NASA/TM—2011-216340

Collocation and Galerkin Time-Stepping Methods

H.T. Huynh
Glenn Research Center, Cleveland, Ohio

August 2011

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at *http://www.sti.nasa.gov*

- E-mail your question via the Internet to *help@sti.nasa.gov*

- Fax your question to the NASA STI Help Desk at 443–757–5803

- Telephone the NASA STI Help Desk at 443–757–5802

- Write to:
  NASA Center for AeroSpace Information (CASI)
  7115 Standard Drive
  Hanover, MD 21076–1320

NASA/TM—2011-216340

Collocation and Galerkin Time-Stepping Methods

H.T. Huynh
Glenn Research Center, Cleveland, Ohio

Prepared for the
19th Computational Fluid Dynamics Conference
sponsored by the American Institute of Aeronautics and Astronautics
San Antonio, Texas, June 22–25, 2009

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

August 2011

Level of Review: This material has been technically reviewed by technical management.

Available from

NASA Center for Aerospace Information
7115 Standard Drive
Hanover, MD 21076–1320

National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

Available electronically at http://www.sti.nasa.gov

Collocation and Galerkin Time-Stepping Methods

H.T. Huynh
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

We study the numerical solutions of ordinary differential equations by one-step methods where the solution at t_n is known and that at t_{n+1} is to be calculated. The approaches employed are collocation, continuous Galerkin (CG) and discontinuous Galerkin (DG). Relations among these three approaches are established. A quadrature formula using s evaluation points is employed for the Galerkin formulations. We show that with such a quadrature, the CG method is identical to the collocation method using quadrature points as collocation points. Furthermore, if the quadrature formula is the right Radau one (including t_{n+1}), then the DG and CG methods also become identical, and they reduce to the Radau IIA collocation method. In addition, we present a generalization of DG that yields a method identical to CG and collocation with arbitrary collocation points. Thus, the collocation, CG, and generalized DG methods are equivalent, and the latter two methods can be formulated using the differential instead of integral equation. Finally, all schemes discussed can be cast as s-stage implicit Runge-Kutta methods.

1.0 Introduction

Collocation is an idea widely applicable to numerical analysis. In the case of numerical solutions for differential equations (or time-stepping schemes), for the one-step methods where the data u_n at t_n is known and the solution u_{n+1} at t_{n+1} is to be calculated, the collocation approach can be formulated as follows (e.g., Hairer, Norsett, and Wanner 1987, Lambert 1991). The solution is first approximated on $[t_n, t_{n+1}]$ by a polynomial P of degree s (for s-stage) interpolating the solution values at s points on $[t_n, t_{n+1}]$ called collocation points together with the value u_n at t_n. The polynomial P is determined by requiring that it satisfies the differential equation at the s collocation points. The solution u_{n+1} is given by $P(t_{n+1})$. For these methods, their accuracy and stability are determined by the choice of collocation points. For example, if the s points are chosen to be the Gauss, Radau, or Lobatto points, then the resulting method is accurate to order $2s$, $2s-1$, or $2s-2$, respectively. Collocation methods were studied in (Cooper 1968, Axelsson 1969). Wright (1970) showed that the collocation process leads to an s-stage implicit Runge-Kutta (IRK) method. His proof will be reproduced and utilized here.

The Galerkin method was introduced in 1915 for the elastic equilibrium of rods and thin plates (Fletcher 1984). It was employed to solve ordinary differential equations by Hulme (1972). An introduction to both continuous Galerkin (CG) and discontinuous Galerkin (DG) methods for differential equations can be found in (Eriksson et al. 1996). The CG method seeks to approximate the solution by a continuous function which, on each interval $[t_n, t_{n+1}]$, is a polynomial of degree s. This polynomial is determined by requiring that on $[t_n, t_{n+1}]$, the weak form of the differential equation holds for all test functions that are polynomials of degree s vanishing at t_n. Hulme (1972) showed that if an s-point quadrature formula is employed, then the resulting CG method is equivalent to a collocation method provided that the step size is bounded by certain norms to ensure the uniqueness of both solutions—a condition which will be removed here. (Solution uniqueness is not always available, e.g., for the three-dimensional Navier-Stokes equations, the problem of existence and uniqueness of the solution is still open.)

The discontinuous Galerkin method (DG) is currently popular for the spatial discretization of conservation laws (see the review paper by Cockburn, Karniadakis, and Shu, 2000). Formulated for differential equations by LeSaint and Raviart (1974), the DG method seeks to approximate the solution by a function, which can be discontinuous across t_n, and is a polynomial of degree k on each $[t_n, t_{n+1}]$. At each t_n where the solution is discontinuous, the value chosen is that just to its left—for conservation laws, such a choice is called ‘upwinding’; it serves the purpose of adding numerical dissipation and results in a more stable method. Here, after an integration by parts, this upwind value is employed to evaluate the boundary term. The polynomial of degree k representing the solution is determined by requiring that on $[t_n, t_{n+1}]$, the weak form of the differential equation (after the integration by parts) holds for all test functions that are polynomials of degree k. Using a quadrature formula with $k + 1$ evaluations including an evaluation at the left boundary t_n, LeSaint and Raviart showed that the DG formulation results in a $(k + 1)$-stage implicit Runge-Kutta (IRK) method accurate to order $2k + 1$ or less. In addition, they proved the strong A-stability property (see also Bauer 1995). The method was generalized by employing the boundary values to the right of t_{n+1} in (Delfour, Hager, and Trochu 1981). The relation between DG and collocation methods, however, has not been established. On a different but related subject, it was proven in (Adjerid et al. 2002) that for conservation laws, the DG method is superconvergent to order $2k + 1$ at the “downwind” boundary of each cell. Concerning the basic formulation, it was shown in (Huynh 2007) that for conservation laws (on a quadrilateral mesh), the DG method can be formulated using the differential form, and the result is a simple and economical algorithm.

In this paper, we first prove that if an s-point quadrature formula is employed, then the CG method using polynomials of degree s is identical to the collocation method using the s quadrature points as collocation points; in other words, the condition on the step size being small enough in (Hulme 1972) is removed. Our proof is constructive; in addition, it shows the equivalence of the integral and differential forms: with appropriate choices of basis functions, one set for the space of trial solutions and another for the space of test functions, the CG (integral) formulation is shown to result in a collocation (differential) formulation. In contrast, a typical CG formulation employs (essentially) the same basis functions for the trial and test spaces. Next, we show that if the quadrature formula is the right Radau one (including the right boundary t_{n+1}), then the DG and CG methods also become identical, and they reduce to a collocation method called Radau IIA. Compared to the proof of the fact that the DG method can be cast in the form of IRK by LeSaint and Raviart, our proof is more direct and leads to a specific member of the IRK class, namely, Radau IIA. In addition, it results in a formulation of DG using the differential instead of integral equation. Such a formulation can simplify the time discretization of the space-time DG scheme (for standard space-time DG methods, see (Van der Vegt and Van der Ven 2002)). Finally, we generalize the DG formulation in a manner that the resulting method becomes identical to CG. Our approach to this generalization does not involve the value to the right of t_{n+1}; therefore, it is simpler than the approach of Delfour, Hager, and Trochu (1981).

Most papers on this subject are written in a highly concise manner. Often, readers can find the motivation and meaning of a technique or an equation only after plowing through complicated algebraic expressions. Such conciseness may make for an elegant style; however, it can sometimes cause misunderstanding. For example, in (Delfour et al. 1981), it was stated that their generalized DG method has the property of “superconvergence of order $2k + 1$” where k is the degree of the discontinuous piecewise polynomial. Concerning the CG method discussed by Hulme (1972), they stated: “Note, however, that these continuous approximations have order $2k$ at the mesh points instead of $2k + 1$”. It will be shown here that the CG method is, in fact, more accurate than DG: if s is the number of stages for the resulting IRK method, then, concerning highest attainable accuracy, CG is of order $2s$, whereas DG, order $2s - 1$. To put it differently, for highest accuracy, CG corresponds to the Gauss quadrature, whereas DG, the right Radau one; as a result, CG is more accurate than DG. Note that for stiff problems, an even more critical criterion is stability, and here, the DG or right Radau collocation method is more advantageous.

This paper is written in an expository manner since several different methods are involved, and a typical reader may be unfamiliar with one or more of them. Another goal of the expository style is to avoid misunderstanding. The paper is organized as follows: the collocation method is discussed in Section 2.0; CG in Section 3.0; and DG in Section 4.0. A brief review of Legendre and Radau polynomials and a few examples of collocation methods can be found in the Appendices.

We now set up the problem and introduce notations and techniques common to all methods. Note that the methods discussed here can be applied to systems of equations; for simplicity of notation, we deal only with the scalar case. Consider the ordinary differential equation (ODE)

$$u'(t) = f(t, u(t)) \tag{1.1}$$

with the initial condition

$$u(t_0) = u_0 \,. \tag{1.2}$$

Let h be the step size and $t_n = t_0 + nh$ where $n = 0,1,\ldots,N$. Recall that a one-step method uses one starting value for each step; i.e., the data u_n at time t_n is assumed to be known; the method provides a solution u_{n+1} at t_{n+1}. For $n = 0$, u_n is the initial condition u_0 in (1.2). Note that these methods can be applied to a variable step size h_n; the assumption of constant step size is only for convenience.

The following two well-known special cases are illuminating. For the first case, f depends only on t:

$$u'(t) = f(t) \,. \tag{1.3}$$

The exact solution is

$$u(t) = u_0 + \int_{t_0}^{t} f(\tau)\, d\tau \,. \tag{1.4}$$

If u_n is known, the exact u_{n+1} is

$$u_{n+1,\,\text{exact}} = u_n + \int_{t_n}^{t_{n+1}} f(t)\, dt \,. \tag{1.5}$$

Thus, each one-step method results in a quantity $u_{n+1} - u_n$, which is a quadrature formula approximating the integral above.

For the second case, $f = \lambda u$ where λ is a complex constant; therefore,

$$u'(t) = \lambda\, u(t) \,. \tag{1.6}$$

The exact solution is again obvious:

$$u(t) = u_0\, e^{\lambda (t - t_0)} \,. \tag{1.7}$$

If u_n is known, the exact u_{n+1} is given by

$$u_{n+1,\,\text{exact}} = u_n\, e^{\lambda h} \,. \tag{1.8}$$

Each one-step method yields a solution u_{n+1}. Define the stability function R by

$$u_{n+1} = u_n\, R(\lambda h) \,. \tag{1.9}$$

Then with $z = \lambda h$, (1.8) implies $R(z)$ approximates e^z. If the method is of order p, then the local error is

$$E(z) = e^z - R(z) = O(z^{p+1}) \,. \tag{1.10}$$

In other words,

$$R(z) = 1 + z + \frac{z^2}{2!} + \ldots + \frac{z^p}{p!} + O(z^{p+1}) \,. \tag{1.11}$$

The converse, however, does not hold: due to nonlinear errors, (1.10) or (1.11) does not imply that the method is of order p. (Note that the quantities $O(z^{p+1})$ in (1.10) and (1.11) are different from each other).

The stability domain is

$$S = \left\{ \text{complex number } z \text{ such that } |R(z)| \le 1 \right\} .$$

With $z = \lambda h$, the solution for (1.6) after n steps is $u_n = u_0 R(z)^n$. If z is in S, then $|R(z)^n| \le 1$; therefore, $|u_n| \le |u_0|$ for all n, i.e., the solution is bounded for all time. Next, a method is A-stable if the corresponding S contains the left half of the complex plane:

$$S \supset \left\{ z \, ; \operatorname{Re}(z) \le 0 \right\} . \tag{1.12}$$

An A-stable method (such as the trapezoidal rule (B.3) below) can have the following property, which is not always desirable,

$$\lim_{z \to \infty} R(z) = 1 .$$

With such a property, for an exact solution that damps quickly (say, e^{-1000t}), the approximate solution damps very slowly. A more desirable property is L-stability: a method is L-stable if it is A-stable and

$$\lim_{z \to \infty} R(z) = 0 . \tag{1.13}$$

L-stability implies that the method is suitable for stiff problems (the λ values or eigenvalues of a stiff problem have magnitudes in a wide range, from very small to very large).

The following rescaling technique is employed extensively below. Denote $I_n = [t_n, t_{n+1}]$ and $I = [0,1]$. Instead of I_n, it is often more convenient to work with I. For t on I_n, set

$$\tau = (t - t_n)/h . \tag{1.14a}$$

Then τ varies on I. The inverse maps I onto I_n,

$$t = t_n + \tau \, h . \tag{1.14b}$$

Each function $g(t)$ on I_n corresponds to a function $\hat{g}(\tau)$ on I, namely, $\hat{g}(\tau) = g(t_n + \tau \, h)$. Here, we use the notation $g(t_n + \tau h)$ instead of $\hat{g}$. Denoting $g' = dg/dt$, we have, by the chain rule,

$$\frac{d}{d\tau} g(t_n + \tau h) = h \, \frac{dg(t)}{dt} = h \, g'(t) . \tag{1.15}$$

As for integrals,

$$\int_{t_n}^{t_{n+1}} g(t) \, dt = h \int_0^1 g(t_n + \tau h) \, d\tau . \tag{1.16}$$

2.0 Collocation Methods

To describe these methods, let c_i, $i = 1,\ldots,s$ be (collocation) points in ascending order on I,

$$0 \le c_i \le 1 \text{ and } c_i < c_j \text{ for } i < j . \tag{2.1}$$

Let the collocation points on I_n be defined by

$$t_{n,\, i} = t_n + c_i\, h\,. \tag{2.2}$$

Suppose, for the moment, the solution values $u_{n,i}$ at $t_{n,i}$, $i = 1,\ldots,s$, are known. These s values together with the value u_n at t_n determine a polynomial $P = P(t)$ of degree s (the case $c_1 = 0$ will be discussed later),

$$P(t_n) = u_n \tag{2.3}$$

and, for $i = 1,\ldots,s$,

$$P(t_{n,\, i}) = u_{n,\, i}\,. \tag{2.4}$$

The quantities $P'(t_{n,i}) = (dP/dt)(t_{n,i})$ and $f(t_{n,i},u_{n,i})$ can then be evaluated for $i = 1,\ldots,s$. The collocation method seeks a polynomial P that satisfies the following implicit equations: for $i = 1,\ldots,s$,

$$P'(t_{n,\, i}) = f(t_{n,\, i}\,, u_{n,\, i}) = f(t_{n,\, i}\,, P(t_{n,\, i})) \tag{2.5}$$

Once P is determined, the solution at t_{n+1} is given by

$$u_{n+1} = P(t_{n+1})\,. \tag{2.6}$$

Two remarks are in order. First, the approximating polynomials for u and f of the ODE are of different degrees. Indeed, if u is approximated by P of degree s, then u' is approximated by P' of degree $s - 1$. Since $u' = f$, we wish to approximate f by a polynomial of degree $s - 1$. For the collocation method, this polynomial is determined by the values of f at $t_{n,1},\ldots,t_{n,s}$ (and not the value at t_n). As an example, for the equation $u' = \lambda u$, the function u of the left hand side is approximated by P, whereas, u of the right hand side, by the values at the collocation points $t_{n,1},\ldots,t_{n,s}$ (a polynomial of degree $s - 1$).

The second remark concerns the case $c_1 = 0$. Here, the derivative $P'(t_{n,1}) = P'(t_n) = f(t_n,u_n)$ can be calculated explicitly. Therefore, P is determined by (2.5) with $i = 2,\ldots,s$ and

$$P(t_n) = u_n \quad \text{and} \quad P'(t_n) = f(t_n\,, u_n)\,. \tag{2.7}$$

2.1 Collocation and Implicit Runge-Kutta (IRK) Methods

It was shown by Wright (1970) that the above collocation method results in an s-stage IRK method. The proof of this fact (Lambert 1991) is reproduced below since it will be employed for our proof of equivalence between CG and collocation methods. It amounts to expressing P' in terms of certain basis functions and then integrating P' to obtain P in IRK form.

Consider the s collocation points $c_1,\ldots,c_s$ on [0,1]. The values (of a function) at these points determine a polynomial of degree $s - 1$. With i fixed, $1 \le i \le s$, let $L_i(\tau)$ be the Lagrange polynomial of degree $s - 1$ defined by $L_i(c_j) = \delta_{ij}$ for $j = 1,\ldots,s$; i.e., L_i takes value 1 at $\tau = c_i$ and 0 at all other c_j, $j \neq i$ (see Fig. 2.1),

$$L_i(\tau) = \prod_{j=1,\ j\neq i}^{s} \frac{\tau - c_j}{c_i - c_j}\,. \tag{2.8}$$

Then L_i, $1 \le i \le s$, form a basis for the space of polynomials of degree $s - 1$ on [0,1].

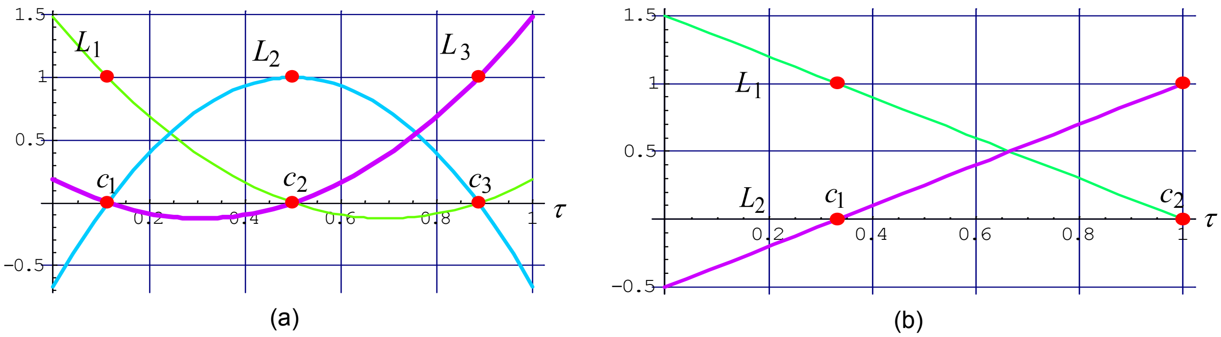


Figure 2.1.—(a) Functions L_j, j = 1,2,3 for the case of 3 Gauss points (s = 3) as collocation points on the interval I = [0,1]. (b) The same functions for the case of 2 right Radau points (c_2 = 1).

For $i = 1,\ldots,s$, set

$$k_i = f(t_{n,\,i}\,, u_{n,\,i})\,. \tag{2.9}$$

Next, observe that $P' = dP/dt$ is a polynomial of degree $s - 1$. Using j for the index instead of i (the reason for this switch will be clear in (2.11)), since P' interpolates the s data points $(t_n + c_jh,k_j)$,

$$P'(t_n + \tau h) = \sum\nolimits_{j=1}^{s} L_j(\tau)\; k_j\ . \tag{2.10}$$

Concerning the above left hand side, with $t = t_n + \tau h$, we have

$$h\int_0^{c_i} P'(t_n + \tau h)\; d\tau = \int_{t_n}^{t_n + c_i h} P'(t)\; dt = P(t_n + c_i\, h) - P(t_n)\ .$$

Equation (2.10) then implies, for the i-th stage,

$$P(t_n + c_i\, h) - P(t_n)\ \ =\ h\sum\nolimits_{j=1}^{s}\left(\int_0^{c_i} L_j(\tau)\; d\tau\right) k_j\,. \tag{2.11}$$

As for the solution u_{n+1},

$$u_{n+1} - u_n\ = P(t_n + h) - P(t_n)\ \ =\ h\sum\nolimits_{j=1}^{s}\left(\int_0^{1} L_j(\tau)\; d\tau\right) k_j\,. \tag{2.12}$$

Motivated by (2.11), set

$$a_{ij}\ = \int_0^{c_i} L_j(\tau)\; d\tau\,, \tag{2.13}$$

and, by (2.12), set

$$b_j\ = \int_0^{1} L_j(\tau)\; d\tau\,. \tag{2.14}$$

Then, with $u_{n,j} = P(t_{n,j})$ and $k_j = f\,(t_{n,j},\, u_{n,j})$, (2.11) implies the following IRK form of the collocation method: for $i = 1,\ldots,s$,

$$u_{n,i} = u_n + h\sum_{j=1}^{s} a_{ij}\,k_j\,. \tag{2.15}$$

After obtaining $u_{n,i}$ (and k_i) by solving the above system of s implicit equations, the solution is given by

$$u_{n+1} = u_n + h\sum_{j=1}^{s} b_j\,k_j\,. \tag{2.16}$$

This completes the proof.

The Butcher array of an IRK method consists of c_i, a_{ij}, and b_j arranged as follows

$$\begin{array}{c|ccc} c_1 & a_{11} & \ldots & a_{1s} \\ \vdots & \vdots & \ldots & \vdots \\ c_s & a_{s1} & \ldots & a_{ss} \\ \hline & b_1 & \ldots & b_s \end{array}$$

Here, for each i-th row, $a_{ij}, j = 1,\ldots,s$ are the weights of a quadrature formula as will be shown in (2.18).

Note that if $c_1 = 0$, then $t_{n,1} = t_n$, $u_{n,1} = u_n$, and $a_{1j} = 0$ for all j; in other words, the first row of the Butcher array is identically zero. In addition, the quantity $k_1 = f(t_n,u_n)$ can be calculated explicitly. Since $u_{n,1} = u_n$ is known, (2.15) for $i = 2,\ldots,s$ then yields the equations to calculate $u_{n,2},\ldots,u_{n,s}$.

Concerning the stability function for the IRK method, define the $s{\times}s$ matrices $\mathbf{A} = [a_{ij}]$ and $\mathbf{I}$ = identity matrix, column vectors of s entries $b = [b_1,\ldots,b_s]^T$ and e = $[1,1,\ldots,1]^T$ where the superscript T stands for transpose, and det = determinant. Then, for the IRK method, the stability function R can be calculated by one of the following two formulas (e.g., Lambert 1991)

$$R(z) = 1 + zb^T(\mathbf{I} - z\mathbf{A})^{-1}e$$

or

$$R(z) = \frac{\det[1 - z\mathbf{A} + zeb^T]}{\det[\mathbf{I} - z\mathbf{A}]}\,.$$

2.2 Quadratures Associated with IRK Method

The above IRK method relates to quadrature formulas as follows. By (2.16) and the definition of k_j,

$$u_{n+1} - u_n = h\sum_{j=1}^{s} b_j\,u'_{n,j}\,.$$

The corresponding quadrature formula is, for any continuous function v on $I = [0,1]$,

$$\int_0^1 v(\tau)\,d\tau \approx \sum_{j=1}^{s} b_j\,v(c_j)\,. \tag{2.17}$$

Here, b_j are the weights given in (2.14), and the collocation points c_j are the evaluation points. Similarly, by (2.15),

$$u_{n,i} - u_n = h\sum_{j=1}^{s} a_{ij}\,u'_{n,j}\,.$$

The corresponding quadrature formula is

$$\int_0^{c_i} v(\tau)\,d\tau \;\approx\; \sum\nolimits_{j=1}^{s} a_{ij}\, v(c_j)\,. \tag{2.18}$$

Here, we use the s collocation points on [0,1] to evaluate the integral from 0 to c_i.

An observation concerning accuracy of these quadratures is in order. With appropriate choice of evaluation points on I, formula (2.17) has a degree of precision of up to $2s - 1$. (Recall that a quadrature is of degree of precision m if it is exact for polynomials of degree m or less.) Formula (2.18) for the stages, however, has a degree of precision no higher than $s - 1$ since special points (say, Gauss points) on [0,1] are generally not special on $[0,c_i]$. For example, if we use 2 Gauss points on [0,1] as collocation points, then the degree of precision for (2.17) is 3, but that for (2.18) is only 1. As will be shown, the resulting collocation method is of order 4.

2.3 Basis Functions $\widetilde{L}_j$

We next introduce the basis functions $\widetilde{L}_j$, which will be employed in the proof of equivalence between the collocation and CG methods. With $t = t_n + \tau h$, similar to (2.11), by (2.10),

$$P(t) - P(t_n) \;=\; h\sum\nolimits_{j=1}^{s}\left(\int_0^{\xi} L_j(\tau)\;d\tau\right) k_j\,. \tag{2.19}$$

Denote, for $j = 1,\ldots,s$,

$$\widetilde{L}_j(\tau) = \int_0^{\tau} L_j(\xi)\;d\xi\,. \tag{2.20}$$

Then $\widetilde{L}_j$ is of degree s since L_j is of degree $s - 1$. Some additional noteworthy properties of $\widetilde{L}_j$ are:

$$d\,\widetilde{L}_j\,/\,d\tau = L_j\,; \tag{2.21a}$$

in addition,

$$\widetilde{L}_j(0) = 0, \qquad \widetilde{L}_j(c_i) = a_{ij}, \quad \text{and} \quad \widetilde{L}_j(1) = b_j\,; \tag{2.21b}$$

moreover, since $\sum\nolimits_{j=1}^{s} L_j = 1$,

$$\sum\nolimits_{j=1}^{s} \widetilde{L}_j(\tau) \;=\; \tau\,. \tag{2.22}$$

Next, by (2.19),

$$P(t) - P(t_n) \;=\; P(t_n + \tau h) - P(t_n) \;=\; h\sum\nolimits_{j=1}^{s} k_j\,\widetilde{L}_j(\tau)\;. \tag{2.23}$$

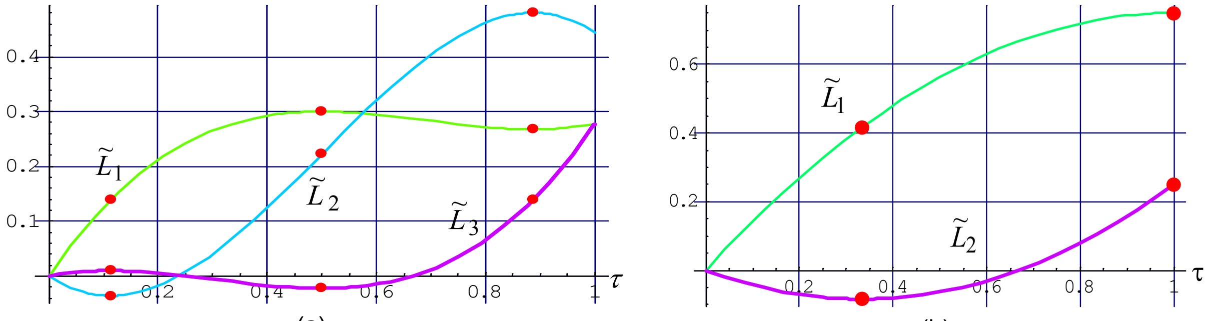


Figure 2.2.—(a) Functions $\tilde{L}_j$ j = 1,2,3 for the case of 3 Gauss points on I = [0,1]. Note that $d\tilde{L}_j(\tau)/d\tau = L_j(\tau)$; therefore, the graph of $\tilde{L}_j$ has slope 1 at c_j and slope 0 at all other c_i as can be seen by the slopes at the dots. (b) Functions $\tilde{L}_j$ for the case of 2 (right) Radau points.

For convenience, set

$$\tilde{L}_0 = 1 \tag{2.24a}$$

and

$$k_0 = P(t_n)/h = u_n/h\,. \tag{2.24b}$$

Then $P(t_n) = h\,k_0\,\tilde{L}_0$, and (2.23) can be expressed as

$$P(t_n + \tau h) \;\; = \; h\sum\nolimits_{j=0}^{s} \; k_j\,\tilde{L}_j(\tau)\;. \tag{2.25}$$

Observe that $\tilde{L}_j, j = 0,\ldots,s$, are s + 1 polynomials of degree no higher than s. We will show that they are linearly independent; thus, they form a basis for the space of polynomials of degree s or less. That is, we wish to show that if

$$\sum\nolimits_{j=0}^{s} \alpha_j\,\tilde{L}_j = 0\,, \tag{2.26}$$

then

$$\alpha_j = 0 \;\text{ for }\; j = 0,\ldots,s\,. \tag{2.27}$$

To this end, observe that (2.26) implies $\sum\nolimits_{j=0}^{s} \alpha_j\,\tilde{L}_j(0) = 0$. Since $\tilde{L}_j(0) = 0$ for j = 1,…,s, we have $\alpha_0\,\tilde{L}_0(0) = 0$. By definition, $\tilde{L}_0 = 1$; as a result,

$$\alpha_0 = 0\,. \tag{2.28}$$

Equation (2.26) then takes the form

$$\sum\nolimits_{j=1}^{s} \alpha_j\,\tilde{L}_j = 0\,. \tag{2.29}$$

Note the starting value of 1 for j. Differentiating the above, we have

$$\sum_{j=1}^{s} \alpha_j L_j = 0 .$$

Since $L_j, j = 1,\ldots,s$, are independent, we conclude that $\alpha_1 = \ldots = \alpha_s = 0$. This fact and (2.28) complete the proof that $\widetilde{L}_j$ are independent.

The above observation implies that $\widetilde{L}_j$, $j= 0,\ldots,s$, form a basis for the space of polynomials of degree no higher than s. Thus, if Q is any polynomial of degree s or less, then Q can be expressed as a linear combination of $\widetilde{L}_j$:

$$Q(t_n + \tau h) \;=\; h\sum_{j=0}^{s} \alpha_j \widetilde{L}_j(\tau) \tag{2.30a}$$

where the coefficients α_j relate to Q by

$$\alpha_0 = Q(t_n)/h \;\text{ and }\; \alpha_j = Q'(t_{n,j}) \;\text{ for }\; j = 1,\ldots,s . \tag{2.30b}$$

Also note that if Q is a polynomial of degree s and $Q(t_n)$ and $Q'(t_{n,j})$, $j = 1,\ldots,s$, are known, then Q is given by (2.30). Finally, examples of the Gauss, Radau, and Lobatto collocation methods can be found in appendix A.

3.0 Continuous Galerkin (CG) Methods

The CG method seeks to approximate the solution by a continuous function which, on each interval $[t_n, t_{n+1}]$, is a polynomial U of degree s determined by using the weak form of the differential equation. Again, assuming that $U(t_n) = u_n$ is known, we wish to calculate $u_{n+1} = U(t_{n+1})$.

We need some preparations. Let $[\alpha,\beta]$ be any interval; here, it is either $I = [0,1]$ or $I_n = [t_n, t_{n+1}]$. For simplicity, unless otherwise stated, we deal only with continuous functions (in fact, polynomials) on $[\alpha,\beta]$. The inner product of two functions v_1 and v_2 is defined by

$$(v_1, v_2) = \int_{\alpha}^{\beta} v_1(t)\, v_2(t)\, dt \;. \tag{3.1}$$

Next, denote by $\mathbf{P}^s[\alpha,\beta]$ the space of polynomials of degree s or less on $[\alpha,\beta]$. In addition, denote by $\mathbf{P}_0^s[\alpha,\beta]$ the subspace of $\mathbf{P}^s[\alpha,\beta]$ consisting of polynomials that vanish at the left boundary α. When there is no confusion concerning the interval, we use the notation $\mathbf{P}^s$ and $\mathbf{P}_0^s$. Note that $\mathbf{P}^s$ is of dimension $s + 1$, and $\mathbf{P}_0^s$, dimension s.

In general, the integrals, e.g., the inner product (3.1), are carried out by approximations rather than by exact integration. We can employ the quadrature (2.17) on I (the c_j are now the evaluation points):

$$\int_0^1 v(\tau)\, d\tau \;\approx\; \sum_{j=1}^{s} b_j\, v(c_j) . \tag{3.2}$$

The quadrature on I_n differs from the above by a factor of h: for any continuous function $v(t)$,

$$\int_{t_n}^{t_{n+1}} v(t)\; dt \;\approx\; h \sum_{j=1}^{s} b_j\, v(t_{n,j}) . \tag{3.3}$$

We often use (3.2) instead of the above (i.e., the factor h is understood) when there is no confusion.

The *trial* space or space of trial solutions on I_n is defined by

$$\mathbf{S} = \{\text{polynomial } U \text{ of degree } s \text{ such that } U(t_n) = u_n\} . \tag{3.4}$$

In spite of its name, **S** is in fact not a space, but is a hyperplane in $\mathbf{P}^s[t_n, t_{n+1}]$.

A *test* space or a space of test functions on I_n, commonly denoted by **V**, is a subspace of $\mathbf{P}^s$ which has dimension s. Among the most commonly used test spaces is the space consisting of polynomials that satisfy the homogeneous boundary condition at t_n

$$\mathbf{V} = \{ v \text{ of degree } s \text{ such that } v(t_n) = 0 \}, \tag{3.5}$$

i.e., $\mathbf{V} = \mathbf{P}_0^s$. Note that if U is in **S**, then $U - u_n$ is an element of the above **V**.

Next, the weak form of the equation $u' = f$ can be written formally as

$$(u', v) = (f, v). \tag{3.6}$$

The CG method seeks a solution U in the trial space **S** that satisfies, for all v in the test space **V**,

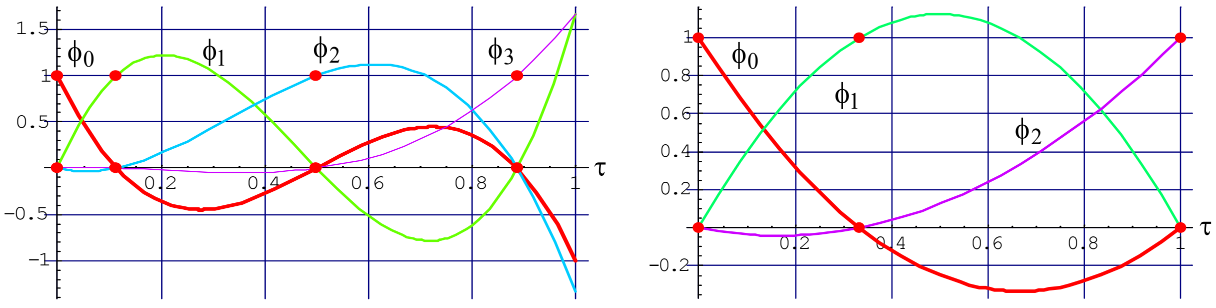
$$(U', v) = (f, v) \ . \tag{3.7}$$

In other words, the projection of U' and $f(t, U(t))$ onto **V** are the same.

We now show that the test space, as opposed to the case of trial space, must be a subspace, and it must have dimension s. Indeed, first, suppose (3.7) holds for functions v_1 and v_2 (which take the place of v). Let α and β be real numbers. Then, one can easily verify that $(U', \alpha v_1 + \beta v_2) = (f, \alpha v_1 + \beta v_2)$; thus, the test space must form a subspace. Next, since U is of degree s, and one condition is known, namely, $\mathrm{U}(t_n) = u_n$, s conditions remain to be determined. Therefore, the test space must be of dimension s.

We next discuss the choice of test spaces. Instead of solving for U, we can solve for $U - u_n$. Since $U - u_n$ is in **V**, and **V** defined by (3.5) has the correct dimension, it is sensible to use **V** as a test space.

The following argument results in another choice of test space. Since U' is of degree $s - 1$, for the two sides of (3.7) to match, we should approximate f by a polynomial also of degree $s - 1$. Such an approximation can be obtained by the projection of f on $\mathbf{P}^{s-1}$. For this projection to be identical to U', it suffices to require that (3.7) holds for all v in $\mathbf{P}^{s-1}$, i.e., the test space be $\mathbf{P}^{s-1}$. It was observed in (Fletcher 1984) and also in (Eriksson et al. 1996) that this test space yields a more accurate solution than the test space $\mathbf{P}_0^s$. We will show in (3.34) below that with the quadrature (3.2), or equivalently, (3.3), the two test spaces $\mathbf{V} = \mathbf{P}_0^s$ and $\mathbf{V} = \mathbf{P}^{s-1}$ yield identical results. The following conclusion can then be drawn: for the CG formulation with the test space $\mathbf{V} = \mathbf{P}_0^s$, the method using Gauss quadrature is more accurate than that using exact integration. (This fact is contrary to the common belief that more accurate integration formulas yield more accurate solutions.)



(a) (b)

Figure 3.1.—(a) Basis functions ϕ_j $j = 0,\ldots,3$ for the case of 3 Gauss points on $I = [0,1]$. (b) Basis functions ϕ_0, ϕ_1, and ϕ_2 for the case of 2 right Radau points. Note that $\phi_1,\ldots, \phi_s$ form a basis for $\mathbf{P}_0^s$.

In addition to $\widetilde{L}_j$ defined by (2.20), we will need the following basis for $\mathbf{P}^s$. On I, assuming $c_1 > 0$, set

$$c_0 = 0 . \tag{3.8}$$

Then, $c_0 > c_1 > \ldots . c_s \leq 1$. Let $\phi_0,\ldots, \phi_s$ be the corresponding basis functions: for each j,

$$\phi_j(\tau) = \prod_{i=0,\, i\neq j}^{s} \frac{\tau - c_i}{c_j - c_i} . \tag{3.9}$$

That is, $\phi_j(c_i) = \delta_{ij}$ as shown in Figure 3.1. (Note that the definition of L_i in (2.8) is different from the current definition in that it does not include c_0).

Since $\phi_0(0) = 1$, we can employ ϕ_0 to deal with the boundary condition at t_n. For all $j \geq 1$, $\phi_j(0) = 0$; as a consequence, $\phi_1,\ldots, \phi_s$ form a basis for $\mathbf{P}_0^s$. Also note that in the case of s Gauss points, ϕ_0 defined above is identical to the Legendre polynomial of degree s on I; in the case of right Radau points, it is identical to the Radau polynomial (see Appendix). In the case of Lobatto points, however, ϕ_0 is not the Lobatto polynomial; this case corresponds to $c_1 = 0$ and will be discussed later.

3.1 CG and Collocation Methods

We claim that if the quadrature formula (3.2) is employed to evaluate the inner product, then the resulting CG method is identical to the collocation method with quadrature points as collocation points.

To prove the above claim, we will start with the CG solution U and show that it can be expressed in collocation form. The key ingredient of the proof is the choice of $\{\widetilde{L}_j\}_{j=0}^{s}$ defined by (2.20) as a basis for $\mathbf{S}$ and $\{\phi_j\}_{j=1}^{s}$ defined by (3.9) as a basis for $\mathbf{V} = \mathbf{P}_0^s$. Since $U - u_n$ is in $\mathbf{V}$, and $\{\widetilde{L}_j\}_{j=1}^{s}$ is a basis for $\mathbf{V}$, $U - u_n$ can be expressed in terms of these basis functions:

$$U(t_n + \tau h) \; - \; u_n \; = \; h \sum\nolimits_{j=1}^{s} \; d_j \, \widetilde{L}_j(\tau) \; . \tag{3.10}$$

As discussed in (2.30), d_j and U are related by, for $j = 1,\ldots,s$,

$$d_j = U'(t_{n,j}) . \tag{3.11}$$

We wish to show, by using the weak form (3.7), that $d_j = f(t_{n,j},U(t_{n,j}))$. To this end, because $d\widetilde{L}_j / d\tau = L_j$, by differentiating (3.10) and employing the chain rule,

$$U'(t_n + \tau h) \; = \; \sum\nolimits_{j=1}^{s} \; d_j \, L_j(\tau) \; . \tag{3.12}$$

Next, recall that U is a CG solution. Noting that $\phi_1,\ldots,\phi_s$ form a basis for $\mathbf{V}$, they can serve as test functions: for $i = 1,\ldots,s$,

$$(U', \phi_i) = (f, \phi_i) . \tag{3.13}$$

We will show that under the quadrature rule, the left hand side above yields $U'(t_{n,i})$, and the right hand side, $f(t_{n,i}, U(t_{n,i}))$. Indeed, by (3.12),

$$(U', \phi_i) \; = \; \sum\nolimits_{j=1}^{s} \; d_j \, (L_j , \phi_i) . \tag{3.14}$$

When the inner product is evaluated by quadrature (3.2), we use the notation $(.,.)_q$, e.g.,

$$(v, w)_q \ = \ \sum_{i=1}^{s} b_i \ v(c_i)\, w(c_i)\,.$$

In (3.14), L_j vanishes at all collocation points except at c_j, and ϕ_i vanishes at all collocation points except at c_i. Consequently, the only nonzero term for the sum is $d_i(L_i,\phi_i)$. Since $L_i(c_i) = \phi_i(c_i) = 1$, employing the quadrature rule, (3.14) implies

$$(U', \phi_i)_q \ = \ b_i\, d_i\,. \tag{3.15}$$

Concerning the right hand side of (3.13), under the quadrature rule, only the values of f at the quadrature points are needed. Since $f\phi_i$ takes on the value $f(t_{n,i},U(t_{n,i}))$ at $\tau = c_i$ and the value 0 at all other c_j,

$$(f, \phi_i)_q \ = \ b_i\, f(t_{n,i}, U(t_{n,i}))\,. \tag{3.16}$$

Again, under the quadrature rule, by (3.15) and (3.16), equation (3.13) implies

$$d_i \ = f(t_{n,i}, U(t_{n,i}))\,.$$

From the above and (3.11), for $i = 1,\ldots,s$,

$$U'(t_{n,i}) \ = f(t_{n,i}, U(t_{n,i}))\,. \tag{3.17}$$

That is, U plays the role of P in the definition of the collocation method (2.23). This completes the proof.

3.2 Standard CG Method (Via Basis Functions)

In (3.7) above, the CG method is formulated via trial and test spaces. It can also be formulated via basis functions (Hulme 1972, Delfour et al. 1981). This formulation is presented here with more explanations and will be employed later in (3.35) to (3.45). Let $\varphi_0,\ldots,\varphi_s$ be a set of basis functions for $\mathbf{P}^s[0,1]$ with the following properties. First,

$$\varphi_0(0) = 1 \tag{3.18}$$

so that φ_0 can deal with the boundary condition at t_n. Next, for $j = 1,\ldots.,s$,

$$\varphi_j(0) = 0 \tag{3.19}$$

so that $\varphi_1,\ldots,\varphi_s$ form a basis for $\mathbf{P}_0^s$. Note that $\widetilde{L}_j$ defined by (2.20) and (2.24a) and ϕ_j by (3.9) possess these properties. Set

$$d_0 \ = \ u_n / h\,. \tag{3.20}$$

Then the trial solution U of (3.4) can be written as

$$U(t_n + \tau h) \ = \ h \sum_{j=0}^{s} d_j\, \varphi_j(\tau) \tag{3.21}$$

where d_0 is given by (3.20) and $d_1,\ldots,d_s$ remain to be determined. Taking $d/d\tau$ of the above,

$$U'(t_n + \tau h) \;=\; \sum\nolimits_{j=0}^{s} d_j \; d\varphi_j(\tau)/d\tau \;. \tag{3.22}$$

Using the test function $v = \varphi_i$ where $i = 1,\ldots,s$, defined by (3.9) (these φ_i form a basis for **V**), the weak form (3.7) implies

$$\sum\nolimits_{j=0}^{s} (\varphi_i,\; d\varphi_j/d\tau)\, d_j \;=\; (\varphi_i, f)\,. \tag{3.23}$$

We thus obtain s implicit equations (since f also depends on d_i) for s unknowns $d_1,\ldots,d_s$.

Note that in general, the expression $f\left(t, \sum\nolimits_{j=0}^{s} d_j\, \varphi_j(t)\right)$, which depends on the unknowns d_j, can be complicated; as a result, evaluating (φ_i,f) via exact integration may become difficult. It is often more convenient to evaluate the inner product by a quadrature formula employing only the values $f(t_{n,j},U(t_{n,j}))$ at the quadrature points.

3.3 Equivalence of Collocation and CG Methods Via Approximating the Dirac Delta Function

If we use the Dirac delta function, the proof of this equivalence is shortened considerably. The basic idea is that on $\mathbf{P}^s$ with an inner product by quadrature (3.2), the Dirac delta function δ_i has the same effect as ϕ_i/b_i where ϕ_i is the basis function given by (3.9) and b_i is the quadrature weight. Indeed, on $I = [0,1]$, let δ_i be the Dirac delta function at c_i defined by, for any v in $\mathbf{P}^s$,

$$(v,\delta_i) \;=\; v(c_i)\,. \tag{3.24}$$

Using the quadrature rule (3.2), again for any v in $\mathbf{P}^s$,

$$(v,\, \phi_i/b_i)_q \;=\; v(c_i) \;=\; (v,\delta_i)\,. \tag{3.25}$$

That is, concerning the inner product on $\mathbf{P}^s$ via the quadrature rule, we have

$$\phi_i/b_i = \delta_i\,. \tag{3.26}$$

The collocation method (2.5), with P defined by (2.3) and (2.4), can be written as

$$(P',\delta_i) \;=\; (f,\delta_i)\,. \tag{3.27}$$

Therefore, by (3.25), for $i = 1,\ldots,s$,

$$(P',\phi_i)_q \;=\; (f,\phi_i)_q\,. \tag{3.28}$$

The above is the CG form (3.13) with the inner product evaluated by the quadrature (3.2) and U replaced by P.

Note that for the above argument to hold, ϕ_i is required to be of degree s and thus, is defined by $s + 1$ conditions, but only the s values of ϕ_i at the collocation points are needed in the above proof. The extra condition can be arbitrary (i.e., the condition $\phi_i(0) = 0$ is not required) as discussed further in (3.33) below.

3.4 The Case $c_1 = 0$ (and $c_2 > 0$)

Examples for this case are the Lobatto and the left Radau quadrature points. Via the collocation approach, as discussed in (2.7), for this case, in addition to the boundary value u_n, the derivative $u_n' = f(x_n, u_n)$ is also known (easily evaluated). Concerning the CG approach, it needs to be modified to incorporate the condition that u_n' is known. To this end, the trial space is defined as

$$\{\text{polynomial } U \text{ of degree } s \text{ such that } U(t_n) = u_n \text{ and } U_n' = f(x_n, u_n)\} . \tag{3.29}$$

This trial space is of dimension $s - 1$. The test space can be modified accordingly:

$$\{v \text{ of degree } s \text{ such that } v(t_n) = 0 \text{ and } v'(t_n) = 0\} . \tag{3.30}$$

This space is also of dimension $s - 1$. For the case $c_1 = 0$, the CG method seeks a solution U in the trial space (3.29) that satisfies, for all v in the test space (3.30),

$$(U', v) = (f, v) .$$

Next, the definition of $\widetilde{L}_j$ in (2.20) remains valid. In addition, the definition of the basis functions ϕ_j in (3.9), except for ϕ_0 and ϕ_1, also remains valid. Since $c_0 = c_1 = 0$, to modify ϕ_0, we define it to be a polynomial of degree s such that

$$\phi_0(0) = 1,\ \phi_0{}'(0) = 0, \text{ and } \phi_0(c_i) = 0 \text{ for } i = 2, \ldots, s . \tag{3.31a}$$

That is

$$\phi_0(\tau) = (1 - a\tau) \prod_{i=2}^{s} \frac{c_i - \tau}{c_i}$$

where (3.31b)

$$a = \left(\prod_{i=2}^{s} \frac{c_i - \tau}{c_i} \right)' (0) .$$

As for ϕ_1, it is defined to be a polynomial of degree s such that

$$\phi_1(0) = 0,\ \phi_1{}'(0) = 1, \text{ and } \phi_1(c_i) = 0 \text{ for } i = 2, \ldots, s . \tag{3.32a}$$

That is

$$\phi_1(\tau) = \tau \prod_{i=2}^{s} \frac{c_i - \tau}{c_i} . \tag{3.32b}$$

Note that ϕ_0 serves the purpose of matching the value u_n, and ϕ_1, the slope $u_n{}' = f(t_n, u_n)$.

The claim of equivalence between CG and collocation methods still holds when $c_1 = 0$. The proof, which employs the basis functions $\phi_2, \ldots, \phi_s$ for the test space (3.30), and the basis functions $\widetilde{L}_j$ for the trial space (3.29), is essentially the same as that of the case $c_1 > 0$ and is omitted.

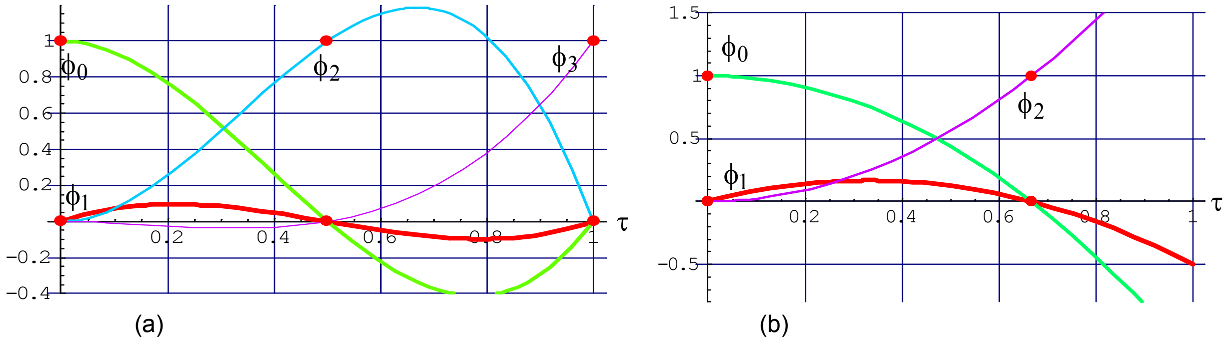


Figure 3.2.—(a) Basis functions $\phi_0,\ldots,\phi_3$ for the case of 3 Lobatto points. (b) Basis functions ϕ_0, ϕ_1, and ϕ_2 for the case of 2 left Radau points. Note that $\phi_2,\ldots,\phi_s$ form a basis for the test space defined in (3.30).

3.5 Test Spaces

We claim that using quadrature (3.2) with evaluation points c_i's, the test spaces $\mathbf{P}^{s-1}$ and $\mathbf{P}_0^s$ (or **V**) yield identical results. The reason is that on $\mathbf{P}^s$ with such a quadrature, as far as the inner product is concerned, ϕ_i defined by (3.9) has the same effect as L_i defined by (2.8) for $i = 1,\ldots,s$. Indeed, for any continuous function v, with i varies between 1 and s,

$$(v,\phi_i)_q = (v,L_i)_q = (v, b_i\delta_i)_q = b_i\, v(c_i)\,. \tag{3.33}$$

Thus, if U is a solution of the standard CG method, then since ϕ_i is in the test space $\mathbf{P}_0^s$, $(U',\phi_i)_q = (\mathrm{f},\phi_i)_q$. As shown above, we can replace ϕ_i by L_i,

$$(U',L_i)_q = (f,L_i)_q\,. \tag{3.34}$$

Noting that L_i, $i = 1,\ldots,s$ form a basis for $\mathbf{P}^{s-1}$, the claim follows.

Readers who are interested only in the main results can omit the rest of this section with no loss of continuity.

3.6 A Dilemma

The question then is: What causes the difference between the established fact that the test space $\mathbf{P}^{s-1}$ yields a more accurate solution than the test space $\mathbf{P}_0^s$ and the above claim that under the quadrature rule, the test spaces $\mathbf{P}^{s-1}$ and $\mathbf{P}_0^s$ yield identical results? To answer this question, we need to derive the solutions using exact integration. This derivation is similar to that in (Fletcher 1984); the key difference, however, is that instead of $u' = u$, we use the equation

$$u' = \lambda u\,. \tag{3.35}$$

As a consequence, we can study not only which test space yields a more accurate numerical solution, but also the stability as well as order of accuracy of the corresponding method. We will show that with $\mathbf{P}_0^s$ as the test space, an appropriate quadrature formula results in a method more accurate than that by exact integration. In other words, such a quadrature provides the cancellation needed for high accuracy whereas, with exact integration, the cancellation no longer occurs. This observation contrasts the common belief that a more accurate integration procedure results in a solution with better accuracy.

To solve the above ODE, consider the following basis functions for $\mathbf{P}^s[0,1]$

$$\varphi_j = \tau^j, \quad j = 0,\ldots,s\,. \tag{3.36}$$

Clearly, $\varphi_j(0) = 0$ for $j = 1,\ldots,s$, and $\varphi_1,\ldots, \varphi_s$ form a basis for $\mathbf{P}_0^s = \mathbf{V}$. Next, set

$$d_0 = u_n / h \tag{3.37a}$$

and

$$U = h \sum_{j=0}^{s} d_j \varphi_j(\tau) \tag{3.37b}$$

as in (3.20) and (3.21) where, $d_1,\ldots,d_s$ are to be determined. For the equation $u' = \lambda u$, the standard CG formulation, namely (3.23), implies

$$\sum_{j=0}^{s} (\varphi_i ,\ d\varphi_j / d\tau)\, d_j \ = \lambda \sum_{j=0}^{s} (\varphi_i , \varphi_j)\, d_j \ . \tag{3.38}$$

Here, for the test space $\mathbf{P}_0^s$, i varies from 1 to s; for the test space $\mathbf{P}^{s-1}$, from 0 to $s - 1$. Moving the two terms corresponding to $j = 0$ to the right hand side and the rest to the left, we have

$$\sum_{j=1}^{s} [(\varphi_i ,\ d\varphi_j / d\tau) - \lambda(\varphi_i ,\ \varphi_j)]\, d_j \ = -(\varphi_i ,\ d\varphi_0 / d\tau) + \lambda(\varphi_i ,\ \varphi_0)\ d_0 . \tag{3.39}$$

Since $\varphi_0 = 1$, the first term on the right hand side above drops out:

$$\sum_{j=1}^{s} [(\varphi_i ,\ d\varphi_j / d\tau) - \lambda(\varphi_i ,\ \varphi_j)]\, d_j \ = \lambda(\varphi_i ,\ \varphi_0)\ d_0 . \tag{3.40}$$

With exact integration, we obtain, for all i and j,

$$(\varphi_i ,\ d\varphi_j / d\tau) \ = \int_0^1 j\tau^{i+j-1} d\tau \ = \frac{j}{i+j} \tag{3.41a}$$

and

$$(\varphi_i , \varphi_j) \ = \int_0^1 \tau^{i+j} d\tau \ = \frac{1}{i+j+1} . \tag{3.41b}$$

Consider now the case $s = 2$. Assuming that $u_n = 1$, $h = 1$, then by (3.37a), $d_0 = 1$. With exact integration, the two cases concerning different test spaces are as follows.

For the test space $\mathbf{P}^{s-1} = \mathbf{P}^1$, using (3.41), the system (3.40) with $i = 0$ and $i = 1$ takes the form

$$\begin{pmatrix} 1-\frac{\lambda}{2} & 1-\frac{\lambda}{3} \\ \frac{1}{2}-\frac{\lambda}{3} & \frac{2}{3}-\frac{\lambda}{4} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = \begin{pmatrix} \lambda \\ \frac{1}{2}\lambda \end{pmatrix} , \tag{3.42}$$

The solutions are $d_1 = [6\lambda(2 - \lambda)]/(12 - 6\lambda + \lambda^2)$ and $d_2 = 6\lambda^2/(12 - 6\lambda + \lambda^2)$. Since $u_{n+1} = 1 + d_1 + d_2$,

$$u_{n+1} = \frac{12 + 6\lambda + \lambda^2}{12 - 6\lambda + \lambda^2} \quad \text{or} \quad R(z) = \frac{1 + z/2 + z^2/12}{1 - z/2 + z^2/12} . \tag{3.43}$$

Thus, the method yields the same solution as the collocation method (2.37) using two Gauss points. It is of order 4, and the error can be found in (2.38). The stability region is shown in Figure 2.3(b).

For the test space $\mathbf{P}_0^s$, using (3.41), the system (3.40) with i = 1 and i = 2 takes the form

$$\begin{pmatrix} \frac{1}{2}-\frac{\lambda}{3} & \frac{2}{3}-\frac{\lambda}{4} \\ \frac{1}{3}-\frac{\lambda}{4} & \frac{2}{4}-\frac{\lambda}{5} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2}\lambda \\ \frac{1}{3}\lambda \end{pmatrix}, \tag{3.44}$$

The solutions are $d_1 = [4\lambda(5-3\lambda)]/(20-12\lambda+3\lambda^2)$, $d_2 = 10\lambda^2/(20-12\lambda+3\lambda^2)$, and

$$u_{n+1} = \frac{20+8\lambda+\lambda^2}{20-12\lambda+3\lambda^2} \quad \text{or} \quad R(z) = \frac{20+8z+z^2}{20-12z+3z^2}\ . \tag{3.45}$$

The error is

$$E(z) \; = \; e^z - R(z) \; = \; 0.016667\,z^3 + O(z^4)\,. \tag{3.46}$$

Thus, the method is of order only 2. As |z|→∞, $R(z)$→1/3; as a result, the method is not L-stable. It is A-stable, however.

Note that if the Gauss quadrature is employed to evaluate the left hand side of (3.38), the result is the same as that by exact integration since (φ_i,d$\varphi_j/d\tau$) is of degree $2s-1$ or less. For the right hand side of (3.38), however, when $i = j = s$, $\varphi_i,\varphi_j = \tau^{2s}$, and the Gauss quadrature yields a result different from that by exact integration. Thus, for the test space $\mathbf{P}_0^s$, since the Gauss quadrature for CG results in a method equivalent to the Gauss point collocation, such a quadrature yields a CG method of order 4. For the same test space, i.e., $\mathbf{P}_0^s$, exact integration, as shown above, results in a CG method of order only 2. Thus, concerning the CG formulation, a well-matched quadrature can provide the cancellation needed for high-order accuracy, whereas exact integration may not, and the result is a less accurate solution.

4.0 Discontinuous Galerkin (DG) Methods

The DG method seeks to approximate the solution by a function which is allowed to be discontinuous across each t_n and, on each interval I_n, is a polynomial of degree $s-1$ denoted by u_h (i.e., one degree lower than that of U, the corresponding CG solution). This polynomial is determined by using the weak form of the ODE together with an integration by parts. Here, we assume that $u_h(t_n^-)$, which plays the role of u_n, is known. For the first interval, $u_h(t_0^-) = u_0$. We wish to calculate $u_h(t_{n+1}^-)$, which plays the role of u_{n+1}.

On I_n, formally (the correct expression is (4.4) below), the weak form for the equation $u' = f$ is, for any v in $\mathbf{P}^{s-1}$,

$$\int_{t_n}^{t_{n+1}} u_h{}'(t)v(t)\;dt \;\; = \;\; \int_{t_n}^{t_{n+1}} f(t,u_h(t))\,v(t)\;dt\,. \tag{4.1}$$

The above equation contains no effect from the initial condition $u_h(t_n^-) = u_n$. To involve this condition, we use integration by parts:

$$\int_{t_n}^{t_{n+1}} u_h{}'(t)v(t)\;dt \;\; = \big[u_h\,v\big]_{t_n}^{t_{n+1}} - \int_{t_n}^{t_{n+1}} u_h(t)v'(t)\;dt\;. \tag{4.2}$$

At each t_n, n = 0,…,N, the value u_h is not well defined since $u_h(t_n^-)$ is generally different from $u_h(t_n^+)$. By assuming that 'time marches forward', the value $u_h(t_n^-)$ is employed for the boundary evaluations.

Such a choice for the case of an advection (where 'the wind blows from the left') is called the 'upwind' choice, a term employed here as well. The choice adds numerical dissipation to the method whereas the 'centered' choice, i.e., $[u_h(t_n^-)+u_h(t_n^+)]/2$ generally results in no numerical dissipation. Thus, between $u_h(t_n^-)$ and $u_h(t_n^+)$, the upwind value is $u_h(t_n^-)$. Note that for conservation laws, we deal with the upwind flux; here, the flux is u itself; therefore, we deal with the upwind value. As for the test function v on I_n, the values $v(t_n)$ and $v(t_{n+1})$ involve no ambiguity. The above equation then takes the form

$$\int_{t_n}^{t_{n+1}} u_h'(t)v(t)\,dt = [u_h(t_{n+1}^-)v(t_{n+1})-u_h(t_n^-)v(t_n)] - \int_{t_n}^{t_{n+1}} u_h(t)v'(t)\,dt\,. \tag{4.3}$$

The DG formulation is the following: with the value $u_h(t_n^-)$ given, we wish to find, on I_n, a polynomial u_h in $\mathbf{P}^{s-1}$ that satisfies, for all v in $\mathbf{P}^{s-1}$,

$$[u_h(t_{n+1}^-)v(t_{n+1})-u_h(t_n^-)v(t_n)] - \int_{t_n}^{t_{n+1}} u_h(t)v'(t)\,dt = \int_{t_n}^{t_{n+1}} f(t,u_h(t))\,v(t)\,dt\,. \tag{4.4}$$

As in (LeSaint and Raviart 1974), we need another integration by parts. On I_n, since u_h is a polynomial, the values at the two boundaries are respectively identical to $u_h(t_n^+)$ and $u_h(t_{n+1}^-)$. Consequently,

$$\int_{t_n}^{t_{n+1}} u_h(t)v'(t)\,dt = [u_h(t_{n+1}^-)v(t_{n+1})-u_h(t_n^+)v(t_n)] - \int_{t_n}^{t_{n+1}} u_h'(t)v(t)\,dt\,.$$

Note that this equation involves no upwinding. Substitute it into (4.4), i.e., after integrating by parts twice, we obtain the following DG formulation: with $u_h(t_n^-)$ given, find u_h in $\mathbf{P}^{s-1}$ that satisfies, for all v in $\mathbf{P}^{s-1}$,

$$-[u_h(t_n^-)-u_h(t_n^+)]v(t_n) + \int_{t_n}^{t_{n+1}} u_h'(t)v(t)\,dt = \int_{t_n}^{t_{n+1}} f(t,u_h(t))\,v(t)\,dt\,. \tag{4.5}$$

Between the above and the formal expression (4.1), the difference is the 'correction' term $-[u_h(t_n^-)-u_h(t_n^+)]v(t_n)$. This term serves the purpose of enforcing the upwind value $u_h(t_n^-)$ at the left boundary of I_n. The upwind value $u_h(t_{n+1}^-)$ at the right boundary is built in as shown in (4.4). It is also the solution we wish to calculate.

As an example, we solve $u'(t) = 2t$ with initial condition $u(0) = 0$ via the above DG formulation. The exact solution is obvious: $u(t) = t^2$. We use a linear element u_h, i.e., $s = 2$. For simplicity of notation, set $a = t_n$ and $b = t_{n+1}$. Assuming that $u_h(a^-)$ is exact, i.e., $u_h(a^-) = a^2$, we wish to calculate u_h on the interval $I_n = [a,b]$. First, since u_h is linear, the problem reduces to calculating $u_h(b^-)$and $u_h(a^+)$. Next, noting that $h = b - a$, on the interval (a,b), $u_h' = [u_h(b^-) - u_h(a^+)]/h$. As a result, (4.5) implies

$$-[a^2-u_h(a^+)]v(a) + \frac{1}{h}[u_h(b^-)-u_h(a^+)]\int_a^b v(t)\,dt = \int_a^b 2t\ v(t)\,dt\,. \tag{4.6}$$

Set $v = 1$, the above results in

$$-[a^2-u_h(a^+)] + [u_h(b^-)-u_h(a^+)] = b^2-a^2\,.$$

Thus, the solution at $t_{n+1} = b$ is

$$u_h(b^-)=b^2\,.$$

This solution is exact. Now, setting $v = t$ for (4.6), it follows that

$$-[a^2 - u_h(a^+)]\ a\ + \frac{1}{(b-a)}[b^2 - u_h(a^+)]\ \frac{1}{2}(b^2 - a^2)\ =\ \frac{2}{3}[b^3 - a^3]\,.$$

Consequently,

$$u_h(a^+) = \frac{1}{3}(2a^2 + 2ab - b^2)\,.$$

Or,

$$u_h(a^+)\ =\ a^2 - \frac{(b-a)^2}{3}\ =\ a^2 - \frac{h^2}{3}\ .$$

This solution is a poor approximation to the exact solution $u(a) = a^2$. However, observe that at the Radau point (away from the right boundary b), namely, $t = a + h/3$, the linear function u_h takes on the value

$$u_h(a + h/3) = (a + h/3)^2\,,$$

which is again exact. See Figure 4.1. The above observation is consistent with the next assertion.

4.1 Equivalence Between DG and Radau IIA Methods

If the right Radau quadrature is employed, then the DG method via (4.5) is identical to the collocation method using s right Radau points as collocation points, i.e., the Radau IIA method. (As a consequence, due to the equivalence result in the previous section, with such a quadrature, the DG, CG, and collocation methods become identical.)

The key idea for the proof is the following. Since u_h has a discontinuity or a jump across t_n, the value $u_h'(t_n)$ is not well defined (it includes the derivative of a jump, which involves the Dirac delta function). Viewing this from another angle, the derivative $u_h'(t)$ on I_n incorporates no effect from the known boundary value $u_h(t_n^-)$. As a result, u' for the ODE will be evaluated not by u_h' but by U' where U, a function to be constructed, is continuous across all t_n, and $U(t_n) = u_h(t_n^-)$. The continuous function U will be obtained by adding a correction to u_h.

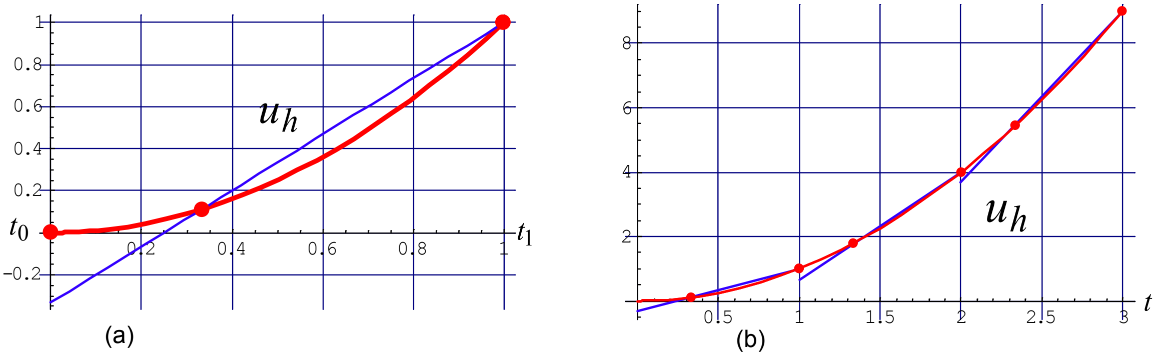


Figure 4.1.—The piecewise linear (s = 2) DG solution for $u'(t) = 2t$ with initial condition $u(0) = 0$ and $h = 1$. (a) After one step, the solution u_h is exact at the Radau points $t = 1$ and $t = 1/3$; (b) the solution after three steps (note the different scales).

To define U, we need the following correction function g, which serves the purpose of eliminating the trial function v (collocation methods do not use trial functions). This correction function, introduced for conservation laws in (Huynh 2006), is where the current approach differs from those in the literature. It leads to a DG formulation using the differential instead of integral equations, and results in a simplified version of the DG method.

To deal with the correction term on the left hand side of (4.5), in view of the expression $\int_{t_n}^{t_{n+1}} u_h{}'(t)v(t)\ dt$, we ask the following question: can we define a polynomial g on I_n which possesses the property that for all v in $\mathbf{P}^{(s-1)}$,

$$\int_{t_n}^{t_{n+1}} g'(t)v(t)\ dt \ = \ -\ v(t_n)\,. \tag{4.7}$$

Such a function will help eliminate the correction term and the test function. The answer to the above question is positive: applying integration by parts to the left hand side of (4.7), we have

$$\int_{t_n}^{t_{n+1}} g'(t)v(t)\ dt \ = \ g(t_{n+1})v(t_{n+1}) - g(t_n)v(t_n) \ - \ \int_{t_n}^{t_{n+1}} g(t)v'(t)\ dt\,. \tag{4.8}$$

Thus, for (4.7) to hold, it suffices that

$$g(t_n) = 1\,,\ \ g(t_{n+1}) = 0\,, \tag{4.9}$$

and

$$\int_{t_n}^{t_{n+1}} g(t)v'(t)\ dt \ = \ 0 \ \text{ for all } \ v \ \text{ in } \ \mathbf{P}^{s-1}\,. \tag{4.10}$$

Since v is in $\mathbf{P}^{s-1}$, v' is in $\mathbf{P}^{s-2}$; in addition, as v spans $\mathbf{P}^{s-1}$, v' spans $\mathbf{P}^{s-2}$. Consequently, the above is equivalent to g being orthogonal to $\mathbf{P}^{s-2}$:

$$\int_{t_n}^{t_{n+1}} g(t)w(t)\ dt \ = \ 0 \ \text{ for all } \ w \ \text{ in } \ \mathbf{P}^{s-2}\,. \tag{4.11}$$

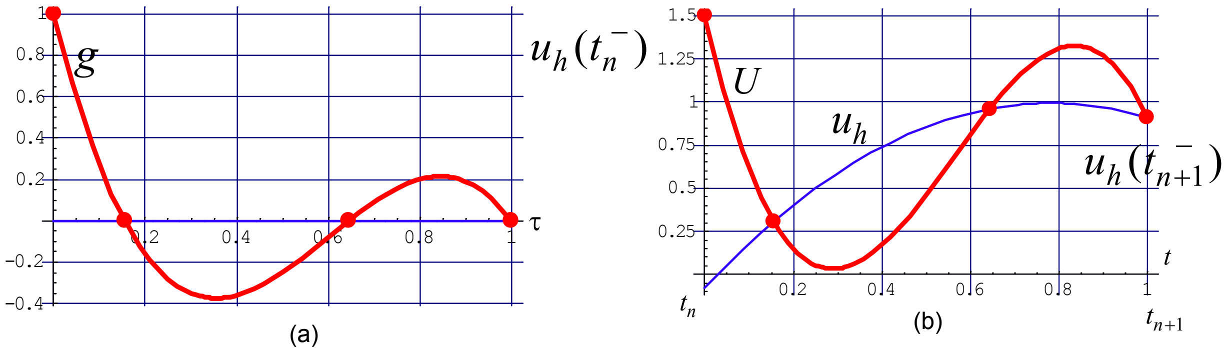
That is,

$$\int_{t_n}^{t_{n+1}} \ g(t)\,(t-t_n)^j\ dt \ = \ 0 \ \text{ for } \ j = 0,1,...,s-2\,. \tag{4.12}$$

In other words, g approximates the function 0 in a least square sense. The requirement (4.11) or (4.12) provides $s - 1$ conditions to define g. Together with the two conditions in (4.9), we have a total of $s + 1$ conditions. Therefore, we require g to be of degree s.

On the interval I_n, the right Radau polynomial of degree s satisfies conditions (4.11) and (4.9) (see Fig. 4.2(a)). Let g be this right Radau polynomial. Then g can also defined by $g(t_n) = 1$ and g vanishes at the s right Radau points on I_n. (For the relation between Legendre and Radau polynomials, see the Appendix.) Next, set

$$U \ = \ u_h \ + \ [u_h(t_n^-) - u_h(t_n^+)]\,g\,. \tag{4.13}$$



(a) (b)

Figure 4.2.—The DG method with s = 3. (a) The correction function g is the Radau polynomial of degree s on I = [0,1]. It is defined by: $g(0)$ =1, $g(1)$ = 0, , and the projection of g onto $\mathbf{P}^{s-2}$ is the zero function. (b) The polynomial u_h is of degree s – 1 and is generally discontinuous across t_n (thin curve); the polynomial U is of degree s and is continuous across t_n (thick curve); U is defined by: $U(t_n) = u_h(t_n^-)$, $U(t_{n+1}) = u_h(t_{n+1}^-)$, and the projections of U and u_h onto P^{s-2} are the same.

Then, U is of degree s (whereas, u_h is of degree s – 1) as shown in Figure 4.2(b). Using (4.9), we obtain

$$U(t_n^+) = u_h(t_n^-) \tag{4.14a}$$

and

$$U(t_{n+1}^-) = u_h(t_{n+1}^-)\,. \tag{4.14b}$$

Since the above two equations hold for all n, replacing n + 1 by n in the latter, we have $U(t_n^-) = u_h(t_n^-)$. Therefore, by (4.14a),

$$U(t_n^+) = U(t_n^-) = u_h(t_n^-)\,.$$

That is, U is continuous across all t_n. Next, by (4.13),

$$U' = u_h' + [u_h(t_n^-) - u_h(t_n^+)]g'\,.$$

As a consequence, using (4.7),

$$\int_{t_n}^{t_{n+1}} U'(t)\ v(t)\ dt = -[u_h(t_n^-) - u_h(t_n^+)]v(t_n) + \int_{t_n}^{t_{n+1}} u_h'(t)v(t)\ dt\,.$$

Therefore, by (4.5),

$$\int_{t_n}^{t_{n+1}} U'(t)\ v(t)\ dt = \int_{t_n}^{t_{n+1}} f(t,u_h(t))\ v(t)\ dt\ . \tag{4.15}$$

Note that there is no correction term in the above weak formulation. Also note that whereas U appears on the left hand side, u_h appears on the right hand side. We need to replace u_h by U. To this end, let $t_{n,j}$, j = 1,…,s, be the s right Radau points. Since g vanishes at these points, by (4.13),

$$U(t_{n,j}) = u_h(t_{n,j})\,. \tag{4.16}$$

We now use the right Radau quadrature. If the weights of the quadrature are b_j, then (4.15) implies

$$\sum_{j=1}^{s} b_j U'(t_{n,j})\, v(t_{n,j}) \;=\; \sum_{j=1}^{s} b_j f(t_{n,j}, u_h(t_{n,j}))\, v(t_{n,j}) .$$

Let v be one of the basis functions L_i, which are the Lagrangian polynomials of degree $s - 1$ defined by (2.8) for the right Radau points. Then

$$b_i U'(t_{n,i}) \;=\; b_i f(t_{n,i}, u_h(t_{n,i})) .$$

Therefore, by (4.16), for $i = 1,\ldots,s$,

$$U'(t_{n,i}) \;=\; f(t_{n,i}, U(t_{n,i})) . \tag{4.17}$$

The fact that U is of degree s, (4.14a), and the above shows that U is the solution by collocation method with the right Radau points as collocation points. This completes the proof.

4.2 Remarks

The following remarks are in order.

The CG and DG solutions relate to each other as follows. Assume that the right Radau quadrature is employed, and let $u_n = u_h(t_n^-)$. For the CG method, the solution polynomial U_{CG} is of degree s and determined by u_n and the values $u_{n,1},\ldots,u_{n,s}$ at the s right Radau points. For the DG method, the polynomial u_h is of degree $s - 1$ and determined by the values $u_{n,1},\ldots,u_{n,s}$ but not u_n. See Figure 4.2(b). The polynomial $U_{DG} = U$ defined by (4.13) above is identical to U_{CG}. Also note that with the right Radau quadrature, since $c_s = 1$, we have $t_{n,s} = t_{n+1}$, and

$$U_{DG}(t_{n+1}) = u_{n,s} = u_h(t_{n+1}^-) = U_{CG}(t_{n+1}) . \tag{4.18}$$

Next, we discuss accuracy. Since u_h is of degree $s - 1$, we expect $u_h(t)$ to have an error of $O(h^s)$. On the other hand, U_{CG} is of degree s; therefore, its error on I_n is $O(h^{s+1})$. Note that u_h and U_{CG} take on the same values at $t_{n,1},\ldots,t_{n,s}$, namely, $u_{n,1},\ldots,u_{n,s}$, respectively (see Fig. 4.2(b)). As a result, for the DG method, as values of u_h, the solutions at the interior right Radau points, namely, $u_{n,1},\ldots,u_{n,s-1}$, have errors $O(h^{s+1})$, one order higher than expected. As for $u_{n,s} = u_h(t_{n+1}^-)$, it has an error of $O(h^{2s})$. For example, with $s = 2$, i.e., a piecewise linear u_h, the value $u_{n,2} = u_h(t_{n+1}^-)$ is accurate to $O(h^4)$, and the method is of order 3, whereas, $u_{n,1} = u_h(t_{n,1})$ has an error of $O(h^3)$, i.e., it is as good as a parabolic approximation.

The next remark concerns the function g defined by (4.7), i.e., for all v in $\mathbf{P}^{s-1}$,

$$\int_{t_n}^{t_{n+1}} g'(t) v(t)\, dt \;=\; -\, v(t_n) .$$

That is, for any v in $\mathbf{P}^{s-1}$, the projection of v onto the line spanned by g' is the value $-v(t_n)$. Using distribution theory, we can view g in the following manner. The above implies that on the subspace $\mathbf{P}^{s-1}$, the function $-g'$ is equivalent to the Dirac delta function at $t = t_n$:

$$-g' \approx \delta(t_n) .$$

Since the integral of the Dirac delta function is the unitstep (or Heaviside) function,

$$-g + \text{constant} \;\approx\; H(t - t_n) = \begin{cases} 0 & \text{if} \quad t \le t_n, \\ 1 & \text{if} \quad t_n < t \le t_{n+1}. \end{cases}$$

By choosing the constant to be 1, we obtain the result that g approximates the 'step-down function':

$$g \approx \begin{cases} 1 & \text{if} \quad t \le t_n, \\ 0 & \text{if} \quad t_n < t \le t_{n+1}. \end{cases}$$

That is, $g(t_n) = 1$, and g approximates the zero function on I_n.

4.3 A Generalization for DG Formulation

The DG formulation above can be generalized. One way is to use, for each boundary t_n, a quantity that blends the upwind value $u_h(t_n^-)$ and the downwind value $u_h(t_n^+)$. Thus, for the left boundary evaluation in (4.3), the value to be employed is, with α a parameter between 0 and 1,

$$u_n \;=\; \alpha u_h(t_n^-) \;+\; (1-\alpha) u_h(t_n^+). \tag{4.19}$$

If α = 1, then $u_n \;=\; u_h(t_n^-)$; α = 1/2, then $u_n \;=\; [u_h(t_n^-) + u_h(t_n^+)]/2$; and, α = 0, then $u_n \;=\; u_h(t_n^+)$. This approach to generalization was presented by Delfour, Hager, and Trochu (1981). For the right boundary t_{n+1}, in addition to $u_h(t_{n+1}^-)$, we also need $u_h(t_{n+1}^+)$. The use of such a value results in a complicated method.

Here, we introduce a generalization that does not employ $u_h(t_{n+1}^+)$ by formulating the CG method in the form of DG. Suppose u_n is given, and u_{n+1} is to be calculated. Consider the CG method using, for the moment, the Gauss quadrature whose evaluation points are the Gauss points denoted by $t_{n,1},\ldots,t_{n,s}$ where $t_n < t_{n,1} < \ldots\; t_{n,s} < t_{n+1}$. As shown in Section 3.0, the solution polynomial $U = U_{CG}$ is identical to the collocation solution with collocation points $t_{n,1},\ldots,t_{n,s}$. That is, U is of degree s and interpolates u_n and the s collocation values $u_{n,1},\ldots,u_{n,s}$. Consider the polynomial of degree $s - 1$ interpolating $u_{n,1},\ldots,u_{n,s}$ (but not u_n) denoted by u_h and shown in Figure 4.3(b). This polynomial plays the same role as u_h of the DG method. In addition, on $I = [0,1]$, let g be a polynomial of degree s that takes on the value 1 at $\tau = 0$ and vanishes at $\tau_{n,1},\ldots,\tau_{n,s}$, i.e., g is the function ϕ_0 defined in (3.9) and shown in Figure 4.3(a) (here, g is the Legendre polynomial on [0,1]). The polynomial $U = U_{CG}$ can be written as, with $t = t_n + \tau h$,

$$U(t) \;=\; u_h(t) + \;[u_n - u_h(t_n^+)]\; g((t-t_n)/h). \tag{4.20}$$

The solution we are seeking is $u_{n+1} = U(t_{n+1})$,

$$u_{n+1} \;=\; U(t_{n+1}) \;=\; U(t_{n+1}^-) \;=\; u_h(t_{n+1}^-) + \;[u_n - u_h(t_n^+)]\; g(1). \tag{4.21}$$

That is, u_{n+1} is obtained by adding to the value $u_h(t_{n+1}^-)$ a correction term which depends only on the jump $u_n - u_h(t_n^+)$ at the left boundary. The question is: if U is the CG solution, then what is the corresponding DG formulation for u_h? Using the test space $\mathbf{P}^{s-1}$ in the CG formulation, for any v in $\mathbf{P}^{s-1}$,

$$(U', v) = (f, v). \tag{4.22}$$

Applying integration by parts to the left hand side above,

$$(U', v) \;=\; u_{n+1}\, v(t_{n+1}) - u_n\, v(t_{n+1}) - \int_{t_n}^{t_{n+1}} U(t)\, v'(t) dt. \tag{4.23}$$

Next, by (4.20),

$$\int_{t_n}^{t_{n+1}} U(t)\,v'(t)dt \;=\; \int_{t_n}^{t_{n+1}} \{u_h(t) + [u_n - u_h(t_n^+)]\; g((t-t_n)/h)\}\; v'(t)dt\,.$$

Using the fact that g is orthogonal to $\mathbf{P}^{s-2}$ (in fact, g is orthogonal to $\mathbf{P}^{s-1}$),

$$\int_{t_n}^{t_{n+1}} U(t)\,v'(t)dt \;=\; \int_{t_n}^{t_{n+1}} u_h(t)\,v'(t)dt\,.$$

where, again, v is in $\mathbf{P}^{s-1}$ Thus, (4.23) implies

$$(U',v) \;=\; u_{n+1}\,v(t_{n+1}) - u_n\,v(t_n) - \int_{t_n}^{t_{n+1}} u_h(t)\,v'(t)dt\,. \tag{4.24}$$

Applying integration by parts on the last term above, we obtain

$$\int_{t_n}^{t_{n+1}} u_h(t)\,v'(t)dt \;=\; u_h(t_{n+1}^-)\,v(t_{n+1}) - u_h(t_n^+)\,v(t_n) - \int_{t_n}^{t_{n+1}} u_h{}'(t)\,v(t)dt\,.$$

The above two equations imply

$$(U',v) \;=\; [u_{n+1} - u_h(t_{n+1}^-)]\,v(t_{n+1}) - [u_n - u_h(t_n^+)]\,v(t_n) \;+\; \int_{t_n}^{t_{n+1}} u_h{}'(t)\,v(t)dt\,. \tag{4.25}$$

Next, by (4.21),

$$u_{n+1} - u_h(t_{n+1}^-) = [u_n - u_h(t_n^+)]\; g(1)\,.$$

Substitute the above into (4.25),

$$(U',v) \;=\; [u_n - u_h(t_n^+)]\; g(1)v(t_{n+1}) - [u_n - u_h(t_n^+)]\,v(t_n) \;+\; \int_{t_n}^{t_{n+1}} u_h{}'(t)\,v(t)dt\,.$$

Or

$$(U',v) \;=\; [u_n - u_h(t_n^+)]\,[\,g(1)v(t_{n+1}) - v(t_n)\,] \;+\; \int_{t_n}^{t_{n+1}} u_h{}'(t)\,v(t)dt\,. \tag{4.26}$$

Therefore, by (4.22),

$$[u_n - u_h(t_n^+)]\,[\,g(1)v(t_{n+1}) - v(t_n)\,] \;+\; (u_h{}',v) \;=\; (f,v)\,. \tag{4.27}$$

Note that the above reduces to the DG formulation (4.5) if $g(1) = 0$, i.e., if $\tau_s = 1$.

This completes the generalization of the DG method. Also note that such a DG formulation is more involved than the CG formulation.

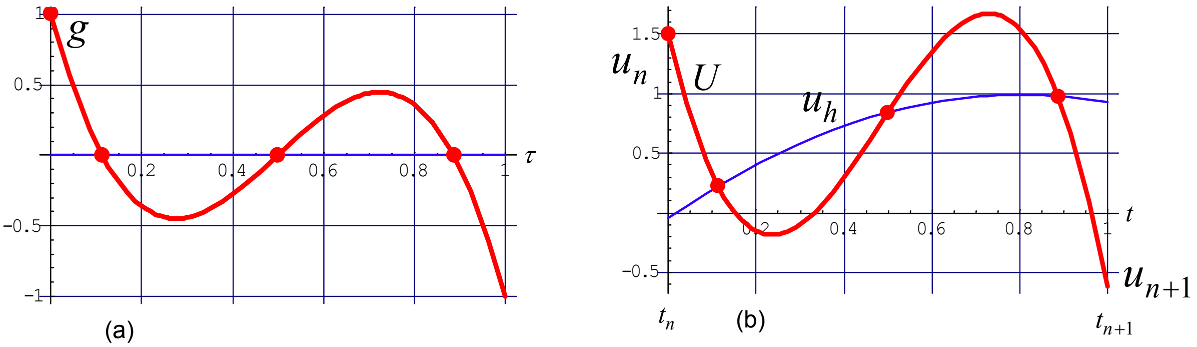


Figure 4.3.—A generalization DG method for $s = 3$. (a) Here, the quadrature points are the Gauss points. The function g on $I = [0,1]$ is of degree s and defined by the conditions that $g(0) = 1$, and g vanishes at the quadrature points; for this case, g is the Legendre polynomial of degree s. (b) The polynomial u_h is of degree $s – 1$ and interpolates the values $u_{n,1},\ldots,u_{n,s}$ at the s quadrature points.

5.0 Conclusions and Discussion

In summary, we studied the numerical solutions for ordinary differential equations using the collocation, continuous Galerkin (CG), and discontinuous Galerkin (DG) methods. It was shown that if a quadrature formula using s evaluation points is employed, then the CG method is identical to the collocation method using quadrature points as collocation points. Furthermore, if the quadrature formula is the right Radau one, then the DG and CG methods also become identical, and they reduce to the Radau IIA collocation method. We also present a generalization of DG that yields a method identical to CG and collocation with arbitrary collocation points. As a result of these findings, both the CG and DG methods can be formulated using the differential instead of integral form.

In addition to clarifying the relations among these methods, the equivalence results can be employed for the high-order time discretization of conservation laws. It can also be applied to simplify the time discretization of the space-time DG methods.

Appendix A.—Radau Polynomials

Since the Radau polynomials are not widely known, we derive them below. The Radau polynomial of degree s, which is determined by conditions (4.9) and (4.11) (or (4.12)), is defined here by using the Legendre polynomials. The advantage of such a definition is that it clarifies the relation between these polynomials as well as the orthogonality properties (to the various space of polynomials). Instead of the interval [0,1], to be consistent with the standard Legendre polynomials, we employ the interval [–1,1]. If g is a polynomial on [–1,1], then the corresponding polynomial on [0,1] can easily be obtained by, for η on [0,1], $G(\eta) = g(2\eta - 1)$.

We now focus on $I = [-1,1]$. For any integer m ≥ 0, let $\mathbf{P}_m$ be the space of polynomials of degree m or less. Then $\mathbf{P}_m$ is a vector space of dimension $m + 1$. A polynomial v is orthogonal to $\mathbf{P}_m$ if, for each l, $0 \le l \le m$,

$$(v, \xi^l) = \int_{-1}^{1} v(\xi)\,\xi^l\, d\xi = 0\,.$$

Clearly, the criterion of being orthogonal to $\mathbf{P}_m$ provides $m + 1$ conditions (or equations).

For $k = 0,1,2,\ldots$, let the Legendre polynomial P_k be defined on $I = [-1,1]$ as the unique polynomial of degree k satisfying the following $k + 1$ conditions: it is orthogonal to $\mathbf{P}_{k-1}$ and $P_k(1) = 1$. The Legendre polynomials are given by a recurrence formula (e.g., Hildebrand 1987):

$$P_0 = 1, \quad P_1 = \xi,$$

and, for $k \ge 2$,

$$P_k(\xi) = \frac{2k-1}{k}\,\xi P_{k-1}(\xi) - \frac{k-1}{k}\,P_{k-2}(\xi)\,. \tag{A.1}$$

The first few Legendre polynomials are plotted in Figure A.1(a). Useful properties of the Legendre polynomials are listed below. If $k > m$, then P_k is orthogonal to $\mathbf{P}_m$. Next, P_k is an even function (involving only even powers of ξ) for even k, and an odd function for odd k. For all k, the values at the boundaries are

$$P_k(-1) = (-1)^k\,, \tag{A.2a}$$

$$P_k(1) = 1\,. \tag{A.2b}$$

The derivative values at the end points are

$$P_k{}'(-1) = (-1)^{k-1} k(k+1)/2\,, \tag{A.3a}$$

$$P_k{}'(1) = k(k+1)/2\,. \tag{A.3b}$$

In addition, for $k \ne l$, $(P_k, P_l) = 0$. Finally, the zeros of P_k are the k Gauss points on [–1,1].

The right Radau polynomial of degree k ($k \ge 1$) is defined by

$$R_{R,k} \;=\; \frac{(-1)^k}{2}(P_k - P_{k-1})\,. \tag{A.4}$$

The letter R stands for 'Radau' and the subscript R for 'right'. The factor $(-1)^k$ is nonstandard and is needed so that (A.6a) below holds. The first few Radau polynomials are plotted in Figure A.1(b). The above definition implies that $R_{R,k}$ is orthogonal to $\mathbf{P}_{k-2}$. In addition, by (A.2),

$$R_{R,k}(-1)=1\,. \tag{A.5a}$$

$$R_{R,k}(1)=0 \tag{A.5b}$$

It is important to note that $R_{R,k}$, which is of degree k, is defined by the above two conditions and the $k-1$ conditions that it is orthogonal to $\mathbf{P}_{k-2}$. This definition of the Radau polynomial shows that it approximates the zero function in the sense of least squares. At the two boundaries, by using (A.3),

$$R_{R,k}{}'(-1)=-k^2/2\,, \tag{A.6a}$$

$$R_{R,k}{}'(1)=(-1)^{k-1}k/2\,. \tag{A.6b}$$

The zeros of the Radau polynomial $R_{R,k}$ are the k right Radau points.

For later use, the Lobatto polynomial of degree k ($k \geq 1$) is defined by

$$\mathrm{Lo}_k \;=\; P_k - P_{k-2}\,. \tag{A.7}$$

The zeros of the Lobatto polynomial of degree k are the k Lobatto points; they include the two boundaries ±1.

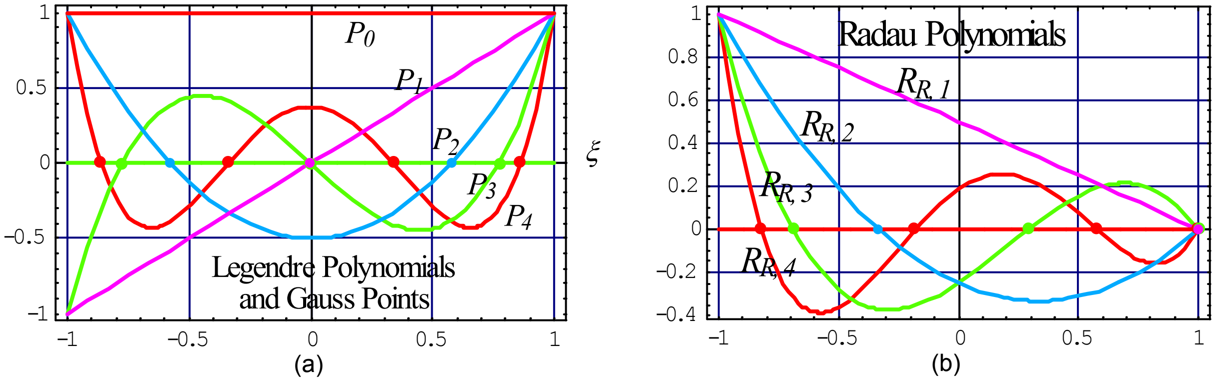


Figure A.1.—(a) Legendre polynomials and (b) right Radau polynomials.

Appendix B.—Examples of Collocation Methods

In the following examples of collocation methods, we make several uncommon observations, which, hopefully, will shed some light on the matter. First, we write the collocation method in IRK form. Then we calculate the solution for the ODE $u' = \lambda u$ by assuming that

$$u_n = 1 \quad \text{and} \quad h = 1 . \tag{B.1}$$

The assumption $h = 1$ means, loosely put, h is absorbed into z where $z = \lambda h = \lambda$ and, for order of accuracy calculation, halving the step size takes the form of halving z. As for the stability function,

$$R(z) = u_{n+1} . \tag{B.2}$$

The case of 2 Lobatto points corresponds to a Butcher array shown in Table B.1(a). The resulting collocation method reduces to the Trapezoidal Rule:

$$u_{n+1} = u_n + \tfrac{1}{2} h (f_n + f_{n+1}) . \tag{B.3}$$

The stability function and its error are, respectively,

$$R(z) = \left(1+\tfrac{1}{2}z\right)\Big/\left(1-\tfrac{1}{2}z\right), \text{ and } E(z) = e^z - R(z) = -\frac{1}{12}z^3 + O(z^4) . \tag{B.4}$$

Thus, the method using 2 Lobatto points is of order 2. It is also A-stable as can be seen by Figure B.1(a).

TABLE B.1.—BUTCHER ARRAYS FOR 2-POINT COLLOCATION METHODS.

0	0	0
1	1/2	1/2
	1/2	1/2

(a) Lobatto

$1/2 - \sqrt{3}/6$	$1/4$	$1/4 - \sqrt{3}/6$
$1/2 + \sqrt{3}/6$	$1/4 + \sqrt{3}/6$	$1/4$
	1/2	1/2

(b) Gauss

For the case of two Gauss points, the Butcher array is shown in Table B.1(b). Using assumption (B.1),

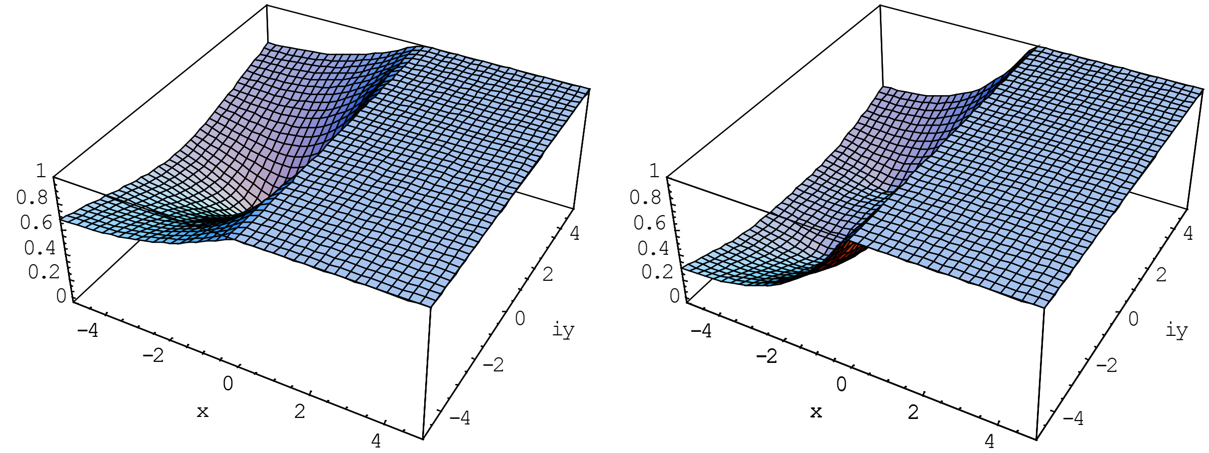
$$u_{n,1} = \frac{1-\lambda\sqrt{3}/6}{1-\lambda/2+\lambda^2/12} \quad \text{and} \quad u_{n,2} = \frac{1+\lambda\sqrt{3}/6}{1-\lambda/2+\lambda^2/12} . \tag{B.5}$$

The solution at t_{n+1} and the stability function are, respectively,

$$u_{n+1} = \frac{1+\lambda/2+\lambda^2/12}{1-\lambda/2+\lambda^2/12} \quad \text{and} \quad R(z) = \frac{1+z/2+z^2/12}{1-z/2+z^2/12} \tag{B.6}$$

At the collocation points, $u_{n,1}$ and $u_{n,2}$ respectively approximate e^{zc_1} and e^{zc_2} ; the errors are,

$$\begin{aligned} e_{n,1} &= e^{zc_1} - u_{n,1} \approx 0.008019z^3 + 0.005167z^4 + 0.002008z^5 + O(z^6), \quad \text{and} \\ e_{n,2} &= e^{zc_2} - u_{n,2} \approx -0.008019z^3 - 0.005167z^4 + 0.002008z^5 + O(z^6). \end{aligned} \tag{B.7}$$



(a) 2 Lobatto points

(b) 2 Gauss points

Figure B.1.—Plot of $|R(z)|$ for 2-point collocation methods where $z = x + iy$ varies on the complex plane. On the region $\{z;\ |R(z)| > 1\}$, the plot is cut off (flat part); the region $z;\ |R(z)| \le 1\}$ is the stability domain S. The two methods are A-stable: (a) 2 Lobatto points and (b) 2 Gauss points.

As expected, these errors are $O(z^3)$. The cancellation of errors yields a solution u_{n+1} with an error of $O(z^5)$:

$$E(z) = e^z - R(z) = z^5/720 + O(z^6) = 0.001389z^5 + O(z^6) \,. \tag{B.8}$$

Thus, the (2-stage) collocation method using 2 Gauss points is of order 4. For comparison, the leading term of the error of the 4-stage explicit RK method is $z^5/5! = z^5/120 = 0.008333z^5$, which is six times larger. Concerning stability, the method using two Gauss points is A-stable as can be seen in Figure B.1(b).

Note that with two points (or two evaluations), the Gauss quadrature is exact for a cubic (its degree of precision is 3). Next, if the exact u' is a cubic, then the exact u is a quartic and, in this case, u_{n+1} is exact. Since u_{n+1} is exact for the case of a quartic u, for the general case, u_{n+1} has an error of $O(h^5)$, a fact consistent with (B.8).

For the case of 2 left Radau points, the Butcher array is shown in Table B.2(a). Using assumption (B.1),

$$u_{n,1} = 1 \qquad \text{and} \qquad u_{n,2} = (1+\lambda/3)/(1-\lambda/3) \,. \tag{B.9}$$

The solution at t_{n+1} and the stability function are, respectively,

$$u_{n+1} = \frac{1+2\lambda/3+\lambda^2/6}{1-\lambda/3} \quad \text{and} \quad R(z) = \frac{1 + 2z/3 + z^2/6}{1-z/3} \tag{B.10}$$

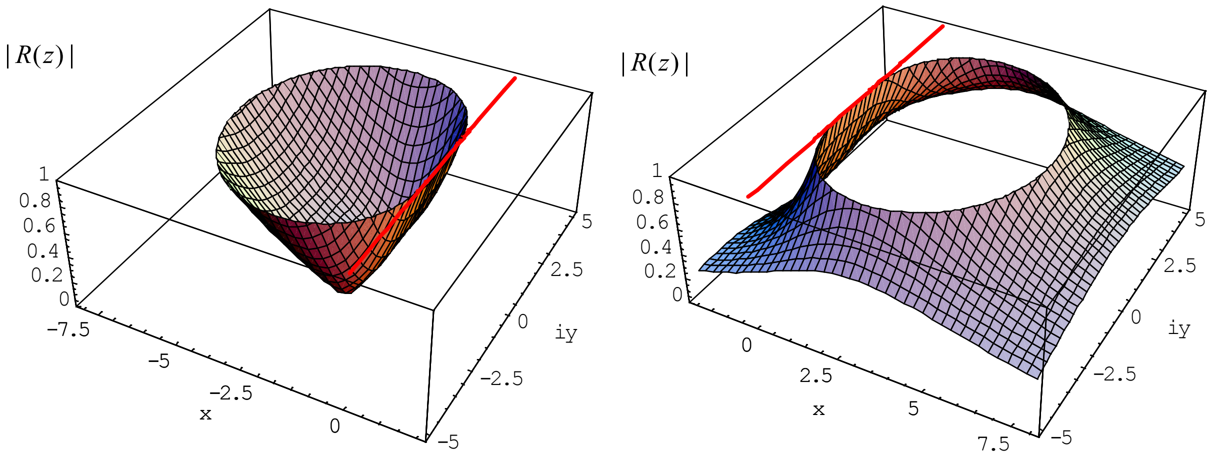
At $c_2 = 2/3$, $u_{n,2}$ approximates $e^{2z/3}$ with an error of

$$e_{n,2} = e^{2z/3} - u_{n,2} = -(2/81)z^3 + O(z^4) \,. \tag{B.11}$$

The solution u_{n+1} approximates e^z with an error of

$$E(z) = e^z - R(z) = -z^4/72 + O(z^5) \,. \tag{B.12}$$

That is, $u_{n,2}$ has an error of $O(z^3)$, whereas u_{n+1}, $O(z^4)$. Thus, the collocation method using 2 left Radau points is of order 3. Concerning stability, this method is *not* A-stable as can be seen in Figure B.2(a).



(a) 2 left Radau points

(b) 2 right Radau points

Figure B.2.—Plot of |*R*(*z*)| for 2-point collocation methods where ***z*** = *x* + *iy* varies on the complex plane. On the region {*z*; |*R*(*z*)| > 1}, the plot is cut off; the region {*z*; |*R*(*z*)| ≤ 1} is the stability domain *S*. (a) The left Radau method is not A-stable. (b) The right Radau method is L-stable, thus, also A-stable.

TABLE B.2.—BUTCHER ARRAYS FOR 2-POINT COLLOCATION METHODS

0	0	0
2/3	1/3	1/3
	1/4	3/4

(a) 2 left Radau points

1/3	5/12	−1/12
1	3/4	1/4
	3/4	1/4

(b) 2 right Radau points

For the case of 2 right Radau points, the Butcher array is shown in Table 2.2(b). Using assumption (B.1),

$$u_{n,1} = \frac{1-\lambda/3}{1-2\lambda/3+\lambda^2/6} \quad \text{and} \quad u_{n,2} = \frac{1+\lambda/3}{1-2\lambda/3+\lambda^2/6}. \tag{B.13}$$

Since $c_2 = 1$, we have $u_{n,2} = u_{n+1}$, and

$$R(z) = \frac{1+z/3}{1-2z/3+z^2/6}. \tag{B.14}$$

At $c_1 = 1/3$, $u_{n,1}$ approximates $e^{z/3}$ with an error of

$$e_{n,1} = e^{z/3} - u_{n,1} = (2/81)z^3 + O(z^4). \tag{B.15}$$

The solution u_{n+1} approximates e^z with an error of

$$E(z) = e^z - R(z) = e^z - u_{n+1} = z^4/72 + O(z^5). \tag{B.16}$$

That is, $u_{n,1}$ has an error of $O(z^3)$, whereas, u_{n+1}, $O(z^4)$. Thus, the (2-stage) collocation method using 2 right Radau points is of order 3. For comparison, the error of the standard 3-stage explicit Runge-Kutta (RK) method is $\lambda^4/(4!) = \lambda^4/24$, which is three times larger. Concerning stability, this Radau method is L-stable as can be seen in Figure B.2(b). Collocation methods using right Radau points are called Radau IIA methods (see, e.g., Lambert 1991 or Hairer and Wanner 1991).

Note that the errors for the left Radau case in (B.11) and (B.12) and those for the right Radau case in (B.15) and (B.16) are of opposite sign and same magnitude. Also note that the curve {*z*; |*R*(*z*)| = 1} for the left Radau method in Figure B.2(a) and that for the right Radau method in Figure B.2(b) are reflections of each other about the origin of the complex plane. Indeed, the equations for these curves are, respectively, by (B.10) and (B.14),

$$|1+\tfrac{2}{3}z+\tfrac{1}{6}z^2| = |1-\tfrac{1}{3}z| \quad \text{and} \quad |1-\tfrac{2}{3}z+\tfrac{1}{6}z^2| = |1+\tfrac{1}{3}z|. \tag{B.17}$$

Replacing z by $-z$ for the equation on the left, we obtain that on the right, and the observation follows. The above observations on the errors as well as the symmetry of the curves $\{z; |R(z)| = 1\}$ between the left and right Radau methods hold for the general case of s collocation points as well.

Finally, loosely put, the more biased toward t_{n+1} are the collocation points (e.g., right Radau points), the more stable is the method.

References

S. Adjerid, K.D. Devine, and J.E. Flaherty and L. Krivodonova, “A posteriori error estimation for discontinuous Galerkin solutions of hyperbolic problems,” Computer Methods in Applied Mechanics and Engineering, 191 (2002), pp. 1097–1112.

O. Axelsson, “A class of A-stable Methods,” Nordisk Tidskr. Informationbehandling (BIT), v. 9, 1969, pp. 185–199.

R.E. Bauer, “Discontinuous Galerkin methods for ordinary differential equations,” Master Thesis, Applied Mathematics, University of Colorado, 1995.

B. Cockburn, G. Karniadakis, and C.-W. Shu, “The development of discontinuous Galerkin methods,” in Discontinuous Galerkin methods: Theory, Computation, and Application, B. Cockburn, G. Karniadakis, and C.-W. Shu, eds., Lecture Notes in Computational Science and Engineering, Springer (2000), pp. 3–50

G. J. Cooper, “Interpolation and quadrature methods for ordinary differential equations,” Math. Comp., v. 22, 1968, pp. 69–76.

M. Delfour, W. Hager, and F. Trochu, “Discontinuous Galerkin methods for ordinary differential equations,” Math. Comp., v. 36, 1981, pp. 455–473.

K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, “Computational Differential Equations,” Cambridge University Press (1996).

C.A.J. Fletcher, “Computational Galerkin Methods,” Springer-Verlag (1984).

E. Hairer, S.P. Norsett, and G. Wanner, “Solving ordinary differential equations I,” 2nd edition, Springer-Verlag (1993).

E. Hairer and G. Wanner, “Solving ordinary differential equations II,” Springer-Verlag (1991).

F.B. Hildebrand, “Introduction to Numerical Analysis,” Dover (1987).

B. Hulme, “Discrete Galerkin and related one-step methods for ordinary differential equations,” Math. Comp., v. 26, 1972, pp. 881–891.

H.T. Huynh, “A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods,” 18th AIAA-CFD Conference, AIAA–2007–4079.

J.D. Lambert, “Numerical methods for ordinary differential systems,” John Wiley and Sons (1991).

P. LeSaint and P.A. Raviart, “On the finite element method for solving the neutron transport equation,” in C. de Boor, ed., “Mathematical aspects of finite elements in partial differential equations,” Academic Press (1974), pp. 89–145.

W.H. Reed and T.R. Hill, Triangular mesh methods for the neutron transport equation, Tech. Report LA–UR–73–479, Los Alamos Scientific Laboratory, 1973.

J.J.W. van der Vegt and H. van der Ven “Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flows, Part I. General formulation,” J. Comput. Phys., 182 (2002) 546–585.

K. Wright, “Some relationship between implicit Runge-Kutta, Collocation and Lanczos τ-methods, and their stability properties,” BIT, v. 10, 1970, pp. 217–227.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
01-08-2011

2. REPORT TYPE
Technical Memorandum

3. DATES COVERED *(From - To)*

4. TITLE AND SUBTITLE
Collocation and Galerkin Time-Stepping Methods

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Huynh, H., T.

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
WBS 599489.02.07.03.03.03.03

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
John H. Glenn Research Center at Lewis Field
Cleveland, Ohio 44135-3191

8. PERFORMING ORGANIZATION REPORT NUMBER
E-17277

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING/MONITOR'S ACRONYM(S)
NASA

11. SPONSORING/MONITORING REPORT NUMBER
NASA/TM-2011-216340

12. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified-Unlimited
Subject Category: 02
Available electronically at http://www.sti.nasa.gov
This publication is available from the NASA Center for AeroSpace Information, 443-757-5802

13. SUPPLEMENTARY NOTES

14. ABSTRACT

We study the numerical solutions of ordinary differential equations by one-step methods where the solution at t_n is known and that at t_{n+1} is to be calculated. The approaches employed are collocation, continuous Galerkin (CG) and discontinuous Galerkin (DG). Relations among these three approaches are established. A quadrature formula using s evaluation points is employed for the Galerkin formulations. We show that with such a quadrature, the CG method is identical to the collocation method using quadrature points as collocation points. Furthermore, if the quadrature formula is the right Radau one (including t_{n+1}), then the DG and CG methods also become identical, and they reduce to the Radau IIA collocation method. In addition, we present a generalization of DG that yields a method identical to CG and collocation with arbitrary collocation points. Thus, the collocation, CG, and generalized DG methods are equivalent, and the latter two methods can be formulated using the differential instead of integral equation. Finally, all schemes discussed can be cast as s-stage implicit Runge-Kutta methods.

15. SUBJECT TERMS
Numerical solution for ordinary differential equations

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	38	STI Help Desk (email:help@sti.nasa.gov)
					19b. TELEPHONE NUMBER *(include area code)* 443-757-5802