

## MULTI-ADAPTIVE GALERKIN METHODS FOR ODES I\*

ANDERS LOGG<sup>†</sup>

**Abstract.** We present *multi-adaptive* versions of the standard continuous and discontinuous Galerkin methods for ODEs. Taking adaptivity one step further, we allow for individual time-steps, order and quadrature, so that in particular each individual component has its own time-step sequence. This paper contains a description of the methods, an analysis of their basic properties, and a posteriori error analysis. In the accompanying paper [A. Logg, *SIAM J. Sci. Comput.*, submitted], we present adaptive algorithms for time-stepping and global error control based on the results of the current paper.

**Key words.** multi-adaptivity, individual time-steps, local time-steps, ODE, continuous Galerkin, discontinuous Galerkin, global error control, adaptivity, mcG(q), mdG(q)

**AMS subject classifications.** 65L05, 65L07, 65L20, 65L50, 65L60, 65L70

**PII.** S1064827501389722

**1. Introduction.** In this paper, we present multi-adaptive Galerkin methods for initial value problems for systems of ODEs of the form

$$(1.1) \quad \begin{cases} \dot{u}(t) &= f(u(t), t), & t \in (0, T], \\ u(0) &= u_0, \end{cases}$$

where  $u : [0, T] \rightarrow \mathbb{R}^N$ ,  $f : \mathbb{R}^N \times (0, T] \rightarrow \mathbb{R}^N$  is a given bounded function that is Lipschitz-continuous in  $u$ ,  $u_0 \in \mathbb{R}^N$  is a given initial condition, and  $T > 0$  is a given final time. We use the term *multi-adaptivity* to describe methods with individual time-stepping for the different components  $u_i(t)$  of the solution vector  $u(t) = (u_i(t))$ , including (i) time-step length, (ii) order, and (iii) quadrature, all chosen adaptively in a computational feedback process. In the companion paper [29], we apply the multi-adaptive methods to a variety of problems to illustrate the potential of multi-adaptivity.

The ODE (1.1) models a very large class of problems, covering many areas of applications. Often different solution components have different time-scales and thus ask for individual time-steps. A prime example to be studied in detail below is our own solar system, where the moon orbits around Earth once every month, whereas the period of Pluto is 250 years. In numerical simulations of the solar system, the time-steps needed to track the orbit of the moon accurately are thus much less than those required for Pluto, the difference in time-scales being roughly a factor 3,000.

Surprisingly, individual time-stepping for ODEs has received little attention in the large literature on numerical methods for ODEs; see, e.g., [4, 21, 22, 3, 34]. For specific applications, such as the  $n$ -body problem, methods with individual time-stepping have been used—see, e.g., [31, 1, 5] or [25]—but a general methodology has been lacking. Our aim is to fill this gap. For time-dependent PDEs, in particular for conservation laws of the type  $\dot{u} + f(u)_x = 0$ , attempts have been made to construct methods with individual (locally varying in space) time-steps. Flaherty et al. [20] have constructed a method based on the discontinuous Galerkin method combined with local forward

---

\*Received by the editors May 23, 2001; accepted for publication (in revised form) November 13, 2002; published electronically May 2, 2003.

<http://www.siam.org/journals/sisc/24-6/38972.html>

<sup>†</sup>Department of Computational Mathematics, Chalmers University of Technology, SE-412 96 Göteborg, Sweden (logg@math.chalmers.se).

Euler time-stepping. A similar approach is taken in [6], where a method based on the original work by Osher and Sanders [33] is presented for conservation laws in one and two space dimensions. Typically the time-steps used are based on local CFL conditions rather than error estimates for the global error and the methods are low order in time (meaning  $\leq 2$ ). We believe that our work on multi-adaptive Galerkin methods (including error estimation and arbitrary order methods) presents a general methodology to individual time-stepping, which will result in efficient integrators also for time-dependent PDEs.

The methods presented in this paper fall within the general framework of adaptive Galerkin methods based on piecewise polynomial approximation (finite element methods) for differential equations, including the continuous Galerkin method  $cG(q)$  of order  $2q$ , and the discontinuous Galerkin method  $dG(q)$  of order  $2q + 1$ ; more precisely, we extend the  $cG(q)$  and  $dG(q)$  methods to their multi-adaptive analogues  $mcG(q)$  and  $mdG(q)$ . Earlier work on adaptive error control for the  $cG(q)$  and  $dG(q)$  methods include [7, 16, 24, 18, 17, 19]. The techniques for error analysis used in these references, developed by Johnson and coworkers (see, e.g., [11, 12, 10, 13, 14, 15], and [8] in particular) naturally carries over to the multi-adaptive methods.

The outline of the paper is as follows: In section 2 we summarize the key features of the multi-adaptive methods, and in section 3 we discuss the benefits of the new methods in comparison to standard ODE codes. We then motivate and present the formulation of the multi-adaptive methods  $mcG(q)$  and  $mdG(q)$  in section 4. Basic properties of these methods, such as order, energy conservation, and monotonicity, are discussed in section 5. In the major part of this paper, section 6, we derive a posteriori error estimates for the two methods based on duality arguments, including Galerkin errors, numerical errors, and quadrature errors. We also prove an a posteriori error estimate for stability factors computed from approximate dual solutions.

**2. Key features.** We summarize the key features of our work on the  $mcG(q)$  and  $mdG(q)$  methods as follows.

**2.1. Individual time-steps and order.** To discretize (1.1), we introduce for each component,  $i = 1, \dots, N$ , a partition of the time-interval  $(0, T]$  into  $M_i$  subintervals,  $I_{ij} = (t_{i,j-1}, t_{ij}]$ ,  $j = 1, \dots, M_i$ , and we seek an approximate solution  $U(t) = (U_i(t))$  such that  $U_i(t)$  is a polynomial of degree  $q_{ij}$  on every local interval  $I_{ij}$ . Each individual component  $U_i(t)$  thus has its own sequence of time-steps,  $\{k_{ij}\}_{j=1}^{M_i}$ . The entire collection of individual time-intervals  $\{I_{ij}\}$  may be organized into a sequence of *time-slabs*, collecting the time-intervals between certain synchronised time-levels common to all components, as illustrated in Figure 2.1.

**2.2. Global error control.** Our goal is to compute an approximation  $U(T)$  of the exact solution  $u(T)$  at final time  $T$  within a given tolerance  $TOL > 0$ , using a minimal amount of computational work. This goal includes an aspect of *reliability* (the error should be less than the tolerance) and an aspect of *efficiency* (minimal computational work). To measure the error we choose a norm, such as the Euclidean norm  $\|\cdot\|$  on  $\mathbb{R}^N$ , or more generally some other quantity of interest (see [32]).

The mathematical basis of global error control in  $\|\cdot\|$  for  $mcG(q)$  is an error representation of the form

$$(2.1) \quad \|U(T) - u(T)\| = \int_0^T (R, \varphi) dt,$$

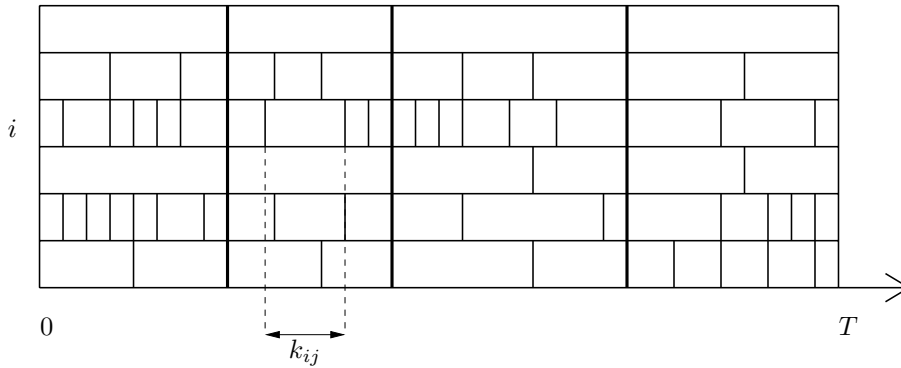


FIG. 2.1. Individual time-discretizations for different components.

where  $R = (R_i) = R(U, t) = \dot{U}(t) - f(U(t), t)$  is the residual vector of the approximate solution  $U(t)$ ,  $\varphi(t)$  is the solution of an associated linearized dual problem, and  $(\cdot, \cdot)$  is the  $\mathbb{R}^N$  scalar product.

Using the Galerkin orthogonality, the error representation can be converted into an error bound of the form

$$(2.2) \quad \|U(T) - u(T)\| \leq \sum_{i=1}^N S_i(T) \max_{0 \leq t \leq T} k_i(t)^{q_i(t)} |R_i(U, t)|,$$

where  $\{S_i(T)\}_{i=1}^N$  are stability factors for the different components, depending on the dual solution  $\varphi(t)$ , and where  $k_i(t) = k_{ij}$ ,  $q_i(t) = q_{ij}$  for  $t \in I_{ij}$ . The error bound may take different forms depending on how  $\int_0^T (R, \varphi) dt$  is bounded in terms of  $R$  and  $\varphi$ .

By solving the dual problem numerically, the individual stability factors  $S_i(T)$  may be determined approximately, and thus the right-hand side of (2.2) may be evaluated. The adaptive algorithm seeks to satisfy the stopping criterion

$$(2.3) \quad \sum_{i=1}^N S_i(T) \max_{0 \leq t \leq T} k_i(t)^{q_i(t)} |R_i(U, t)| \leq \text{TOL},$$

with maximal time-steps  $k = (k_i(t))$ .

**2.3. Iterative methods.** Both mcG( $q$ ) and mdG( $q$ ) give rise to systems of nonlinear algebraic equations, coupling the values of  $U(t)$  over each time-slab. Solving these systems with full Newton may be quite heavy, and we have instead successfully used diagonal Newton methods of more explicit nature.

**2.4. Implementation of higher-order methods.** We have implemented mcG( $q$ ) and mdG( $q$ ) in C++ for arbitrary  $q$ , which in practice means  $2q \leq 50$ . The implementation, *Tanganyika*, is described in more detail in [29] and is publicly (GNU GPL) available for Linux/Unix [30].

**2.5. Applications.** We have applied mcG( $q$ ) and mdG( $q$ ) to a variety of problems to illustrate their potential; see [29]. (See also [27] and [26].) In these applications, including the Lorenz system, the solar system, and a number of time-dependent PDE problems, we demonstrate the use of individual time-steps, and for each system we solve the dual problem to collect extensive information about the problems stability features, which can be used for global error control.

**3. Comparison with standard ODE codes.** Standard ODE codes use time-steps which are variable in time but the same for all components, and the time-steps are adaptively chosen by keeping the “local error” below a given local error tolerance set by the user. The global error connects to the local error through an estimate, corresponding to (2.2), of the form

$$(3.1) \quad \{\text{global error}\} \leq S \max\{\text{local error}\},$$

where  $S$  is a stability factor. Standard codes do not compute  $S$ , which means that the connection between the global error and the local error is left to be determined by the clever user, typically by computing with a couple of different tolerances.

Comparing the adaptive error control of standard ODE codes with the error control presented in this paper and the accompanying paper [29], an essential difference is thus the technique to estimate the global error: either by clever trial-and-error or, as we prefer, by solving the dual problem and computing the stability factors. Both approaches carry extra costs and what is best may be debated; see, e.g., [32] for a comparison.

However, expanding the scope to multi-adaptivity with individual stability factors for the different components, trial-and-error becomes very difficult or impossible, and the methods for adaptive time-stepping and error control presented below based on solving the dual problem seem to bring clear advantages in efficiency and reliability.

For a presentation of the traditional approach to error estimation in ODE codes, we refer to [2], where the following rather pessimistic view is presented: *Here we just note that a precise error bound is often unknown and not really needed.* We take the opposite view: *global error control is always needed and often possible to obtain at a reasonable cost.* We hope that multi-adaptivity will bring new life to the discussion on efficient and reliable error control for ODEs.

**4. Multi-adaptive Galerkin.** In this section we present the multi-adaptive Galerkin methods, mcG( $q$ ) and mdG( $q$ ), based on the discretization presented in section 2.1.

**4.1. The mcG( $q$ ) method.** The mcG( $q$ ) method for (1.1) reads as follows: Find  $U \in V$  with  $U(0) = u_0$ , such that

$$(4.1) \quad \int_0^T (\dot{U}, v) dt = \int_0^T (f(U, \cdot), v) dt \quad \forall v \in W,$$

where

$$(4.2) \quad \begin{aligned} V &= \{v \in [C([0, T])]^N : v_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}}(I_{ij}), j = 1, \dots, M_i, i = 1, \dots, N\}, \\ W &= \{v : v_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}-1}(I_{ij}), j = 1, \dots, M_i, i = 1, \dots, N\}, \end{aligned}$$

and where  $\mathcal{P}^q(I)$  denotes the linear space of polynomials of degree  $\leq q$  on  $I$ . The trial functions in  $V$  are thus continuous piecewise polynomials, locally of degree  $q_{ij}$ , and the test functions in  $W$  are discontinuous piecewise polynomials that are locally of degree  $q_{ij} - 1$ .

Noting that the test functions are discontinuous, we can rewrite the global problem (4.1) as a number of successive local problems for each component: For  $i = 1, \dots, N$ ,  $j = 1, \dots, M_i$ , find  $U_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}}(I_{ij})$  with  $U_i(t_{i,j-1})$  given, such that

$$(4.3) \quad \int_{I_{ij}} \dot{U}_i v dt = \int_{I_{ij}} f_i(U, \cdot) v dt \quad \forall v \in \mathcal{P}^{q_{ij}-1}(I_{ij}).$$

We notice the presence of the vector  $U(t) = (U_1(t), \dots, U_N(t))$  in the local problem for  $U_i(t)$  on  $I_{ij}$ . If thus component  $U_{i_1}(t)$  couples to component  $U_{i_2}(t)$  through  $f$ , this means that in order to solve the local problem for component  $U_{i_1}(t)$  we need to know the values of component  $U_{i_2}(t)$  and vice versa. The solution is thus implicitly defined by (4.3). Notice also that if we define the *residual*  $R$  of the approximate solution  $U$  as  $R_i(U, t) = \dot{U}_i(t) - f_i(U(t), t)$ , we can rewrite (4.3) as

$$(4.4) \quad \int_{I_{ij}} R_i(U, \cdot)v \, dt = 0 \quad \forall v \in \mathcal{P}^{q_{ij}-1}(I_{ij}),$$

i.e., the residual is orthogonal to the test space on every local interval. We refer to this as the *Galerkin orthogonality* for the mcG( $q$ ) method.

Making an ansatz for every component  $U_i(t)$  on every local interval  $I_{ij}$  in terms of a nodal basis for  $\mathcal{P}^{q_{ij}}(I_{ij})$  (see the appendix), we can rewrite (4.3) as

$$(4.5) \quad \xi_{ijm} = \xi_{ij0} + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) f_i(U(t), t) \, dt, \quad m = 1, \dots, q_{ij},$$

where  $\{\xi_{ijm}\}_{m=0}^{q_{ij}}$  are the nodal degrees of freedom for  $U_i(t)$  on the interval  $I_{ij}$ ,  $\{w_m^{[q]}\}_{m=1}^q \subset \mathcal{P}^{q-1}(0, 1)$  are corresponding polynomial weight functions, and  $\tau_{ij}$  maps  $I_{ij}$  to  $(0, 1]$ :  $\tau_{ij}(t) = (t - t_{i,j-1})/(t_{ij} - t_{i,j-1})$ . Here we assume that the solution is expressed in terms of a nodal basis with the end-points included, so that by the continuity requirement  $\xi_{ij0} = \xi_{i,j-1,q_{i,j-1}}$ .

Finally, evaluating the integral in (4.5) using nodal quadrature, we obtain a fully discrete scheme in the form of an implicit Runge–Kutta method: For  $i = 1, \dots, N$ ,  $j = 1, \dots, M_i$ , find  $\{\xi_{ijm}\}_{m=0}^{q_{ij}}$ , with  $\xi_{ij0}$  given by the continuity requirement, such that

$$(4.6) \quad \xi_{ijm} = \xi_{ij0} + k_{ij} \sum_{n=0}^{q_{ij}} w_{mn}^{[q_{ij}]} f_i(U(\tau_{ij}^{-1}(s_n^{[q_{ij}]})), \tau_{ij}^{-1}(s_n^{[q_{ij}]})), \quad m = 1, \dots, q_{ij},$$

for certain weights  $\{w_{mn}^{[q]}\}$  and certain nodal points  $\{s_n^{[q]}\}$  (see the appendix).

**4.2. The mdG( $q$ ) method.** The mdG( $q$ ) method in local form, corresponding to (4.3), reads as follows: For  $i = 1, \dots, N$ ,  $j = 1, \dots, M_i$ , find  $U_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}}(I_{ij})$ , such that

$$(4.7) \quad [U_i]_{i,j-1}v(t_{i,j-1}^+) + \int_{I_{ij}} \dot{U}_i v \, dt = \int_{I_{ij}} f_i(U, \cdot)v \, dt \quad \forall v \in \mathcal{P}^{q_{ij}}(I_{ij}),$$

where  $[\cdot]$  denotes the jump, i.e.,  $[v]_{ij} = v(t_{ij}^+) - v(t_{ij}^-)$ , and the initial condition is specified for  $i = 1, \dots, N$ , by  $U_i(0^-) = u_i(0)$ . On a global level, the trial and test spaces are given by

$$(4.8) \quad V = W = \{v : v_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}}(I_{ij}), j = 1, \dots, M_i, i = 1, \dots, N\}.$$

In the same way as for the continuous method, we define the residual  $R$  of the approximate solution  $U$  as  $R_i(U, t) = \dot{U}_i(t) - f_i(U(t), t)$ , defined on the inner of every local interval  $I_{ij}$ , and we rewrite (4.7) in the form

$$(4.9) \quad [U_i]_{i,j-1}v(t_{i,j-1}^+) + \int_{I_{ij}} R_i(U, \cdot)v \, dt = 0 \quad \forall v \in \mathcal{P}^{q_{ij}}(I_{ij}).$$

We refer to this as the Galerkin orthogonality for the mdG( $q$ ) method. Notice that this is similar to (4.4) if we extend the integral in (4.4) to include the left end-point of the interval  $I_{ij}$ . (The derivative of the discontinuous solution is a Dirac delta function at the end-point.)

Making an ansatz for the solution in terms of some nodal basis, we get, as for the continuous method, the following explicit version of (4.7) on every local interval:

$$(4.10) \quad \xi_{ijm} = \xi_{ij0}^- + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) f_i(U(t), t) dt, \quad m = 0, \dots, q_{ij},$$

or, applying nodal quadrature,

$$(4.11) \quad \xi_{ijm} = \xi_{ij0}^- + k_{ij} \sum_{n=0}^{q_{ij}} w_{mn}^{[q_{ij}]} f_i(U(\tau_{ij}^{-1}(s_n^{[q_{ij}]}), \tau_{ij}^{-1}(s_n^{[q_{ij}]}), \quad m = 0, \dots, q_{ij},$$

where the weight functions, the nodal points, and the weights are not the same as for the continuous method.

**4.3. The multi-adaptive mcG( $q$ )-mdG( $q$ ) method.** The discussion above for the two methods extends naturally to using different methods for different components. Some of the components could therefore be solved for using the mcG( $q$ ) method, while for others we use the mdG( $q$ ) method. We can even change methods between different intervals.

Although the formulation thus includes adaptive orders and methods, as well as adaptive time-steps, our focus will be mainly on adaptive time-steps.

**4.4. Choosing basis functions and quadrature.** What remains in order to implement the two methods specified by (4.6) and (4.11) is to choose basis functions and quadrature. For simplicity and efficiency reasons, it is desirable to let the nodal points for the nodal basis coincide with the quadrature points. It turns out that for both methods, the mcG( $q$ ) and the mdG( $q$ ) methods, this is possible to achieve in a natural way. We thus choose the  $q + 1$  Lobatto quadrature points for the mcG( $q$ ) method, i.e., the zeros of  $xP_q(x) - P_{q-1}(x)$ , where  $P_q$  is the  $q$ th-order Legendre polynomial on the interval; for the mdG( $q$ ) method, we choose the Radau quadrature points, i.e., the zeros of  $P_q(x) + P_{q+1}(x)$  on the interval (with time reversed so as to include the right end-point). See [28] for a detailed discussion on this subject. The resulting discrete schemes are related to the implicit Runge–Kutta methods referred to as Lobatto and Radau methods; see, e.g., [3].

**5. Basic properties of the multi-adaptive Galerkin methods.** In this section we examine some basic properties of the multi-adaptive methods, including order, energy conservation, and monotonicity.

**5.1. Order.** The standard cG( $q$ ) and dG( $q$ ) methods are of order  $2q$  and  $2q + 1$ , respectively. The corresponding properties hold for the multi-adaptive methods, i.e., mcG( $q$ ) is of order  $2q$  and mdG( $q$ ) is of order  $2q + 1$ , assuming that the exact solution  $u$  is smooth. We examine this in more detail in subsequent papers.

**5.2. Energy conservation for mcG( $q$ ).** The standard cG( $q$ ) method is energy-conserving for Hamiltonian systems. We now prove that also the mcG( $q$ ) method has this property, with the natural restriction that we should use the same time-steps for every pair of positions and velocities. We consider a Hamiltonian system,

$$(5.1) \quad \ddot{x} = -\nabla_x P(x),$$

on  $(0, T]$  with  $x(t) \in \mathbb{R}^N$ , together with initial conditions for  $x$  and  $\dot{x}$ . Here  $\ddot{x}$  is the acceleration, which by Newton's second law is balanced by the force  $F(x) = -\nabla_x P(x)$  for some potential field  $P$ . With  $u = x$  and  $v = \dot{x}$  we rewrite (5.1) as

$$(5.2) \quad \begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v \\ F(u) \end{bmatrix} = \begin{bmatrix} f_u(v) \\ f_v(u) \end{bmatrix} = f(u, v).$$

The total energy  $E(t)$  is the sum of the kinetic energy  $K(t)$  and the potential energy  $P(x(t))$ ,

$$(5.3) \quad E(t) = K(t) + P(x(t)),$$

with

$$(5.4) \quad K(t) = \frac{1}{2} \|\dot{x}(t)\|^2 = \frac{1}{2} \|v(t)\|^2.$$

Multiplying (5.1) with  $\dot{x}$  it is easy to see that energy is conserved for the continuous problem, i.e.,  $E(t) = E(0)$  for all  $t \in [0, T]$ . We now prove the corresponding property for the discrete mcG( $q$ ) solution of (5.2).

**THEOREM 5.1.** *The multi-adaptive continuous Galerkin method conserves energy in the following sense: Let  $(U, V)$  be the mcG( $q$ ) solution to (5.2) defined by (4.3). Assume that the same time-steps are used for every pair of positions and corresponding velocities. Then at every synchronized time-level  $\bar{t}$ , such as, e.g.,  $T$ , we have*

$$(5.5) \quad K(\bar{t}) + P(\bar{t}) = K(0) + P(0),$$

with  $K(t) = \frac{1}{2} \|V(t)\|^2$  and  $P(t) = P(U(t))$ .

*Proof.* If every pair of positions and velocities have the same time-step sequence, then we may choose  $\dot{V}$  as a test function in the equations for  $U$ , to get

$$\int_0^{\bar{t}} (\dot{U}, \dot{V}) dt = \int_0^{\bar{t}} (V, \dot{V}) dt = \frac{1}{2} \int_0^{\bar{t}} \frac{d}{dt} \|V\|^2 dt = K(\bar{t}) - K(0).$$

Similarly,  $\dot{U}$  may be chosen as a test function in the equations for  $V$  to get

$$\int_0^{\bar{t}} (\dot{V}, \dot{U}) dt = \int_0^{\bar{t}} -\nabla P(U) \dot{U} dt = - \int_0^{\bar{t}} \frac{d}{dt} P(U) dt = -(P(\bar{t}) - P(0)),$$

and thus  $K(\bar{t}) + P(\bar{t}) = K(0) + P(0)$ .  $\square$

*Remark 5.1.* Energy conservation requires exact integration of the right-hand side  $f$  (or at least that  $\int_0^t (\dot{U}, \dot{V}) dt + (P(t) - P(0)) = 0$ ) but can also be obtained in the case of quadrature; see [23].

**5.3. Monotonicity.** We shall prove that the mdG( $q$ ) method is  $B$ -stable (see [3]).

**THEOREM 5.2.** *Let  $U$  and  $V$  be the mdG( $q$ ) solutions of (1.1) with initial data  $U(0^-)$  and  $V(0^-)$ , respectively, defined by (4.7) on the same partition. If the right-hand side  $f$  is monotone, i.e.,*

$$(5.6) \quad (f(u, \cdot) - f(v, \cdot), u - v) \leq 0 \quad \forall u, v \in \mathbb{R}^N,$$

then, at every synchronized time-level  $\bar{t}$ , such as, e.g.,  $T$ , we have

$$(5.7) \quad \|U(\bar{t}^-) - V(\bar{t}^-)\| \leq \|U(0^-) - V(0^-)\|.$$

*Proof.* Choosing the test function as  $v = W = U - V$  in (4.7) for  $U$  and  $V$ , summing over the local intervals, and subtracting the two equations, we have

$$\sum_{ij} \left[ [W_i]_{i,j-1} W_{i,j-1}^+ + \int_{I_{ij}} \dot{W}_i W_i dt \right] = \int_0^T (f(U, \cdot) - f(V, \cdot), U - V) dt \leq 0.$$

Noting that

$$\begin{aligned} [W_i]_{i,j-1} W_{i,j-1}^+ + \int_{I_{ij}} \dot{W}_i W_i dt &= \frac{1}{2} (W_{i,j-1}^+)^2 + \frac{1}{2} (W_{ij}^-)^2 - W_{i,j-1}^- W_{i,j-1}^+ \\ &= \frac{1}{2} [W_i]_{i,j-1}^2 + \frac{1}{2} ((W_{ij}^-)^2 - (W_{i,j-1}^-)^2), \end{aligned}$$

we get

$$-\frac{1}{2} \|W(0^-)\|^2 + \frac{1}{2} \|W(T^-)\|^2 \leq \sum_{ij} [W_i]_{i,j-1} W_{i,j-1}^+ + \int_{I_{ij}} \dot{W}_i W_i dt \leq 0,$$

so that

$$\|W(T^-)\| \leq \|W(0^-)\|.$$

The proof is completed noting that the same analysis applies with  $T$  replaced by any other synchronized time-level  $\bar{t}$ .  $\square$

*Remark 5.2.* The proof extends to the fully discrete scheme, using the positivity of the quadrature weights.

**6. A posteriori error analysis.** In this section we prove a posteriori error estimates for the multi-adaptive Galerkin methods, including quadrature and discrete solution errors. Following the procedure outlined in the introduction, we first define the dual linearized problem and then derive a representation formula for the error in terms of the dual and the residual.

**6.1. The dual problem.** The dual problem comes in two different forms: a continuous and a discrete. For the a posteriori error analysis of this section, we will make use of the continuous dual. The discrete dual problem is used to prove a priori error estimates.

To set up the continuous dual problem, we define, for given functions  $v_1(t)$  and  $v_2(t)$ ,

$$(6.1) \quad J^*(v_1(t), v_2(t), t) = \left( \int_0^1 \frac{\partial f}{\partial u}(sv_1(t) + (1-s)v_2(t), t) ds \right)^*,$$

where  $*$  denotes the transpose, and we note that

$$(6.2) \quad \begin{aligned} J(v_1, v_2, \cdot)(v_1 - v_2) &= \int_0^1 \frac{\partial f}{\partial u}(sv_1 + (1-s)v_2, \cdot) ds (v_1 - v_2) \\ &= \int_0^1 \frac{\partial f}{\partial s}(sv_1 + (1-s)v_2, \cdot) ds = f(v_1, \cdot) - f(v_2, \cdot). \end{aligned}$$

The continuous dual problem is then defined as the following system of ODEs:

$$(6.3) \quad \begin{cases} -\dot{\varphi} &= J^*(u, U, \cdot)\varphi + g & \text{on } [0, T], \\ \varphi(T) &= \varphi_T, \end{cases}$$

with data  $\varphi_T$  and right-hand side  $g$ . Choosing the data and right-hand side appropriately, we obtain error estimates for different quantities of the computed solution. We shall assume below that the dual solution has  $q$  continuous derivatives ( $\varphi_i^{(q_{ij})} \in C(I_{ij})$  locally on interval  $I_{ij}$ ) for the continuous method and  $q + 1$  continuous derivatives ( $\varphi_i^{(q_{ij}+1)} \in C(I_{ij})$  locally on interval  $I_{ij}$ ) for the discontinuous method.



**6.2. Error representation.** The basis for the error analysis is the following error representation, expressing the error of an approximate solution  $U(t)$  in terms of the residual  $R(U, t)$  via the dual solution  $\varphi(t)$ . We stress that the result of the theorem is valid for any piecewise polynomial approximation of the solution to the initial value problem (1.1) and thus in particular the mcG( $q$ ) and mdG( $q$ ) approximations.

**THEOREM 6.1.** *Let  $U$  be a piecewise polynomial approximation of the exact solution  $u$  of (1.1), and let  $\varphi$  be the solution to (6.3) with right-hand side  $g(t)$  and initial data  $\varphi_T$ , and define the residual of the approximate solution  $U$  as  $R(U, t) = \dot{U}(t) - f(U(t), t)$ , defined on the open intervals of the partitions  $\cup_j I_{ij}$  as*

$$R_i(U, t) = \dot{U}_i(t) - f_i(U(t), t), \quad t \in (k_{i,j-1}, k_{ij}),$$

$j = 1, \dots, M_i, i = 1, \dots, N$ . Assume also that  $U$  is right-continuous at  $T$ . Then the error  $e = U - u$  satisfies

$$(6.4) \quad L_{\varphi_T, g}(e) \equiv (e(T), \varphi_T) + \int_0^T (e, g) dt = \sum_{i=1}^N \sum_{j=1}^{M_i} \left[ \int_{I_{ij}} R_i(U, \cdot) \varphi_i dt + [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) \right].$$

*Proof.* By the definition of the dual problem, we have using (6.2)

$$\begin{aligned} \int_0^T (e, g) dt &= \int_0^T (e, -\dot{\varphi} - J^*(u, U, \cdot)\varphi) dt \\ &= \sum_{ij} \int_{I_{ij}} -e_i \dot{\varphi}_i dt + \int_0^T (-J(u, U, \cdot)e, \varphi) dt \\ &= \sum_{ij} \int_{I_{ij}} -e_i \dot{\varphi}_i dt + \int_0^T (f(u, \cdot) - f(U, \cdot), \varphi) dt \\ &= \sum_{ij} \int_{I_{ij}} -e_i \dot{\varphi}_i dt + \sum_{ij} \int_{I_{ij}} (f_i(u, \cdot) - f_i(U, \cdot)) \varphi_i dt. \end{aligned}$$

Integrating by parts, we get

$$\int_{I_{ij}} -e_i \dot{\varphi}_i dt = e_i(t_{i,j-1}^+) \varphi(t_{i,j-1}) - e_i(t_{ij}^-) \varphi(t_{ij}) + \int_{I_{ij}} \dot{e}_i \varphi_i dt,$$

so that

$$\begin{aligned} \sum_{ij} \int_{I_{ij}} -e_i \dot{\varphi}_i dt &= \sum_{ij} [e_i]_{i,j-1} \varphi_i(t_{i,j-1}) - (e(T^-), \varphi_T) + \int_0^T (\dot{e}, \varphi) dt \\ &= \sum_{ij} [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) - (e(T), \varphi_T) + \int_0^T (\dot{e}, \varphi) dt. \end{aligned}$$

Thus, with  $L_{\varphi_T, g}(e) = (e(T), \varphi_T) + \int_0^T (e, g) dt$ , we have

$$\begin{aligned} L_{\varphi_T, g}(e) &= \sum_{ij} \left[ \int_{I_{ij}} (\dot{e}_i + f_i(u, \cdot) - f_i(U, \cdot)) \varphi_i dt + [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) \right] \\ &= \sum_{ij} \left[ \int_{I_{ij}} (\dot{U}_i - f_i(U, \cdot)) \varphi_i dt + [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) \right] \\ &= \sum_{ij} \left[ \int_{I_{ij}} R_i(U, \cdot) \varphi_i dt + [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) \right], \end{aligned}$$

which completes the proof.  $\square$

We now apply this theorem to represent the error in various norms. As before, we let  $\|\cdot\|$  denote the Euclidean norm on  $\mathbb{R}^N$  and define  $\|v\|_{L^1([0, T], \mathbb{R}^n)} = \int_0^T \|v\| dt$ .

**COROLLARY 6.2.** *If  $\varphi_T = e(T)/\|e(T)\|$  and  $g = 0$ , then*

$$(6.5) \quad \|e(T)\| = \sum_{i=1}^N \sum_{j=1}^{M_i} \left[ \int_{I_{ij}} R_i(U, \cdot) \varphi_i dt + [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) \right].$$

COROLLARY 6.3. *If  $\varphi_T = 0$  and  $g(t) = e(t)/\|e(t)\|$ , then*

$$(6.6) \quad \|e\|_{L^1([0,T],\mathbb{R}^N)} = \sum_{i=1}^N \sum_{j=1}^{M_i} \left[ \int_{I_{ij}} R_i(U, \cdot) \varphi_i \, dt + [U_i]_{i,j-1} \varphi_i(t_{i,j-1}) \right].$$

**6.3. Galerkin errors.** To obtain expressions for the Galerkin errors, i.e., the errors of the mcG( $q$ ) or mdG( $q$ ) approximations, assuming exact quadrature and exact solution of the discrete equations, we use two ingredients: the error representation of Theorem 6.1 and the Galerkin orthogonalities, (4.4) and (4.9). We first prove the following interpolation estimate.

LEMMA 6.4. *If  $f \in C^{q+1}([a, b])$ , then there is a constant  $C_q$ , depending only on  $q$ , such that*

$$(6.7) \quad |f(x) - \pi^{[q]} f(x)| \leq C_q k^{q+1} \frac{1}{k} \int_a^b |f^{(q+1)}(y)| \, dy \quad \forall x \in [a, b],$$

where  $\pi^{[q]} f(x)$  is the  $q$ th-order Taylor expansion of  $f$  around  $x_0 = (a+b)/2$ ,  $k = b-a$ , and  $C_q = 1/(2^q q!)$ .

*Proof.* Using Taylor’s formula with the remainder in integral form, we have

$$\begin{aligned} |f(x) - \pi^{[q]} f(x)| &= \left| \frac{1}{q!} \int_{x_0}^x f^{(q+1)}(y) (y - x_0)^{(q)} \, dy \right| \\ &\leq \frac{1}{2^q q!} k^{q+1} \frac{1}{k} \int_a^b |f^{(q+1)}(y)| \, dy. \quad \square \end{aligned}$$

Note that since we allow the polynomial degree to change between different components and between different intervals, the interpolation constant will change in the same way. We thus have  $C_{q_i} = C_{q_i}(t) = C_{q_{ij}}$  for  $t \in I_{ij}$ .

We can now prove a posteriori error estimates for the mcG( $q$ ) and mdG( $q$ ) methods. The estimates come in a number of different versions. We typically use  $E_2$  or  $E_3$  to adaptively determine the time-steps and  $E_0$  or  $E_1$  to evaluate the error. The quantities  $E_4$  and  $E_5$  may be used for qualitative estimates of error growth. We emphasize that all of the estimates derived in Theorems 6.5 and 6.6 below may be of use in an actual implementation, ranging from the very sharp estimate  $E_0$  containing only local quantities to the more robust estimate  $E_5$  containing only global quantities.

THEOREM 6.5. *The mcG( $q$ ) method satisfies the following estimates:*

$$(6.8) \quad |L_{\varphi_T, g}(e)| = E_0 \leq E_1 \leq E_2 \leq E_3 \leq E_4$$

and

$$(6.9) \quad |L_{\varphi_T, g}(e)| \leq E_2 \leq E_5,$$

where

$$(6.10) \quad \begin{aligned} E_0 &= \left| \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) \, dt \right|, \\ E_1 &= \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} |R_i(U, \cdot)| |\varphi_i - \pi_k \varphi_i| \, dt, \\ E_2 &= \sum_{i=1}^N \sum_{j=1}^{M_i} C_{q_{ij}-1} k_{ij}^{q_{ij}+1} r_{ij} s_{ij}^{[q_{ij}]}, \\ E_3 &= \sum_{i=1}^N S_i^{[q_i]} \max_{[0,T]} \{C_{q_i-1} k_i^{q_i} r_i\}, \\ E_4 &= S^{[q],1} \sqrt{N} \max_{i,[0,T]} \{C_{q_i-1} k_i^{q_i} r_i\}, \\ E_5 &= S^{[q],2} \|C_{q-1} k^q R(U, \cdot)\|_{L^2(\mathbb{R}^N \times [0,T])}, \end{aligned}$$

with  $C_q$  as in Lemma 6.4,  $k_i(t) = k_{ij}$ ,  $r_i(t) = r_{ij}$ , and  $s_i^{[qi]}(t) = s_{ij}^{[qij]}$  for  $t \in I_{ij}$ ,

$$(6.11) \quad \begin{aligned} r_{ij} &= \frac{1}{k_{ij}} \int_{I_{ij}} |R_i(U, \cdot)| \, dt, & s_{ij}^{[qij]} &= \frac{1}{k_{ij}} \int_{I_{ij}} |\varphi^{(qij)}| \, dt, \\ S_i^{[qi]} &= \int_0^T |\varphi_i^{(qi)}| \, dt, & S^{[q],1} &= \int_0^T \|\varphi^{(q)}\| \, dt, \\ S^{[q],2} &= \left( \int_0^T \|\varphi^{(q)}\|^2 \, dt \right)^{1/2}, \end{aligned}$$

and where  $\pi_k \varphi$  is any test space approximation of the dual solution  $\varphi$ . Expressions such as  $C_{q-1} k^q R$  are defined componentwise, i.e.,  $(C_{q-1} k^q R(U, \cdot))_i = C_{q_{ij}-1} k_{ij}^{q_{ij}} R_i(U, \cdot)$  for  $t \in I_{ij}$ .

*Proof.* Using the error representation of Theorem 6.1 and the Galerkin orthogonality (4.4), noting that the jump terms disappear since  $U$  is continuous, we have

$$|L_{\varphi_T, g}(e)| = \left| \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) \, dt \right| = E_0,$$

where  $\pi_k \varphi$  is any test space approximation of  $\varphi$ . By the triangle inequality, we have

$$E_0 \leq \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} |R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i)| \, dt = E_1.$$

Choosing  $\pi_k \varphi_i$  as in Lemma 6.4 on every interval  $I_{ij}$ , we have

$$\begin{aligned} E_1 &\leq \sum_{ij} C_{q_{ij}-1} k_{ij}^{q_{ij}} \int_{I_{ij}} |R_i(U, \cdot)| \, dt \frac{1}{k_{ij}} \int_{I_{ij}} |\varphi_i^{(qij)}| \, dt \\ &= \sum_{ij} C_{q_{ij}-1} k_{ij}^{q_{ij}+1} r_{ij} s_{ij}^{[qij]} = E_2. \end{aligned}$$

Continuing, we have

$$\begin{aligned} E_2 &\leq \sum_{i=1}^N \max_{[0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} \sum_{j=1}^{M_i} k_{ij} s_{ij}^{[qij]} \\ &= \sum_{i=1}^N \max_{[0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} \sum_{j=1}^{M_i} \int_{I_{ij}} |\varphi_i^{(qij)}| \, dt \\ &= \sum_{i=1}^N \max_{[0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} \int_0^T |\varphi_i^{(qi)}| \, dt \\ &= \sum_{i=1}^N S_i^{[qi]} \max_{[0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} = E_3, \end{aligned}$$

and, finally,

$$\begin{aligned} E_3 &\leq \max_{i, [0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} \sum_{i=1}^N \int_0^T |\varphi_i^{(qi)}| \, dt \\ &\leq \max_{i, [0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} \sqrt{N} \int_0^T \|\varphi^{(q)}\| \, dt \\ &= \max_{i, [0,T]} \{C_{q_i-1} k_i^{q_i} r_i\} \sqrt{N} S^{[q],1} = E_4. \end{aligned}$$

As an alternative we can use Cauchy's inequality in a different way. Continuing from  $E_2$ , we have

$$\begin{aligned} E_2 &= \sum_{i=1}^N \sum_{j=1}^{M_i} C_{q_{ij}-1} k_{ij}^{q_{ij}+1} r_{ij} s_{ij}^{[qij]} \\ &= \sum_{i=1}^N \sum_{j=1}^{M_i} C_{q_{ij}-1} k_{ij}^{q_{ij}} s_{ij}^{[qij]} \int_{I_{ij}} |R_i(U, \cdot)| \, dt \\ &= \sum_{i=1}^N \int_0^T C_{q_i-1} k_i^{q_i} |R_i(U, \cdot)| s_i^{[qi]} \, dt \\ &= \int_0^T (C_{q-1} k^q |R(U, \cdot)|, s^{[q]}) \, dt \\ &\leq \int_0^T \|C_{q-1} k^q R(U, \cdot)\| \|s^{[q]}\| \, dt \\ &\leq \left( \int_0^T \|C_{q-1} k^q R(U, \cdot)\|^2 \, dt \right)^{1/2} \left( \int_0^T \|s^{[q]}\|^2 \, dt \right)^{1/2}, \end{aligned}$$

where  $|R(U, \cdot)|$  denotes the vector-valued function with components  $|R|_i = |R_i| = |R_i(U, \cdot)|$ . Noting now that  $s$  is the  $L^2$ -projection of  $|\varphi^{(q)}|$  onto the piecewise constants on the partition, we have

$$\left( \int_0^T \|s^{[q]}\|^2 dt \right)^{1/2} \leq \left( \int_0^T \|\varphi^{(q)}\|^2 dt \right)^{1/2},$$

so that

$$|L_{\varphi_T, g}(e)| \leq \|C_{q-1} k^q R(U, \cdot)\|_{L^2(\mathbb{R}^N \times [0, T])} \|\varphi^{[q]}\|_{L^2(\mathbb{R}^N \times [0, T])} = E_5,$$

completing the proof.  $\square$

The proof of the estimates for the mdG( $q$ ) method is obtained similarly. Since in the discontinuous method the test functions are on every interval of one degree higher order than the test functions in the continuous method, we can choose a better interpolant. Thus, in view of Lemma 6.4, we obtain an extra factor  $k_{ij}$  in the error estimates.

**THEOREM 6.6.** *The mdG( $q$ ) method satisfies the following estimates:*

$$(6.12) \quad |L_{\varphi_T, g}(e)| = E_0 \leq E_1 \leq E_2 \leq E_3 \leq E_4$$

and

$$(6.13) \quad |L_{\varphi_T, g}(e)| \leq E_2 \leq E_5,$$

where

$$(6.14) \quad \begin{aligned} E_0 &= \left| \sum_{ij} \int_{I_{ij}} R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) dt + [U_i]_{i,j-1} (\varphi_i(t_{i,j-1}) - \pi_k \varphi_i(t_{i,j-1}^+)) \right|, \\ E_1 &= \sum_{ij} \int_{I_{ij}} |R_i(U, \cdot)| |\varphi_i - \pi_k \varphi_i| dt + |[U_i]_{i,j-1}| |\varphi_i(t_{i,j-1}) - \pi_k \varphi_i(t_{i,j-1}^+)|, \\ E_2 &= \sum_{i=1}^N \sum_{j=1}^{M_i} C_{q_{ij}} k_{ij}^{q_{ij}+2} \bar{r}_{ij} s_{ij}^{[q_{ij}+1]}, \\ E_3 &= \sum_{i=1}^N S_i^{[q_i+1]} \max_{[0, T]} \left\{ C_{q_i} k_i^{q_i+1} \bar{r}_i \right\}, \\ E_4 &= S^{[q+1], 1} \sqrt{N} \max_{i, [0, T]} \left\{ C_{q_i} k_i^{q_i+1} \bar{r}_i \right\}, \\ E_5 &= S^{[q+1], 2} \|C_q k^{q+1} \bar{R}(U, \cdot)\|_{L^2(\mathbb{R}^N \times [0, T])}, \end{aligned}$$

with

$$(6.15) \quad \bar{r}_{ij} = \frac{1}{k_{ij}} \int_{I_{ij}} |R_i(U, \cdot)| dt + \frac{1}{k_{ij}} |[U_i]_{i,j-1}|, \quad \bar{R}_i(U, \cdot) = |R_i(U, \cdot)| + \frac{1}{k_{ij}} |[U_i]_{i,j-1}|,$$

and we otherwise use the notation of Theorem 6.5.

*Proof.* As in the proof for the continuous method, we use the error representation of Theorem 6.1 and the Galerkin orthogonality (4.9) to get

$$|L_{\varphi_T, g}(e)| = \left| \sum_{ij} \int_{I_{ij}} R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) dt + [U_i]_{i,j-1} (\varphi_i(t_{i,j-1}) - \pi_k \varphi_i(t_{i,j-1}^+)) \right| = E_0.$$

By Lemma 6.4 we obtain

$$\begin{aligned} E_0 &\leq \sum_{ij} \int_{I_{ij}} |R_i(U, \cdot)| |\varphi_i - \pi_k \varphi_i| dt + |[U_i]_{i,j-1}| |\varphi_i(t_{i,j-1}) - \pi_k \varphi_i(t_{i,j-1}^+)| = E_1 \\ &\leq \sum_{ij} C_{q_{ij}} k_{ij}^{q_{ij}+1} \left( \int_{I_{ij}} |R_i(U, \cdot)| dt + |[U_i]_{i,j-1}| \right) \frac{1}{k_{ij}} \int_{I_{ij}} |\varphi_i^{(q_{ij}+1)}| dt \\ &\leq \sum_{ij} C_{q_{ij}} k_{ij}^{q_{ij}+2} \bar{r}_{ij} s_{ij}^{[q_{ij}+1]} = E_2. \end{aligned}$$

Continuing now in the same way as for the continuous method, we have  $E_2 \leq E_3 \leq E_4$  and  $E_2 \leq E_5$ .  $\square$

*Remark 6.1.* When evaluating the expressions  $E_0$  or  $E_1$ , the interpolant  $\pi_k \varphi$  does not have to be chosen as in Lemma 6.4. This is only a convenient way to obtain the interpolation constant. In section 6.6 below we discuss a more convenient choice of interpolant.

*Remark 6.2.* If we replace  $\frac{1}{k_{ij}} \int_{I_{ij}} |R_i| dt$  by  $\max_{I_{ij}} |R_i|$ , we may replace  $C_q$  by a smaller constant  $C'_q$ . The value of the constant thus depends on the specific way the residual is measured.

**6.4. Computational errors.** The error estimates of Theorems 6.5 and 6.6 are based on the Galerkin orthogonalities (4.4) and (4.9). If the corresponding discrete equations are not solved exactly, there will be an additional contribution to the total error. Although Theorem 6.1 is still valid, the first steps in Theorems 6.5 and 6.6 are not. Focusing on the continuous method, the first step in the proof of Theorem 6.5 is the subtraction of a test space interpolant. This is possible, since by the Galerkin orthogonality we have

$$\sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} R_i(U, \cdot) \pi_k \varphi_i dt = 0$$

for all test space interpolants  $\pi_k \varphi$ . If the residual is no longer orthogonal to the test space, we add and subtract this term to get to the point where the implications of Theorem 6.5 are valid for one of the terms. Assuming now that  $\varphi$  varies slowly on each subinterval, we estimate the remaining extra term as follows:

$$\begin{aligned} (6.16) \quad E_C &= \left| \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} R_i(U, \cdot) \pi_k \varphi_i dt \right| \leq \sum_{i=1}^N \sum_{j=1}^{M_i} \left| \int_{I_{ij}} R_i(U, \cdot) \pi_k \varphi_i dt \right| \\ &\approx \sum_{i=1}^N \sum_{j=1}^{M_i} k_{ij} |\bar{\varphi}_{ij}| \frac{1}{k_{ij}} \left| \int_{I_{ij}} R_i(U, \cdot) dt \right| = \sum_{i=1}^N \sum_{j=1}^{M_i} k_{ij} |\bar{\varphi}_{ij}| |\mathcal{R}_{ij}^C| \\ &\leq \sum_{i=1}^N \bar{S}_i^{[0]} \max_j |\mathcal{R}_{ij}^C|, \end{aligned}$$

where  $\bar{\varphi}$  is a piecewise constant approximation of  $\varphi$  (using, say, the mean values on the local intervals),

$$(6.17) \quad \bar{S}_i^{[0]} = \sum_{j=1}^{M_i} k_{ij} |\bar{\varphi}_{ij}| \approx \int_0^T |\varphi_i| dt = S_i^{[0]}$$

is a stability factor, and we define the *discrete* or *computational* residual as

$$(6.18) \quad \mathcal{R}_{ij}^C = \frac{1}{k_{ij}} \int_{I_{ij}} R_i(U, \cdot) dt = \frac{1}{k_{ij}} \left( (\xi_{ijq} - \xi_{ij0}) - \int_{I_{ij}} f_i(U, \cdot) dt \right).$$

More precise estimates may be used if needed.

For the mdG( $q$ ) method, the situation is similar with the computational residual now defined as

$$(6.19) \quad \mathcal{R}_{ij}^C = \frac{1}{k_{ij}} \left( (\xi_{ijq} - \xi_{ij0}^-) - \int_{I_{ij}} f_i(U, \cdot) dt \right).$$

Thus, to estimate the computational error, we evaluate the computational residuals and multiply with the computed stability factors.

**6.5. Quadrature errors.** We now extend our analysis to take into account also quadrature errors. We denote integrals evaluated by quadrature with  $\tilde{\int}$ . Starting from the error representation as before, we have for the mcG( $q$ ) method

$$(6.20) \quad \begin{aligned} L_{\varphi_T, g}(e) &= \int_0^T (R, \varphi) dt \\ &= \int_0^T (R, \varphi - \pi_k \varphi) dt + \int_0^T (R, \pi_k \varphi) dt \\ &= \int_0^T (R, \varphi - \pi_k \varphi) dt + \tilde{\int}_0^T (R, \pi_k \varphi) dt + \left[ \int_0^T (R, \pi_k \varphi) dt - \tilde{\int}_0^T (R, \pi_k \varphi) dt \right] \\ &= \int_0^T (R, \varphi - \pi_k \varphi) dt + \tilde{\int}_0^T (R, \pi_k \varphi) dt + \left( \tilde{\int}_0^T - \int_0^T \right) (f(U, \cdot), \pi_k \varphi) dt \end{aligned}$$

if the quadrature is exact for  $\dot{U}v$  when  $v$  is a test function. The first term of this expression was estimated in Theorem 6.5 and the second term is the computational error discussed previously (where  $\tilde{\int}$  denotes that in a real implementation, (6.18) is evaluated using quadrature). The third term is the quadrature error, which may be nonzero even if  $f$  is linear, if the time-steps are different for different components. To estimate the quadrature error, notice that

$$(6.21) \quad \begin{aligned} \left( \tilde{\int}_0^T - \int_0^T \right) (f(U, \cdot), \pi_k \varphi) dt &= \sum_{ij} \left( \tilde{\int}_{I_{ij}} - \int_{I_{ij}} \right) f_i(U, \cdot) \pi_k \varphi_i dt \\ &\approx \sum_{ij} k_{ij} \tilde{\varphi}_{ij} \mathcal{R}_{ij}^Q \leq \sum_{i=1}^N \tilde{S}_i^{[0]} \max_j |\mathcal{R}_{ij}^Q|, \end{aligned}$$

where  $\{\tilde{S}_i^{[0]}\}_{i=1}^N$  are the same stability factors as in the estimate for the computational error and

$$(6.22) \quad \mathcal{R}_{ij}^Q = \frac{1}{k_{ij}} \left( \tilde{\int}_{I_{ij}} f_i(U, \cdot) dt - \int_{I_{ij}} f_i(U, \cdot) dt \right)$$

is the *quadrature residual*. A similar estimate holds for the mdG( $q$ ) method.

We now make a few comments on how to estimate the quadrature residual. The Lobatto quadrature of the mcG( $q$ ) method is exact for polynomials of degree less than or equal to  $2q - 1$ , and we have an order  $2q$  estimate for  $\tilde{\int} - \int$  in terms of  $f^{(2q)}$ , and so we make the assumption  $\mathcal{R}_{ij}^Q \propto k_{ij}^{2q_{ij}}$ . If, instead of using the standard quadrature rule over the interval with quadrature residual  $\mathcal{R}_{ij}^Q$ , we divide the interval into  $2^m$  parts and use the quadrature on every interval, summing up the result, we will get a different quadrature residual, namely

$$(6.23) \quad \mathcal{R}^{\mathcal{Q}^m} = \frac{1}{k} C 2^m (k/2^m)^{2q+1} = 2^{m(-2q)} C k^{2q} = 2^{-2q} \mathcal{R}^{\mathcal{Q}^{m-1}},$$

where we have dropped the  $ij$  subindices. Thus, since  $|\mathcal{R}^{\mathcal{Q}_m}| \leq |\mathcal{R}^{\mathcal{Q}_m} - \mathcal{R}^{\mathcal{Q}_{m+1}}| + |\mathcal{R}^{\mathcal{Q}_{m+1}}| = |\mathcal{R}^{\mathcal{Q}_m} - \mathcal{R}^{\mathcal{Q}_{m+1}}| + 2^{-2q}|\mathcal{R}^{\mathcal{Q}_m}|$ , we have the estimate

$$(6.24) \quad |\mathcal{R}^{\mathcal{Q}_m}| \leq \frac{1}{1 - 2^{-2q}} |\mathcal{R}^{\mathcal{Q}_m} - \mathcal{R}^{\mathcal{Q}_{m+1}}|.$$

Thus, by computing the integrals at two or more dyadic levels, we may estimate quadrature residuals and thus the quadrature error.

For the mdG( $q$ ) method the only difference is that the basic quadrature rule is one order better, i.e., instead of  $2q$  we have  $2q + 1$ , so that

$$(6.25) \quad |\mathcal{R}^{\mathcal{Q}_m}| \leq \frac{1}{1 - 2^{-1-2q}} |\mathcal{R}^{\mathcal{Q}_m} - \mathcal{R}^{\mathcal{Q}_{m+1}}|.$$

**6.6. Evaluating  $E_G$ .** We now present an approach to estimating the quantity  $\int_0^T (R(U, \cdot), \varphi - \pi_k \varphi) dt$  by direct evaluation, with  $\varphi$  a computed dual solution and  $\pi_k \varphi$  a suitably chosen interpolant. In this way we avoid introducing interpolation constants and computing derivatives of the dual. Note, however, that although we do not explicitly compute any derivatives of the dual, the regularity assumed in section 6.1 for the dual solution is still implicitly required for the computed quantities to make sense. Starting now with

$$(6.26) \quad E_G = \left| \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) dt \right|$$

for the continuous method, we realize that the best possible choice of interpolant, if we want to prevent cancellation, is to choose  $\pi_k \varphi$  such that  $R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) \geq 0$  (or  $\leq 0$ ) on every local interval  $I_{ij}$ . With such a choice of interpolant, we would have

$$(6.27) \quad E_G = \left| \sum_{i=1}^N \sum_{j=1}^{M_i} \int_{I_{ij}} R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i) dt \right| = \sum_{i=1}^N \sum_{j=1}^{M_i} \alpha_{ij} \int_{I_{ij}} |R_i(U, \cdot) (\varphi_i - \pi_k \varphi_i)| dt$$

with  $\alpha_{ij} = \pm 1$ . The following lemmas give us an idea of how to choose the interpolant.

**LEMMA 6.7.** *If, for  $i = 1, \dots, N$ ,  $f_i = f_i(U(t), t) = f_i(U_i(t), t)$  and  $f_i$  is linear or, alternatively,  $f = f(U(t), t)$  is linear and all components have the same time-steps and order, then every component  $R_i(U, \cdot)$  of the mcG( $q$ ) residual is a Legendre polynomial of order  $q_{ij}$  on  $I_{ij}$ , for  $j = 1, \dots, M_i$ .*

*Proof.* On every interval  $I_{ij}$  the residual component  $R_i(U, \cdot)$  is orthogonal to  $\mathcal{P}^{q_{ij}-1}(I_{ij})$ . Since the conditions assumed in the statement of the lemma guarantee that the residual is a polynomial of degree  $q_{ij}$  on every interval  $I_{ij}$ , it is clear that on every such interval it is the  $q_{ij}$ th-order Legendre polynomial (or a multiple thereof).  $\square$

Even if the rather strict conditions of this lemma do not hold, we can say something similar. The following lemma restates this property in terms of approximations of the residual.

**LEMMA 6.8.** *Let  $\tilde{R}$  be the local  $L^2$ -projection of the mcG( $q$ ) residual  $R$  onto the trial space, i.e.,  $\tilde{R}_i(U, \cdot)|_{I_{ij}}$  is the  $L^2(I_{ij})$ -projection onto  $\mathcal{P}^{q_{ij}}(I_{ij})$  of  $R_i(U, \cdot)|_{I_{ij}}$ ,  $j = 1, \dots, M_i$ ,  $i = 1, \dots, N$ . Then every  $\tilde{R}_i(U, \cdot)|_{I_{ij}}$  is a Legendre polynomial of degree  $q_{ij}$ .*

*Proof.* Since  $\tilde{R}_i(U, \cdot)$  is the  $L^2$ -projection of  $R_i(U, \cdot)$  onto  $\mathcal{P}^{q_{ij}}(I_{ij})$  on  $I_{ij}$ , we have

$$\int_{I_{ij}} \tilde{R}_i(U, \cdot)v \, dt = \int_{I_{ij}} R_i(U, \cdot)v \, dt = 0$$

for all  $v \in \mathcal{P}^{q_{ij}-1}(I_{ij})$ , so that  $\tilde{R}_i(U, \cdot)$  is the  $q_{ij}$ th-order Legendre polynomial on  $I_{ij}$ .  $\square$

To prove the corresponding results for the discontinuous method, we first note some basic properties of Radau polynomials.

LEMMA 6.9. *Let  $P_q$  be the  $q$ th-order Legendre polynomial on  $[-1, 1]$ . Then the  $q$ th-order Radau polynomial,  $Q_q(x) = (P_q(x) + P_{q+1}(x))/(x + 1)$ , has the following property:*

$$(6.28) \quad I = \int_{-1}^1 Q_q(x)(x + 1)^p \, dx = 0$$

for  $p = 1, \dots, q$ . Conversely, if  $f$  is a polynomial of degree  $q$  on  $[-1, 1]$  and has the property (6.28), i.e.,  $\int_{-1}^1 f(x)(x + 1)^p \, dx = 0$  for  $p = 1, \dots, q$ , then  $f$  is a Radau polynomial.

*Proof.* We can write the  $q$ th-order Legendre polynomial on  $[-1, 1]$  as  $P_q(x) = \frac{1}{q!2^q} D^q((x^2 - 1)^q)$ . Thus, integrating by parts, we have

$$\begin{aligned} I &= \int_{-1}^1 \frac{P_q(x)+P_{q+1}(x)}{x+1} (x + 1)^p \, dx \\ &= \frac{1}{q!2^q} \int_{-1}^1 D^q((x^2 - 1)^q + x(x^2 - 1)^q)(x + 1)^{p-1} \, dx \\ &= \frac{1}{q!2^q} \int_{-1}^1 D^q((x + 1)(x^2 - 1)^q)(x + 1)^{p-1} \, dx \\ &= \frac{1}{q!2^q} (-1)^p \int_{-1}^1 D^{q-p}((x + 1)(x^2 - 1)^q) D^p(x + 1)^{p-1} \, dx = 0, \end{aligned}$$

since  $D^l((x + 1)(x^2 - 1)^q)$  is zero at  $-1$  and  $1$  for  $l < q$ . Assume now that  $f$  is a polynomial of degree  $q$  on  $[-1, 1]$  with the property (6.28). Since  $\{(x + 1)^p\}_{p=1}^q$  are linearly independent on  $[-1, 1]$  and orthogonal to the Radau polynomial  $Q_q$ ,  $\{Q_q(x), (x + 1), (x + 1)^2, \dots, (x + 1)^q\}$  form a basis for  $\mathcal{P}^q([-1, 1])$ . If then  $f$  is orthogonal to the subspace spanned by  $\{(x + 1)^p\}_{p=1}^q$ , we must have  $f = cQ_q$  for some constant  $c$ , and the proof is complete.  $\square$

LEMMA 6.10. *If, for  $i = 1, \dots, N$ ,  $f_i = f_i(U(t), t) = f_i(U_i(t), t)$  and  $f_i$  is linear or, alternatively,  $f = f(U(t), t)$  is linear and all components have the same time-steps and order, then every component  $R_i(U, \cdot)$  of the mdG( $q$ ) residual is a Radau polynomial of order  $q_{ij}$  on  $I_{ij}$  for  $j = 1, \dots, M_i$ .*

*Proof.* Note first that by assumption the residual  $R_i(U, \cdot)$  is a polynomial of degree  $q_{ij}$  on  $I_{ij}$ . By the Galerkin orthogonality, we have

$$0 = \int_{I_{ij}} R_i(U, \cdot)v \, dt + [U]_{i,j-1}v(t_{i,j-1}^+) \quad \forall v \in \mathcal{P}^{q_{ij}}(I_{ij}),$$

which holds especially for  $v(t) = (t - t_{i,j-1})^p$  with  $p = 1, \dots, q$ , for which the jump terms disappear. Rescaling to  $[-1, 1]$ , it follows from Lemma 6.9 that the residual  $R_i(U, \cdot)$  must be a Radau polynomial on  $I_{ij}$ .  $\square$

Also for the discontinuous method there is a reformulation in terms of approximations of the residual.

LEMMA 6.11. *Let  $\tilde{R}$  be the local  $L^2$ -projection of the mdG( $q$ ) residual  $R$  onto the trial space, i.e.,  $\tilde{R}_i(U, \cdot)|_{I_{ij}}$  is the  $L^2(I_{ij})$ -projection onto  $\mathcal{P}^{q_{ij}}(I_{ij})$  of  $R_i(U, \cdot)|_{I_{ij}}$ ,*



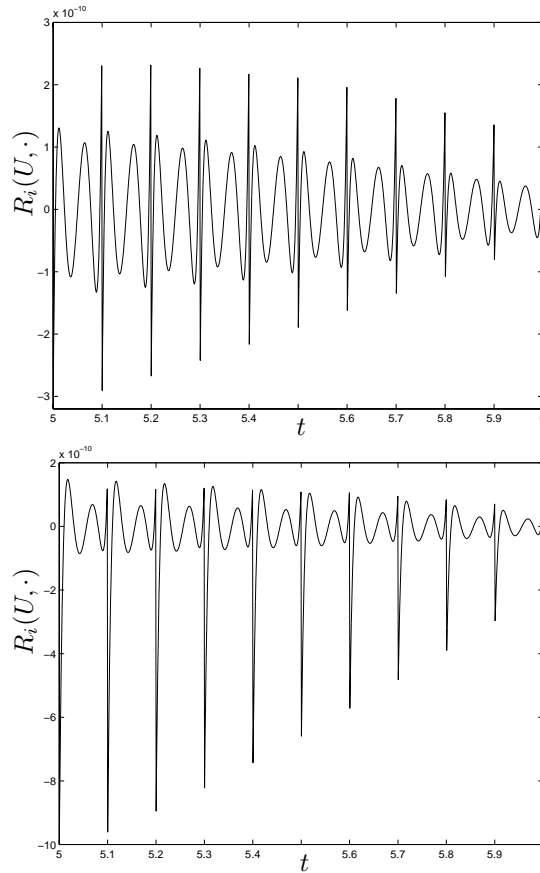


FIG. 6.1. The Legendre-polynomial residual of the mcG( $q$ ) method (left) and the Radau-polynomial residual of the mdG( $q$ ) method (right), for polynomials of degree five, i.e., methods of order 10 and 11, respectively.

$j = 1, \dots, M_i, i = 1, \dots, N$ . Then every  $\tilde{R}_i(U, \cdot)|_{I_{ij}}$  is a Radau polynomial of degree  $q_{ij}$ .

*Proof.* Since  $\tilde{R}_i(U, \cdot)$  is the  $L^2$ -projection of  $R_i(U, \cdot)$  onto  $\mathcal{P}^{q_{ij}}(I_{ij})$  on  $I_{ij}$ , it follows from the Galerkin orthogonality that

$$\int_{I_{ij}} \tilde{R}_i(U, \cdot)v \, dt = \int_{I_{ij}} R_i(U, \cdot)v \, dt = 0$$

for any  $v(t) = (t - t_{i,j-1})^p$  with  $1 \leq p \leq q$ . From Lemma 6.9 it then follows that  $\tilde{R}_i(U, \cdot)$  is a Radau polynomial on  $I_{ij}$ .  $\square$

We thus know that the mcG( $q$ ) residuals are (in the sense of Lemma 6.8) Legendre polynomials on the local intervals and that the mdG( $q$ ) residuals are (in the sense of Lemma 6.11) Radau polynomials. This is illustrated in Figure 6.1.

From this information about the residual, we now choose the interpolant. Assume that the polynomial order of the method on some interval is  $q$  for the continuous method. Then the dual should be interpolated by a polynomial of degree  $q - 1$ , i.e., we have freedom to interpolate at exactly  $q$  points. Since a  $q$ th-order Legendre polynomial has  $q$  zeros on the interval, we may choose to interpolate the dual exactly at those

points where the residual is zero. This means that if the dual can be approximated well enough by a polynomial of degree  $q$ , the product  $R_i(U, \cdot)(\varphi_i - \pi_k \varphi_i)$  does not change sign on the interval.

For the discontinuous method, we should interpolate the dual with a polynomial of degree  $q$ , i.e., we have freedom to interpolate at exactly  $q + 1$  points. To get rid of the jump terms that are present in the error representation for the discontinuous method, we want to interpolate the dual at the beginning of every interval. This leaves  $q$  degrees of freedom. We then choose to interpolate the dual at the  $q$  points within the interval where the Radau polynomial is zero.

As a result, we may choose the interpolant in such a way that we have

$$(6.29) \quad |L_{\varphi_T, g}(e)| = \left| \sum_{ij} \int_{I_{ij}} R_i(U, \cdot)(\varphi_i - \pi_k \varphi_i) dt \right| = \sum_{ij} \alpha_{ij} \int_{I_{ij}} |R_i(U, \cdot)(\varphi_i - \pi_k \varphi_i)| dt,$$

with  $\alpha_{ij} = \pm 1$ , for both the mcG( $q$ ) method and the mdG( $q$ ) method (but the interpolants are different). Notice that the jump terms for the discontinuous method have disappeared.

There is now a simple way to compute the integrals  $\int_{I_{ij}} R_i(U, \cdot)(\varphi_i - \pi_k \varphi_i) dt$ . Since the integrands are, in principle, products of two polynomials for which we know the positions of the zeros, the product is a polynomial with known properties. There are then constants  $C_q$  (which can be computed numerically), depending on the order and the method, such that

$$(6.30) \quad \int_{I_{ij}} |R_i(U, \cdot)(\varphi_i - \pi_k \varphi_i)| dt = C_{q_{ij}} k_{ij} |R_i(U, t_{ij}^-)| |\varphi_i(t_{ij}) - \pi_k \varphi_i(t_{ij}^-)|.$$

Finally, note that there are “computational” counterparts also for the estimates of type  $E_3$  in Theorems 6.5 and 6.6, namely

$$(6.31) \quad \begin{aligned} |L_{\varphi_T, g}(e)| &\leq \sum_{ij} \int_{I_{ij}} |R_i(U, \cdot)| |\varphi_i - \pi_k \varphi_i| dt \\ &= \sum_{ij} C'_{q_{ij}} k_{ij}^{q_{ij}} |R_i(U, t_{ij}^-)| \int_{I_{ij}} \frac{1}{k_{ij}^{q_{ij}}} |\varphi_i - \pi_k \varphi_i| dt \\ &\leq \sum_{i=1}^N \tilde{S}_i \max_{j=1, \dots, M_i} C'_{q_{ij}} k_{ij}^{q_{ij}} |R_i(U, t_{ij}^-)|, \end{aligned}$$

with  $\tilde{S}_i = \int_0^T \frac{1}{k_i^{q_i}} |\varphi_i - \pi_k \varphi_i| dt$  for the continuous method and similarly for the discontinuous method.

**6.7. The total error.** The total error is composed of three parts—the Galerkin error,  $E_G$ , the computational error,  $E_C$  and the quadrature error,  $E_Q$ :

$$(6.32) \quad |L_{\varphi_T, g}(e)| \leq E_G + E_C + E_Q.$$

As an example, choosing estimate  $E_3$  of Theorems 6.5 and 6.6 we have the following (approximate) error estimate for the mcG( $q$ ) method:

$$(6.33) \quad |L_{\varphi_T, g}(e)| \leq \sum_{i=1}^N \left[ S_i^{[q_i]} \max_{[0, T]} \{C_{q_i-1} k_i^{q_i} r_i\} + \bar{S}_i^{[0]} \max_{[0, T]} |\mathcal{R}_i^C| + \bar{S}_i^{[0]} \max_{[0, T]} |\mathcal{R}_i^Q| \right];$$

for the mdG( $q$ ) method we have

$$(6.34) \quad |L_{\varphi T, g}(e)| \leq \sum_{i=1}^N \left[ S_i^{\lfloor q_i+1 \rfloor} \max_{[0, T]} \left\{ C_{q_i} k_i^{q_i+1} \bar{r}_i \right\} + \bar{S}_i^{[0]} \max_{[0, T]} |\mathcal{R}_i^C| + \bar{S}_i^{[0]} \max_{[0, T]} |\mathcal{R}_i^Q| \right].$$

These estimates containing Galerkin errors, computational errors, and quadrature errors also include numerical round-off errors (included in the computational error). *Modelling errors* could also be similarly accounted for since these are closely related to quadrature errors, in that both errors can be seen as arising from integrating the wrong right-hand side.

The true global error may thus be estimated in terms of computable stability factors and residuals. We expect the estimate for the Galerkin error,  $E_G$ , to be quite sharp, while  $E_C$  and  $E_Q$  may be less sharp. Even sharper estimates are obtained using estimates  $E_0$ ,  $E_1$ , or  $E_2$  of Theorems 6.5 and 6.6.

**6.8. An a posteriori error estimate for the dual.** We conclude this section by proving a computable a posteriori error estimate for the dual problem. To compute the stability factors used in the error estimates presented above, we solve the dual problem numerically, and we thus face the problem of estimating the error in the stability factors.

To demonstrate how relative errors of stability factors can be estimated using the same technique as above, we compute the relative error for the stability factor  $S_\varphi(T)$ , defined as

$$(6.35) \quad S_\varphi(T) = \sup_{\|\varphi(T)\|=1} \int_0^T \|\varphi\| dt$$

for a computed approximation  $\Phi$  of the dual solution  $\varphi$ .

To estimate the relative error of the stability factor, we use the error representation of Theorem 6.1 to represent the  $L^1([0, T], \mathbb{R}^N)$ -error of  $\Phi$  in terms of the residual of  $\Phi$  and the dual of the dual,  $\omega$ . In [28] we prove the following lemma, from which the estimate follows.

LEMMA 6.12. *Let  $\varphi$  be the dual solution with stability factor  $S_\varphi(t)$ , i.e., with data  $\|\varphi(t)\| = 1$  specified at time  $t$ , and let  $\omega$  be the dual of the dual. We then have the following estimate:*

$$(6.36) \quad \|\omega(t)\| \leq S_\varphi(T - t) \quad \forall t \in [0, T].$$

THEOREM 6.13. *Let  $\Phi$  be a continuous approximation of the dual solution with residual  $R_\Phi$ , and assume that  $S_\varphi(t)/S_\varphi(T)$  is bounded by  $C$  on  $[0, T]$ . Then the following estimate holds for the relative error of the stability factor  $S_\Phi(T)$ :*

$$(6.37) \quad |S_\Phi(T) - S_\varphi(T)|/S_\varphi(T) \leq C \int_0^T \|R_\Phi\| dt,$$

and for many problems we may take  $C = 1$ .

*Proof.* By Corollary 6.3, we have an expression for the  $L^1([0, T], \mathbb{R}^N)$ -error of the

dual, so that

$$\begin{aligned}
 |S_\Phi(T) - S_\varphi(T)| &= \left| \int_0^T \|\Phi\| dt - \int_0^T \|\varphi\| dt \right| \\
 (6.38) \qquad &= \left| \int_0^T (\|\Phi\| - \|\varphi\|) dt \right| \leq \int_0^T \|\Phi - \varphi\| dt \\
 &= \|\Phi - \varphi\|_{L^1([0,T],\mathbb{R}^n)} = \int_0^T (R_\Phi, \omega(T - \cdot)) dt \\
 &\leq \int_0^T \|R_\Phi\| \|\omega(T - \cdot)\| dt.
 \end{aligned}$$

With  $C$  defined as above it now follows by Lemma 6.12 that

$$|S_\Phi(T) - S_\varphi(T)| \leq C \int_0^T \|R_\Phi\| dt S_\varphi(T),$$

and the proof is complete.  $\square$

*Remark 6.3.* We also have to take into account quadrature errors when evaluating (6.35). This can be done in many ways; see, e.g., [9].

**Appendix A. Derivation of the methods.**

This section contains some details left out of the discussion of section 4.

**A.1. The mcG( $q$ ) method.** To rewrite the local problem in a more explicit form, let  $\{s_n\}_{n=0}^q$  be a set of nodal points on  $[0, 1]$ , with  $s_0 = 0$  and  $s_q = 1$ . A good choice for the cG( $q$ ) method is the Lobatto points of  $[0, 1]$ . Now, let  $\tau_{ij}$  be the linear mapping from the interval  $I_{ij}$  to  $(0, 1]$ , defined by

$$(A.1) \qquad \tau_{ij}(t) = \frac{t - t_{i,j-1}}{t_{ij} - t_{i,j-1}},$$

and let  $\{\lambda_n^{[q]}\}_{n=0}^q$  be the  $\{s_n\}_{n=0}^q$  Lagrange basis functions for  $\mathcal{P}^q([0, 1])$  on  $[0, 1]$ , i.e.,

$$(A.2) \qquad \lambda_n^{[q]}(s) = \frac{(s - s_0) \cdots (s - s_{n-1})(s - s_{n+1}) \cdots (s - s_q)}{(s_n - s_0) \cdots (s_n - s_{n-1})(s_n - s_{n+1}) \cdots (s_n - s_q)}.$$

We can then express  $U_i$  on  $I_{ij}$  in the form

$$(A.3) \qquad U_i(t) = \sum_{n=0}^q \xi_{ijn} \lambda_n^{[q_{ij}]}(\tau_{ij}(t)),$$

and choosing the  $\lambda_n^{[q-1]}$  as test functions we can formulate the local problem (4.3) as follows: Find  $\{\xi_{ijn}\}_{n=0}^{q_{ij}}$ , with  $\xi_{ij0} = \xi_{i,j-1,q_{i,j-1}}$ , such that for  $m = 0, \dots, q_{ij} - 1$

$$(A.4) \qquad \int_{I_{ij}} \sum_{n=0}^{q_{ij}} \xi_{ijn} \frac{d}{dt} \left[ \lambda_n^{[q_{ij}]}(\tau_{ij}(t)) \right] \lambda_m^{[q_{ij}-1]}(\tau_{ij}(t)) dt = \int_{I_{ij}} f_i(U(t), t) \lambda_m^{[q_{ij}-1]}(\tau_{ij}(t)) dt.$$

To simplify the notation, we drop the  $ij$  subindices and assume that the time-interval is  $[0, k]$ , keeping in mind that, although not visible, all other components are present in  $f$ . We thus seek to determine the coefficients  $\{\xi_n\}_{n=1}^q$  with  $\xi_0$  given, such that for  $m = 1, \dots, q$  we have

$$(A.5) \qquad \sum_{n=0}^q \xi_n \frac{1}{k} \int_0^k \dot{\lambda}_n^{[q]}(\tau(t)) \lambda_{m-1}^{[q-1]}(\tau(t)) dt = \int_0^k f \lambda_{m-1}^{[q-1]}(\tau(t)) dt,$$

or simply

$$(A.6) \quad \sum_{n=1}^q a_{mn}^{[q]} \xi_n = b_m,$$

where

$$(A.7) \quad a_{mn}^{[q]} = \int_0^1 \dot{\lambda}_n^{[q]}(t) \lambda_{m-1}^{[q-1]}(t) dt$$

and

$$(A.8) \quad b_m = \int_0^k f \lambda_{m-1}^{[q-1]}(\tau(t)) dt - a_{m0} \xi_0.$$

We explicitly compute the inverse  $\bar{A}^{[q]} = (\bar{a}_{mn}^{[q]})$  of the matrix  $A^{[q]} = (a_{mn}^{[q]})$ . Thus, switching back to the full notation, we get

$$(A.9) \quad \xi_{ijm} = -\xi_0 \sum_{n=1}^q \bar{a}_{mn}^{[q]} a_{n0} + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) f_i(U(t), t) dt, \quad m = 1, \dots, q_{ij},$$

where the *weight functions*  $\{w_m^{[q]}\}_{m=1}^q$  are given by

$$(A.10) \quad w_m^{[q]} = \sum_{n=1}^q \bar{a}_{mn}^{[q]} \lambda_{n-1}^{[q-1]}, \quad m = 1, \dots, q.$$

Following Lemma A.1 below, this relation may be somewhat simplified.

LEMMA A.1. *For the mcG(q) method, we have*

$$\sum_{n=1}^q \bar{a}_{mn}^{[q]} a_{n0} = -1.$$

*Proof.* Assume the interval to be  $[0, 1]$ . The value is independent of  $f$  so we may take  $f = 0$ . We thus want to prove that if  $f = 0$ , then  $\xi_n = \xi_0$  for  $n = 1, \dots, q$ , i.e.,  $U = U_0$  on  $[0, 1]$  since  $\{\lambda_n^{[q]}\}_{n=0}^q$  is a nodal basis for  $\mathcal{P}^q([0, 1])$ . Going back to the Galerkin orthogonality (4.4), this amounts to showing that if

$$\int_0^1 \dot{U} v dt = 0 \quad \forall v \in \mathcal{P}^{q-1}([0, 1]),$$

with  $U \in \mathcal{P}^q([0, 1])$ , then  $U$  is constant on  $[0, 1]$ . This follows by taking  $v = \dot{U}$ . Thus,  $\xi_n = \xi_0$  for  $n = 1, \dots, q$ , so that the value of  $\sum_{n=1}^q \bar{a}_{mn}^{[q]} a_{n0}$  must be  $-1$ . This completes the proof.  $\square$

The mcG(q) method thus reads as follows: For every local interval  $I_{ij}$ , find  $\{\xi_{ijn}\}_{n=0}^{q_{ij}}$ , with  $\xi_{ij0} = \xi_{i,j-1,q_{i,j-1}}$ , such that

$$(A.11) \quad \xi_{ijm} = \xi_{ij0} + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) f_i(U(t), t) dt, \quad m = 1, \dots, q_{ij},$$

for certain weight functions  $\{w_n^{[q]}\}_{m=1}^q \subset \mathcal{P}^{q-1}(0, 1)$ , and where the initial condition is specified by  $\xi_{i00} = u_i(0)$  for  $i = 1, \dots, N$ .

The weight functions may be computed analytically for small  $q$ , and for general  $q$  they are easy to compute numerically.

**A.2. The mdG( $q$ ) method.** We now make the same ansatz as for the continuous method,

$$(A.12) \quad U_i(t) = \sum_{n=0}^q \xi_{ijn} \lambda_n^{[q_{ij}]}(\tau_{ij}(t)),$$

where the difference is that we now have  $q + 1$  degrees of freedom on every interval, since we no longer have the continuity requirement for the trial functions. We make the assumption that the nodal points for the nodal basis functions are chosen so that

$$(A.13) \quad s_q = 1,$$

i.e., the end-point of every subinterval is a nodal point for the basis functions.

With this ansatz, we get the following set of equations for determining  $\{\xi_{ijn}\}_{n=0}^{q_{ij}}$ :

$$(A.14) \quad \left( \sum_{n=0}^{q_{ij}} \xi_{ijn} \lambda_n^{[q_{ij}]}(0) - \xi_{ij0}^- \right) \lambda_m^{[q_{ij}]}(0) + \int_{I_{ij}} \sum_{n=0}^{q_{ij}} \xi_{ijn} \frac{d}{dt} \left[ \lambda_n^{[q_{ij}]}(\tau_{ij}(t)) \right] \lambda_m^{[q_{ij}]}(\tau_{ij}(t)) dt$$

$$(A.15) \quad = \int_{I_{ij}} f_i(U(t), t) \lambda_m^{[q_{ij}]}(\tau_{ij}(t)) dt$$

for  $m = 0, \dots, q_{ij}$ , where we use  $\xi_{ij0}^-$  to denote  $\xi_{i,j-1,q_{i,j-1}}$ , i.e., the value at the right end-point of the previous interval. To simplify the notation, we drop the subindices  $ij$  again and rewrite to  $[0, k]$ . We thus seek to determine the coefficients  $\{\xi_n\}_{n=0}^q$  such that for  $m = 0, \dots, q$  we have

$$(A.16) \quad \left( \sum_{n=0}^q \xi_n \lambda_n^{[q]}(0) - \xi_0^- \right) \lambda_m^{[q]}(0) + \sum_{n=0}^q \xi_n \frac{1}{k} \int_0^k \dot{\lambda}_n^{[q]}(\tau(t)) \lambda_m^{[q]}(\tau(t)) dt = \int_0^k f \lambda_m^{[q]}(\tau(t)) dt,$$

or simply

$$(A.17) \quad \sum_{n=0}^q a_{mn}^{[q]} \xi_n = b_m,$$

where

$$(A.18) \quad a_{mn}^{[q]} = \int_0^1 \dot{\lambda}_n^{[q]}(t) \lambda_m^{[q]}(t) dt + \lambda_n^{[q]}(0) \lambda_m^{[q]}(0)$$

and

$$(A.19) \quad b_m^{[q]} = \int_0^k f \lambda_m^{[q]}(\tau(t)) dt + \xi_0^- \lambda_m^{[q]}(0).$$

Now, let  $A^{[q]}$  be the  $(q+1) \times (q+1)$  matrix  $A^{[q]} = (a_{mn}^{[q]})$  with inverse  $\bar{A}^{[q]} = (\bar{a}_{mn}^{[q]})$ . Then, switching back to the full notation, we have

$$(A.20) \quad \xi_{ijm} = \xi_{ij0}^- \sum_{n=0}^q \bar{a}_{mn}^{[q]} \lambda_n^{[q]}(0) + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) f_i(U(t), t) dt, \quad m = 0, \dots, q_{ij},$$

where the *weight functions*  $\{w_n^{[q]}\}_{n=0}^q$  are given by

$$(A.21) \quad w_m^{[q]} = \sum_{n=0}^q \bar{a}_{mn}^{[q]} \lambda_n^{[q]}, \quad m = 0, \dots, q.$$

As for the continuous method, this may be somewhat simplified.

LEMMA A.2. *For the mdG( $q$ ) method, we have*

$$\sum_{n=0}^q \bar{a}_{mn}^{[q]} \lambda_n^{[q]}(0) = 1.$$

*Proof.* As in the proof for the mcG( $q$ ) method, assume that the interval is  $[0, 1]$ . Since the value of the expression is independent of  $f$  we can take  $f = 0$ . We thus want to prove that if  $f = 0$ , then the solution  $U$  is constant. By the Galerkin orthogonality, we have

$$[U]_0 v(0) + \int_0^1 \dot{U} v \, dt = 0 \quad \forall v \in \mathcal{P}^q(0, 1),$$

with  $U \in \mathcal{P}^q(0, 1)$ . Taking  $v = U - U(0^-)$ , we have

$$\begin{aligned} 0 &= ([U]_0)^2 + \int_0^1 \dot{U}(U - U(0^-)) \, dt = ([U]_0)^2 + \frac{1}{2} \int_0^1 \frac{d}{dt} (U - U(0^-))^2 \, dt \\ &= \frac{1}{2} (U(0^+) - U(0^-))^2 + \frac{1}{2} (U(1) - U(0^-))^2, \end{aligned}$$

so that  $[U]_0 = 0$ . Now take  $v = \dot{U}$ . This gives  $\int_0^1 (\dot{U})^2 \, dt = 0$ . Since then both  $[U]_0 = 0$  and  $\dot{U} = 0$  on  $[0, 1]$ ,  $U$  is constant and equal to  $U(0^-)$ , and the proof is complete.  $\square$

The mdG( $q$ ) method thus reads as follows: For every local interval  $I_{ij}$ , find  $\{\xi_{ijn}\}_{n=0}^{q_{ij}}$ , such that for  $m = 0, \dots, q_{ij}$  we have

$$(A.22) \quad \xi_{ijm} = \xi_{ij0}^- + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) f_i(U(t), t) \, dt$$

for certain weight functions  $\{w_n^{[q]}\}_{n=0}^q \subset \mathcal{P}^q(0, 1)$ .

#### REFERENCES

- [1] S. ALEXANDER AND C. AGNOR, *n-body simulations of late stage planetary formation with a simple fragmentation model*, ICARUS, 132 (1998), pp. 113–124.
- [2] U. M. ASCHER AND L. R. PETZOLD, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, Philadelphia, 1998.
- [3] J. BUTCHER, *The Numerical Analysis of Ordinary Differential Equations. Runge-Kutta and General Linear Methods*, John Wiley, Chichester, UK, 1987.
- [4] G. DAHLQUIST, *Stability and Error Bounds in the Numerical Integration of Ordinary Differential Equations*, Kungl. Tekn. Högsk. Handl. Stockholm. No., 130 (1959).
- [5] R. DAVÉ, J. DUBINSKI, AND L. HERNQUIST, *Parallel treesph*, New Astronomy, 2 (1997), pp. 277–297.
- [6] C. DAWSON AND R. KIRBY, *High resolution schemes for conservation laws with locally varying time steps*, SIAM J. Sci. Comput., 22 (2001), pp. 2256–2281.
- [7] M. DELFOUR, W. HAGER, AND F. TROCHU, *Discontinuous Galerkin methods for ordinary differential equations*, Math. Comp., 36 (1981), pp. 455–473.
- [8] K. ERIKSSON, D. ESTEP, P. HANSBO, AND C. JOHNSON, *Introduction to adaptive methods for differential equations*, in Acta Numerica, 1995, Acta Numer., Cambridge University Press, Cambridge, 1995, pp. 105–158.

- [9] K. ERIKSSON, D. ESTEP, P. HANSBO, AND C. JOHNSON, *Computational Differential Equations*, Cambridge University Press, Cambridge, 1996.
- [10] K. ERIKSSON AND C. JOHNSON, *Adaptive Finite Element Methods for Parabolic Problems III: Time Steps Variable in Space*, manuscript, Chalmers University of Technology, Göteborg, Sweden.
- [11] K. ERIKSSON AND C. JOHNSON, *Adaptive finite element methods for parabolic problems I: A linear model problem*, SIAM J. Numer. Anal., 28 (1991), pp. 43–77.
- [12] K. ERIKSSON AND C. JOHNSON, *Adaptive finite element methods for parabolic problems II: Optimal error estimates in  $L_\infty L_2$  and  $L_\infty L_\infty$* , SIAM J. Numer. Anal., 32 (1995), pp. 706–740.
- [13] K. ERIKSSON AND C. JOHNSON, *Adaptive finite element methods for parabolic problems IV: Nonlinear problems*, SIAM J. Numer. Anal., 32 (1995), pp. 1729–1749.
- [14] K. ERIKSSON AND C. JOHNSON, *Adaptive finite element methods for parabolic problems V: Long-time integration*, SIAM J. Numer. Anal., 32 (1995), pp. 1750–1763.
- [15] K. ERIKSSON, C. JOHNSON, AND S. LARSSON, *Adaptive finite element methods for parabolic problems VI: Analytic semigroups*, SIAM J. Numer. Anal., 35 (1998), pp. 1315–1325.
- [16] K. ERIKSSON, C. JOHNSON, AND V. THOMÉE, *Time discretization of parabolic problems by the discontinuous Galerkin method*, RAIRO Modél. Math. Anal. Numér., 19 (1985), pp. 611–643.
- [17] D. ESTEP, *A posteriori error bounds and global error control for approximation of ordinary differential equations*, SIAM J. Numer. Anal., 32 (1995), pp. 1–48.
- [18] D. ESTEP AND D. FRENCH, *Global error control for the continuous Galerkin finite element method for ordinary differential equations*, RAIRO Modél. Math. Anal. Numér., 28 (1994), pp. 815–852.
- [19] D. ESTEP AND R. WILLIAMS, *Accurate parallel integration of large sparse systems of differential equations*, Math. Models Methods Appl. Sci., 6 (1996), pp. 535–568.
- [20] J. FLAHERTY, R. LOY, M. SHEPHARD, B. SZYMANSKI, J. TERESCO, AND L. ZIANTZ, *Adaptive local refinement with octree load balancing for the parallel solution of three-dimensional conservation laws*, J. Parallel Distrib. Comput., 47 (1997), pp. 139–152.
- [21] E. HAIRER AND G. WANNER, *Solving Ordinary Differential Equations. I. Nonstiff Problems*, Springer Ser. Comput. Math. 8, Springer-Verlag, Berlin, 1987.
- [22] E. HAIRER AND G. WANNER, *Solving Ordinary Differential Equations. II. Stiff and Differential-Algebraic Problems*, Springer Ser. Comput. Math. 14, Springer-Verlag, Berlin, 1991.
- [23] P. HANSBO, *A note on energy conservation for hamiltonian systems using continuous time finite elements*, Comm. Numer. Methods Engrg., 17 (2001), pp. 863–869.
- [24] C. JOHNSON, *Error estimates and adaptive time-step control for a class of one-step methods for stiff ordinary differential equations*, SIAM J. Numer. Anal., 25 (1988), pp. 908–926.
- [25] O. KESSEL-DEYNET, *Berücksichtigung ionisierender Strahlung im Smoothed-Particle-Hydrodynamics-Verfahren und Anwendung auf die Dynamik von Wolkenkernen im Strahlungsfeld massiver Sterne*, Ph.D. thesis, Naturwissenschaftlich-Mathematischen Gesamtfakultät, Ruprecht-Karls-Universität, Heidelberg, 1999.
- [26] A. LOGG, *Multi-Adaptive Error Control for ODES*, NA Group Report 98/20, Oxford University Computing Laboratory, Oxford, UK, 1998; also available online from <http://www.phi.chalmers.se/preprints/abstracts/preprint-2000-03.html>.
- [27] A. LOGG, *A Multi-Adaptive ODE-Solver*, M.Sc. thesis, Department of Mathematics, Chalmers University of Technology, Göteborg, Sweden, 1998; also available online from <http://www.phi.chalmers.se/preprints/abstracts/preprint-2000-02.html>.
- [28] A. LOGG, *Multi-Adaptive Galerkin Methods for ODES I: Theory & Algorithms*, Chalmers Finite Element Center Preprint 2001–09, <http://www.phi.chalmers.se/preprints/abstracts/preprint-2000-09.html> (25 February 2001).
- [29] A. LOGG, *Multi-adaptive Galerkin methods for ODES II: Implementation and applications*, SIAM J. Sci. Comput., submitted.
- [30] A. LOGG, *Tanganyika, version 1.2.1*, <http://www.phi.chalmers.se/tanganyika/> (10 May 2001).
- [31] J. MAKINO AND S. AARSETH, *On a hermite integrator with Ahmad–Cohen scheme for gravitational many-body problems*, Publ. Astron. Soc. Japan, 44 (1992), pp. 141–151.
- [32] K.-S. MOON, A. SZEPESSY, R. TEMPONE, AND G. ZOURAKIS, *Adaptive Approximation of Differential Equations Based on Global and Local Errors*, preprint. TRITA-NA-0006 NADA, KTH, Stockholm, Sweden, 2000.
- [33] S. OSHER AND R. SANDERS, *Numerical approximations to nonlinear conservation laws with locally varying time and space grids*, Math. Comp., 41 (1983), pp. 321–336.
- [34] L. SHAMPINE, *Numerical Solution of Ordinary Differential Equations*, Chapman and Hall, London, 1994.