MULTIRATE PARTITIONED RUNGE–KUTTA METHODS

M. GÜNTHER, A. KVÆRNO, and P. RENTROP

1 IWRMM, Universität Karlsruhe (TH), D-76128 Karlsruhe, Germany
   email: guenther@iwrmm.math.uni-karlsruhe.de, rentrop@iwrmm.math.uni-karlsruhe.de
2 Department of Mathematical Sciences, Norwegian University of Science and Technology,
   N-7491 Trondheim, Norway. email: anne@math.ntnu.no

Abstract.

The coupling of subsystems in a hierarchical modelling approach leads to different
time constants in the dynamical simulation of technical systems. Multirate schemes
exploit the different time scales by using different time steps for the subsystems. The
stiffness of the system or at least of some subsystems in chemical reaction kinetics or
network analysis, for example, forbids the use of explicit integration schemes. To cope
with stiff problems, we introduce multirate schemes based on partitioned Runge–Kutta
methods which avoid the coupling between active and latent components based on
interpolating and extrapolating state variables. Order conditions and test results for
such a lower order MPRK method are presented.

AMS subject classification: 65L05.

Key words: Coupled systems, stiff systems, multirate, partitioned Runge–Kutta
schemes, partitioning, latency, inverter chain.

1 Introduction.

The dynamical behaviour of coupled systems is characterized by different time
scales of the subsystems. In radio frequency applications, for example, one has
to deal with coupled systems of analogue and digital networks. Whereas the
analogue part runs in the microsecond scale, the digital part is measured in
nanoseconds. This requires multirate integration schemes using the inherent step
size of each subsystem.

In general, the dynamics of the whole system can be described by an initial
value problem of ordinary differential equations

\[ \dot{y} = f(t, y), \quad y(t_0) = y_0, \quad y \in \mathbb{R}^n. \]  

Due to a hierarchical modeling process, the system is often given partitioned into
p subsystems:

\[
\begin{align*}
\dot{y}_1 &= f_1(t, y_1, \ldots, y_p), \\
&\vdots \\
\dot{y}_p &= f_p(t, y_1, \ldots, y_p),
\end{align*}
\]

(1.2) with initial value \(y(t_0) = y_0\) for the partitioned solution vector \(y = (y_1, \ldots, y_p)^T\), \(y_0 \in \mathbb{R}^{n_i}, i = 1, \ldots, n, \sum_{i=1}^p n_i = n\).

For the numerical solution of (1.2) multirate schemes have been derived, where the different step sizes are introduced explicitly in the discretization. Basic work on backward differentiation formulas (BDF) can be found in Gear and Wells [4], Skelboe [11], and Skelboe and Anderson [10]. Multirate extrapolation [2] and Runge–Kutta [3] schemes are successfully used in stellar problems by Engstler and Lubich. In Günther and Rentrop [5, 6] multirate Rosenbrock–Wanner (MROW) methods are used for VLSI applications of electrical networks. One shortcoming of all these multirate methods derived so far is the coupling between active and latent components by interpolating and extrapolating state variables, which makes their implementation very difficult into existing simulation packages.

Recently, Kværnø, and Rentrop [8] have derived a concise theory of explicit multirate Runge–Kutta methods which avoids the large overhead caused by coupling subsystems via interpolation and extrapolation. However, in many technical applications such as chemical reaction kinetics or electrical network simulation, the stiffness of the system or at least of some subsystems forbids the use of explicit schemes. To cope also with stiff problems, we generalize in Section 2 the former approach to Multirate Partitioned Runge–Kutta (MPRK) methods.

Section 3 includes the order conditions with and without simplifying conditions, and a coefficient set for an embedded method of order (2)3 is derived. Finally, an inverter chain taken from electric circuit simulation, shows the performance of the approach.

2 Generalized multirate methods.

In the following we concentrate our investigations on autonomous initial value problems, whose state vector \(y \in \mathbb{R}^n\) is partitioned into only two parts: latent components \(y_L \in \mathbb{R}^{n_L}\) and a small number of active components \(y_A \in \mathbb{R}^{n_A}\) with \(n_A + n_L = n\) and \(n_A \ll n_L\),

\[
\begin{align*}
\dot{y}_A &= f_A(y_A, y_L), & y_A(t_0) &= y_{A0}, \\
\dot{y}_L &= f_L(y_A, y_L), & y_L(t_0) &= y_{L0}.
\end{align*}
\]

The active components \(y_A\) are integrated with a small step size \(h\), the latent components \(y_L\) with a large step size \(H\). A synchronization of the micro steps \(h\) and the macro step \(H\) is performed after each macrostep. In each macro step the micro steps are assumed to be constant: \(H = mh\).

Stiffness is only expected in the latent part, whereas the active part is characterized by rapidly changing solution courses. The assumption that all the