BIT Numerical Mathematics **44:** 711–720, 2004.

GENERALIZED GAUSS–RADAU AND GAUSS–LOBATTO FORMULAE

WALTER GAUTSCHI[1]

[1] *Department of Computer Sciences, Purdue University, West Lafayette, IN 47907, USA.*
email: wxg@cs.purdue.edu

Abstract.

Computational methods are developed for generating Gauss-type quadrature formulae having nodes of arbitrary multiplicity at one or both end points of the interval of integration. Positivity properties of the boundary weights are investigated numerically, and related conjectures are formulated. Applications are made to moment-preserving spline approximation.

AMS subject classification (2000): 65D32, 41A15.

Key words: Gauss–Radau quadrature, Gauss–Lobatto quadrature, multiple boundary node(s), positivity, moment-preserving spline approximation.

1 Introduction.

Gauss–Radau and Gauss–Lobatto formulae, as is well known, are quadrature formulae of Gauss type involving function values not only at interior points of the interval of integration, but also at one or both end points of this interval. The phrase "of Gauss type" means that the polynomial degree of exactness of these formulae is as large as possible subject to the constraints on the nodes. Similarly, one can define *generalized* Gauss–Radau and Gauss–Lobatto formulae in which not only function values at one or both end points appear, but also consecutive derivative values up to an arbitrary finite order $r-1$. These are of interest, e.g., in moment-preserving spline approximation on a compact interval (cf. Section 4). Thus, in the case of Gauss–Radau formulae, they have the form

$$\int_a^\infty f(t)\,\mathrm{d}\lambda(t) = \sum_{\rho=0}^{r-1} \lambda_0^{(\rho)} f^{(\rho)}(a) + \sum_{\nu=1}^{n} \lambda_\nu^R f(\tau_\nu^R) + R_{n,r}^R(f), \tag{1.1}$$

where $r > 1$ is the multiplicity of the end point a and $\mathrm{d}\lambda$ a positive measure with (bounded or unbounded) support contained in $[a,\infty]$. The degree of exactness is $2n-1+r$,

$$R_{n,r}^R(f) = 0 \quad \text{for all } f \in \mathbb{P}_{2n-1+r}. \tag{1.2}$$

Here and in the following, $\mathbb{P}_m$ denotes the space of real polynomials of degree $\le m$. The internal nodes τ_ν^R and weights λ_ν^R, as well as the weights $\lambda_0^{(\rho)}$, although not expressed in our notation, all depend on n and r.

Naturally, there is an analogous formula for $\int_{-\infty}^{b} f(t)\,\mathrm{d}\lambda(t)$ with fixed multiple node at $t = b$.

In the case of the generalized Gauss–Lobatto formula, the support of $\mathrm{d}\lambda$ is assumed to be contained in a finite interval $[a, b]$, $a < b$, and the formula takes the form

$$\begin{aligned}\int_a^b f(t)\,\mathrm{d}\lambda(t) \;=\; &\sum_{\rho=0}^{r-1} \lambda_0^{(\rho)} f^{(\rho)}(a) + \sum_{\nu=1}^{n} \lambda_\nu^L f(\tau_\nu^L) + \\ &+ \sum_{\rho=0}^{r-1} (-1)^\rho \lambda_{n+1}^{(\rho)} f^{(\rho)}(b) + R_{n,r}^L(f),\end{aligned} \tag{1.3}$$

where

$$R_{n,r}^L(f) = 0 \quad \text{for all } f \in \mathbb{P}_{2n-1+2r}. \tag{1.4}$$

We have included signs $(-1)^\rho$ in the weights of the derivative values at $t = b$ in anticipation of the fact that $\lambda_0^{(\rho)} = \lambda_{n+1}^{(\rho)}$ in the case of symmetric formulae, that is, formulae with $a + b = 0$ and $\mathrm{d}\lambda(-t) = \mathrm{d}\lambda(t)$.

The characterization of the internal nodes and weights of generalized Gauss–Radau and Gauss–Lobatto formulae is well known; see, e.g., [1, Theorems 3.9 and 3.12]. In the former case,

$$\tau_\nu^R = \tau_\nu^{[r]}, \qquad \lambda_\nu^R = \frac{\lambda_\nu^{[r]}}{(\tau_\nu^R - a)^r}, \quad \nu = 1, 2, \ldots, n, \tag{1.5}$$

where $\tau_\nu^{[r]}$, $\lambda_\nu^{[r]}$ are the nodes and weights of the n-point Gauss quadrature formula for the measure

$$\mathrm{d}\lambda^{[r]}(t) = (t - a)^r\,\mathrm{d}\lambda(t) \quad \text{(Radau)}, \tag{1.6}$$

while in the latter case,

$$\tau_\nu^L = \tau_\nu^{[r]}, \qquad \lambda_\nu^L = \frac{\lambda_\nu^{[r]}}{[(\tau_\nu^L - a)(b - \tau_\nu^L)]^r}, \quad \nu = 1, 2, \ldots, n, \tag{1.7}$$

with $\tau_\nu^{[r]}$, $\lambda_\nu^{[r]}$ the Gaussian nodes and weights of

$$\mathrm{d}\lambda^{[r]}(t) = [(t - a)(b - t)]^r\,\mathrm{d}\lambda(t) \quad \text{(Lobatto)}. \tag{1.8}$$

All internal weights are therefore positive, and the same is true for the boundary weights if $r = 2$ (cf. Sections 2.2 and 3.2). In this latter case, explicit formulae are known for the Legendre measure $\mathrm{d}\lambda(t) = \mathrm{d}t$ on $[-1, 1]$ ([1, Examples 3.10 and 3.13]) and for all four Chebyshev measures [2]. Other than that, little is known about generalized Gauss–Radau and Gauss–Lobatto formulae.

In this paper, we develop a procedure for computing such formulae for arbitrary r. Respective Matlab routines `gradau.m` and `globatto.m` are downloadable from the Web site `http://www.cs.purdue.edu/archives/2002/wxg/codes/` which contains a suite of many other useful routines, in part assembled as a companion piece to the book in [1]. All Matlab routines referred to in this paper are downloadable individually from this site, in particular also routines developed for testing `gradau.m` and `globatto.m`. Both routines are used to investigate (numerically) the positivity of the boundary weights. In Section 4 we illustrate the use of these routines in the context of moment-preserving spline approximation.

2 The generalized Gauss–Radau formula.

2.1 Computational method.

The measure relevant for the internal nodes and weights, by (1.6), is obtained by r successive modifications of the measure $\mathrm{d}\lambda(t)$ by the linear factor $t-a$. The respective orthogonal polynomials can thus be generated by r applications of the routine `chri1.m` (cf. [1, §2.4.2]). For the computation of $\tau_\nu^{[r]}$ and $\lambda_\nu^{[r]}$ in (1.5) one then applies the Gauss quadrature routine `gauss.m`.

In order to compute the boundary weights $\lambda_0^{(\rho)}$ in (1.1), we use (1.1) with $f(t)=(t-a)^{i-1}\pi_n^2(t)$, $i=1,2,\ldots,r$, where $\pi_n(t)=\prod_{\nu=1}^n(t-\tau_\nu^R)$, and note, since $f\in\mathbb{P}_{2n-1+r}$, that the remainder is zero, and by the choice of f, that all terms in the quadrature sum are zero except the boundary terms with $\rho\ge i-1$. Therefore,

$$\sum_{\rho=i-1}^{r-1}\lambda_0^{(\rho)}[(t-a)^{i-1}\pi_n^2(t)]_{t=a}^{(\rho)}=b_i, \tag{2.1}$$

$$b_i=\int_a^\infty(t-a)^{i-1}\pi_n^2(t)\,\mathrm{d}\lambda(t),\ i=1,2,\ldots,r.$$

Here, the integrals on the right are computable (exactly) by $(n+\lfloor(r+1)/2\rfloor)$-point Gauss quadrature with respect to the measure $\mathrm{d}\lambda$. Writing $\rho=j-1$, $j=i,i+1,\ldots,r$, Equations (2.1) represent an upper triangular system of linear algebraic equations

$$\boldsymbol{A}\boldsymbol{x}=\boldsymbol{b},\quad \boldsymbol{A}\in\mathbb{R}^{r\times r},\ \boldsymbol{x},\boldsymbol{b}\in\mathbb{R}^r, \tag{2.2}$$

where

$$\boldsymbol{A}=[a_{ij}],\quad a_{ij}=[(t-a)^{i-1}\pi_n^2(t)]_{t=a}^{(j-1)},\quad j\ge i;\qquad a_{ij}=0,\ j<i, \tag{2.3}$$

$$\boldsymbol{x}=[x_j],\quad x_j=\lambda_0^{(j-1)};\qquad \boldsymbol{b}=[b_i].$$

When applying Leibniz's rule to compute a_{ij}, only one term survives, namely the one in which the first factor $(t-a)^{i-1}$ is differentiated $i-1$ times and the other factor $j-1-(i-1)=j-i$ times. Thus,

$$a_{ij}=\binom{j-1}{j-i}(i-1)![\pi_n^2(t)]_{t=a}^{(j-i)},\quad j\ge i. \tag{2.4}$$

It remains to compute the derivatives $[\pi_n^2(t)]_{t=a}^{(s)}$. Since $\pi_n^2(t) = \prod_{\nu=1}^n (\tau_\nu^R - t)^2$, we have

$$[\pi_n^2(t)]' = -2\sum_{\nu=1}^n (\tau_\nu^R - t)\prod_{\mu\neq\nu}(\tau_\mu^R - t)^2 = -2\pi_n^2(t)\sum_{\nu=1}^n (\tau_\nu^R - t)^{-1}.$$

Differentiating this $j-i-1$ times by Leibniz's rule, and then putting $t = a$, yields, for $j > i$,

$$[\pi_n^2(t)]_{t=a}^{(j-i)} = -2\sum_{\sigma=0}^{j-i-1}\binom{j-i-1}{\sigma}[\pi_n^2(t)]_{t=a}^{(j-i-1-\sigma)}\cdot\sigma!\sum_{\nu=1}^n(\tau_\nu^R - a)^{-(\sigma+1)}.$$

With (2.4) used both on the left and on the right, this gives, for $i = r-1, r-2, \dots, 1$,

$$a_{ij} = -\frac{2}{j-i}\sum_{\sigma=0}^{j-i-1} a_{i+1+\sigma,j}\sum_{\nu=1}^n(\tau_\nu^R - a)^{-(\sigma+1)}, \quad j = i+1, i+2, \dots, r, \tag{2.5}$$

while for $j = i$,

$$a_{ii} = (i-1)!\pi_n^2(a), \quad i = r, r-1, \dots, 1. \tag{2.6}$$

This allows us to compute the elements of the upper triangular matrix $\boldsymbol{A}$ from the bottom up, and then to solve the system (2.2) by backward substitution. The procedure is implemented in the routine `gradau.m`.

Tests against the known formulae for $r = 2$ in the case of the Legendre and Chebyshev measures turned out to be satisfactory, even for values of n as large as $n = 160$; see the routine `test_gradau_req2.m`. A meaningful test for $r \geq 1$ is to have the generalized Gauss–Radau formula for arbitrary r, say $r = 1{:}10$, and with n internal points, compute (exactly) all moments of orders up to $2n-1+r$ for relatively small values of n, say $n = 1{:}5$. This, too, turned out very satisfactory; see the routine `test_gradau_rgt0.m`. Finally, for $r = 1$, the generalized Gauss–Radau formula reduces to the ordinary Gauss–Radau formula, so that `gradau.m` can be tested against `radau.m`. This is done successfully in the routine `test_gradau_radau.m`.

2.2 Positivity.

It is clear from (2.6) that all diagonal elements of $\boldsymbol{A}$ are positive, and from (2.1) that the same is true for the elements of $\boldsymbol{b}$. In particular, $x_r = b_r/a_{rr} > 0$. From (2.5) it can also be seen that $a_{r-1,r} < 0$, from which $x_{r-1} = (b_{r-1} - a_{r-1,r}x_r)/a_{r-1,r-1} > 0$. There follows

$$\lambda_0^{(r-1)} > 0, \qquad \lambda_0^{(r-2)} > 0. \tag{2.7}$$

Thus, if $r = 2$, the boundary weights of the generalized Gauss–Radau formula are positive. Whether or not the same is true for arbitrary $r > 2$ is an open question. Using the routine `gradau.m`, we found numerically that positivity holds for Jacobi measures with parameters $\alpha = -.9{:}.2{:}.9$, $\beta = \alpha{:}.2{:}.9$ and with $\alpha = -.75{:}.25{:}1$, $\beta = 1, 1.5, 2, 5, 10$, each time for values of $n = 2{:}20$, $n = 25{:}5{:}40$ and $r = 2{:}10$. For the same values of α, n, and r, positivity was also observed for generalized Laguerre measures. For these two measures, therefore, it seems safe to conjecture that the generalized Gauss–Radau formula has positive boundary weights. Additional tests involving the "elliptic" Chebyshev measure ([1, Example 2.29]) and the half-range Hermite measure ([1, Example 2.31]) suggest that positivity may hold for arbitrary measures. At any rate, the situation appears to be rather different from what is known in the case of Gauss–Turán formulae, which also involve derivative values, but at internal nodes. Here one can prove positivity of the weights only for derivative terms of *even* order (cf. [1, §3.1.3.1]), whereas those of odd order may have weights of either sign.

3 The generalized Gauss–Lobatto formula.

3.1 Computational method.

The measure (1.8) determining the internal nodes and weights of the generalized Gauss–Lobatto formula is now obtained by $2r$ consecutive modifications of the measure $\mathrm{d}\lambda$, half of them with the shift a, and the other half with shift b. When r is odd, the resulting measure is negative definite, which is easily corrected by changing the sign of the coefficient β_0 produced by `chri1.m`.

The boundary weights are computed similarly as in the case of generalized Gauss–Radau formulae except for some special attention being required when r is odd. The formula (1.3) is applied once to $f(t) = (t-a)^{i-1}\pi_n^2(t)(b-t)^r$ to find the weights $\lambda_0^{(\rho)}$, and once to $f(t) = (t-b)^{i-1}\pi_n^2(t)(a-t)^r$ to determine the weights $(-1)^\rho\lambda_{n+1}^{(\rho)}$, where now $\pi_n(t) = \prod_{\nu=1}^n (t-\tau_\nu^L)$. In both cases, the remainder is zero, since $f \in \mathbb{P}_{2n-1+2r}$. Suppose first that $r \geq 2$ is even. The first choice of f then yields for $x_j = \lambda_0^{(j-1)}$ the upper triangular system (2.2) with

$$\boldsymbol{A} = [a_{ij}], \quad a_{ij} = ((t-a)^{i-1}[\pi_n^2(t)(b-t)^r])_{t=a}^{(j-1)},$$

$$\boldsymbol{b} = [b_i], \quad b_i = \int_a^b (t-a)^{i-1}\pi_n^2(t)(b-t)^r\,\mathrm{d}\lambda(t).$$

The integrals defining b_i can all be evaluated (exactly) by $(n+r)$-point Gauss quadrature relative to the measure $\mathrm{d}\lambda$.

To compute a_{ij}, we proceed similarly as in Section 2.1, writing

$$\pi_n^2(t)(b-t)^r = \prod_{\nu=1}^{n+r/2} (\tau_\nu - t)^2 \quad (r \text{ even}),$$

where $\tau_\nu = \tau_\nu^L$ for $1 \le \nu \le n$, and $\tau_\nu = b$ for $n+1 \le \nu \le n + r/2$. Then, exactly as before, one gets for $i = r, r-1, \ldots, 1$,

$$a_{ii} = (i-1)!\pi_n^2(a)(b-a)^r,$$

$$a_{ij} = -\frac{2}{j-i}\sum_{\sigma=0}^{j-i-1} a_{i+1+\sigma,j}\sum_{\nu=1}^{n+r/2}(\tau_\nu - a)^{-(\sigma+1)}, \quad j = i+1, i+2, \ldots, r, \tag{3.1}$$

and the computation is the same as in Section 2.1, except for the extended summation on the right.

When r is odd, one defines as before $\tau_\nu = \tau_\nu^L$ for $1 \le \nu \le n$ and $\tau_\nu = b$, but now for $n+1 \le \nu \le n + \lfloor (r+1)/2 \rfloor$ (which is valid also when r is even). Then

$$\pi_n^2(t)(b-t)^r = \left[\prod_{\nu=1}^{n+(r-1)/2}(\tau_\nu - t)^2\right](b-t) \quad (b = \tau_{n+(r+1)/2}),$$

and one obtains

$$\begin{aligned}
&[\pi_n^2(t)(b-t)^r]' \\
&= -2\sum_{\nu=1}^{n+(r-1)/2}(\tau_\nu - t)\prod_{\mu\ne\nu}(\tau_\mu - t)^2 \cdot (b-t) - \prod_{\nu=1}^{n+(r-1)/2}(\tau_\nu - t)^2 \\
&= -2\sum_{\nu=1}^{n+(r-1)/2}(\tau_\nu - t)^{-1}\prod_{\mu=1}^{n+(r-1)/2}(\tau_\mu - t)^2 \cdot (b-t) - \prod_{\nu=1}^{n+(r-1)/2}(\tau_\nu - t)^2 \\
&= -2\pi_n^2(t)(b-t)^r\left\{\sum_{\nu=1}^{n+(r-1)/2}(\tau_\nu - t)^{-1} + \frac{1}{2}(b-t)^{-1}\right\}.
\end{aligned}$$

Thus,

$$[\pi_n^2(t)(b-t)^r]' = -2\pi_n^2(t)(b-t)^r \sideset{}{'}\sum_{\nu=1}^{n+(r+1)/2}(\tau_\nu - t)^{-1},$$

where the prime on the summation sign indicates that the last term in the summation is to be halved. In the same manner as in Section 2.1, differentiating $j-i-1$ times, one obtains for $j > i$

$$a_{ij} = -\frac{2}{j-i}\sum_{\sigma=0}^{j-i-1} a_{i+1+\sigma,j}\sideset{}{'}\sum_{\nu=1}^{n+(r+1)/2}(\tau_\nu - a)^{-(\sigma+1)}, \quad j = i+1, i+2, \ldots, r,$$

while for $j = i$, as before,

$$a_{ii} = (i-1)!\pi_n^2(a)(b-a)^r.$$

For the choice $f(t) = (t-b)^{i-1}\pi_n^2(t)(a-t)^r$, the same procedure applies, with the roles of a and b in (3.1) interchanged. The routine implementing all of this is `globatto.m`.

Tests analogous to those in Section 2.1 were performed on the routine `globatto.m` with equal success; see the routines `test_globatto_req2.m`, `test_globatto_rgt0.m`, and `test_globatto_lobatto.m`.

3.2 Positivity.

As in Section 2.2, one shows from (3.1) that

$$\lambda_0^{(r-1)} > 0, \qquad \lambda_0^{(r-2)} > 0.$$

With regard to the boundary weights at the point b, one must interchange a and b both in (3.1) and in the integrand of b_i. One then has $\operatorname{sign} a_{ii} = (-1)^r$, $\operatorname{sign} b_i = (-1)^{i-1+r}$, hence $\operatorname{sign} x_r = \operatorname{sign}(b_r/a_{rr}) = (-1)^{r+1}$, so that $\operatorname{sign}((-1)^{r-1}\lambda_{n+1}^{(r-1)}) = (-1)^{r+1}$, that is, $\lambda_{n+1}^{(r-1)} > 0$. Similarly, $\operatorname{sign} a_{r-1,r} = (-1)^r$, $\operatorname{sign} x_{r-1} = \operatorname{sign}[(b_{r-1} - a_{r-1,r}x_r)/a_{r-1,r-1}] = \operatorname{sign} a_{r-1,r-1} = (-1)^r$, so that $\operatorname{sign}[(-1)^{r-2}\lambda_{n+1}^{(r-2)}] = (-1)^r$, and $\lambda_{n+1}^{(r-2)} > 0$. Thus, generalized Gauss–Lobatto formulae, when $r = 2$, have boundary weights satisfying

$$\lambda_0^{(\rho)} > 0, \qquad \lambda_{n+1}^{(\rho)} > 0, \quad \rho = 0, 1, \ldots, r-1 \ (r = 2). \tag{3.2}$$

Positivity in the case $r > 2$ is still an open question. Numerical tests with `globatto.m` for the same Jacobi measures as in Section 2.2, and for the same values of $n \geq 3$ and r, revealed positivity in all cases (cf. the routine `globatto_pos.m`). We therefore conjecture that the generalized Gauss–Lobatto formula for Jacobi measures has boundary weights satisfying (3.2) for arbitrary $r > 2$. Judging from additional tests with the elliptic Chebyshev measure, we believe that positivity holds for other, if not all, measures as well.

4 Examples.

It is known (cf. [1, §3.3]) that Gauss-type quadrature formulae, and in particular generalized Gauss–Lobatto and Gauss–Radau formulae, are useful in generating spline approximations to functions f that reproduce as many consecutive moments of f as possible. For approximation on the positive real line $\mathbb{R}_+$, the measure involved is

$$\mathrm{d}\lambda^{[m]}(t) = \frac{(-1)^{m+1}}{m!} t^{m+1} f^{(m+1)}(t)\,\mathrm{d}t, \quad t \in \mathbb{R}_+, \tag{4.1}$$

where m is the degree of the spline. The spline, in this case, has the form

$$s_{n,m}(t) = \sum_{\nu=1}^{n} a_\nu (t_\nu - t)_+^m, \quad t \in \mathbb{R}_+, \tag{4.2}$$

where $u_+(t) = \max(0, u(t))$ and $0 < t_1 < t_2 < \cdots < t_n < \infty$ are the knots of the spline. These are assumed freely variable (subject to ordering), and so are the coefficients $a_\nu \in \mathbb{R}$. Given the first $2n$ moments $\mu_j = \int_{\mathbb{R}_+} f(t)t^j \, \mathrm{d}t$, $j = 0, 1, \dots, 2n-1$, of f, the problem on $\mathbb{R}_+$ is to find $s_{n,m}$ such that

$$\int_{\mathbb{R}_+} s_{n,m}(t)t^j \, \mathrm{d}t = \mu_j, \quad j = 0, 1, \dots, 2n-1. \tag{4.3}$$

This has a unique solution precisely if the measure (4.1) admits a Gaussian quadrature formula

$$\int_{\mathbb{R}_+} g(t) \, \mathrm{d}\lambda^{[m]}(t) = \sum_{\nu=1}^{n} \lambda_\nu^G g(\tau_\nu^G), \quad g \in \mathbb{P}_{2n-1}, \tag{4.4}$$

satisfying $0 < \tau_1^G < \tau_2^G < \cdots < \tau_n^G$, in which case

$$t_\nu = \tau_\nu^G, \qquad a_\nu = \frac{\lambda_\nu^G}{[\tau_\nu^G]^{m+1}}, \quad \nu = 1, 2, \dots, n, \tag{4.5}$$

yields the desired spline approximant (cf. [1, Theorem 3.57]).

On a compact interval $[0, 1]$, the measure involved is

$$\mathrm{d}\lambda^{[m]}(t) = \frac{(-1)^{m+1}}{m!} f^{(m+1)}(t) \, \mathrm{d}t, \quad t \in [0, 1]. \tag{4.6}$$

In this case, a polynomial $p \in \mathbb{P}_m$ may be added to the spline in (4.2), so that

$$s_{n,m}(t) = p(t) + \sum_{\nu=1}^{n} a_\nu (t_\nu - t)_+^m, \quad t \in [0, 1], \tag{4.7}$$

where $0 < t_1 < t_2 < \cdots < t_n < 1$. Two problems are then of interest:

Problem I. Determine $s_{n,m}$ in (4.7) such that

$$\int_0^1 s_{n,m}(t)t^j \, \mathrm{d}t = \mu_j, \quad j = 0, 1, \dots, 2n+m. \tag{4.8}$$

Since we have $m+1$ more parameters at our disposal (the coefficients of p), we can impose $m+1$ additional moment conditions compared to (4.3).

Problem II. Determine $s_{n,m}$ in (4.7) such that

$$\int_0^1 s_{n,m}(t)t^j \, \mathrm{d}t = \mu_j, \quad j = 0, 1, \dots, 2n-1, \tag{4.9}$$

and

$$s_{n,m}^{(\mu)}(1) = f^{(\mu)}(1), \quad \mu = 0, 1, \dots, m. \tag{4.10}$$

Here the extra $m+1$ parameters are used to enforce the derivative conditions (4.10), which, incidentally, immediately determine p, since $s_{n,m}^{(\mu)}(1) = p^{(\mu)}(1)$.

4.1 Solution of Problem I.

By [1, Theorem 3.61], Problem I has a unique solution if and only if the measure $\mathrm{d}\lambda^{[m]}$ in (4.6) admits a generalized Gauss–Lobatto formula (1.3), with $[a,b]=[0,1]$, satisfying $0<\tau_1^L<\tau_2^L<\cdots<\tau_n^L<1$ and having boundary points of multiplicity $r=m+1$. The solution of Problem I is then given by

$$t_\nu=\tau_\nu^L, \qquad a_\nu=\lambda_\nu^L, \quad \nu=0,1,\dots,n, \tag{4.11}$$

with the polynomial p uniquely determined by its derivative values at $t=1$,

$$p^{(\mu)}(1)=f^{(\mu)}(1)+(-1)^m m!\lambda_{n+1}^{(m-\mu)}, \quad \mu=0,1,\dots,m. \tag{4.12}$$

Even though the emphasis of these approximations is on preserving as many moments as possible, it is still interesting to observe how well they do with regard to pointwise approximation.

Example 4.1. The exponential function $f(t)=\mathrm{e}^{-t}$.

This example was solved in [1, Example 3.59] on the interval $\mathbb{R}_+$ using the Gauss formula (4.4) for the measure (4.1), and (4.5). We now compare the restriction of that solution to the interval $[0,1]$ with direct solution on $[0,1]$ via (4.11) and (4.12).

Note, first of all, that the measure (4.6) in this case is

$$\mathrm{d}\lambda^{[m]}(t)=\frac{1}{m!}\,\mathrm{e}^{-t}\,\mathrm{d}t, \quad 0<t<1, \tag{4.13}$$

the Laguerre measure on the finite interval $[0,1]$. Its first $n+2m+2$ recurrence coefficients needed in the $2r=2m+2$ applications of the routine `chri1.m` to the measure $\mathrm{d}\lambda^{[m]}$ (cf. the first paragraph of Section 3.1) are not known explicitly, but can easily be computed by a discretization procedure using Gauss–Legendre quadrature on $[0,1]$ to discretize the inner product for $\mathrm{d}\lambda^{[m]}$ (cf. [1, §2.2.4]). This, together with the procedure in Section 3.1, is implemented in the routine `ex3_3.m`, which also computes the restriction to $[0,1]$ of the spline approximation on $\mathbb{R}_+$ for comparison. The results are shown in the first four columns of Table 4.1, where the third column contains the maximum errors on $[0,1]$ of the spline approximation (4.2), (4.5), and the fourth column the analogous information for the spline (4.7), (4.11)–(4.12). (The maximum errors are computed on a set of 100 equally spaced points on $[0,1]$.) It is seen, not surprisingly, that the latter are significantly smaller than the former.

4.2 Solution of Problem II.

By [1, Theorem 3.62], the solution, if it exists, is now provided by the generalized Gauss–Radau formula (1.1) for the measure $\mathrm{d}\lambda^{[m]}$ in (4.6), where $[a,\infty]$ is to be replaced by $[0,1]$. Indeed,

$$t_\nu=\tau_\nu^R, \qquad a_\nu=\lambda_\nu^R, \quad \nu=1,2,\dots,n, \tag{4.14}$$

Table 4.1: Maximum errors in Examples 4.1 and 4.2

m	n	err	err	err
1	5	5.0419e−02	2.3346e−03	2.9070e−03
1	10	2.8150e−02	7.5711e−04	9.5130e−04
1	20	1.4824e−02	2.5198e−04	2.4060e−04
1	40	7.1401e−03	6.4979e−05	7.2096e−05
1	80	3.7475e−03	1.5633e−05	1.9889e−05
2	5	1.7857e−02	3.9962e−05	6.8379e−05
2	10	3.4965e−03	8.5681e−06	1.1922e−05
2	20	1.0938e−03	1.5137e−06	1.8741e−06
2	40	3.6171e−04	2.3831e−07	2.6307e−07
2	80	1.2197e−04	3.3245e−08	3.5524e−08
3	5	7.9365e−03	9.6683e−07	2.4463e−06
3	10	9.9900e−04	1.4155e−07	2.4701e−07
3	20	1.3962e−04	1.5061e−08	2.1292e−08
3	40	3.0058e−05	1.3686e−09	1.6194e−09
3	80	7.8536e−06	9.9495e−11	1.1154e−10

in (4.7), and p is given (trivially) by

$$p(t)=\sum_{\mu=0}^{m}\frac{f^{(\mu)}(1)}{\mu!}(t-1)^{\mu}. \tag{4.15}$$

Example 4.2. The exponential function $f(t)=\mathrm{e}^{-t}$.

We now need only $r=m+1$ applications of the routine `chri1.m` to the measure (4.13) (cf. the first paragraph of Section 2.1), which is done analogously as in Example 4.1. The results, also produced by the routine `ex3_3.m`, are shown in the last column of Table 4.1. They are very similar, and only slightly worse, than those in the fourth column.

REFERENCES

1. W. Gautschi, *Orthogonal Polynomials: Computation and Approximation*, Numerical Mathematics and Scientific Computation, Oxford University Press, Oxford, 2004.
2. W. Gautschi and S. Li, *Gauss–Radau and Gauss–Lobatto quadratures with double end points*, J. Comput. Appl. Math., 34 (1991), pp. 343–360.