A proper solution can be to ensure that the iteration does not influence the local and global error estimates dramatically. This means that the error of the iteration must be of order $s+1$ at least, with respect to the grid diameter τ, at each point of the generated grid. In other words, it must show the same asymptotic behaviour as the neglected terms of error estimators (13) and (14).

First, Theorems 6,7 and 8 in [3] are applicable to the error evaluation technique developed for the IPP methods (2). We have only to replace the order s of multistep methods considered in [3] with the order $s-1$ of the IPP schemes. We point out that Theorems 6,7 and 8 work for numerical methods of order 1 or higher. It follows from the corrections made in [7] and [8]. Second, we recall that the improved values $\tilde{x}_{k-1,j}$, $j=1,2,\ldots,s$ are used to calculated a good initial guess for an iteration applied to discrete problem (2), as explained in Sect. 7 of [16]. Then we conclude that one iteration step per grid point will be sufficient in the iterative schemes considered in [3], to ensure proper stepsize selection. We remark that the nontrivial predictor mentioned above is vital for this.

Nevertheless, we recommend two modified Newton iteration steps per grid point in order to treat discrete problem (2) in a practical situation because of the following reasons:

- Newton-type iterations can be applied to a wider variety of ODE's (1) than the fixed-point one. That is why we prefer Newton-type iterations.

- The modified Newton iteration, when the Jacobian is calculated and decomposed once per grid point, is cheaper than the full Newton iteration.

- Two modified Newton iteration steps are not much more time-consuming than the single iteration step because the second iteration step implies the solution of a linear problem with decomposed coefficient matrix. On the other hand, two iteration steps provide more confidence that the error of the modified Newton iteration does not influence crucially the accuracy of the global error estimation in the IPP methods. Notice that Theorems 6,7 and 8 in [3], which give the lower bound restriction on the number of iterates at each time point for the iterative schemes considered there, will work properly when the diameter of the grid is sufficiently small; i.e., as $\tau \to 0$. In practice, we deal with finite step sizes produced by a stepsize selection algorithm, which might not be small enough for the fast iteration convergence. Therefore, it is wise to do one more iteration step to increase robustness of the code. One more reason to do that is to preserve stability of numerical integration, especially, when applied to stiff initial value problems.

Certainly, a more sophisticated mechanism can be implemented to control the error of iterations in the course of integration. Corresponding results are available in the literature on numerical integration of ODE's by adaptive implicit methods. However, we restrict ourselves to two modified Newton iteration steps per grid point in this paper. We recall that the more accurate solution $\tilde{x}_{ki}$ is taken as the output of our numerical scheme. In other words, we generate the grid for a method of order $s-1$ but perform the integration by a method of order s. So we usually obtain a numerical solution that is more accurate than is required. This supply of accuracy is also expected to compensate for the iterative error as well as for other errors appearing in numerical integration.

Next, it is important to mention here that our local and global error evaluations will work correctly only if the starting values x_{0i}, $i = 1, 2, \dots, s$, are found with errors of $O(\tau^s)$ and the errors $\Delta_1 \bar{x}_{0i}$, $i = 1, 2, \dots, s$, in the starting values are known accurate to $O(\tau^{s+1})$ (see Theorem 2 in Sect. 2). Moreover, we desire the starting values to be calculated with errors not exceeding a given tolerance ϵ_g. Therefore an additional starting procedure is required. Two approaches are possible. First, we can follow the technique presented for multistep formulas with global error control facility in [5] (see also [7] for more detail). Its principal idea is to determine a numerical solution to the given problem (1) with an error of $\mathcal{O}(\tau^{s+2})$ in a number of equidistant grid points, as explained in the cited papers, and to consider the initial errors to be equal to zero. The only difference to be taken into account here is an accommodation of the starting procedure designed earlier for multistep formulas to the IPP method (2), where stage values are treated as separate multistep formulas working on the fixed non-equidistant mesh $c_i\tau_0$, $i = 1, 2, \dots, s$, where τ_0 is a default value of the initial step size. Having computed the improved initial values $\tilde{x}_{0i}$, $i = 1, 2, \dots, s$ with an error of $\mathcal{O}(\tau^{s+1})$, because we deal with numerical schemes of order $s-1$ in this paper, we simply set the initial errors to be zero, i.e. $\Delta_1 \bar{x}_{0i} := 0$, $i = 1, 2, \dots, s$. Second, we merely apply a Runge-Kutta method of order s (or higher) with local error control. So, the initial values x_{0i}, $i = 1, 2, \dots, s$ are computed by s calls of the code implementing the above-mentioned method in one step of the size $c_i\tau_0$ and for the tolerance ϵ_g. In this paper, we choose the second approach and use code `DOP853` [2] for the starting. The initial errors are also considered to be zero because of the order of the Runge-Kutta formula used.

Now we consider a stepsize selection algorithm that aims to produce a numerical solution satisfying a user-supplied accuracy requirement in automatic mode. It means that a global error tolerance ϵ_g is fixed by the user. Here, we deal with the local-global error control mechanism presented in [3] in the first time (see also [5] or [7] for a pseudo-code). That error control uses reducing the maximum stepsize border to attain the required accuracy of computation. However, it was found later that better results (in terms of CPU time) can be obtained via the technique of reducing the local error tolerance. Therefore, we implement the Stepsize Selection IV algorithm developed in [11] in the adaptive IPP methods as follows:

Stepsize Selection IV

```
Step 0. Initially, we set $\epsilon_l := \epsilon_g^{s/(s-1)}$, $\Gamma \in (0, \infty]$ and suppose that $t_0 + \tau_0 \le t_{end}$;
Step 1.    $k := 0$, $M := 0$;
Step 2.    If $t_k < t_{end}$,
               then go to Step 3,
               else go to Step 10;
```

```
Step 3.    $t_{k+1} := t_k + \tau_k$, compute $\tilde{X}^*_{k+1}$, $\Delta_1\tilde{X}_{k+1}$;
Step 4.    $\tau^*_k := \tau_k \Big(\delta_1\epsilon_l/\|\Delta_1\tilde{X}_{k+1}\|\Big)^{1/s}$;
Step 5.    If $\|\Delta_1\tilde{X}_{k+1}\| > \epsilon_l$,
             then $\tau_k := \tau^*_k$, go to Step 3;
Step 6.    Compute $X_{k+1}$, $\Delta_1\bar{X}_{k+1}$;
Step 7.    If $\|\Delta_1\bar{X}_{k+1}\| > \epsilon_g$,
             then $M := 1$;
Step 8.    If $M = 1$ and $\|\Delta_1\bar{X}_{k+1}\| > \Gamma$,
             then go to Step 11;
Step 9.    $\tau_{k+1} := \min\{\tau^*_k, t_{end} - t_{k+1}\}$, $\bar{X}_{k+1} := X_{k+1} + \Delta_1\bar{X}_{k+1}$
           $k := k + 1$, go to Step 2;
Step 10.   if $M = 0$,
             then go to Step 12;
Step 11.   $\epsilon_l := \epsilon_l\Big(\delta_2\epsilon_g/\max_k\|\Delta_1\bar{X}_k\|\Big)^{s/(s-1)}$, go to Step 1;
Step 12. Stop.
```

The long vector X_{k+1} contains all stage values of method (2) calculated in the $(k+1)$-th step, i.e. $X_{k+1} := (x_{k1}, x_{k2}, \dots, x_{ks})$. Other long vectors $\tilde{X}^*_{k+1}$, $\Delta_1\tilde{X}_{k+1}$, $\Delta_1\bar{X}_{k+1}$ are defined similarly. The vector norm used in the algorithm is the usual sup-norm. In other words, we first calculate all the numerical solutions in the $(k+1)$-th step and only after that we check the accuracy of computation of all the stage values together. It matches to the parallel implementation of the numerical schemes under consideration.

Parameters δ_1, δ_2 and Γ have some default values, which can be changed by the user. δ_1 means a usual safety factor in our local error control. Parameter δ_2 plays the same role in the global error control mechanism. More precisely, they are set to be $\delta_1 = \delta_2 := 0.5$ in the experiments, below (see Sect. 4). The bound $\Gamma := 1$. It was introduced in the local-global stepsize selection algorithms in [11] in the first time. The idea is to allow the global error estimate $\Delta_1\bar{X}_{k+1}$ to exceed the user-supplied accuracy requirement ϵ_g in the course of integration. In this way, we get more information about its behaviour and, hence, calculate a better local error tolerance ϵ_l (control parameter) in *Step 11* of Stepsize Selection IV. This reduces the execution time of the code (see [11] for numerical examples).

We stress that the stepsize change in *Step 4* must provide a stable integration. This means that the factor to change a step size τ_k must be in the stability interval of the implemented IPP method, as indicated in (3). We refer to [16] for more information on stability intervals of the IPP methods under consideration. So, when the step size calculated in *Step 4* is outside the stability interval we will simply replace it with the new one calculated by $\tau^*_k := \omega\tau_k$. It is also important to set up the minimum step size τ_{min} and the maximum step size $\tau_{max} := \tau$ barriers in order to check after each change that $\tau^*_k \in [\tau_{min}, \tau_{max}]$. We use $\tau_{min} := 10^{-15}$ and $\tau_{max} := 10^{-02}$ in our codes. This is necessary to ensure condition (5) in practice. If $\tau^*_k > \tau_{max}$ we will set $\tau^*_k := \tau_{max}$. If $\tau^*_k < \tau_{min}$ the code reports that it fails to compute a numerical solution for the given accuracy requirement ϵ_g.

On the other hand, the global error evaluation developed in this paper is asymptotically correct only. It will work properly if the global error estimate obtained is sufficiently small. We recall that terms of $\mathcal{O}(\tau_k(\Delta_1 X_k)^2)$ have been omitted in the derivation of the error equation (7) (see Sect. 2). So, if $\|\Delta_1 \bar{X}_k\|$ is too big then we have no reason to consider the global error estimate computed as valid. Consequently, the improved numerical solution $\tilde{x}_k$ might not be sufficiently accurate to ensure the proper work of the global error evaluation algorithm. Eventually, the local error tolerance ϵ_l will not correlate with the real situation and might be unnecessarily small.

4 Numerical experiments

To examine the above-discussed error evaluation and stepsize selection algorithms in practice we take the following two test problems:

$$x_1'(t) = 2t x_2^{1/5}(t) x_4(t), \quad x_2'(t) = 10t \exp\Big(5\big(x_3(t) - 1\big)\Big) x_4(t), \tag{16a}$$

$$x_3'(t) = 2t x_4(t), \quad x_4'(t) = -2t \ln\big(x_1(t)\big), \tag{16b}$$

where $t \in [0, 3]$ and $x(0) = (1, 1, 1, 1)^T$;

$$x_1''(t) = x_1(t) + 2x_2'(t) - \mu_1 \frac{x_1(t) + \mu_2}{y_1(t)} - \mu_2 \frac{x_1(t) - \mu_1}{y_2(t)}, \tag{17a}$$

$$x_2''(t) = x_2(t) - 2x_1'(t) - \mu_1 \frac{x_2(t)}{y_1(t)} - \mu_2 \frac{x_2(t)}{y_2(t)}, \tag{17b}$$

$$y_1(t) = \Big((x_1(t) + \mu_2)^2 + x_2^2(t)\Big)^{3/2}, \tag{17c}$$

$$y_2(t) = \Big((x_1(t) - \mu_1)^2 + x_2^2(t)\Big)^{3/2} \tag{17d}$$

where $t \in [0, T]$, $T = 17.0652165601579625588917$, $\mu_1 = 1 - \mu_2$ and $\mu_2 = 0.012277471$. The initial values of problem (17) are: $x_1(0) = 0.994$, $x_1'(0) = 0$, $x_2(0) = 0$, $x_2'(0) = -2.0015851063790825224053$. Problem (16) has the exact solution

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

Therefore we can calculate exact errors of numerical integrations and compare them with error estimates obtained by formula (13). Such information will allow to comment on the quality of the error estimation procedure designed in this paper. Problem (17) has no analytic solution, but it is still useful to gain experience because its solution-path is periodic with the period T. Thus, we can monitor the error at the point T to verify the quality of numerical solutions computed.

In our tests, we take two samples of IPP methods (2). The first of them has 4 stages and is of order 3 (denoted as IPP3). The second one has 6 stages and is of order 5 (denoted as IPP5). In both cases we choose the coefficient matrix A to be zero. This is recommended for the best stability at infinity (see [16] for more detail). The other constant coefficients are:

- for IPP3:

$$c_1 = 0.1, \quad c_2 = 0.3, \quad c_3 = 0.7, \quad c_4 = 1, \quad \gamma_1 = 0.5924710362,$$

$$\gamma_2 = 0.6732567086, \quad \gamma_3 = 0.8348280534, \quad \gamma_4 = 0.9560065620;$$

- for IPP5:

$$c_1 = 0.1, \quad c_2 = 0.2, \quad c_3 = 0.3, \quad c_4 = 0.6, \quad c_5 = 0.8, \quad c_6 = 1,$$

$$\gamma_1 = 0.05000000000, \quad \gamma_2 = 0.07480736013, \quad \gamma_3 = 0.09961472026,$$
$$\gamma_4 = 0.17403680065, \quad \gamma_5 = 0.22365152091, \quad \gamma_6 = 0.27326624117.$$

Remaining coefficients b_{ij}, $i, j = 1, 2, \dots, s$, depend on the stepsize ratios θ_k and are calculated as shown in [16] (see Lemma 2). We point out that the above methods are chosen only for testing the global error estimation theory and are not the best in the class of IPP methods (2).

We want to check quality of the error estimation in our first experiment. For that, we apply IPP3 and IPP5 to problems (16) and (17) on equidistant grids. The numbers of steps to integrate numerically our tests problems are shown in Tables 1 and 2. For each numerical integration, we display the maximum of the exact error in the first row, the maximum of the error estimate (13) in the second row and the quality of the error estimation procedure (i.e. the maximum of their differences) in the third row.

Table 1: Global errors and their estimates for IPP3 and IPP5 applied to problem (16)

IPP	Number of Steps				
Method	300	600	1200	2400	4800
IPP3	4.089E+0000	5.411E-0001	6.847E-0002	8.592E-0003	1.075E-0003
	3.946E+0000	5.491E-0001	6.934E-0002	8.652E-0003	1.079E-0003
	4.295E-0001	1.147E-0002	8.729E-0004	6.026E-0005	3.874E-0006
IPP5	2.271E-0002	6.712E-0004	2.012E-0005	6.477E-0007	2.451E-0007
	2.277E-0002	6.726E-0004	2.009E-0005	6.455E-0007	3.188E-0008
	3.055E-0004	6.940E-0006	1.092E-0007	2.152E-0009	2.186E-0007

We see that the error evaluation mechanism developed in this paper works well. It gives us true information about the size of the global errors committed in the numerical integrations under discussion. However, we have to stress that our procedure is designed to calculate the principal term of the global error only. This means that the accuracy of the error estimation must raise as the step size tends to zero. This is exactly what we observe in our first experiment for both methods and test problems (16) and (17). On the other hand, round-off errors influence the results obtained for the higher order IPP method when the number of steps is sufficiently large (see Tables 1 and 2).

Table 2: Errors and their estimates evaluated at the point T for IPP3 and IPP5 applied to problem (17)

IPP	Number of Steps				
Method	40000	80000	160000	320000	640000
IPP3	5.144E-0001	2.936E-0001	6.041E-0002	9.010E-0003	1.222E-0003
	1.370E+0001	4.770E-0001	6.124E-0002	8.968E-0003	1.218E-0003
	1.413E+0001	1.834E-0001	1.538E-0003	4.979E-0005	4.694E-0006
IPP5	3.343E-0002	3.093E-0003	8.546E-0005	1.325E-0006	8.221E-0008
	1.464E-0001	6.191E-0003	1.363E-0004	2.295E-0006	2.873E-0008
	1.672E-0001	3.098E-0003	5.085E-0005	9.688E-0007	5.348E-0008

Thus, we have checked for two examples that the global error evaluation presented for the IPP methods (2) in this paper indeed works in practice. In our second experiment, we now want to verify the capacity of the adaptive IPP3 and IPP5 methods with Stepsize Selection IV algorithm to compute the numerical solution for a user-supplied accuracy requirement in automatic mode. Again, we apply the above-mentioned methods to problems (16) and (17) for a range of global error tolerances ϵ_g and collect the data in Tables 3 and 4. The computations are performed for $\Gamma := 1$. We also set $\omega := 1.6$ for IPP3 and $\omega := 1.3$ for IPP5 to provide stability of integration (see formula (3)).

Table 3: Global errors of the adaptive IPP3 and IPP5 methods with Stepsize Selection IV algorithm applied to problem (16)

IPP	Global Error Tolerance				
Method	ϵ_g=1.0E-02	ϵ_g=1.0E-03	ϵ_g=1.0E-04	ϵ_g=1.0E-05	ϵ_g=1.0E-06
IPP3	4.966E-0003	4.978E-0004	5.391E-0005	4.925E-0006	4.947E-0007
IPP5	6.203E-0003	5.271E-0004	5.032E-0005	5.712E-0006	9.178E-0007

Table 4: Errors evaluated at the point T for the adaptive IPP3 and IPP5 methods with Stepsize Selection IV algorithm applied to problem (17)

IPP	Global Error Tolerance				
Method	ϵ_g=1.0E-02	ϵ_g=1.0E-03	ϵ_g=1.0E-04	ϵ_g=1.0E-05	ϵ_g=1.0E-06
IPP3	3.898E-0003	4.973E-0004	5.006E-0005	3.817E-0006	4.221E-0007
IPP5	3.358E-0003	4.513E-0004	6.454E-0005	5.056E-0006	7.907E-0007

The second experiment confirms clearly the power of Stepsize Selection IV in controlling the global error of numerical integration. We see that the errors of all numerical solutions computed are below the global error tolerances set.

Sometimes, round-off errors and errors introduced by the iterative scheme might cause our technique to fail to calculate the numerical solution for some global error tolerances.

Nevertheless, it works very well in most cases and even when it fails the numerical solution is computed with the accuracy that is close to the requirement. Thus, all numerical experiments of Sect. 4 confirm practically the theory of global error estimation developed in the IPP methods (2) that creates a basis for further study in this area.

5 Conclusion

Our paper is the first step towards global error estimation in recently presented two-step peer methods [16]. One of their important features is the capacity for parallel implementation. This allows high-performance computers to be used for the discussed methods. The global error evaluation technique presented here is also designed for parallel implementation. It means that the adaptive IPP methods (2) with the developed global error control facility can be very efficient in terms of accuracy and execution time.

The numerical examples of this paper confirm our theoretical analysis and show how the global error estimation works in practice. They also confirm that the Stepsize Selection IV algorithm is able to control the global error of numerical solutions in automatic mode. All the numerical solutions were computed for the assigned accuracy requirements.

Certainly, this first paper does not cover everything in the area of error estimation for IPP methods. There is a space for further research. For example, we considered in the above theory that the method's nodes $c_i \in [0, 1]$. On the other hand, Schmitt et al. [16] presented powerful IPP methods satisfying the property $c_i \in [-1, 1]$. So, our global error control must be accommodated to the best IPP methods. We will also address the issue of efficiency of the error control algorithm (in terms of CPU time) in our future papers. Especially, we plan to cover the global error estimation and control in stiff ODE's. In addition, we want to boost the accuracy of our error estimation technique and to make it more robust to other errors involved in numerical integration. This can be done on the basis of multistep extrapolation technique presented in [4] for ODE's and in [8] for differential-algebraic systems of index 1.

Acknowledgements

The first author thanks the National Research Foundation of South Africa for partial support of his work under grant No. FA2004033000016.

References

[1] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, John Wiley and Sons, Chichester, 2003.

[2] E. Hairer, S.P. Nørsett and G. Wanner, Solving ordinary differential equations I: Nonstiff problems, Springer-Verlag, Berlin, 1993.

[3] G.Yu. Kulikov and S.K. Shindin, A technique for controlling the global error in multistep methods, (*in Russian*) Zh. Vychisl. Mat. Mat. Fiz. 40 (2000), pp. 1308–1329; *translation in* Comput. Math. Math. Phys. 40 (2000), pp. 1255–1275.

[4] G.Yu. Kulikov and S.K. Shindin, On multistep extrapolation methods for ordinary differential equations, (*in Russian*) Dokl. Akad. Nauk., 372 (2000), pp. 301–304; *translation in* Doklady Mathematics, 61 (2000), pp. 357–360.

[5] G.Yu. Kulikov and S.K. Shindin, On automatic global error control in multistep methods with polynomial interpolation of numerical solution, in: Computational Science and Its Applications — ICCSA 2004, International Conference, Assisi, Italy, May 2004, Proceedings, Part III, Antonio Lagana et al. (Eds.), Lecture Notes in Computer Science, vol. 3045, Springer-Verlag, Berlin, 2004, pp. 345–354.

[6] G.Yu. Kulikov and S.K. Shindin, 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), pp. 847–868; *translation in* Comput. Math. Math. Phys. 44 (2004), pp. 794–814.

[7] G.Yu. Kulikov and S.K. Shindin, On interpolation-type multistep methods with automatic global error control, (*in Russian*) Zh. Vychisl. Mat. Mat. Fiz. 44 (2004), pp. 1400–1421; *translation in* Comput. Math. Math. Phys. 44 (2004), pp. 1314–1333.

[8] G.Yu. Kulikov and S.K. Shindin, Global error estimation and extrapolated multistep methods for index 1 differential-algebraic systems, BIT 45 (2005), pp. 517–542.

[9] G.Yu. Kulikov and S.K. Shindin, One-leg variable-coefficient formulas for ordinary differential equations and local-global step size control, Numer. Algorithms, 43 (2006), pp. 99–121.

[10] G.Yu. Kulikov and S.K. Shindin, Local and global error estimation in Nordsieck methods, Russ. J. Numer. Anal. Math. Model. (*to appear*)

[11] G.Yu. Kulikov and S.K. Shindin, Automatic global error control in Nordsieck methods. (*in preparation*)

[12] P. Lancaster, Theory of Matrices, Academic Press, New York and London, 1969.

[13] H. Podhaisky, R. Weiner and B.A. Schmitt, Rosenbrock-type ”peer” methods, Appl. Numer. Math., 53 (2005), pp. 409–420.

[14] H. Podhaisky, R. Weiner and B.A. Schmitt, Linearly-implicit two-step methods and their implementation in Nordsieck form, Appl. Numer. Math., 56 (2006), pp. 374–387.

[15] B.A. Schmitt and R. Weiner, Parallel two-step W-methods with peer variables, SIAM J. Numer. Anal., 42 (2004), pp. 265–286.

[16] B.A. Schmitt, R. Weiner and K. Erdmann, Implicit parallel peer methods for initial value problems, Appl. Numer. Math., 53 (2005), pp. 457–470.

[17] R. Weiner, B.A. Schmitt and H. Podhaisky, Parallel two-step W-methods and their application to MOL-systems, Appl. Numer. Math., 48 (2004), pp. 425–439.

Reports of the Institutes 2008

01-08. R. I. Boţ, A. Löhne, *On totally Fenchel unstable functions in finite dimensional spaces.*

02-08. G. Warnecke, *Schulen und Schulverläufe bei Julius Plücker (1801 -1868) und seinem Studenten August Beer (1825 - 1863) in einer Gesellschaft im Prozess Grundlegender Änderungen, Teil I.*

03-08. G. Warnecke, *Schulen und Schulverläufe bei Julius Plücker (1801 -1868) und seinem Studenten August Beer (1825 - 1863) in einer Gesellschaft im Prozess Grundlegender Änderungen, Teil II.*

04-08. J. Bruder, *Construction and Consistency of Krylov-W-methods for DAE's.*

05-08. P. Brückmann, P. Winkert, *T-symmetrical Tensor Differential Forms with Logarithmic Poles along a Hypersurface Section.*

06-08. R. Denk, J. Prüss, R. Zacher, *Maximal L_p-Regularity of Parabolic Problems with Boundary Dynamics of Relaxation Type.*

07-08. P. Winkert, *Entire extremal solutions for elliptic inclusions of Clarke's gradient type*

08-08. M. Goebel, Ch. Schlensag, *Hans Brandes (1883-1965) Promotion in Halle - Lehrer in Braunschweig*

09-08. J. Prüss, G. Simonett, R. Zacher, *On convergence of solutions to equilibria for quasilinear parabolic problems*

10-08. R. Zacher, *Weak solutions of abstract evolutionary integro-differential equations in Hilbert spaces*

11-08. R. Zacher, *Boundedness of weak solutions to evolutionary partial integro-differential equations with discontinuous coefficients*

12-08. J. Prüss, G. Simonett, R. Zacher, *On normal stability for nonlinear parabolic equations*

13-08. Th. Krohn, *Über die Schrift "Prodromus Conjunctionis Magnae, anno 1623. futurae. Das ist: Kurtzes und Einfeltiges, doch in Gottes Wort und der Astrologischen Kunst gegründets Bedencken von dem grossen Cometstern, der in abgewichenem 1618. Jahre im Novembri sich erst recht sehen lassen [...]" von Erasmus Schmidt (1570-1637), Professor für Mathematik an der Wittenberger Universität*

14-08. R. Zacher, *The Harnack inequality for the Riemann-Liouville fractional derivation operator*

15-08. R. Zacher, *Convergence to equilibrium for second order differential equations with weak damping of memory type*

16-08. G. Yu. Kulikov, R. Weiner, *Global Error Control in Implicit Parallel Peer Methods*

Reports are available via WWW: http://www2.mathematik.uni-halle.de/institut/reports/