ROWMAP – a ROW-code with Krylov techniques for large stiff ODEs *

R. Weiner, B.A. Schmitt and H. Podhaisky

Abstract

We present a Krylov-W-code ROWMAP for the integration of stiff initial value problems. It is based on the ROW-methods of the code ROS4 of Hairer and Wanner and uses Krylov techniques for the solution of linear systems. A special multiple Arnoldi process ensures order $p = 4$ already for fairly low dimensions of the Krylov subspaces independently of the dimension of the differential equations. Numerical tests and comparisons with the multistep code VODPK illustrate the efficiency of ROWMAP for large stiff systems. Furthermore, the application to nonautonomous systems is discussed in more detail.

Key words. ROW-methods, stiff initial value problems, Krylov subspaces, multiple Arnoldi process
AMS(MOS) subject classifications. 65L06, 65F10

1 Introduction

For the numerical solution of stiff initial value problems

$$\begin{aligned} y'(t) &= f(t, y(t)) \\ y(t_0) &= y_0 \in \mathbb{R}^n, \end{aligned} \tag{1.1}$$

implicit or linearly implicit methods have to be used due to stability requirements. For large dimensions n these methods spend most of their computing time for the evaluation of the Jacobian and in the solution of linear equations. Recently, Krylov methods have been used extensively in implicit methods for stiff ordinary differential equations to accelerate the linear algebra in these integration schemes, e.g. [3], [4] [5], [16]. An alternative approach for large stiff equations are methods containing matrix exponentials as exact solutions of linearized equations. Krylov approximations for such matrix functions multiplied by a vector were discussed in [7], [10], [11]. The main advantages of Krylov techniques in ODE codes are

- the explicit computation and storage of the full Jacobian is not necessary ("matrix free" methods)

*This work was supported by Deutsche Forschungsgemeinschaft

- the methods exploit sparsity but do not require special structure of the systems (banded Jacobian, ...)

We will study the use of Krylov techniques in Rosenbrock-type methods like ROW- and W-methods. ROW-methods (e.g. [13]) have shown to be very efficient solution methods for stiff problems. The code ROS4 of Hairer and Wanner [9] performs well for low and medium accuracy requirements. A ROW-method for autonomous systems (1.1) is given by

$$(1.2)\qquad \begin{aligned} (I - h\gamma A)(k_i + \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j) &= f_i + \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j \\ f_i &= f(u_{m+1}^{(i)}) \\ u_{m+1}^{(i)} &= u_m + h \sum_{j=1}^{i-1} a_{ij} k_j \\ u_{m+1} &= u_m + h \sum_{i=1}^{s} b_i k_i \end{aligned}$$

where

$$(1.3)\qquad A = f_y(u_m).$$

If A is replaced by some approximation T, (1.2) becomes a so-called *W-method* ([18]).
We consider Krylov-W-methods where the matrix A is replaced by low-rank approximations

$$(1.4)\qquad T_i = Q_i Q_i^T A$$

at each stage [4], [16]. The orthogonal matrices $Q_i = \{q_1, ..., q_{\varkappa_i}\} \in \mathbb{R}^{n,\varkappa_i}$, where Q_i $(i > 1)$ is an extension of Q_{i-1}, are computed by a special multiple Arnoldi process (MAP) which will be presented in section 2. The stage equation (1.2) then becomes

$$(1.5)\qquad (I - h\gamma T_i)(k_i + \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j) = f_i + \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j =: w_i$$

Since

$$(1.6)\qquad (I - h\gamma Q_i Q_i^T A)^{-1} = I + h\gamma Q_i (I - h\gamma H_i)^{-1} Q_i^T A, \quad H_i = Q_i^T A Q_i \in \mathbb{R}^{\varkappa_i,\varkappa_i},$$

the stage increment k_i may be computed by

$$(1.7)\qquad \begin{aligned} k_i &= (I - h\gamma T_i)^{-1} w_i - \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j \\ &= [I + h\gamma Q_i (I - h\gamma H_i)^{-1} Q_i^T A] w_i - \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j, \end{aligned}$$

and we have to solve equations of dimension $\varkappa_i$ only. The special multiple Arnoldi process guarantees, that the right-hand side w_i at the i-th stage belongs to the subspace

$$(1.8)\qquad \mathcal{K}_i = \operatorname{span}\{q_1, ..., q_{\varkappa_i}\} = \operatorname{range} Q_i,$$

i.e.,

$$Q_i Q_i^T w_i = w_i. \tag{1.9}$$

In this case the relation (1.7) simplifies further leading to the *Krylov-W-method*

$$\begin{aligned} k_i &= Q_i(I - h\gamma H_i)^{-1} Q_i^T w_i - \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j \\ w_i &= f_i + \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j, \quad f_i = f(u_{m+1}^{(i)}) \\ u_{m+1}^{(i)} &= u_m + h \sum_{j=1}^{i-1} a_{ij} k_j, \quad u_{m+1} = u_m + h \sum_{i=1}^{s} b_i k_i. \end{aligned} \tag{1.10}$$

Extensive numerical tests have been performed previously with a 3-stage method of order 2 (e.g. [4], [16]). The main results of these tests on large problems obtained by MOL-semidiscretization of 2- and 3-dimensional parabolic equations were the following:

- Comparisons with the multistep code VODPK (see [6]) have shown the efficiency of the Krylov-W-method for low accuracy requirements.

- The dimensions of Krylov subspaces have to be restricted in a practical implementation due to memory requirements of the Arnoldi process. The stability of the method was satisfactory already for low Krylov dimensions and was mainly influenced by the first stage. An analysis of the linear case showed that the dimension of the Krylov subspace has to be increased at least by one for each new function evaluation at a higher stage, and this property has also been observed in numerical tests [4]. This rule has to be obeyed in the restrictions on $\varkappa_i$ at different stages.

- One of the main difficulties of the overall scheme was the proper control of the stage residuals (*ktol*) of the Krylov process in relation to the accuracy *tol* prescribed for the overall integration scheme. The error analysis in [5] suggested a choice $ktol \approx tol/h$. Due to the practical restriction of the Krylov subspace dimensions, however, this tolerance was missed sometimes. Surprisingly, the code still performed well in cases where the residuals of the stage systems were much larger than *ktol*. Here the overall accuracy of the code was maintained by the stepsize control alone. This observation suggested, that the basis of the stepsize control mechanism still was sound for certain restrictions on the dimensions $\varkappa_i$, which means a correct relation between the errors of the propagated and the embedded solutions.

On the basis of these results our aims were

1. the construction of higher order Krylov-W-methods, where the order of the underlying method (which could also be a ROW method) is preserved for Krylov dimensions $\varkappa_i \ll n$,

2. the implementation and test of a corresponding code.

In [21] we have shown, that the order of any ROW method of order 4 can be preserved in the Krylov-W-method with fixed problem-independent subspace dimensions $\varkappa_i$ if a certain multiple Arnoldi process is employed. On the other hand, it is not realistic to assume that such a restricted scheme can be stable. As a compromise we decided to choose the dimension $\varkappa_1$ of the first Krylov subspace by requiring the residual to satisfy the condition (2.4) in order to provide stability, but to restrict the following dimension increases by the rules derived from the consistency requirements. The result of our investigations is the code ROWMAP that is based on the ROW-methods of order 4(3) from the code ROS4 [9]. The special multiple Arnoldi process preserves these orders for subspace dimensions $\varkappa_i$ (in the range 4–11) independent of n. In section 2 we briefly describe this special order preserving multiple Arnoldi process.
In section 3 we present numerical results for ROWMAP applied to several large stiff problems obtained by MOL-discretization of parabolic equations. Comparisons with VODPK show the efficiency and reliability of the code.
In section 4 we discuss various possibilities to treat nonautonomous problems. In contrast to classical ROW- and W-methods different versions (using f_t or a slightly larger autonomous problem) are not equivalent for Krylov-W-methods.

2 A special multiple Arnoldi process

For the computation of the approximate solution u_{m+1} in (1.2) s systems of linear equations with the same coefficient matrix have to be solved. At the first stage, for the solution of

$$(I - h\gamma A)\tilde{k}_1 = w_1,$$

the classical Arnoldi method (cf. [14]) may be used to construct a sequence of orthonormal vectors $q_1, q_2, \ldots, q_{\varkappa_1}$, that constitute a basis for the Krylov subspace

$$\mathcal{K}_1 = \mathcal{K}(A, q_1, \varkappa_1) := \operatorname{span}\{q_1, ..., A^{\varkappa_1 - 1} q_1\}.$$

This process mixes Krylov steps Aq_j with Gram-Schmidt-orthogonalization

$$q_{j+1} = \frac{v}{\|v\|_2}, \quad v = (I - \sum_{l=1}^{j} q_l q_l^T) A q_j.$$

With these vectors q_j the matrices $Q_1 = (q_1, ..., q_{\varkappa_1})$ and $H_1 = Q_1^T A Q_1$ are formed. Note, that the matrix H_1 in (1.10) is in Hessenberg form. Of practical importance is the fact, that the residual of the Krylov approximation from (1.10) can be computed from subspace data alone by

$$(2.1)\quad r := (I - h\gamma A)k_1 - w_1 = -h\gamma(I - Q_1 Q_1^T) A q_{\varkappa_1} e_{\varkappa_1}^T l_1, \quad l_1 = (I - h\gamma H_1)^{-1} Q_1^T w_1,$$

whithout knowing $k_1 = Q_1 l_1$. At higher stages, $i > 1$, we will re-use the known information about A contained in Q_1 and H_1 and continue the Arnoldi process in an appropriate way. In an earlier paper [5] a first attempt was discussed which simply continues the initial Arnoldi process. Although this method performs surprisingly well for different test problems, it has the drawback that the right hand sides w_i, $i \geq 2$, are no longer contained in the Krylov subspace $\mathcal{K}_i$, in general. This leads to an additional term of the form $(I - QQ^T)w_i$ in the residual which is difficult to handle in theoretical investigations and can lead to high Krylov dimensions in practical computations for some problems. The following multiple Arnoldi process avoids this drawback. It is a special variant of the multiple Arnoldi process discussed in [16], [21].

We consider as basic integration methods the 4-stage ROW-methods of order 4 from ROS4 [9] (with the exception of GRK4A), characterized by the following properties:

- $b_3 = 0$
- $f_4 = f_3$
- there exists an embedded method of order $p = 3$.

In the context of these ROW-methods the Krylov-W-method ROWMAP employs the following special multiple Arnoldi process. In order to improve the readibility of this method we formalize the Gram-Schmidt orthogonalization part by the notation

$$v \overset{GS(i)}{\longleftarrow} w \qquad \Longleftrightarrow \qquad v := \Big(I - \sum_{m=1}^{i} q_m q_m^T\Big) w.$$

Special Multiple Arnoldi Process (MAP) for ROWMAP:

Stage 1: Classical Arnoldi process,

$$\begin{aligned} q_1 &:= v/\|v\|, \quad v := w_1 = f(u_m), \\ q_{j+1} &:= v/\|v\|, \quad v \overset{GS(j)}{\longleftarrow} Aq_j \quad j = 1, ..., \varkappa_1 - 1, \quad \varkappa_1 \geq 4. \end{aligned}$$

Stage 2: $\varkappa_2 = \varkappa_1 + 3$, include w_2, one block Arnoldi step with block size 2.

$$\begin{aligned} q_{\varkappa_1+1} &:= v/\|v\|, \quad v \overset{GS(\varkappa_1)}{\longleftarrow} w_2 \\ q_{\varkappa_1+2} &:= v/\|v\|, \quad v \overset{GS(\varkappa_1+1)}{\longleftarrow} Aq_{\varkappa_1} \\ q_{\varkappa_2} &:= v/\|v\|, \quad v \overset{GS(\varkappa_1+2)}{\longleftarrow} Aq_{\varkappa_1+1} \end{aligned}$$

Stage 3: $\varkappa_3 = \varkappa_2 + 1$, include w_3

$$q_{\varkappa_3} := v/\|v\|, \quad v \overset{GS(\varkappa_2)}{\longleftarrow} w_3$$

Stage 4: Finally, $\varkappa_4 = \varkappa_3 + 3$, one block Arnoldi step with block-size 3.

$$\begin{aligned} q_{\varkappa_3+1} &:= v/\|v\|, \quad v \overset{GS(\varkappa_3)}{\longleftarrow} Aq_{\varkappa_1+2} \\ q_{\varkappa_3+2} &:= v/\|v\|, \quad v \overset{GS(\varkappa_3+1)}{\longleftarrow} Aq_{\varkappa_2} \\ q_{\varkappa_4} &:= v/\|v\|, \quad v \overset{GS(\varkappa_3+2)}{\longleftarrow} Aq_{\varkappa_3} \end{aligned}$$

We collect the basis vectors in matrices $Q_i = (q_1, \dots, q_{\varkappa_i})$ and denote their column spaces by $\mathcal{K}_i = \text{span} Q_i$, $i = 1, \dots, 4$.

Remark 2.1 1. To summarize, the first dimension $\varkappa_1$ is at least 4 and determined by accuracy requirements for the residual r. The other dimensions grow by at most $\varkappa_2 = \varkappa_1 + 3, \quad \varkappa_3 = \varkappa_2 + 1, \quad \varkappa_4 = \varkappa_3 + 3$.
2. If one of the vectors v vanishes, however, the corresponding step is skipped reducing the Krylov dimensions.
3. Since the weight b_3 is zero the subspace is extended by w_3 only at the third stage (cf. Theorem 2.1).
4. In step 4 the right hand side w_4 is already contained in the Krylov subspace $\mathcal{K}_3$ since $f_4 = f_3$. $\square$

Lemma 2.1 *The strategy for choosing the sequence of basis vectors yields the following properties:*

1. $w_i \in \mathcal{K}_i, \quad i = 1, ..., 4$
2. $Aq_l \in \text{span}\{q_1, ..., q_{\varkappa_4}\}, \quad l = 1, ..., \varkappa_3$
3. *If at some stage i and some $k \leq i$ holds*

$$(T_i^l - A^l) w_k = 0 \quad \textit{for} \quad l = 1, ..., q,$$

then the same is true for stage $j > i \geq k$:

$$(T_j^l - A^l) w_k = 0 \quad \textit{for} \quad l = 1, ..., q.$$

Proof. The statement follows by construction and

$$Q_j = \{Q_i, q_{\varkappa_i + 1}, ..., q_{\varkappa_j}\} \quad \text{for} \quad j > i. \quad \blacksquare$$

Remark 2.2 The matrices $H_i = Q_i^T A Q_i$, $i > 1$, produced by the MAP will no longer be in strict Hessenberg form. At the second and third stage additional subdiagonals appear starting in columns $\varkappa_1$ and $\varkappa_2$. $\square$

The Multiple Arnoldi Process described above indeed ensures that the Krylov-W-method ROWMAP is of order $p = 4$.

Theorem 2.1 *If $\varkappa_1 \geq 4$ holds, the method ROWMAP based on a method of the ROS4 class with $b_3 = 0$ has consistency order $p = 4$, too.*

Proof. The Theorem is a consequence of a more general result in [21], but we will give a direct proof here. We compare the stage increments k_i of (1.10) with the stage increments $\widetilde{k}_i$

from the true ROW-method (1.2) with $A = f_y(u_m)$ using the same initial value $u_m = y(t_m)$. As usual we denote $\beta_{ij} = a_{ij} + \gamma_{ij}$, $\beta_i = \sum_{j=1}^{i-1} \beta_{ij}$. Taylor expansion for $h \to 0$ shows

$$k_i - \widetilde{k}_i = \sum_{l=1}^{\infty} h^l \gamma^l (T_i^l - A^l) w_i + hA \sum_{j=1}^{i-1} \beta_{ij} (k_j - \widetilde{k}_j) + \mathcal{O}(h^2) \sum_{j=1}^{i-1} a_{ij} (k_j - \widetilde{k}_j) + \mathcal{O}(h^2) \sum_{j=1}^{i-1} \gamma_{ij} (k_j - \widetilde{k}_j) \tag{2.2}$$

At the first stage the classical Arnoldi process provides for $k_1 - \widetilde{k}_1 = \mathcal{O}(h^4)$ if $\varkappa_1 \geq 4$. From the relation

$$w_2 = (1 + \frac{\gamma_{21}}{\gamma}) f + h\beta_2 A f + \mathcal{O}(h^2), \quad w_i = \delta_{i1} w_1 + \delta_{i2} w_2 + \mathcal{O}(h^2),\ i = 3, 4$$

and Lemma 2.1 it is readily seen that the first sum in (2.2) is $\mathcal{O}(h^4)$ for $i = 2, 4$ and $\mathcal{O}(h^3)$ for $i = 3$. Since $b_3 = 0$ the error between the ROWMAP and true ROW solution is

$$u_{m+1} - \widetilde{u}_{m+1} = \mathcal{O}(h^5).$$

Thus, ROWMAP also has order $p = 4$ and the embedded method is of order $p = 3$. ■

Remark 2.3 1. The order result has been obtained under the condition $\|hA\| \to 0$ (classical consistency order), which may be not realistic for very stiff systems. We note, however, that the stepsize control in ROS4 is performed also with respect to the classical orders 4(3) and that the special MAP preserves these classical orders of the basic ROW method. For other Krylov processes this is not clear for dimensions $\varkappa_i < n$.
2. In [21] we have proved B-consistency order 2 for Krylov-W-methods for semi-linear systems with constant stiff part. □

Theorem 2.1 shows that ROWMAP has order $p = 4$ with Krylov dimensions

$$\varkappa_1 \geq 4, \quad \varkappa_2 = \varkappa_1 + 3, \quad \varkappa_3 = \varkappa_2 + 1, \quad \varkappa_4 = \varkappa_3 + 3 \tag{2.3}$$

independent of n. The minimal subspace dimensions according to these rules are 4,7,8,11. Stability of the method, however, usually requires a larger dimension $\varkappa_1 > 4$. In ROWMAP for the step $t_m \to t_{m+1}$ the dimension $\varkappa_1$ is determined by the condition

$$h \cdot \sqrt{\frac{1}{n} \sum_{i=1}^{n} \Big(\frac{r_i}{atol_i + rtol_i |u_i(t_m)|} \Big)^2} \leq \vartheta. \tag{2.4}$$

Here, r is the stage residual (2.1), $atol_i$ and $rtol_i$ are the components of an absolute and a relative tolerance vector and $u(t_m)$ is the numerical solution at t_m. In the following tests we used $\vartheta = 0.1$.
If (2.4) cannot be satisfied during the Arnoldi iteration due to memory restrictions on $\varkappa_1$ the orders 4(3) of ROWMAP still are valid if the rules (2.3) are obeyed. So, the stepsize control still should be able to take care of the situation by reducing the stepsize.
For stability investigations of Krylov-W-methods see [4], [5], [17].

3 Numerical tests

We have implemented ROWMAP with the Multiple Arnoldi Process in FORTRAN 77. The code can be obtained from *http://www.mathematik.uni-halle.de/institute/numerik/software*. At the first stage the Arnoldi process is continued until (2.4) is fulfilled or some maximal dimension is reached. The residual is given by

$$r = -h\gamma(I - Q_1Q_1^T)Aq_{\varkappa_1}l_{\varkappa_1},$$

where $(I - Q_1Q_1^T)Aq_{\varkappa_1}$ would have been the next Arnoldi basis vector and $l_{\varkappa_1}$ is the last component of

$$l = (I - h\gamma H_1)^{-1}e_1\|w_1\|, \quad e_1 = (1, 0, ..., 0)^T.$$

All terms can be computed along with the Arnoldi process without computing k_1 itself since the QR-decomposition of $I - h\gamma H_1$ is updated during the process.
An attractive feature of many Krylov processes is the property that the matrix A is used through matrix-vector products Av only which may be approximated by finite differences

$$Av = f_y(u_m)v = \frac{1}{\delta}(f(u_m + \delta v) - f(u_m)) + \mathcal{O}(\delta\|v\|^2). \tag{3.1}$$

So, the exact computation of the Jacobian is not necessary ('matrix-free' methods). However, ROWMAP has an option for an exact Av-subroutine coded by hand or automatically generated by ADIFOR [1], for instance.
We have tested ROWMAP on numerous semidiscretized parabolic problems with the coefficients of the basic ROW-methods of ROS4. In these tests GRK4T [13] with $\gamma = 0.231$ and the method of Veldhuizen [20] with $\gamma = 0.225708$ showed the best performance on almost all problems. We suppose that the reason for this behaviour is a smaller error constant in the approximation of e^z. Both methods are only A(α)-stable with $\alpha > 89^o$. So, for problems with imaginary eigenvalues other basic methods may be better suited. The numerical results in the following experiments were obtained with the Krylov-W-method based on GRK4T.
We compare ROWMAP with the code VODPK (version of May 1994) [6] which is based on the BDF-code VODE [2] and uses GMRES(m) [15] for the solution of the linear systems occuring in the Newton iteration. We present results for the following 6 parabolic problems which are semidiscretized by the method-of-lines with central differences leading to first order systems of ordinary differential equations. The actual dimensions n are given in the diagrams below.
BRUSSELATOR: The 2-dimensional Brusselator (cf. [11])

$$\begin{aligned} u_t &= 1 + u^2v - 4u + \alpha(u_{xx} + u_{yy}) \\ v_t &= 3u - u^2v + \alpha(u_{xx} + u_{yy}), \quad \alpha = 0.02 \\ u(0,x,y) &= 0.5 + y, \quad v(0,x,y) = 1 + 5x \\ \frac{\partial u}{\partial n} &= \frac{\partial v}{\partial n} = 0 \text{ on } \partial\Omega, \quad \Omega = [0,1]^2, \quad t \in [0,1]. \end{aligned}$$

HEAT: The simple 2-dimensional heat equation [6]

$$
\begin{aligned}
u_t &= u_{xx} + u_{yy} \\
u(0,x,y) &= 16x(1-x)y(1-y) \\
u(t,x,y) &= \text{ on } \partial\Omega, \quad \Omega = [0,1]^2, \quad t \in [0, 10.24]
\end{aligned}
$$

NILIDI: The nonlinear diffusion equation ([4])

$$
u_t = e^u(u_{xx} + u_{yy}) + u(18e^u - 1)
$$

on $[0,\pi/3] \times [0,\pi/3]$, $\quad 0 \le t \le 1$.
Initial and boundary values are taken from the exact solution: $u = e^{-t}\sin(3x)\sin(3y)$.
FOOD: 2-dimensional predator-prey model [3]

$$
\begin{aligned}
u_t^i &= d_i \triangle u^i + f^i(u^1, u^2), \quad i = 1,2 \\
f^1(u^1,u^2) &= u^1(b_1 - a_1 u^2), \quad f^2(u^1,u^2) = u^2(b_2 - a_2 u^1) \\
d_1 &= 0.05,\ d_2 = 1,\ b_1 = 1,\ b_2 = -1000, \\
a_1 &= 0.1,\ a_2 = -100 \\
\frac{\partial u^i}{\partial n} &= 0 \text{ on } \partial\Omega,\ \Omega = [0,1]^2, \quad t \in [0,3], \\
u^1(0,x,y) &= 10 - 5\cos(\pi x)\cos(10\pi y) \\
u^2(0,x,y) &= 17 + 5\cos(10\pi x)\cos(\pi y).
\end{aligned}
$$

FOOD3: 3-dimensional predator-prey model [3]

$$
\begin{aligned}
u_t^i &= d_i \triangle u^i + f^i(u^1, u^2), \quad i = 1,2 \\
f^1(u^1,u^2) &= u^1(b_1 - a_{11}u^1 - a_{12}u^2), \\
f^2(u^1,u^2) &= u^2(b_2 - a_{21}u^1 - a_{22}u^2) \\
d_1 &= 0.05,\ d_2 = 1,\ b_1 = b_2 = (1 + 0.2xyz)(10^6 - 1 + 10^{-6}), \\
a_{11} &= 10^6,\ a_{12} = 1,\ a_{21} = 10^6 - 1,\ a_{22} = 10^6 \\
\frac{\partial u^i}{\partial n} &= 0 \text{ on } \partial\Omega,\ \Omega = [0,1]^3, \quad t \in [0,10], \\
u^1(0,x,y,z) &= 500 + 250\cos(\pi x)\cos(3\pi y)\cos(10\pi z) \\
u^2(0,x,y,z) &= 200 + 150\cos(10\pi x)\cos(\pi y)\cos(3\pi z)
\end{aligned}
$$

VDPL: The van der Pol equation with diffusion [12]

$$
\begin{aligned}
u_t^i &= d_i \triangle u^i + f^i(u^1, u^2), \quad i = 1,2 \\
f^1(u^1,u^2) &= u^2, \quad f^2(u^1,u^2) = 100(1 - (u^1)^2)u^2 - u^1 \\
d_1 &= 0.05,\ d_2 = 1 \\
\frac{\partial u^i}{\partial n} &= 0 \text{ on } \partial\Omega,\ \Omega = [0,1]^2, \quad t \in [0,1000], \\
u^1(0,x,y) &= 1 - \cos(\pi x)\cos(2\pi y) \\
u^2(0,x,y) &= 1 + 2\cos(2\pi x)\cos(\pi y).
\end{aligned}
$$

At $t = t_{m+1}$ the new stepsize in ROWMAP is computed by

$$h_{\text{new}} = h \cdot \min(2, \max(0.8c^{-0.25}, 0.25)),$$

where

$$c = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(\frac{u_i(t_{m+1}) - u3_i(t_{m+1})}{atol_i + rtol_i|u_i(t_m)|}\Big)^2}.$$

Here u is the numerical solution of order 4 and $u3$ the embedded solution of order 3. For $c > 1$ the step is repeated.
All problems were solved for six tolerances $tol = 10^{-2} \dots 10^{-7}$ with $rtol_i = atol_i = tol$. The marks in Figures 3.1–3.6 show the computing times (in seconds, logarithmic scale) plotted against the logarithm of the relative accuracy ERR achieved at the endpoint, where

$$\text{ERR} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\Big(\frac{u_i - uex_i}{1 + |uex_i|}\Big)^2}.$$

The different marks in each series of tolerances are connected by lines. Marks for cruder tolerances may be missing if the code failes there. The reference solution uex for the semidiscretized system was computed with VODPK and $tol = 10^{-12}$.

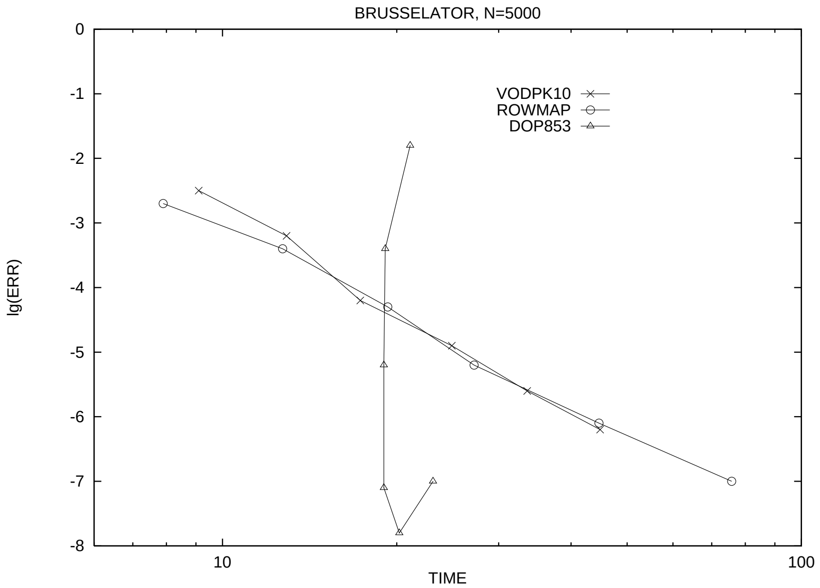

Figure 3.1: Work-precision diagram for the Brusselator

The computations were performed on a workstation IBM RS6000 with the BLAS-routines of the ESSL-library. ROWMAP and VODPK were run without preconditioning, matrix-vector products Av were approximated by finite differences.

Remark 3.1 VODPK has an option for preconditioning. However, the preconditioning method has to be provided by the user for each special problem. We have restricted our investigations to the fully matrix-free case here. Investigations with an automatic polynomial preconditioning for Krylov-W-methods will be the topic of future work. □

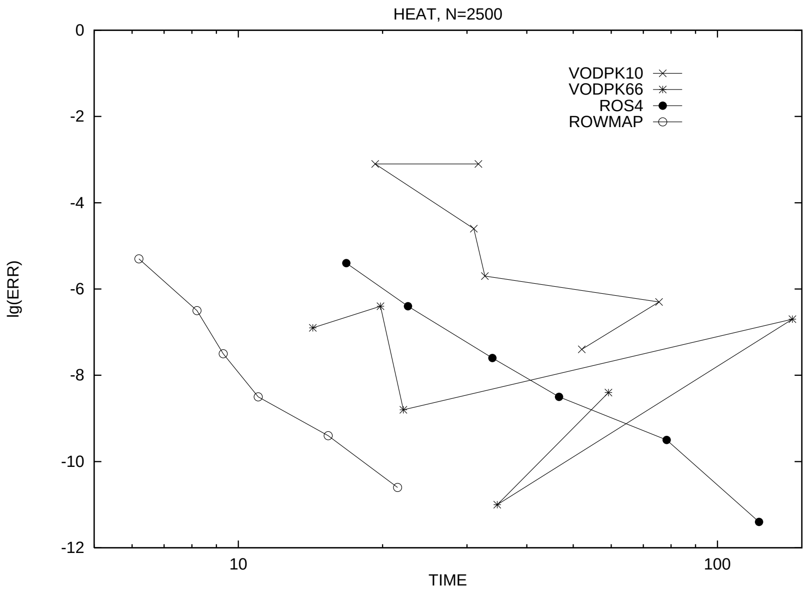

Figure 3.2: Work-precision diagram for the heat equation

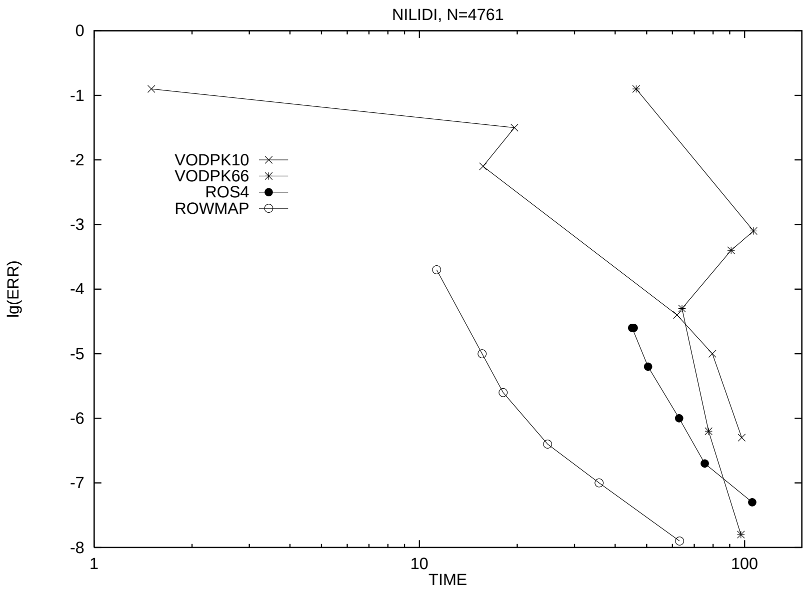

Figure 3.3: Work-precision diagram for NILIDI

In ROWMAP we restricted the maximal Krylov dimension to 66, in VODPK this dimension was set to $maxl = 10$ (VODPK10).

Remark 3.2 In VODPK the Krylov method GMRES(maxl) is used to solve the linear equations arising in the Newton method. When the accuracy is not sufficient after $maxl$ steps a restart of GMRES is performed. Therefore, a low maximal Krylov dimension is the default value in VODPK, $maxl = 5$. For our test problems $maxl = 10$ usually has given better results than $maxl = 5$. For comparison we also present results with the same

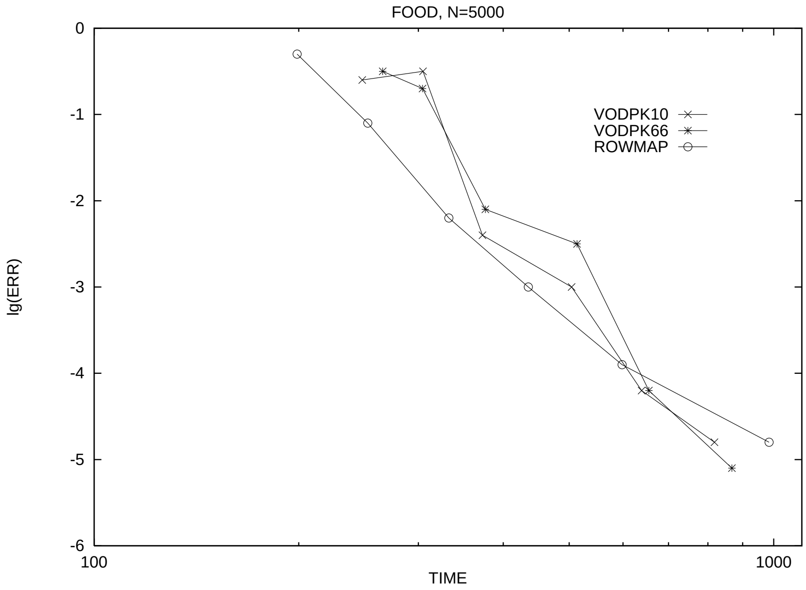

Figure 3.4: Work-precision diagram for FOOD

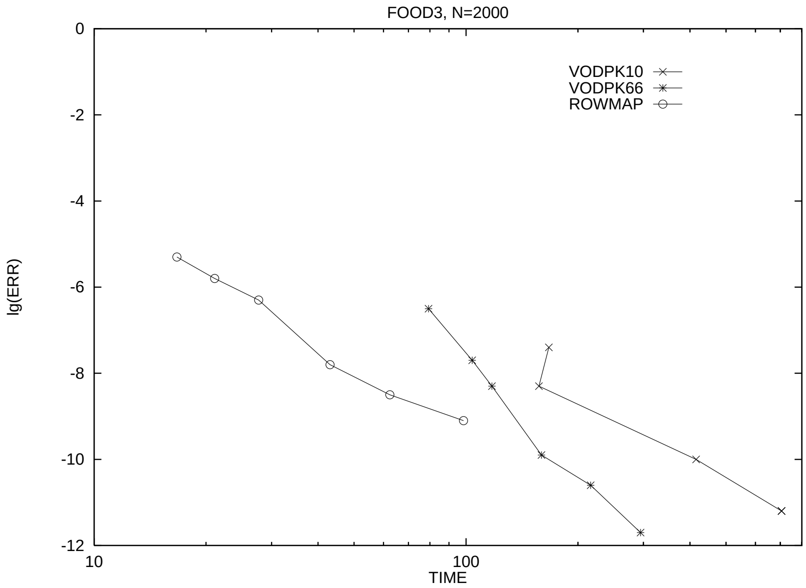

Figure 3.5: Work-precision diagram for FOOD3

restriction as in ROWMAP, i.e. $maxl = 66$ (VODPK66). $\square$

Furthermore, for comparison we applied the explicit Runge-Kutta code DOP853 [8] and the code ROS4 [9] (method GRK4T) with band solver for the linear equations. With exception of the Brusselator the computing times of DOP853 were much larger than for the Krylov methods. For low accuracy requirements ROS4 was often comparable, except for FOOD and FOOD3, where it was very expensive.
The Brusselator is only mildly stiff. For this problem and more stringent tolerances the

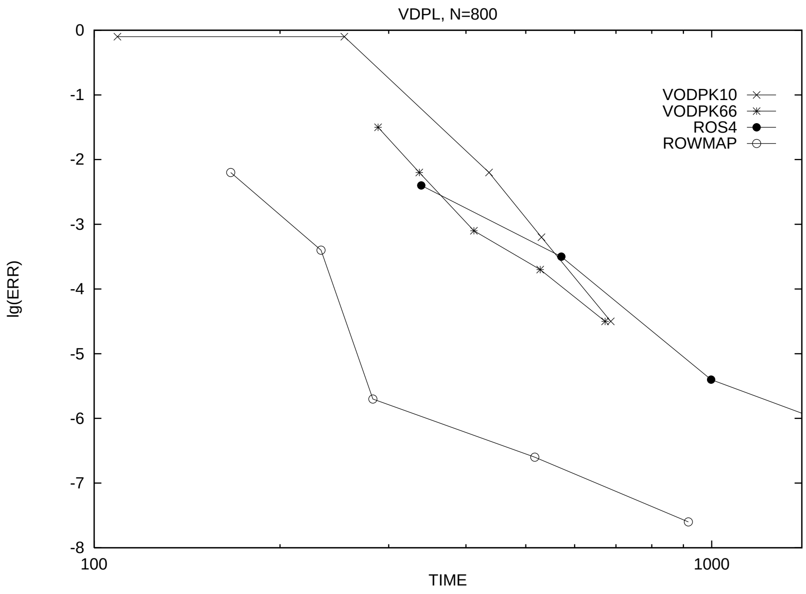

Figure 3.6: Work-precision diagram for the Van der Pol equation

explicit Runge-Kutta code is more efficient than the other methods, its computational effort is nearly independent of the tolerance. VODPK10 and VODPK66 give nearly identical results here.
On the heat and nonlinear diffusion equations VODPK shows a nonregular behaviour for low accuracy requirements, wheras ROWMAP performs well for all tolerances.
On FOOD for low accuracy ROWMAP is superior, for more stringent tolerances VODPK is more efficient.
While ROWMAP solves FOOD3 (with a $10 \times 10 \times 10$ grid) for all tolerances, VODPK10 is successful only for $tol \leq 10^{-8}$ and VODPK66 for $tol \leq 10^{-6}$. So, in Figure3.5 we have included further results down to $tol = 10^{-11}$ for the multistep code. For finer grids ($20 \times 20 \times 20$) ROWMAP again has no difficulties, whereas VODPK requires still more stringent tolerances to solve the problem with success.
On the Van der Pol equation all codes give reliable results only for $tol \leq 10^{-3}$, VODPK10 for $tol \leq 10^{-5}$. Here ROWMAP is clearly superior.
These results and other tests showed that ROWMAP is a reliable and efficient code. For moderate accuracy requirements it is usually faster than VODPK. Its behaviour depends more smoothly on the tolerance than that of VODPK, which sometimes has difficulties for mild tolerances. For more stringent tolerances the higher order of VODPK begins to pay off. We note that in the present examples the function evaluations are quite cheap. For costlier right hand sides the performance of VODPK may improve in relation to ROWMAP. We also want to remark that ROWMAP uses the BLAS subroutines for all matrix-vector operations and therefore may profit more from the optimized ESSL-library than VODPK.

4 Treatment of nonautonomous problems

So far we have only considered autonomous problems. ROW-methods (1.2) can be adapted to solve nonautonomous problems
$$y' = f(t, y)$$
either by incorporating the partial derivative $f_t(t_m, u_m)$ into the scheme ("nonautonomous ROW-methods", cf. [9], [19]) or by transforming the system into autonomous form by adding one differential equation for the time

$$y'_{n+1} = 1, \quad y_{n+1}(t_0) = t_0.$$

Both possibilities also apply to Krylov-W-methods. The nonautonomous Krylov-W-method reads

$$\begin{aligned} k_i &= Q_i(I - h\gamma H_i)^{-1} Q_i^T w_i - \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j \\ w_i &= f_i + \sum_{j=1}^{i-1} \frac{\gamma_{ij}}{\gamma} k_j + h d_i f_t, \quad f_i = f(t_m + c_i h, u_{m+1}^{(i)}) \\ u_{m+1}^{(i)} &= u_m + h \sum_{j=1}^{i-1} a_{ij} k_j, \quad u_{m+1} = u_m + h \sum_{i=1}^{s} b_i k_i \end{aligned} \tag{4.1}$$

with $d_i = \gamma + \sum_{j=1}^{i-1} \gamma_{ij}$, $c_i = \sum_{j=1}^{i-1} a_{ij}$.
It was shown in [21] that the MAP can be adapted to this case and analogous results on consistency of the Krylov-W-method are proved. The nonautonomous version of ROWMAP will be of order $p = 4$ for

$$\varkappa_1 \geq 4, \quad \varkappa_2 = \varkappa_1 + 5, \quad \varkappa_3 = \varkappa_2 + 1, \quad \varkappa_4 = \varkappa_3 + 5.$$

For the underlying ROW-methods both measures are equivalent. Unfortunately, this is no longer true for Krylov-W-methods. Here, the additional differential equation for the time is not integrated exactly. This leads to significiant differences in the numerical performance. It turns out that the simple transformation into an autonomous problem gives inferior results, in general. However, improvements by additional actions are possible. For instance, at the end of an integration step the $(n+1)$-st component of the numerical solution can be set to the actual time. Furthermore the stage increments k_i can be adapted.
We tested different measures and present numerical results for an earlier version ROWMAP for two nonautonomous problems with the following strategies:

$sft = 2:$ nonautonomous form (4.1)
$sft = 3:$ $y'_{n+1} = 1 \quad \Rightarrow \quad$ autonomous system
$sft = 4:$ like $sft = 3$, additionally $u_{m+1,n+1} = t_{m+1}$
$sft = 5:$ like $sft = 4$, additionally $k_{i,n+1} = 1$.

The test problems are

DIFKON: The diffusion-convection equation [16]

$$u_t = \varepsilon \triangle u - y u_x + x u_y + g(t,x,y)$$
$$\varepsilon = 0.1, \quad \Omega = [0,1]^2, \quad t \in [0,1].$$

The inhomogeneity g is chosen appropriately to yield

$$u(t,x,y) = \sin(\pi x)\sin(\pi y)(1 + 4xye^{-2\varepsilon\pi^2 t})$$

as the solution. Initial and boundary conditions are taken from the exact solution.
DIFFU2: The 2-dimensional diffusion equation

$$u_t = \triangle u + f(t,x,y)$$
$$\Omega = [0,1]^2, \quad t \in [0,1].$$

Here f is determined by the exact solution

$$u(t,x,y) = \sin(\pi x)\sin(\pi y)(1 + 4xy\sin t).$$

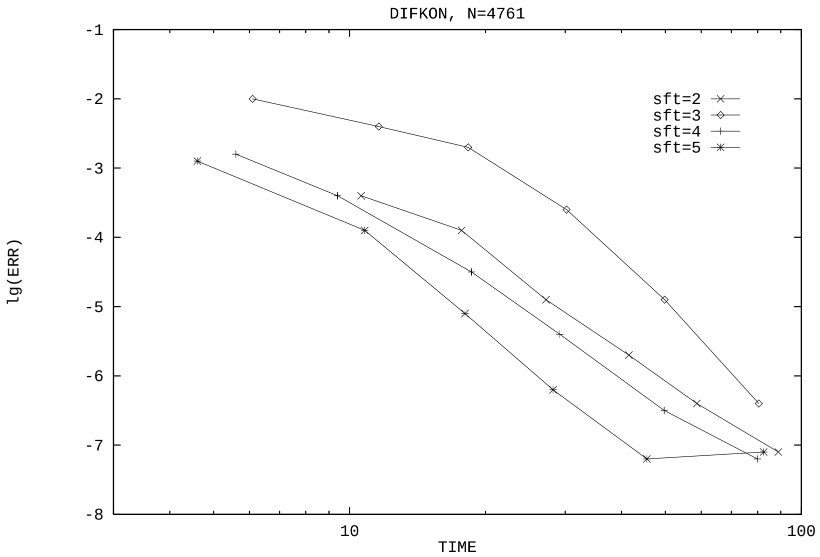

Figure 4.7: Work-precision diagram for DIFKON

The results in the Figures 4.7 and 4.8 clearly show, that the simple transformation into autonomous form is inferior to using the nonautonomous form (4.1). Additional adaption of the component y_{n+1} significiantly improves the performance.
On the basis of these results the actual version of ROWMAP uses the strategy $sft = 2$ for nonautonomous systems.

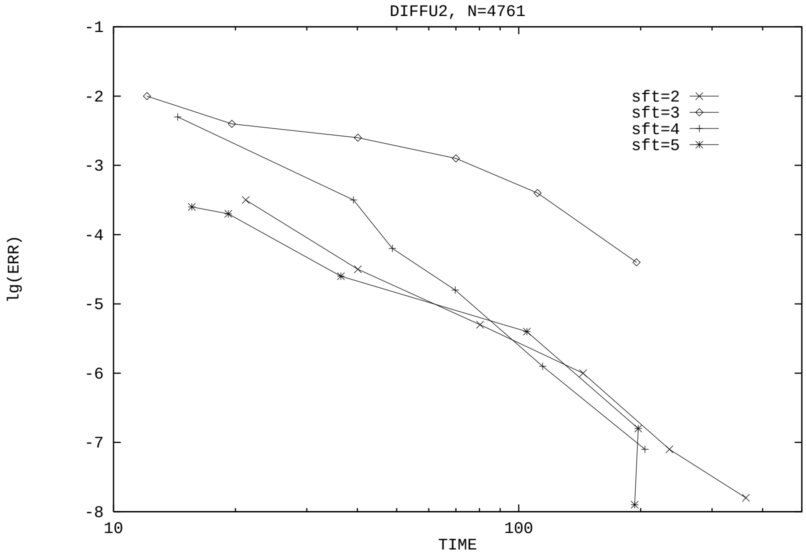

Figure 4.8: Work-precision diagram for the nonautonomous diffusion equation

5 Conclusion

Krylov-W-methods combining ROW- or W-methods with Krylov-techniques for the solution of the linear stage equations are efficient integration methods for large stiff systems since they do not require a special structure of the Jacobian. By a special multiple Arnoldi process the order of the basic ROW method is preserved and the stepsize control of this basic method can be used. The code ROWMAP which is of order 4 already for fairly low Krylov dimensions performs well for large systems obtained by MOL-discretization of parabolic equations and is competitive with the multistep code VODPK.

References

[1] Ch. Bischof, A. Carle, P. Khademi, A. Mauer, and P. Hovland, *ADIFOR 2.0 User's Guide*, Tech. Report CRPC-95516-S, Center for Research on Parallel Computation, 1995.

[2] P.N. Brown, G.D. Byrne, and A.C. Hindmarsh, *VODE: a variable coefficient ODE solver*, SIAM J. Sci. Stat. Comput. **10** (1989), 1038–1051.

[3] P.N. Brown and A.C. Hindmarsh, *Matrix-free methods for stiff systems of ODEs*, SIAM J. Numer. Anal. **23** (1986), 610–638.

[4] M. Büttner, B.A. Schmitt, and R. Weiner, *Automatic partitioning in linearly-implicit Runge-Kutta methods*, Applied Numerical Mathematics **13** (1993), 41–55.

[5] ———, *W-methods with automatic partitioning by Krylov techniques for large stiff systems*, SIAM J. Numer. Anal. **32** (1995), 260–284.

[6] G.D. Byrne, *Pragmatic experiments with Krylov methods in the stiff ODE setting*, Computational ordinary differential equations, Clarendon Press, Oxford, 1992, pp. 323–356.

[7] E. Gallopoulos and Y. Saad, *Efficient solution of parabolic equations by Krylov approximation methods*, SIAM J. Sci. Stat. Comp. **13** (1992), 1236–1264.

[8] E. Hairer, S.P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I*, 2. ed., Springer-Verlag, 1993.

[9] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II*, Springer-Verlag, 1991.

[10] M. Hochbruck and Ch. Lubich, *On Krylov subspace approximations to the matrix exponential operator*, Preprint no. 5, Mathematisches Institut Tübingen, 1995.

[11] M. Hochbruck, Ch. Lubich, and H. Selhofer, *Exponential integrators for large systems of differential equations*, SIAM J. Sci. Comp. (1995), to appear.

[12] K.R. Jackson and W.L. Seward, *Adaptive linear equation solvers in codes for large stiff systems of ODEs*, SIAM J. Sci. Comput. **14** (1993), no. 4, 800–823.

[13] P. Kaps and P. Rentrop, *Generalized Runge-Kutta methods of order four with stepsize control for stiff ordinary differential equations*, Numer. Math. **38** (1979), 55–68.

[14] Y. Saad, *Krylov subspace methods for solving large unsymmetric linear systems*, Math. Comp. **37** (1981), 105–126.

[15] Y. Saad and M.H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. **7** (1986), 856–869.

[16] B.A. Schmitt and R. Weiner, *Matrix-free W-methods using a multiple Arnoldi iteration*, APNUM **18** (1995), 307–320.

[17] ______, *Equilibrium attractivity of Krylov-W-methods for nonlinear stiff ODEs*, Tech. Report 44, FB Mathematik, Universität Marburg, 1996.

[18] T. Steihaug and A. Wolfbrandt, *An attempt to avoid exact Jacobian and nonlinear equations in the numerical solution of stiff ordinary differential equations*, Math. Comp. **33** (1979), 521–534.

[19] K. Strehmel and R. Weiner, *Linear-implizite Runge-Kutta Methoden und ihre Anwendung*, Teubner-Verlag Stuttgart-Leipzig, 1992.

[20] M. van Veldhuizen, *D-stability and Kaps-Rentrop methods*, Computing **32** (1984), 229–237.

[21] R. Weiner and B.A. Schmitt, *Consistency of Krylov-W-methods in initial value problems*, Tech. Report 14, FB Mathematik und Informatik, Universität Halle, 1996.

Rüdiger Weiner, Helmut Podhaisky
FB Mathematik und Informatik
Martin-Luther-Universität
Halle-Wittenberg
Postfach
D-06099 Halle/Saale
Germany
e-mail: rw@mail.mathematik.uni-halle.de

Bernhard A. Schmitt
Fachbereich Mathematik
Universität Marburg
Lahnberge
D-35032 Marburg
Germany
e-mail: schmitt@mathematik.uni-marburg.de