
MARTIN–LUTHER–UNIVERSITÄT
HALLE–WITTENBERG
INSTITUT FÜR MATHEMATIK



Peer methods for the one-dimensional shallow
water equations with CWENO space
discretization

G. Steinebach, R. Weiner

Report No. 09 (2009)

Editors:

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

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

**Peer methods for the one-dimensional shallow
water equations with CWENO space
discretization**

G. Steinebach, R. Weiner

Report No. 09 (2009)

Gerd Steinebach
Hochschule Bonn-Rhein-Sieg
Grantham-Allee 20
D-53757 Sankt Augustin
Email: gerd.steinebach@h-bonn-rhein-sieg.de

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

Peer methods for the one-dimensional shallow water equations with CWENO space discretization

Gerd Steinebach¹& Rüdiger Weiner²

Abstract

For many practical problems an efficient solution of the one-dimensional shallow water equations (Saint-Venant equations) is important, especially when large networks of rivers, channels or pipes are considered. In order to test and develop numerical methods four test problems are formulated. These tests include the well known dam break and hydraulic jump problems and two steady state problems with varying channel bottom, channel width and friction.

The space discretization of the partial differential equations is based on a finite volume approach with central WENO interpolation and local Lax-Friedrich fluxes [7]. For time-integration new linearly-implicit two-step peer methods of orders three and four are developed. These methods are especially adapted to the usage within the method of lines framework. They show a good performance compared to the well established methods like `ode15s`, `radau5` or `rodasp`.

Keywords: Shallow water equations, method of lines, Peer methods, WENO-schemes
AMS classification: 65M20, 65L06, 76B15

1 Introduction

The shallow water equations arise in many problems concerning water flow in channels, rivers or pipes. In industrial applications usually large networks are considered and often an one-dimensional modelling is sufficient for the problem solution [14].

The numerical solution of flood flow problems in rivers has a long tradition [17]. Since friction has a great influence on the behaviour of the water flow, mainly implicit solution methods like the Preisman-scheme have been applied to those problems [2]. Modern explicit time-integration schemes for hyperbolic problems like SSP Runge-Kutta methods [4] are not preferable in these cases.

The aim of this paper is the consideration of suitable time-stepping methods for the one-dimensional shallow water equations in a method of lines (MOL) framework. In order to investigate these methods four test problems are formulated in Section 2. In Section 3 a suitable space-discretization is chosen. This discretization is based on a finite volume

¹Hochschule Bonn-Rhein-Sieg, Grantham-Allee 20, 53757 Sankt Augustin, Germany

²Martin-Luther-Universität Halle-Wittenberg, Institut für Mathematik, Theodor-Lieser-Str. 5, 06120 Halle, Germany

approach with the central WENO-interpolation. The numerical fluxes with local Lax-Fridrich corrections are defined according to [7].

In Section 4 a new subclass of linearly implicit two-step peer methods is constructed. The strategy for the computation of the methods coefficients is adapted to its usage within method of lines applications with moderately accuracy requirements. Finally in Section 5 the new and already well known methods are applied to the test problems. Their efficiency with respect to computing time and accuracy is compared. The paper ends with some conclusion.

2 Test problems for the one-dimensional shallow-water equations

The new methods shall be applied in applications arising in fluid flow problems in large networks [14]. These networks can consist of pipes, channels or natural rivers. The flow can be water flow with free surface or in pressure and gas flow. In this paper we concentrate on free surface water flow problems in one space dimension.

This flow type is described by the Saint-Venant equations or shallow-water equations (SWE) [17]:

$$\partial_t A + \partial_x Q = 0 \quad (2.1)$$

$$\partial_t Q + \partial_x \left(\frac{Q^2}{A} \right) + gA \partial_x z = -gAS_f \quad (2.2)$$

$Q(x, t)$ denotes the volume flow along the channel, $A(x, t)$ the flooded cross-sectional area, g the gravitational acceleration. $z(x, t)$ is the water surface elevation above a reference level, it must be a given function of x and A : $z(x, t) = \tilde{f}(x, A(x, t))$. Then it follows $\partial_x z = \partial_x \tilde{f}(x, A) + \partial_A \tilde{f}(x, A) \cdot \partial_x A$.

S_f is an empirical formula for the friction due to channel roughness, i.e. Manning-Strickler:

$$S_f = \frac{|u|u}{K_{St}^2 h^{4/3}}$$

The flow velocity is given by $u = \frac{Q}{A}$, $h = z - S_0$ is the water depth, $S_0(x)$ the bottom elevation and K_{St} is a roughness coefficient.

The SWE-equations are hyperbolic but not in conservation law form $\partial_t q + \partial_x f(q) = S(q)$ due to the term $gA \partial_x z$.

Simplifications can be obtained in case of rectangular cross-sections $A(x, t) = B \cdot h(x, t)$ with constant width B . If additionally friction is neglected and the bottom elevation $S_0(x) = 0$ is assumed we obtain the equations

$$\partial_t h + \partial_x (uh) = 0 \quad (2.3)$$

$$\partial_t (uh) + \partial_x (u^2 h) + g \partial_x \left(\frac{h^2}{2} \right) = 0 \quad (2.4)$$

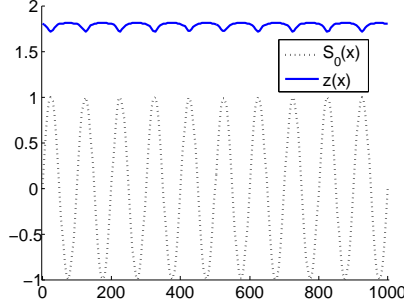


Figure 2.1: Problem 1: Water surface elevation $z(x)$ and bottom elevation $S_0(x)$

Four numerical test problems are considered. Tests 1 and 2 are based on equations (2.1,2.2). For steady flow conditions they read:

$$\begin{aligned} \partial_x Q &= 0 \\ \partial_x \left(\frac{Q^2}{A} \right) + gA \partial_x z &= -gAS_f \end{aligned}$$

Thus $Q(x, t) = Q_0$ is a given constant and the second equation is an ODE for the unknown $A = A(x)$:

$$Q_0^2 \left(\frac{1}{A} \right)' + gA (\partial_x \tilde{f}(x, A) + \partial_A \tilde{f}(x, A) \cdot A') = -gAS_f \quad (2.5)$$

If the geometry of the channel is given by a function $A = \tilde{g}(x, h)$ instead of $z = \tilde{f}(x, A)$ and if the water depth $h(x)$ is used as the unknown one gets the ODE

$$-\frac{Q_0^2}{\tilde{g}(x, h)^2} (\partial_x \tilde{g}(x, h) + \partial_h \tilde{g}(x, h) h') + g \tilde{g}(x, h) (S_0' + h') = -g \tilde{g}(x, h) S_f \quad (2.6)$$

Solving (2.5) or (2.6) for the initial condition $A(b) = A_b$ or $h(b) = h_b$ in the space-interval $x \in [a, b]$ gives the steady state solution of (2.1, 2.2).

In order to check the numerical solution of equations (2.1,2.2) we start with an initial condition which does not solve equation (2.5). Together with the correct boundary condition $Q(a, t) = Q_0$, $A(b, t) = A_b$ the steady state solution should be reached after some time.

In the first test problem friction is neglected ($S_f = 0$) and the bottom elevation is given by $S_0(x) = \sin(\frac{2\pi}{100}x)$ in the space interval $[0, 1000]$. A rectangular channel of constant width $B = 1$ is assumed and the initial conditions are chosen according to $Q(x, 0) = 1$ and $z(x, t) = 1.8$. The boundary conditions are consistent with the initial conditions: $Q(0, t) = 1$, $z(1000, t) = 1.8$. The simulation is computed in the time interval $t \in [0, 28800]$. Figure 2.1 shows the steady state solution $z(x)$ and the bottom elevation $S_0(x)$.

In the second test problem we assume a trapezoidal cross-section of varying width $B(x) = 1 + 0.2 \sin(\frac{3}{1000} 2\pi x)$ and angle $\alpha(x) = \frac{\pi}{24} (1 + 0.2 \sin(\frac{6}{1000} 2\pi x))$, see Figure 2.2. In this case it follows that $A = \tilde{g}(x, h) = h(B + h \tan(\alpha))$. Moreover friction with a coefficient $K_{St} = 30$

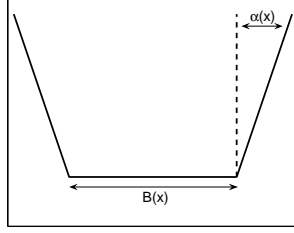


Figure 2.2: Problem 2: trapezoidal cross-section

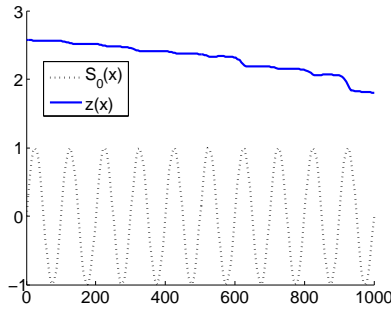


Figure 2.3: Problem 2: Water surface elevation $z(x)$ and bottom elevation $S_0(x)$

is assumed. The initial conditions, boundary conditions and bottom elevation are identical to problem 1, the time interval is unchanged. Figure 2.3 shows the steady state solution $z(x)$ and the bottom elevation $S_0(x)$.

The third problem is the well known dam break problem based on equations (2.3, 2.4). The initial conditions for the water depth are discontinuous:

$$Q(x, 0) = (hu)(x, 0) = 0; \quad h(x, 0) = \begin{cases} h_1 & \text{for } x < x_M \\ h_4 & \text{for } x \geq x_M \end{cases}.$$

For $t > 0$ a shock is travelling to the right and a rarefaction wave to the left, see Figure 2.4. The analytical solution is given by [17, 15]:

$$h(x, t) = \begin{cases} \frac{1}{9g} \left(2c_1 - \frac{x-x_M}{t} \right)^2 & \text{if } x < x_M - c_1 t \\ h_3 & \text{if } x_M - c_1 t \leq x < x_M + (u_3 - c_3)t \\ h_4 & \text{if } x_M + (u_3 - c_3)t \leq x < x_M + s t \\ 0 & \text{if } x \geq x_M + s t \end{cases}, \quad u(x, t) = \begin{cases} 0 & \text{if } x < x_M - c_1 t \\ \frac{2}{3} \left(c_1 + \frac{x-x_M}{t} \right) & \text{if } x_M - c_1 t \leq x < x_M + (u_3 - c_3)t \\ 2(c_1 - c_3) & \text{if } x_M + (u_3 - c_3)t \leq x < x_M + s t \\ 0 & \text{if } x \geq x_M + s t \end{cases}$$

with

$$c_4 = \sqrt{gh_4}; \quad c_1 = \sqrt{gh_1}; \quad c_3 = c_4 \sqrt{\frac{1}{2}(\sqrt{1+8\eta^2} - 1)}; \quad h_3 = \frac{c_3^2}{g}; \quad u_3 = c_4 \left(\eta - \frac{1}{4\eta} (1 + \sqrt{1+8\eta^2}) \right)$$

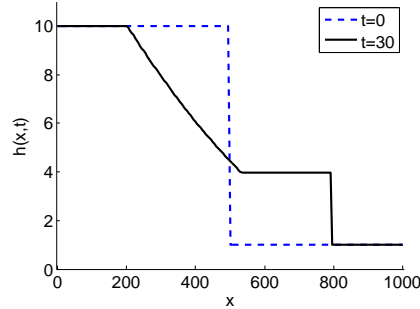


Figure 2.4: Dam break problem: shock and rarefaction wave

and η is the solution of the nonlinear equation

$$0 = \eta - \frac{1}{4\eta}(1 + \sqrt{1 + 8\eta^2}) + 2\sqrt{\frac{1}{2}(\sqrt{1 + 8\eta^2} - 1)} - 2\sqrt{\frac{h_1}{h_4}}$$

The shock velocity of the jump is given by $s = \eta c_4$. In this example we have defined $h_1 = 10$, $h_4 = 1$, $x_M = 500$, $t \in [0, 30]$. The boundary conditions are chosen according to the analytical solution.

The last problem is the hydraulic jump with initial conditions for equations (2.3, 2.4):

$$h(x, 0) = \begin{cases} h_0 & \text{if } x \leq 200 \\ h_1 & \text{if } x > 200 \end{cases}, \quad u(x, 0) = \begin{cases} u_0 & \text{if } x \leq 200 \\ u_1 & \text{if } x > 200 \end{cases}$$

A hydraulic jump with shock velocity s originates, if the jump condition

$$\begin{pmatrix} h_1 - h_0 \\ u_1 h_1 - u_0 h_0 \end{pmatrix} = s \begin{pmatrix} u_1 h_1 - u_0 h_0 \\ h_1 u_1^2 + \frac{1}{2} g h_1^2 - h_0 u_0^2 - \frac{1}{2} g h_0^2 \end{pmatrix}$$

is fulfilled. For given $h_0 = \frac{1}{4}$, $u_0 = 4$ and $s = 1$ a possible solution is $h_1 = 0.56372388976300$, $u_1 = 2.33043855976250$. Figure 2.5 shows the initial conditions and the solution at time $t = 500$.

Obviously three of the problems are frictionless and explicit integrators would be more efficient. On the other hand practical problems usually do contain friction which does introduce some kind of stiffness to the equations. Table 2.1 shows the CPU-time and the number of steps for the second problem with friction and 100 space discretization cells. The applied integrators are the explicit Runge-Kutta MATLAB solver `ode45`, the explicit Runge-Kutta Chebyshev method with extended stability region `rock4` [1, 6] and the semi-implicit ROW-method `rodasp` [5, 15]. The results show clearly the advantage of implicit methods which should be applied as general purpose methods in fluid flow problems for networks. Usually the coupling conditions in such networks are modelled as algebraic equations which leads to a system of differential algebraic equations (DAEs) [16]. Therefore the integration methods should also be applicable to DAEs of index 1.

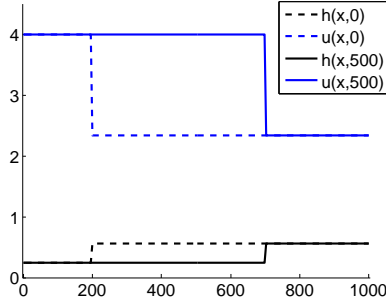


Figure 2.5: Hydraulic jump problem

	NSTEP	CPU
rodasp	91	6
rock4	5326	323
ode45	9383	286

Table 2.1: CPU-times and number of steps for problem 2

3 Space discretization

Equations (2.1, 2.2) can be expressed as a hyperbolic system of type

$$\partial_t q + \partial_x f(q) + M(q) \partial_x g(q) = S(q) \quad (3.1)$$

with $q = (A, Q)^T$, $f(q) = (Q, \frac{Q^2}{A})^T$, $M(q) = \begin{pmatrix} 0 & 0 \\ gA & 0 \end{pmatrix}$, $g(q) = (z, 0)^T$, $S(q) = (0, -gAS_f)^T$.

The proposed finite volume space discretization is based on the method of Kurganov and Levy [7]. In the first step the state variables q are interpolated onto the cell boundaries $x_{i+1/2}$ of the finite volume mesh. In the second step the resulting numerical fluxes are corrected by some kind of local Lax-Friedrich ansatz.

Let q_i denote the average over cell i with mesh width Δx : $q_i = q_i(t) = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} q(x, t) dx$. Then two interpolations for $q_{i+1/2} = q(x_{i+1/2}, t)$ are computed:

$$q_{i+1/2}^+ = f_{weno}^+(q_i, q_{i+1}, q_{i+2}), \quad q_{i+1/2}^- = f_{weno}^-(q_{i-1}, q_i, q_{i+1})$$

The functions f_{weno}^+ , f_{weno}^- are interpolation polynomials $p^+(x)$, $p^-(x)$ evaluated at $x = x_{i+1/2}$ that preserve the given cell averages. Each polynomial is a linear combination of two linear functions f_L , f_R and one quadratic function f_C :

$$p^+(x) = \frac{1}{4} f_L(x) + \frac{1}{2} f_C(x) + \frac{1}{4} f_R(x)$$

with

$$f_L(x) = q_i + \frac{q_{i+1} - q_i}{\Delta x}(x - x_i)$$

$$f_R(x) = q_{i+1} + \frac{q_{i+2} - q_{i+1}}{\Delta x}(x - x_{i+1})$$

$$f_C(x) = q_{i+1} - \frac{1}{12}(q_{i+2} - 2q_{i+1} + q_i) + \frac{q_{i+2} - q_i}{2\Delta x}(x - x_{i+1}) + \frac{q_{i+2} - 2q_{i+1} + q_i}{\Delta x^2}(x - x_{i+1})^2$$

$f_C(x)$ is chosen in such a way that the resulting quadratic function $p^+(x)$ preserves the cell averages q_i, q_{i+1}, q_{i+2} , which gives a third order accuracy. $p^-(x)$ is constructed similarly and preserves the cell averages q_{i-1}, q_i, q_{i+1} .

If nonsmooth solutions occur, the weights $w_L = \frac{1}{4}, w_C = \frac{1}{2}, w_R = \frac{1}{4}$ are adapted in such a way, that the resulting function $p^+(x)$ generates the least oscillatory reconstruction [7]. When all cell boundary values $q_{i+1/2}^\pm$ are defined the numerical fluxes and flux corrections must be computed:

$$\begin{aligned} f_{i+1/2} &= \frac{1}{2}(f(q_{i+1/2}^+) + f(q_{i+1/2}^-)) \\ g_{i+1/2} &= \frac{1}{2}(g(q_{i+1/2}^+) + g(q_{i+1/2}^-)) \\ LF_{i+1/2} &= \frac{1}{2}\lambda_{i+1/2}(q_{i+1/2}^+ - q_{i+1/2}^-) \end{aligned}$$

$\lambda_{i+1/2}$ is an estimation of the largest absolute value of the eigenvalues of $f'(q) + M(q)g'(q)$ in the range $x \in [x_{i-3/2}, x_{i+5/2}]$, $f'(q)$ and $g'(q)$ denote the Jacobi matrices of the flux functions f and g . The eigenvalues are given by $\lambda = u \pm \sqrt{\frac{gA}{B}}$.

Finally this semidiscretization leads to the system of ODEs

$$q_i' = -\frac{1}{\Delta x} \left(f_{i+1/2} - f_{i-1/2} + M(q_i)(g_{i+1/2} - g_{i-1/2}) - (LF_{i+1/2} - LF_{i-1/2}) \right) + S(q_i) \quad (3.2)$$

Table 3.1 shows the convergence order p of this space discretization with different numbers n_x of finite volume cells for the second test problem.

The error is computed by

$$err = \sqrt{\frac{1}{neq} \sum_{i=1}^{neq} (q_{ana}(i) - q_{num}(i))^2}$$

with the number $neq = 2n_x$ of ODEs. $q_{num}(i)$ denotes the solution of (3.2) at time t_{end} and $q_{ana}(i)$ the corresponding solution component computed from a high accurate solution of (2.6) and $Q(x, t) = Q_0$. The maximum theoretical order three for very smooth problems is not completely achieved. On the other hand the nonoscillatory behaviour is very good, see figure 3.1 as an example.

nx	err	p
50	$4.12 \cdot 10^{-2}$	
100	$8.89 \cdot 10^{-3}$	2.2
200	$1.52 \cdot 10^{-3}$	2.5
400	$1.88 \cdot 10^{-4}$	3.0
800	$5.47 \cdot 10^{-5}$	1.8

Table 3.1: Error and convergence order of space discretization for problem 2

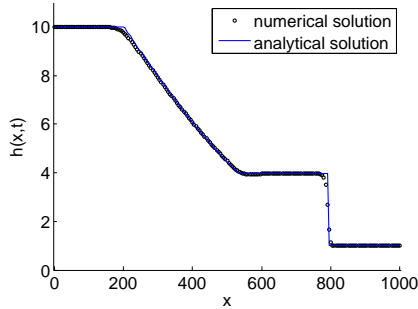


Figure 3.1: Analytical and numerical solution of the dambreak problem with $n_x = 200$ cells.

4 Linearly-implicit two-step peer methods

Linearly-implicit two-step peer methods were introduced first in parallel form in [11] and applied to MOL-problems in [18]. On s processors all s stages of these methods can be computed in parallel. Due to their two-step character the methods do not suffer from order reduction like ROW methods. In a sequential implementation the stability and robustness of the methods can be improved by using the actually computed approximations in later stages. These sequential peer methods were first studied in [8] where they were applied to stiff ODEs. For an initial value problem

$$y' = f(t, y), \quad y(t_0) = y_0 \in \mathbb{R}^n$$

an s -stage sequential linearly-implicit peer method is given by

$$\begin{aligned}
(I - h_m \gamma T_m) Y_{m,i} &= \sum_{j=1}^s b_{ij} Y_{m-1,j} + h_m \sum_{j=1}^s a_{ij} (F_{m-1,j} - T_m Y_{m-1,j}) \\
&+ h_m T_m \sum_{j=1}^{i-1} q_{ij} Y_{m,j}, \quad i = 1, \dots, s,
\end{aligned} \tag{4.1}$$

for a timestep $t_m \rightarrow t_{m+1}$ of size h_m , where $F_{m,i} := f(t_m + c_i h_m, Y_{m,i})$ and $\gamma > 0$. The values $Y_{m,i}$ are approximations to the solution $y(t)$ at points $t_{mi} = t_m + h_m c_i$. The

matrix T_m is supposed to be an approximation to the Jacobian $f_y(t_m, y(t_m))$ for stability reasons. We collect the coefficients b_{ij} , a_{ij} and q_{ij} in matrices B_m , A_m and Q and set $G = Q + \gamma I$. The index m here indicates that the matrices may depend on the stepsize ratio $\sigma = h_m/h_{m-1}$ of two consecutive steps, γ , Q and the nodes c_i are chosen constant. It looks natural to add in the $i - th$ stage in (4.1) a new term with function values F_{mj} for $j = 1, \dots, i - 1$. However, it is not obvious how to choose these additional coefficients and which properties of the method can be influenced by them. This problem will be discussed in the following. We consider now new peer methods of the form

$$(I - h_m \gamma T_m) Y_{m,i} = \sum_{j=1}^s b_{ij} Y_{m-1,j} + h_m \sum_{j=1}^s a_{ij} (F_{m-1,j} - T_m Y_{m-1,j}) \\ + h_m T_m \sum_{j=1}^{i-1} q_{ij} Y_{m,j} + h_m \sum_{j=1}^{i-1} r_{ij} F_{m,j}, \quad i = 1, \dots, s. \quad (4.2)$$

To study the order of the method we consider the local residual Δ_m by substituting the exact solution into the method. With $c = (c_i)$ and $R = (r_{ij})$ (R strictly lower triangular, constant) this gives (for simplicity with notation for scalar equations)

$$\Delta_m = (I - h_m G T_m) y(t_m + ch_m) - B_m y(t_{m-1} + ch_{m-1}) \\ - h_m A (y'(t_{m-1} + ch_{m-1}) - T_m y(t_{m-1} + ch_{m-1})) - h_m R y'(t_m + ch_m). \quad (4.3)$$

By Taylor expansion we get conditions for the order of the method which can be written with the notations $AB(l)$ and $G(l)$ corresponding to the terms without and with T_m :

$$AB(l) = c^l - \frac{1}{\sigma^l} B(c - \mathbb{1})^l - \frac{l}{\sigma^{l-1}} A(c - \mathbb{1})^{l-1} - l R c^{l-1} \quad (4.4)$$

$$G(l) = G c^l - \frac{1}{\sigma^l} A(c - \mathbb{1})^l, \quad (4.5)$$

with the notations $\mathbb{1} = (1, \dots, 1)^T$ and $c^l = (c_1^l, \dots, c_s^l)^T$. As in [8] we obtain

Lemma 4.1 *Let $AB(l) = 0$ be satisfied for $l = 0, \dots, p$, and $G(l) = 0$ for $l = 0, \dots, p - 1$. Then we have $\Delta_m = \mathcal{O}(h_{m-1}^{p+1})$, i.e. the method is consistent of order p . ■*

For convergence of order p we need in addition the zero stability of the method. When applying (4.2) with constant stepsize h to the test equation of A-stability

$$y' = \lambda y, \quad \text{Re } \lambda \leq 0, \quad (4.6)$$

we obtain with $T = \lambda$

$$Y_m = M(z) Y_{m-1}, \quad z = h\lambda.$$

Here the stability matrix $M(z)$ is given by

$$M(z) = (I - z(G + R))^{-1} B_m. \quad (4.7)$$

For $\lambda = 0$ we thus have $M(0) = B_m$. Zero stability requires that

$$\|B_{m+l} B_{m+l-1} \cdots B_{m+1} B_m\| \leq L \quad (4.8)$$

holds for some constant $L < \infty$ and for all m and $l \geq 0$.

Analogously to [8] we can prove

Theorem 4.1 *Let the peer method (4.2) be zero-stable. Let the conditions $AB(l) = 0$ be satisfied for $l = 0, \dots, p$ and let $G(l) = 0$ be satisfied for $l = 0, \dots, p - 1$. Then the method is convergent of order p . ■*

Consider now the determination of the coefficients. By requiring $G(l) = 0$ for $l = 0, \dots, s-1$ for given c, γ and Q the matrix A_m is uniquely defined by

$$A_m = GV_0S_mV_1^{-1}, \quad (4.9)$$

with $S_m = \text{diag}(1, \sigma, \dots, \sigma^{s-1})$ and the Vandermonde matrices

$$V_0 = (c_i^{j-1})_{i,j=1}^s, \quad V_1 = ((c_i - 1)^{j-1})_{i,j=1}^s.$$

From $AB(l) = 0$ for $l = 0, \dots, s - 1$ we then can compute the matrix B_m :

$$B_m = (V_0 - (G + R)V_0DF^T)S_mV_1^{-1} \quad (4.10)$$

where

$$D = \text{diag}(1, 2, \dots, s), \quad F = [e_2, e_3, \dots, e_s, 0],$$

with e_i the i -th unit vector.

In [8] optimally zero stable methods of order $p = s - 1$ were constructed. Optimal zero stability means that one eigenvalue of B_m is one, all others are zero. For given γ and c this condition determines the matrix Q for methods (4.1). On the other side we observe, that B_m and therefore the zero stability depend only on the sum $Q + R$ but not on Q and R separately. For optimally zero stable methods the sum $Q + R$ is therefore fixed. Furthermore, we cannot use the freedom in R for satisfying $AB(s) = 0$:

$$\begin{aligned} AB(s) &= c^s - \frac{1}{\sigma} B(c - \mathbb{1})^s - \frac{s}{\sigma^{s-1}} A(c - 1)^{s-1} - sRc^{s-1} \\ &= c^s - \frac{1}{\sigma} (V_0 - (G + R)V_0DF^T)S_mV_1^{-1}(c - \mathbb{1})^s - \frac{s}{\sigma^{s-1}} GV_0S_mV_1^{-1}(c - \mathbb{1})^{s-1} - sRc^{s-1} \\ &= c^s - \frac{1}{\sigma} (V_0 - (G + R)V_0DF^T)S_mV_1^{-1}(c - \mathbb{1})^s - s(G + R)c^{s-1}, \end{aligned}$$

because of

$$\frac{1}{\sigma^{s-1}} V_0S_mV_1^{-1}(c - 1)^{s-1} = c^{s-1}.$$

Since $G + R$ is determined by optimal zero stability the choice of R has no influence on $AB(s)$ and on the stability function $M(z)$. We therefore try to exploit the free parameters in R in two directions:

1. We want to decrease the error in $G(s + 1)$ (for $\sigma = 1$), i.e. we want to minimize $\|Gc^s - A(c - \mathbb{1})^s\|$.
2. We want to improve the stability for an inexact Jacobian. This is important if the Jacobian is kept constant for some steps. We consider the test equation (4.6) and assume $T = \lambda(1 + \varepsilon)$ with some small perturbation ε . In the limit $z \rightarrow \infty$ this leads to

$$M(\infty) = \varepsilon(G + R)^{-1}A + \mathcal{O}(\varepsilon^2).$$

Note, that this value can be changed by changing R leaving $G + R$ fixed, because A depends on G only.

Our strategy for the construction of peer methods (4.2) is then the following:

1. We use γ and c from the optimal zero stable methods in [8]. The nodes were chosen there as stretched Chebyshev points

$$c_i = -\frac{\cos\left(\left(i - \frac{1}{2}\right)\pi/s\right)}{\cos\left(\frac{1}{2}\pi/s\right)}.$$

$G + R$ is defined by optimal zero stability, A_m by (4.9) and B_m by (4.10).

2. By a random walk we search for R to minimize

$$zf = \|Gc^s - A(c - \mathbf{1})^s\|_\infty + \|(G + R)^{-1}A\|_\infty.$$

We obtained the following 4- and 5-stage methods of orders 3 and 4:

Peer4:

$$\begin{aligned} \gamma &= 0.3007483621794201, \\ c_1 &= -1.0000000000000000, & c_2 &= -0.4142135623730950, & c_3 &= 0.4142135623730950, \\ c_4 &= 1.0000000000000000, \\ q_{21} &= -0.2557258620204163, & q_{31} &= -2.8092538725265510, & q_{32} &= 2.1053752703389605, \\ q_{41} &= -4.7025185354231889, & q_{42} &= 3.5223722849260954, & q_{43} &= -0.6979859142803965, \\ r_{21} &= 0.8415122996473219, & r_{31} &= 3.2234674348996486, & r_{32} &= -1.1053752703389621, \\ r_{41} &= 4.9549716397167654, & r_{42} &= -2.2462299100107046, & r_{43} &= 1.1693904350714293. \end{aligned}$$

The method is $L(89.9)$ -stable, $zf = 4.5417$.

Peer5:

$$\begin{aligned} \gamma &= 0.2603688140218181, \\ c_1 &= -1.0000000000000000, & c_2 &= -0.6180339887498949, & c_3 &= -0.0000000000000001, \\ c_4 &= 0.6180339887498949, & c_5 &= 1.0000000000000000, \\ q_{21} &= 0.2018074171582183, & q_{31} &= 1.3863117322687473, & q_{32} &= -0.3533240600576059, \\ q_{41} &= -0.9637361710327425, & q_{42} &= 2.5686345355497875, & q_{43} &= -0.8575357035708924, \\ q_{51} &= 0.5816833109209432, & q_{52} &= 1.3001874720331683, & q_{53} &= 1.1563212061222223, \\ q_{54} &= -0.5846373640764505, \\ r_{21} &= 0.1801585940918885, & r_{31} &= -1.1953287266436965, & r_{32} &= 1.1623410544325599, \\ r_{41} &= 1.0424254968659954, & r_{42} &= -1.6959565392998046, & r_{43} &= 1.5242023702375591, \\ r_{51} &= -0.4937059700875543, & r_{52} &= -0.5745038109915331, & r_{53} &= -0.2199822079972364, \\ r_{54} &= 0.8296373640764505. \end{aligned}$$

The method is $L(89.3)$ -stable, $zf = 14.0592$.

Although we study peer methods for ODE problems here they can be adapted easily to DAE problems as well ([3], [12]).

5 Numerical experiments

The constructed peer methods are now applied to the four test problems of Section 2 and compared with other approved codes. All computations are performed in MATLAB.

The implementation of the peer methods differs in several aspects from that in [8].

In [12] implicit two-step peer methods were considered. The numerical results have shown that for higher accuracy the additional effort of the Newton iteration does not pay off. However for crude tolerances there was an improvement with respect to robustness for large stepsize changes. Because for the shallow water equations no very high accuracy is required we want to improve the robustness of the peer methods. We follow the idea of [12] and consider the linearly-implicit method (4.2) as the result of one Newton step and we allow one additional Newton step if necessary, a strategy that was also used in [10]. This means, we consider the implicit method

$$Y_{mi} = \sum_{j=1}^s b_{ij} Y_{m-1,j} + h\gamma F_{mi} + h \sum_{j=1}^{i-1} (r_{ij} + q_{ij}) F_{mj}.$$

The Newton iteration reads

$$(I - h\gamma T)(Y_{mi}^{l+1} - Y_{mi}^l) = -Y_{mi}^l + h\gamma F_{mi}^l + \sum_{j=1}^s b_{ij} Y_{m-1,j} + h \sum_{j=1}^{i-1} (r_{ij} + q_{ij}) F_{mj}, l = 0, 1. \quad (5.1)$$

With the special starting values

$$Y_{mi}^0 = \frac{1}{\gamma} \left(\sum_{j=1}^s a_{ij} Y_{m-1,j} - \sum_{j=1}^{i-1} q_{ij} Y_{mj} \right)$$

and replacing $F(Y_{mi}^0)$ by $\frac{1}{\gamma} \sum_{j=1}^s a_{ij} F_{m-1,j} - \frac{1}{\gamma} \sum_{j=1}^{i-1} q_{ij} F_{mj}$ (see [9]) we obtain just method (4.2). In the case

$$\max(\|Y_{mi}^1 - Y_{mi}^0\|_\infty, \|uy\|_\infty) > ktol \cdot atol,$$

we perform one additional Newton step. Here uy is the right hand side in (5.1). In our tests we use $ktol = \min(10^{-3}/atol, 10)$.

For stepsize control we compute a solution of order $\tilde{p} = s - 2$

$$\tilde{Y}_m = Y_{m-1,s} + h \sum_{j=1}^{s-1} d_j F_{mj}, \quad d = W^{-1}r,$$

where W is the left upper submatrix of V_0 of dimension $s - 1$ and $r^T = (1, 1/2, \dots, 1/(s - 2), 0)$. The stepsize control is done as usual with the weighted error

$$err = \left(\frac{1}{n} \sum_{i=1}^n \left(\frac{Y_{ms,i} - \tilde{Y}_{mi}}{sk_i} \right)^2 \right)^{1/2}$$

where

$$sk_i = atol + rtol \cdot \max(|Y_{m-1,s,i}|, |Y_{ms,i}|).$$

The starting values Y_{0i} are computed with `rodasp`.

In contrast to [8] we keep the Jacobian constant for several steps. It is recomputed after at most 10 steps or if

$$err \geq 0.9.$$

For comparison we use the state of the art codes `ode15s` [13], `rodasp` [15] and `radau5` of Ch. Engstler (a MATLAB version of the FORTRAN code `radau5` [5]). The problems 1–4 of Section 2 with the space discretization described in Section 3 are solved for $atol = rtol$ with $atol = 10^{-2}, 10^{-3}, \dots, 10^{-6}$. For all problems we used 200 space grid cells.

The computations were performed on an Intel Core2 Duo PC with 2.2 GHz using MATLAB R2007a. The following figures (in double-logarithmic scale) give the achieved accuracy versus computing time (in seconds). The error is computed by

$$ERR = \max_i \frac{|Y_{m,s,i} - Y_{ref,i}|}{1 + |Y_{ref,i}|}.$$

Here $Y_{m,s}$ is the numerical solution at the endpoint t_{end} and the reference solution Y_{ref} was computed with `ode15s` and $tol = 1.e - 12$.

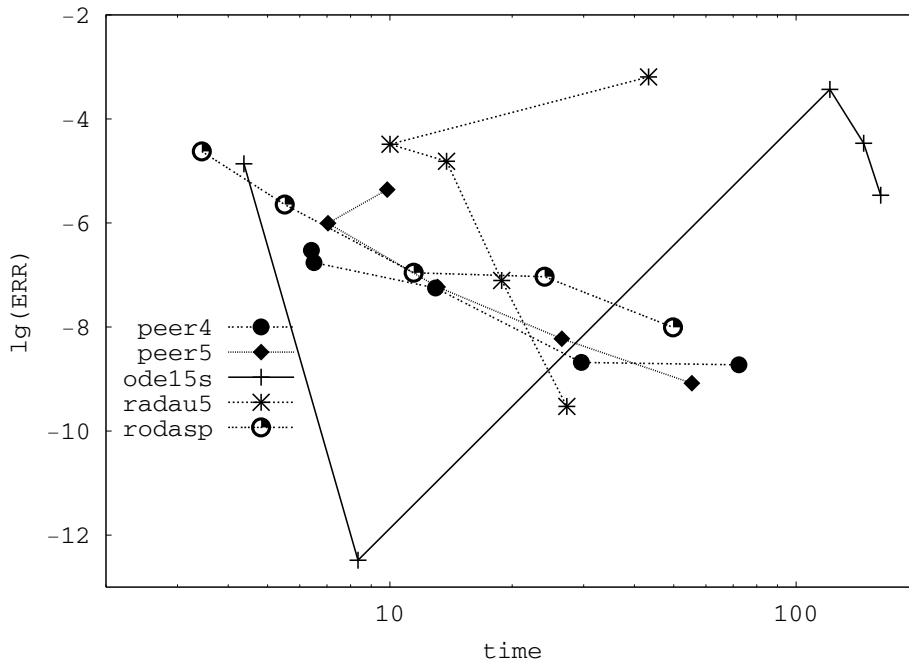


Figure 5.1: Results for Problem 1

The results of the tests define no “best” method. For Problems 3 and 4 `ode15s` is clearly superior, however it’s results for Problems 1 and 2 are unacceptable. `radau5` has difficulties for crude tolerances, which in weaker form is also the case for `peer5`. Summarized over all

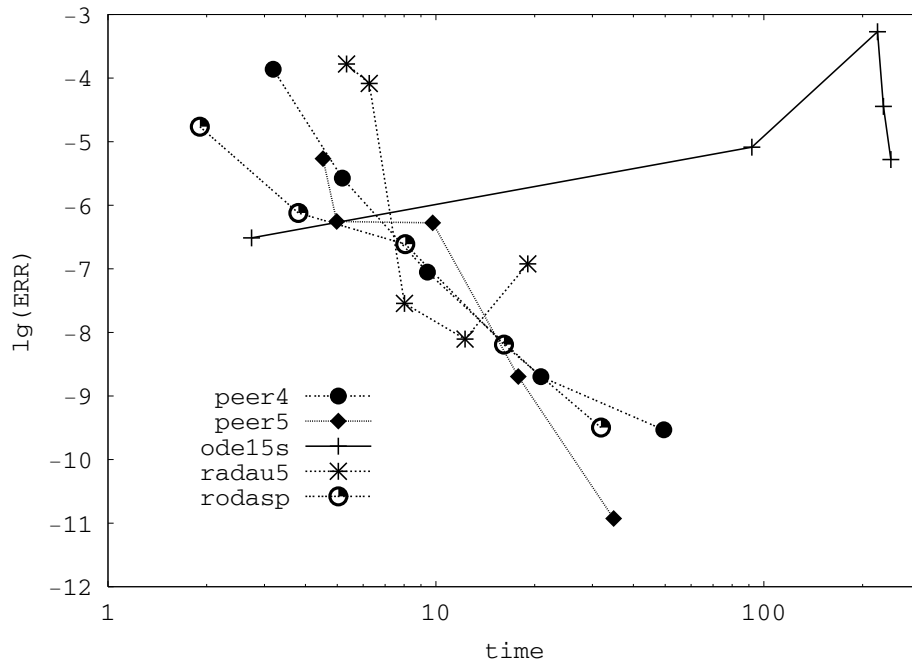


Figure 5.2: Results for Problem 2

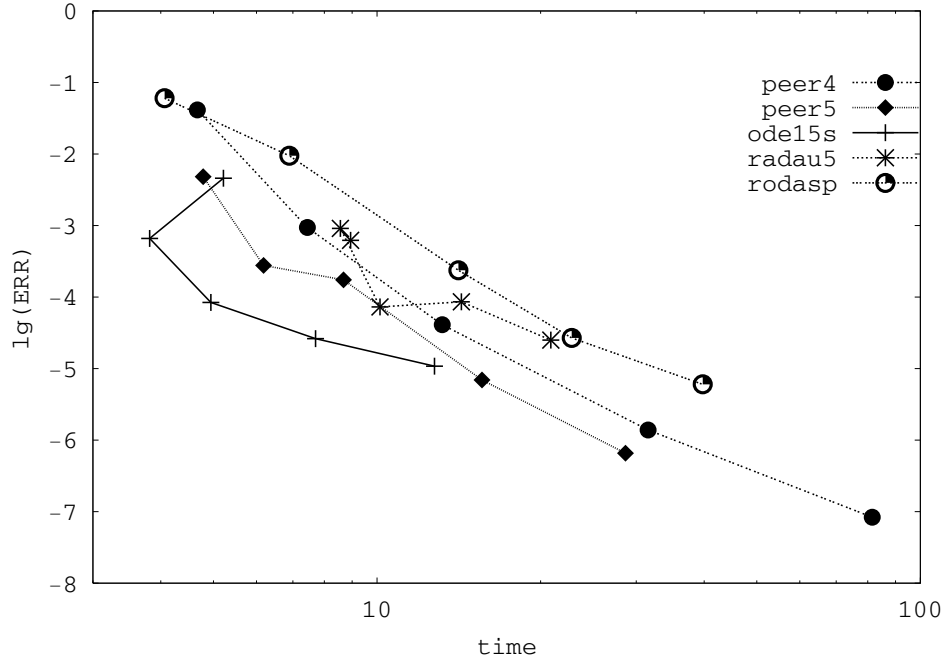


Figure 5.3: Results for Problem 3

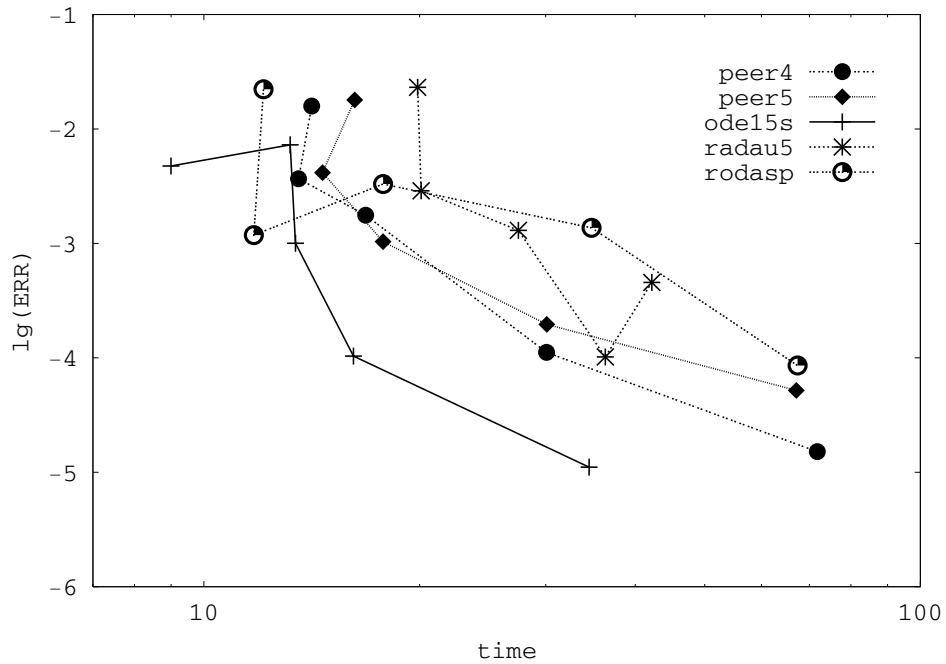


Figure 5.4: Results for Problem 4

problems `rodasp` and `peer4` show the best performance. They are reliable and efficient for all tolerances and all problems.

For completeness we give the average number of Newton steps for the peer methods over all stages in Table 5.1. Their number strongly decreases for more stringent tolerances.

	<i>atol</i>	Problem 1	Problem 2	Problem 3	Problem 4
peer4	$1.e - 2$	1.92	1.89	1.99	1.97
	$1.e - 3$	1.49	1.41	1.64	1.69
	$1.e - 4$	1.13	1.02	1.03	1.35
	$1.e - 5$	1.08	1.02	1.02	1.29
	$1.e - 6$	1.03	1.01	1.01	1.14
peer5	$1.e - 2$	1.93	1.91	1.95	1.95
	$1.e - 3$	1.51	1.40	1.76	1.65
	$1.e - 4$	1.15	1.04	1.37	1.29
	$1.e - 5$	1.17	1.09	1.27	1.33
	$1.e - 6$	1.13	1.08	1.16	1.29

Table 5.1: Number of Newton steps for the peer methods.

6 Conclusions

Numerical solution methods for the Saint-Venant equations are under consideration since more than 50 years [17]. While in the beginning mainly smooth solutions were considered today many schemes are available, that can cope also with discontinuities or shocks. The chosen space discretization scheme in this study according to the central WENO method is reliable and efficient. The main objective was the choice of a suitable time-integration scheme. The considered test problems for the shallow water equations make different demands on the integration methods. Not all integrators are suitable for all problems. In our tests `rodasp` and the new method `peer4` have shown to be most universally applicable to all problems considered.

References

- [1] A. Abdulle, *Fourth order Chebyshev methods with recurrence relation*, SIAM Journal on Scientific Computing, **23**, (2002), 2041-2054.
- [2] J.A. Cunge, F.M. Holly, A. Verwey, *Practical aspects of computational river hydraulics*, Pitman (1980).
- [3] A. Gerisch, J. Lang, H. Podhaisky, and R. Weiner, *High-order linearly implicit two-step peer - finite element methods for time-dependent PDEs*, APNUM 59, 624–638 (2009).
- [4] S. Gottlieb, D.I. Ketcheson, C-W. Shu, *High order strong stability time discretizations*, J.Sci.Comput 38, 251-289 (2009).
- [5] E. Hairer, G. Wanner, *Solving Ordinary Differential Equations II, Stiff and differential algebraic Problems*, (2nd ed.), Springer-Verlag, Berlin, 1996.
- [6] W. Hundsdorfer, J.G. Verwer, *Numerical solution of advection-diffusion-reaction equations*, Springer Series in Comput. Math. **33**, (2003).
- [7] A. Kurganov, D. Levy, *A third-order semidiscrete central scheme for conservation laws and convection-diffusion equations*, SIAM J. Sci. Comput. Vol.22m No4, 1461-1488 (2000).
- [8] H. Podhaisky, R. Weiner and B.A. Schmitt, *Rosenbrock-type 'Peer' two-step methods*, APNUM 53, 409–420 (2005).
- [9] H. Podhaisky, R. Weiner and B.A. Schmitt, *Linearly-implicit two-step methods and their implementation in Nordsieck form*, APNUM 56, 374-387 (2006).
- [10] T. Schierz, *Konstruktion und Test impliziter Peer-Methoden*, Diplomarbeit, Institut für Mathematik, Martin-Luther-Universität Halle-Wittenberg (2008).
- [11] B.A. Schmitt, R. Weiner, *Parallel two-step W-methods with peer variables*, SIAM J. Numer. Anal. 42, 265–282 (2004).

- [12] B.A. Schmitt, R. Weiner and K. Erdmann, *Implicit parallel peer methods for stiff initial value problems*, APNUM 53, 457–470 (2005).
- [13] L.F. Shampine, M. W. Reichelt, *The MATLAB ODE Suite*, SIAM Journal on Scientific Computing, Vol. 18, 1–22, (1997).
- [14] G. Steinebach, *Mathematische Modellbildung und numerische Methoden zur Strömungs-, Transport- und Reaktionssimulation in Netzwerken*, in Zacharias, Ch. et. al.(Hrsg.): *Forschungsspitzen und Spitzenforschung, Innovationen an der Fachhochschule Bonn-Rhein-Sieg*, S. 151-163, Physica-Verlag (2008).
- [15] G. Steinebach, P. Rentrop, *An adaptive method of lines approach for modelling flow and transport in rivers*, in: A.Vande Wouwer, Ph. Saucy, W.E. Schiesser (eds.), *Adaptive method of lines*, Chapman & Hall/CRC, 2001, 181–205. Chapman & Hall/CRC (2001).
- [16] G. Steinebach, S. Rademacher, P. Rentrop and M. Schulz, *Mechanisms of coupling in river flow simulation systems*, J. Comput. Appl. Math. 168(1–2), 459–470 (2004).
- [17] J.J. Stoker, *Water waves, the mathematical theory with applications*, Interscience Publishers Inc., New York, 1957.
- [18] R. Weiner, B.A. Schmitt and H. Podhaisky, *Parallel ‘Peer’ two-step W-methods and their application to MOL-systems*, APNUM 48, 425–439 (2004).

Reports of the Institutes 2009

- 01-09.** Jan Prüss, Vicente Vergara, Rico Zacher, *Well-Posedness and Long-Time Behaviour for the Non-Isothermal Cahn-Hilliard Equation with Memory*
- 02-09.** Siegfried Carl, Patrick Winkert, *General comparison principle for variational-hemivariational inequalities*
- 03-09.** Patrick Winkert, *L^∞ -Estimates for Nonlinear Elliptic Neumann Boundary Value Problems*
- 04-09.** Wilma Di Palma, *John Neper's rods: Calculations are boring and tiring! Leon Battista Alberti's Cipher Wheel*
- 05-09.** Stefan Sperlich, *An integration calculus for stochastic processes with stationary increments and spectral density with applications to parabolic Volterra equations*
- 06-09.** Stefan Sperlich, *A Regularity Theory for Stochastic Processes with Stationary Increments and Spectral Density with Applications to Anomalous Diffusion*
- 07-09.** Andreas Löhne, Thomas Riedrich, Christiane Tammer, *Festschrift zum 75. Geburtstag von Prof. Dr. Alfred Göpfert*
- 08-09.** Sadaf Manzoor and Salahuddin, *Outliers in Uniform Distribution*