Algorithm 793: GQRAT—Gauss Quadrature for Rational Functions

WALTER GAUTSCHI
Purdue University

The concern here is with Gauss-type quadrature rules that are exact for a mixture of polynomials and rational functions, the latter being selected so as to simulate poles that may be present in the integrand. The underlying theory is presented as well as methods for constructing such rational Gauss formulae. Relevant computer routines are provided and applied to a number of examples, including Fermi-Dirac and Bose-Einstein integrals of interest in solid state physics.

Categories and Subject Descriptors: D.3.2 [**Programming Languages**]: Language Classifications—*FORTRAN 77*; G.1.4 [**Numerical Analysis**]: Quadrature and Numerical Differentiation

General Terms: Algorithms

Additional Key Words and Phrases: Construction of quadrature rules, Gaussian quadrature exact for rational functions, generalized Fermi-Dirac and Bose Einstein integrals

1. INTRODUCTION

Traditionally, Gaussian quadrature rules are designed to integrate exactly polynomials of degrees as large as possible. This is meaningful if the integrand behaves like a polynomial. In applications it is not unusual, however, that the integrand has poles away from the interval of integration. This then would suggest to also require exactness for elementary rational functions having the same poles, or at least the more important ones among them (those closest to the interval of integration). Gauss quadrature rules of this rational type have recently been developed in Gautschi [1993a]. (For special cases, see also Van Assche and Vanherwegen [1993] and Rovba [1996].) In particular, two procedures were proposed to construct such formulae, which in a sense complement each other. The first is based on partial fraction decomposition and modification algorithms

Author's address: Department of Computer Sciences, Purdue University, West Lafayette, IN 47907-1398; email: wxg@cs.purdue.edu.

[Gautschi 1993a, Section 2.1], the other on discretization. Whereas the former has the peculiarity of relying on discrete orthogonal polynomials relative to a *nondefinite* inner product, and thus being open to possible instabilities or even breakdowns, it does very well at handling poles very close to the interval of integration. The second procedure uses only *definite* (discrete) inner products and is conceptually much simpler, but has difficulty with poles very close to the interval of integration, which cause convergence to slow down. It was felt, therefore, that both procedures had their right to exist.

In the meantime, it has occurred to us that the problem with slow convergence can be mitigated by treating the difficult poles separately, after the others have been incorporated. With this provision, the second procedure seems to become more attractive, and we therefore concentrate on, and implement only, this second procedure in this article. There are different variants to this procedure; they all, however, use two basic computer routines, called `abmod` and `gqrat`, the former providing the necessary orthogonal polynomials, the latter the desired quadrature rules. Their different uses are illustrated in a number of examples, some of considerable interest in applications.

2. GAUSS QUADRATURE RULE FOR RATIONAL FUNCTIONS

2.1 Basic Result

The quadrature rule to be considered is of the usual form

$$\int_{\mathbb{R}} f(t)d\lambda(t) = \sum_{\nu=1}^{n} \lambda_\nu f(t_\nu) + R_n(f), \tag{2.1}$$

where $d\lambda$ is a given (positive) measure supported on an interval (finite, half-infinite, or infinite). The generalized Gauss principle requires the remainder R_n to vanish on a linear space of functions $\mathbb{S}_{2n}$ of dimension $2n$,

$$R_n(g) = 0, \quad \text{all } g \in \mathbb{S}_{2n}. \tag{2.2}$$

In the classical case, we have $\mathbb{S}_{2n} = \mathbb{P}_{2n-1}$, the space of polynomials of degree $\leq 2n - 1$. In this article we choose

$$\mathbb{S}_{2n} = \mathbb{Q}_m \oplus \mathbb{P}_{2n-1-m}, \quad 0 \leq m \leq 2n, \tag{2.3}$$

where the integer m is a parameter, and where

$$\mathbb{Q}_m = \text{span}\{g : g(t) = (1 + \zeta_\mu t)^{-s},\ s = 1, 2, \ldots, s_\mu,\ \mu = 1, 2, \ldots, M;$$

$$\sum_{\mu=1}^{M} s_\mu = m\}. \tag{2.4}$$

Here ζ_μ are real or complex numbers satisfying

$$\zeta_\mu \neq 0, \quad 1 + \zeta_\mu t \neq 0 \text{ for } t \in \overline{\text{supp}(d\lambda)}, \quad \mu = 1,2, \ldots, M. \tag{2.5}$$

Thus, $\mathbb{Q}_m$ consists of exactly m elementary rational functions having prescribed poles at $-1/\zeta_\mu$ of multiplicities up to s_μ. In the limit cases $m = 0$ and $m = 2n$, the spaces S_{2n} are respectively $\mathbb{P}_{2n-1}$ and $\mathbb{Q}_{2n}$, the former consisting only of polynomials, the latter only of genuinely rational functions.

The basic result, proved in Gautschi [1993a], is as follows.

THEOREM. *Let*

$$\omega_m(t) = \prod_{\mu=1}^{M} (1 + \zeta_\mu t)^{s_\mu} \tag{2.6}$$

(a polynomial of exact degree m). Assume that the measure $d\lambda/\omega_m$ admits an n-point (ordinary) Gaussian quadrature formula

$$\int_{\mathbb{R}} p(t) \frac{d\lambda(t)}{\omega_m(t)} = \sum_{\nu=1}^{n} w_\nu^G p(t_\nu^G), \quad \text{all } p \in \mathbb{P}_{2n-1}, \tag{2.7}$$

having nodes t_ν^G contained in the support of $d\lambda$. Define

$$t_\nu = t_\nu^G, \quad \lambda_\nu = w_\nu^G \omega_m(t_\nu^G). \tag{2.8}$$

Then we have

$$\begin{aligned} &\int_{\mathbb{R}} g(t) d\lambda(t) = \sum_{\nu=1}^{n} \lambda_\nu g(t_\nu) + R_n(g), \\ &R_n(g) = 0, \quad \text{all } g \in \mathbb{Q}_m \oplus \mathbb{P}_{2n-1-m}. \end{aligned} \tag{2.9}$$

Conversely, if (2.9) holds with $t_\nu \in \text{supp}(d\lambda)$, then so does (2.7) with t_ν^G, w_ν^G as obtained from (2.8).

Since ζ_μ in general are complex, the "modified measure" $d\lambda/\omega_m$ in (2.7) may be complex-valued, and the existence of a Gauss formula is not ensured.

Remark 2.1. It is sometimes useful to consider in place of (2.1) (and (2.9)) the more specific quadrature rule

$$\int_{\mathbb{R}} f(t)s(t)d\lambda(t) = \sum_{\nu=1}^{n} \lambda_\nu f(t_\nu) + R_n(f), \tag{2.1'}$$

where $s(t)$ is a function that exhibits difficult behavior unrelated to the poles of f, while $d\lambda$ is one of the standard integration measures. For example, s may have an algebraic singularity near the support interval of $d\lambda$. The theorem then remains valid if (2.7) is replaced by

$$\int_{\mathbb{R}} p(t)\frac{s(t)d\lambda(t)}{\omega_m(t)} = \sum_{\nu=1}^{n} w_\nu^G p(t_\nu^G), \quad \text{all } p \in \mathbb{P}_{2n-1}. \tag{2.7'}$$

We take advantage of this option in Examples 4.3 and 4.4 dealing with generalized Fermi-Dirac and Bose-Einstein integrals. See also Example 4.5.

2.2 Special Cases

The hypothesis of the theorem can be verified in a number of special cases. We assume $m > 0$ and $d\lambda$ a positive measure all of whose moments exist.

Case 1 (Simple Real Poles). All $s_\mu = 1$ (hence $M = m$) and $\zeta_\mu = \xi_\mu$ real and distinct, with $\xi_\mu \neq 0$, $\mu = 1, 2, \ldots, m$.

In this case,

$$\omega_m(t) = \prod_{\mu=1}^{m}(1 + \xi_\mu t).$$

Since ω_m does not vanish on $\text{supp}(d\lambda)$, and $\text{supp}(d\lambda)$ is an interval, the polynomial ω_m has constant sign on $\text{supp}(d\lambda)$, so that $d\lambda/\omega_m$ is a definite measure. All of its moments evidently exist. The Gauss formula (2.7) therefore exists uniquely for any n, and all its nodes are in $\text{supp}(d\lambda)$.

Case 2 (Simple Conjugate Complex Poles). All $s_\mu = 1$, m even, and

$$\zeta_\mu = \xi_\nu + i\eta_\nu, \quad \zeta_{\mu+1} = \xi_\nu - i\eta_\nu \quad (\nu = 1 + \lfloor \mu/2 \rfloor), \quad \mu \text{ (odd)} = 1, 3, \ldots, m-1,$$

where all ξ_ν are real and all η_ν positive.

In this case,

$$\omega_m(t) = \prod_{\nu=1}^{m/2}[(1 + \xi_\nu t)^2 + \eta_\nu^2 t^2], \quad 2 \leq m \text{ (even)} \leq 2n.$$

This is clearly positive on $\mathbb{R}$, and $d\lambda/\omega_m$ is a positive measure. Again, the hypotheses of the theorem are satisfied.

Cases 3 and 4. Analogous to Case 1 and Case 2, respectively, but with all $s_\mu = 2$ (hence $m = 2M$).

Case 5 (Simple Conjugate Complex Poles Plus a Real Simple Pole). All $s_\mu = 1$, m (odd) ≥ 3, and

$$\zeta_\mu = \xi_\nu + i\eta_\nu, \quad \zeta_{\mu+1} = \xi_\nu - i\eta_\nu \quad (\nu = 1 + \lfloor \mu/2 \rfloor), \quad \mu \text{ (odd)} = 1, 3, \ldots, m-2,$$

$$\zeta_m = \xi_m \in \mathbb{R},$$

with $\xi_\nu \in \mathbb{R}$, $\eta_\nu > 0$.

Here,

$$\omega_m(t) = (1 + \xi_m t) \prod_{\nu=1}^{(m-1)/2} [(1 + \xi_\nu t)^2 + \eta_\nu^2 t^2],$$

which again is sign-preserving on $\text{supp}(d\lambda)$.

Case 6. Analogous to Case 5, but with two simple poles instead of one.

Note that all weights λ_ν in (2.9) are positive in each of these cases. This is clear if $\omega_m(t) > 0$ on $\text{supp}(d\lambda)$, but also if $\omega_m(t) < 0$, since then $w_\nu^G < 0$ in (2.7) and thus $\lambda_\nu > 0$ by (2.8).

Many other special cases could of course be considered. Van Assche and Vanherwegen [1993], for example, consider real poles, all of multiplicity 2 except for one, which is simple, and they take n such that $2n - 1 - m = 0$, so that (2.9) is exact on $\mathbb{Q}_m \oplus \mathbb{P}_0$. Rovba [1996], on the other hand, considers mixtures of real and conjugate complex poles, all of order 2, and a value of n such that $2n - 1 - m = 1$, making $R_n(g) = 0$ if $g \in \mathbb{Q}_m \oplus \mathbb{P}_1$.

2.3 Remainder Term

If ω_m has constant sign on $\text{supp}(d\lambda)$, it follows from a well-known expression for the remainder term of (ordinary) Gaussian quadrature rules that (cf. Gautschi [1993a, Eq. (1.18)])

$$R_n(g) = \gamma_{n,m}[\omega_m g]^{(2n)}(\tau), \quad \tau \in \text{int}(\text{supp}(d\lambda)), \tag{2.10}$$

where

$$\gamma_{n,m} = \frac{1}{(2n)!} \int_{\mathbb{R}} [\hat{\pi}_{2n}(t)]^2 \frac{d\lambda(t)}{\omega_m(t)}, \tag{2.11}$$

and

$$\hat{\pi}_{2n}(\cdot) = \pi_{2n}\left(\cdot\,; \frac{d\lambda}{\omega_m}\right)$$

is the monic orthogonal polynomial of degree $2n$ relative to the modified measure $d\lambda/\omega_m$.

For measures $d\lambda$ supported on a finite interval, a bound on the error can also be obtained in terms of best rational approximation (cf. Rovba [1996]). Using the partial fraction decomposition

$$\frac{p(t)}{\omega_m(t)} = \sum_{\mu=1}^{M} \sum_{s=1}^{s_\mu} \frac{c_{s,\mu}}{(1 + \zeta_\mu t)^s} + p_0(t),$$

where $p \in \mathbb{P}_{2n-1}$ is an arbitrary polynomial of degree $\leq 2n - 1$ and $p_0 \in \mathbb{P}_{2n-1-m}$, one has from (2.3) and (2.4) that

$$\int_{\mathbb{R}} \frac{p(t)}{\omega_m(t)} d\lambda(t) = \sum_{\nu=1}^{n} \lambda_\nu \frac{p(t_\nu)}{\omega_m(t_\nu)}, \quad p \in \mathbb{P}_{2n-1}. \tag{2.12}$$

Let

$$\mathscr{E}_{n,m}(g) = \inf_{p \in \mathbb{P}_{2n-1}} \left\| \frac{p}{\omega_m} - g \right\|_\infty = \left\| \frac{p^*}{\omega_m} - g \right\|_\infty, \tag{2.13}$$

where $\|\cdot\|_\infty$ is the maximum norm on the support interval of $d\lambda$. Then, by a standard argument, using (2.12), one gets

$$|R_n(g)| = \left| \int_{\mathbb{R}} g(t) d\lambda(t) - \sum_{\nu=1}^{n} \lambda_\nu g(t_\nu) \right|$$

$$= \left| \int_{\mathbb{R}} \left[g(t) - \frac{p^*(t)}{\omega_m(t)} \right] d\lambda(t) - \sum_{\nu=1}^{n} \lambda_\nu \left[g(t_\nu) - \frac{p^*(t_\nu)}{\omega_m(t_\nu)} \right] \right|,$$

so that, by the positivity of the λ_ν,

$$|R_n(g)| \leq \mathscr{E}_{n,m}(g) \left\{ \int_{\mathbb{R}} d\lambda(t) + \sum_{\nu=1}^{n} \lambda_\nu \right\}. \tag{2.14}$$

If $2n - 1 - m \geq 0$, then $\sum_{\nu=1}^{n} \lambda_\nu = \int_{\mathbb{R}} d\lambda(t)$, and the bound simplifies to

$$|R_n(g)| \leq 2\mathscr{E}_{n,m}(g) \int_{\mathbb{R}} d\lambda(t), \quad m \leq 2n - 1. \tag{2.14$'$}$$

When $m = 0$, this becomes the well-known error estimate for (ordinary) Gaussian quadrature in terms of best polynomial approximation.

3. CONSTRUCTION OF THE RATIONAL GAUSS FORMULA

We now describe a discretization procedure for computing the quadrature formula (2.7). As in (2.3), we assume

$$0 \le m \le 2n, \tag{3.1}$$

where the case $m = 0$ reduces to a method that was previously discussed in Gautschi [1994, Section 4.3]. We also assume that the zeros of ω_m are either real or occurring in conjugate complex pairs, and we asssume that ω_m has constant sign on $\mathrm{supp}(d\lambda)$.

It suffices to compute the recursion coefficients for the measure

$$d\hat{\lambda}(t) = \frac{d\lambda(t)}{\omega_m(t)}, \tag{3.2}$$

that is, the coefficients $\hat{\alpha}_k$, $\hat{\beta}_k$, $k = 0, 1, \ldots, n-1$, in the three-term recurrence relation

$$\hat{\pi}_{k+1}(t) = (t - \hat{\alpha}_k)\hat{\pi}_k(t) - \hat{\beta}_k\hat{\pi}_{k-1}(t), \quad \hat{\pi}_{-1}(t) = 0, \quad \hat{\pi}_0(t) = 1,$$

satisfied by the (monic) polynomials $\hat{\pi}_k$ orthogonal with respect to $d\hat{\lambda}$. In terms of the coefficients $\hat{\alpha}_k$, $\hat{\beta}_k$ the Gauss formula

$$\int_{\mathbb{R}} f(t)d\hat{\lambda}(t) = \sum_{\nu=1}^{n} w_\nu^G f(t_\nu^G) + R_n^G(f) \tag{3.3}$$

can be computed by well-known eigenvalue/eigenvector techniques (cf. Gautschi [1994, Section 6.1]). If we also compute $\hat{\beta}_n$, we can obtain the coefficient $\gamma_{n,m}$ in the error formula (2.10), (2.11) by means of

$$\gamma_{n,m} = \frac{\hat{\beta}_0\hat{\beta}_1 \cdots \hat{\beta}_n}{(2n)!}. \tag{3.4}$$

3.1 The Discretization Method

The desired recursion coefficients $\hat{\alpha}_k$, $\hat{\beta}_k$ can be expressed in terms of the inner product

$$(u, v) = \int_{\mathbb{R}} u(t)v(t)d\hat{\lambda}(t) \tag{3.5}$$

by the well-known formulae

$$\hat{\alpha}_k = \frac{(t\hat{\pi}_k, \hat{\pi}_k)}{(\hat{\pi}_k, \hat{\pi}_k)}, \quad 0 \le k \le n,$$

$$\hat{\beta}_0 = (\hat{\pi}_0, \hat{\pi}_0), \quad \hat{\beta}_k = \frac{(\hat{\pi}_k, \hat{\pi}_k)}{(\hat{\pi}_{k-1}, \hat{\pi}_{k-1})}, \quad 1 \le k \le n. \tag{3.6}$$

The basic idea now is simply to discretize the inner product in (3.5),

$$(u, v) \approx (u, v)_N := \sum_{k=1}^{N} \omega_k^{(N)} u(\tau_k^{(N)}) v(\tau_k^{(N)}), \quad N > n, \tag{3.7}$$

and to approximate the $\hat{\alpha}_k$, $\hat{\beta}_k$ by the formulae (3.6) in which $(\cdot\,, \cdot)$ is replaced by $(\cdot\,, \cdot)_N$ and $\hat{\pi}_k$ by $\hat{\pi}_{k,N}$, the polynomials orthogonal with respect to the inner product (3.7):

$$\hat{\alpha}_k \approx \hat{\alpha}_{k,N}, \quad \hat{\beta}_k \approx \hat{\beta}_{k,N}, \quad 0 \le k \le n. \tag{3.8}$$

The weights $\omega_k^{(N)}$ in (3.7) are assumed to have constant sign (the same sign as ω_m on $\mathrm{supp}(d\lambda)$). In effect we are approximating the orthogonal polynomials $\hat{\pi}_k$ by means of the polynomials orthogonal with respect to the discrete inner product $(\cdot\,, \cdot)_N$. For any reasonable choice of discretization, the procedure is expected to converge as $N \to \infty$.

The recursion coefficients $\hat{\alpha}_{k,N}$, $\hat{\beta}_{k,N}$ for the discrete inner product in (3.7) can be computed by either the Stieltjes procedure or the Lanczos algorithm, which are implemented in the ORTHPOL[1] routines `sti` and `lancz`, respectively (cf. Gautschi [1994, Sections 4.1 and 4.2]).

Assuming that the recursion coefficients α_k, β_k of the measure $d\lambda$ are known, we propose to do the discretization by applying the Gauss formula for $d\lambda$ to the integral in (3.3), i.e., by taking

$$\tau_k^{(N)} = t_k^{(N)}(d\lambda), \quad \omega_k^{(N)} = \frac{w_k^{(N)}(d\lambda)}{\omega_m(\tau_k^{(N)})}, \quad k = 1,2, \ldots, N, \tag{3.9}$$

where $t_k^{(N)}(d\lambda)$ are the zeros of the orthogonal polynomial $\pi_N(\cdot\,; d\lambda)$ and $w_k^{(N)}(d\lambda)$ the respective Christoffel numbers.

[1]Here and in the following, we refer to the routines in Algorithm 726 of Gautschi [1994] as ORTHPOL routines.

3.2 The Basic Routines

The discretization method described in Section 3.1 for generating (approximations to) the recursion coefficients (3.6) is implemented in the routine `abmod`, the computation of the desired quadrature rule in `gqrat`.

The presence of real poles very close to the support interval of $d\lambda$ causes the method of Section 3.1 to converge slowly as $N \to \infty$. It is expedient, therefore, to treat these "difficult" poles separately, after all the other poles (or approximations thereof) have been incorporated. We discuss the necessary procedures in the cases where there is one difficult simple pole, and a pair of such poles located symmetrically with respect to the origin. In applications, these are probably the cases of most interest. As a preparatory step, we describe in the next subsection certain nonlinear recursive algorithms related to modifying a measure by division with appropriate factors.

3.3 Nonlinear Modification Algorithms

Let $d\tilde{\lambda}$ be a given measure supported on $\mathrm{supp}(d\lambda)$. (In special cases, $d\tilde{\lambda} = d\lambda$.) We assume that sufficiently many recursion coefficients $\tilde{\alpha}_k$, $\tilde{\beta}_k$ of $d\tilde{\lambda}$ be known. The problem then is to compute the recursion coefficients $\hat{\alpha}_k$, $\hat{\beta}_k$, respectively, $\hat{\hat{\alpha}}_k$, $\hat{\hat{\beta}}_k$, for the modified measures

$$d\hat{\lambda}(t) = \frac{d\tilde{\lambda}(t)}{t - x} \quad \text{and} \quad d\hat{\hat{\lambda}}(t) = \frac{d\tilde{\lambda}(t)}{t^2 - x^2}, \tag{3.10}$$

where in either case $x \in \mathbb{R}$ is close to, but outside of, the support interval of $d\tilde{\lambda}$.

With regard to the first modification in (3.10), the solution has been given in Gautschi [1982] and implemented in the ORTHPOL routine `chri` under option 4. (For stability aspects, see also Galant [1992].) The method requires knowledge of the quantity

$$h(x;d\tilde{\lambda}) = \int_{\mathbb{R}} \frac{d\tilde{\lambda}(t)}{x - t}, \tag{3.11}$$

whose computation will be discussed in the next subsection. Once $h(x;d\tilde{\lambda})$ has been obtained, the algorithm is (cf. Gautschi [1982, Eq. (5.2)])

$$\hat{\alpha}_0 = x - \frac{\tilde{\beta}_0}{h(x;d\tilde{\lambda})}, \quad \hat{\beta}_0 = -h(x;d\tilde{\lambda}), \quad q_0 = \hat{\alpha}_0 - x$$

$$\left.\begin{array}{c} e_{k-1} = \tilde{\alpha}_{k-1} - x - q_{k-1} \\ \hat{\beta}_k = q_{k-1} e_{k-1} \\ q_k = \tilde{\beta}_k / e_{k-1} \\ \hat{\alpha}_k = q_k + e_{k-1} + x \end{array}\right\} k = 1, 2, \ldots, n-1. \tag{3.12}$$

The second modification in (3.10) can be accomplished by two successive applications of the first,

$$d\hat{\lambda}(t) = \frac{d\tilde{\lambda}(t)}{t - x}, \quad d\hat{\hat{\lambda}}(t) = \frac{d\hat{\lambda}(t)}{t + x} = \frac{d\tilde{\lambda}(t)}{t^2 - x^2}. \tag{3.13}$$

The result can be written in the form

$$\hat{\hat{\alpha}}_0 = x \frac{h(x;d\tilde{\lambda}) + h(-x;d\tilde{\lambda})}{h(x;d\tilde{\lambda}) - h(-x;d\tilde{\lambda})}, \quad \hat{\hat{\beta}}_0 = -\frac{1}{2x}(h(x;d\tilde{\lambda}) - h(-x;d\tilde{\lambda}))$$

$$q_0 = -\frac{\tilde{\beta}_0}{h(x;d\tilde{\lambda})}, \quad \hat{q}_0 = -\frac{h(x;d\tilde{\lambda})}{\hat{\hat{\beta}}_0}$$

$$\hat{e}_{-1} = 0$$

$$\left.\begin{array}{c} \hat{e}_{k-1} = q_{k-1} + e_{k-2} + 2x - \hat{q}_{k-1} \\ \hat{\hat{\beta}}_k = \hat{q}_{k-1} \hat{e}_{k-1} \\ e_{k-1} = \tilde{\alpha}_{k-1} - x - q_{k-1} \\ \hat{q}_k = q_{k-1} e_{k-1} / \hat{e}_{k-1} \\ \hat{\hat{\alpha}}_k = \hat{q}_k + \hat{e}_{k-1} - x \\ q_k = \tilde{\beta}_k / e_{k-1} \end{array}\right\} k = 1, 2, \ldots, n-1, \tag{3.14}$$

with the same definition of $h(x;d\tilde{\lambda})$ as in (3.11).

Both algorithms (3.12) and (3.14) seem to be stable only for x very close to the support interval of $d\tilde{\lambda}$ and become rapidly unstable as x moves away from that interval. Thus, they should only be used when x is very close to the support of $d\tilde{\lambda}$, which of course is precisely the case of interest in our application.

Algorithms (3.12) and (3.14) are implemented in the routine `gchris`.

3.4 Real Poles Near the Support of $d\lambda$

Since we assume only one, or a symmetric pair, of difficult real poles, we can use the algorithms (3.12), respectively, (3.14), once the quantity $h(x;d\tilde{\lambda})$ in (3.11) is known. Here,

$$d\tilde{\lambda}(t) = \frac{d\lambda(t)}{\tilde{\omega}_m(t)}, \tag{3.15}$$

where $\tilde{\omega}_m(t)$ is the polynomial (2.6) with the factors corresponding to the difficult poles removed. If, as it sometimes happens, the remaining poles vary slightly as a function of the real poles that are being removed, we may, in the interest of efficiency, replace them by fixed poles located nearby. In any case, the recursion coefficients $\tilde{\alpha}_k$, $\tilde{\beta}_k$, $k = 0, 1, \ldots, n$, for $d\tilde{\lambda}$ are easily computed by means of the routine `abmod` (where n is to be replaced by $n + 1$). It suffices, therefore, to discuss the computation of $h(x;d\tilde{\lambda})$.

One possible way to do this is by interpreting $h(x;\tilde{\lambda})$ as the first member (for $k = 0$) of the sequence

$$\rho_k(x;d\tilde{\lambda}) = \int_{\mathbb{R}} \frac{\pi_k(t;d\tilde{\lambda})}{x - t} d\tilde{\lambda}(t), \quad k = 0, 1, 2, \ldots, \tag{3.16}$$

which is known to be a minimal solution of the recurrence relation satisfied by the orthogonal polynomials $\pi_k(x;d\tilde{\lambda})$ (cf. Gautschi [1981]). Thus, known iterative algorithms (for example, Gautschi [1981, Eq. (5.2)]; see also (3.20)) can be used to compute $h(x;d\tilde{\lambda}) = \rho_0(x;d\tilde{\lambda})$. The drawback of this approach is the necessity of computing as many recursion coefficients $\tilde{\alpha}_k$, $\tilde{\beta}_k$ as are required for the iteration process to converge.

We prefer a technique that relies on the original measure $d\lambda$. It is based on the decomposition of $\tilde{\omega}_m^{-1}$ in (3.15) into partial fractions. To describe it, we assume for simplicity that $\tilde{\omega}_m$ has only simple zeros, and we consider separately the case where all zeros of $\tilde{\omega}_m$ are real and the case where they are all complex occurring in conjugate complex pairs.

In the first case,

$$\frac{1}{\tilde{\omega}_m(t)} = \frac{1}{\prod_\nu (1 + \xi_\nu t)} = \sum_\nu \frac{c_\nu / \xi_\nu}{\frac{1}{\xi_\nu} + t}, \tag{3.17}$$

where the sum and product are extended over all "benign" poles, or approximations thereof, and

$$c_\nu = \prod_{\mu \neq \nu} \frac{1}{1 - \frac{\xi_\mu}{\xi_\nu}}, \tag{3.18}$$

empty products (if there is only *one* or *no* benign pole) being defined as having the value 1. Then

$$h(x;d\tilde{\lambda}) = \int_{\mathbb{R}} \frac{d\lambda(t)}{(x-t)\tilde{\omega}_m(t)} = \sum_{\nu} \int_{\mathbb{R}} \frac{1}{x-t} \frac{c_\nu / \xi_\nu}{\dfrac{1}{\xi_\nu} + t} d\lambda(t).$$

Since

$$\frac{1}{x-t} \frac{1}{\dfrac{1}{\xi_\nu} + t} = \frac{\xi_\nu}{1 + \xi_\nu x} \left(\frac{1}{x-t} + \frac{1}{\dfrac{1}{\xi_\nu} + t} \right),$$

we get

$$h(x;d\tilde{\lambda}) = \sum_{\nu} \frac{c_\nu}{1 + \xi_\nu x} [h(x;d\lambda) - h(-1/\xi_\nu;d\lambda)]. \tag{3.19}$$

Both terms in brackets can be computed by the iterative process alluded to above, which for $z \notin \operatorname{supp}(d\lambda)$ takes the form

$$h(z;d\lambda) = \rho_0(z;d\lambda) = \lim_{\kappa \to \infty} r_{-1}^{[\kappa]}(z), \tag{3.20}$$

$$r_\kappa^{[\kappa]}(z) = 0, \quad r_{k-1}^{[\kappa]}(z) = \frac{\beta_k}{z - \alpha_k - r_k^{[\kappa]}(z)}, \quad k = \kappa, \kappa - 1, \ldots, 1, 0.$$

It converges rapidly if z is a good distance away from the support interval, as is the case for $z = -1/\xi_\nu$, one of the benign poles of (3.17). For $z = x$, which is close to the support interval, convergence may be slow, but the recurrence relation involved is the one for the original measure $d\lambda$ for which the recursion coefficients are usually easy to generate. In some cases, such as the Lebesgue measure $d\lambda(t) = dt$ (cf. Examples 4.1 and 4.2), the quantity $h(x;d\lambda)$ is known explicitly; in others (cf. Example 4.4), it may be expressed analytically. The procedure (3.20) (more generally for ρ_k) is implemented in the ORTHPOL routine `knum`.

The case of complex (simple) poles is similar, but a bit more complicated. Here (cf. Gautschi [1993a, Eqs. (2.11)–(2.13)])

$$\frac{1}{\tilde{\omega}_m(t)} = \sum_{\nu} \frac{c_\nu + d_\nu t}{\left(t + \dfrac{1}{\zeta_\nu}\right)\left(t + \dfrac{1}{\bar{\zeta}_\nu}\right)}, \tag{3.21}$$

the sum being extended over all pairs of conjugate complex poles, where

$$c_\nu = \frac{1}{\operatorname{Im}\zeta_\nu} \operatorname{Im}\left(\frac{1}{\bar{\zeta}_\nu} p_\nu\right), \quad d_\nu = \frac{1}{\operatorname{Im}\zeta_\nu} \operatorname{Im} p_\nu,$$

and

$$p_\nu = \prod_{\mu \neq \nu} \frac{1}{\left(1 - \frac{\zeta_\mu}{\zeta_\nu}\right)\left(1 - \frac{\overline{\zeta}_\mu}{\zeta_\nu}\right)}.$$

There follows

$$h(x;d\tilde{\lambda}) = \int_{\mathbb{R}} \frac{d\lambda(t)}{(x - t)\tilde{\omega}_m(t)} = \sum_\nu \int_{\mathbb{R}} \frac{1}{x - t} \frac{c_\nu + d_\nu t}{\left(t + \frac{1}{\zeta_\nu}\right)\left(t + \frac{1}{\overline{\zeta}_\nu}\right)} d\lambda(t),$$

where

$$\frac{1}{x - t} \frac{c_\nu + d_\nu t}{\left(t + \frac{1}{\zeta_\nu}\right)\left(t + \frac{1}{\overline{\zeta}_\nu}\right)} = \frac{A_\nu}{x - t} + \frac{B_\nu}{t + \frac{1}{\zeta_\nu}} + \frac{C_\nu}{t + \frac{1}{\overline{\zeta}_\nu}} \tag{3.22}$$

with

$$A_\nu = \frac{c_\nu + d_\nu x}{\left|x + \frac{1}{\zeta_\nu}\right|^2}, \quad B_\nu = \frac{i}{2\mathrm{Im}\left(\frac{1}{\zeta_\nu}\right)} \frac{c_\nu - \frac{d_\nu}{\zeta_\nu}}{x + \frac{1}{\zeta_\nu}}, \quad C_\nu = \overline{B}_\nu.$$

Thus,

$$h(x;d\tilde{\lambda}) = \sum_\nu (A_\nu\, h(x;d\lambda) - 2\mathrm{Re}[B_\nu\, h(-1/\zeta_\nu;d\lambda)]). \tag{3.23}$$

Both terms in the sum can again be computed as discussed above.

4. EXAMPLES

In the following examples, the driver program names refer to files of the same name which form a part of the available CALGO code.

Example 4.1: $I_1(\omega) = \int_{-1}^{1} (\pi t/\omega)/(\sin(\pi t/\omega))dt$, $\omega > 1$. The integrand has simple real poles located at the integer multiples of ω. If we take the $m(= M)$ poles closest to (and on either side of) the interval $[-1, 1]$ as those determining the space $\mathbb{Q}_m$ of rational functions (cf. (2.4)), we are led to define

$$\xi_\nu = \frac{(-1)^\nu}{\lfloor(\nu + 1)/2\rfloor\omega}, \quad \nu = 1, 2, \ldots, m. \tag{4.1}$$

(Symmetry suggests an even value of m.) For ω not very close to 1, all poles are at a fair distance from the interval $[-1, 1]$, and the discretization method of Section 3.1 provides a satisfactory means of computing the desired n-point Gauss-type quadrature formula. Typical choices of m are $m = 2n$, $m = 2\lfloor (n + 1)/2 \rfloor$, and $m = 2$.

The driver program `driver1` implements this approach and computes rational n-point Gauss quadrature approximations to the integral $I_1(\omega)$ for $\omega = 1.1$ and $n = 1, 2, \ldots, n_{max}$. The core of the program, generating the Gauss-type formula, is as follows:

```
      call recur(ncapm,1,0.,0.,a,b,ierr)
      m=2*n
c     m=2*((n+1)/2)
c     m=2
c     m=0
      sgn=1.
      do 10 mu=1,m
        sgn=-sgn
        xir(mu)=sgn/(om*float((mu+1)/2))
        xii(mu)=0.
        is(mu)=1
   10 continue
      call abmod(n+1,ncapm,m,eps,irout,a,b,xir,xii,is,alpha,
     *  beta,ncap,kount,ierab,be,x,w,e,p0,p1,p2)
      call gqrat(n,m,alpha,beta,xir,xii,is,zg,wg,const,
     *    iergq,e)
```

The first command calls upon the ORTHPOL routine `recur` to provide sufficiently many (`ncapm`) recursion coefficients of the Lebesgue measure $d\lambda = dt$. The choice `ncapm` = 50 (100 in double precision) turned out to be adequate in all cases. The error tolerance `eps` was taken to be the machine precision multiplied by 10^2, and we used both the Stieltjes procedure (`irout` = 1) and the Lanczos algorithm (`irout` = 0) in the routine `abmod`. The results were the same; some of them, including relative errors and the error constants in the quadrature rules, are shown in Table I.

It can be seen that incorporating the maximum number $m = 2n$ of poles produces the best results. The cases $m = 2\lfloor (n + 1)/2 \rfloor$ and $m = 2$ give results of comparable accuracy, while, as expected, $m = 0$ (ordinary Gauss-Legendre quadrature) is distinctly inferior.

For $\omega > 1$ very close to $[-1, 1]$, it is better to treat the two poles $\pm\omega$ separately as described in Section 3.4. More precisely, we define $d\tilde{\lambda}$ as in (3.15), where $\tilde{\omega}_m(t) = \Pi_{\nu=1}^{m-2}(1 + \tilde{\xi}_\nu t)$, $m \geq 2$ ($\tilde{\omega}_m(t) = 1$ if $m = 2$), and $\tilde{\xi}_\nu$ are approximations to $\xi_{\nu+2}$ obtained by letting $\omega = 1$:

$$\tilde{\xi}_\nu = \frac{(-1)^\nu}{\lfloor (\nu + 3)/2 \rfloor}, \quad \nu = 1, 2, \ldots, m - 2 \tag{4.2}$$

Since (approximately)

Table I.

n	m	Integral	Relative Error	Error Constant
1	2	0.3348973E+01	0.2504E+00	0.8161E+00
2	4	0.4436920E+01	0.6906E-02	0.1730E-01
3	6	0.4467407E+01	0.8201E-04	0.1524E-03
4	8	0.4467774E+01	0.5997E-07	0.7027E-06
1	2	0.3348973E+01	0.2504E+00	0.8161E+00
2	2	0.4373305E+01	0.2114E-01	0.1553E-01
3	4	0.4466662E+01	0.2487E-03	0.1452E-03
4	4	0.4467742E+01	0.7091E-05	0.6524E-06
5	6	0.4467768E+01	0.1221E-05	0.1910E-08
6	6	0.4467776E+01	0.5936E-06	0.3633E-11
1	2	0.3348973E+01	0.2504E+00	0.8161E+00
2	2	0.4373305E+01	0.2114E-01	0.1553E-01
3	2	0.4461204E+01	0.1471E-02	0.1297E-03
4	2	0.4467364E+01	0.9173E-04	0.5826E-06
5	2	0.4467741E+01	0.7198E-05	0.1627E-08
6	2	0.4467773E+01	0.4675E-07	0.3094E-11
1	0	0.2000000E+01	0.5523E+00	0.3333E+00
2	0	0.3307903E+01	0.2596E+00	0.7407E-02
3	0	0.3956728E+01	0.1144E+00	0.6349E-04
4	0	0.4248042E+01	0.4918E-01	0.2879E-06
5	0	0.4374579E+01	0.2086E-01	0.8079E-09
6	0	0.4428596E+01	0.8769E-02	0.1541E-11

$$\omega_m(t) = \left(1 - \frac{1}{\omega^2}\, t^2\right)\tilde{\omega}_m(t) = -\frac{1}{\omega^2}(t^2 - \omega^2)\tilde{\omega}_m(t),$$

and $d\lambda(t) = dt$, we have

$$d\hat{\lambda}(t) = \frac{dt}{\omega_m(t)} = -\omega^2\frac{dt}{(t^2 - \omega^2)\tilde{\omega}_m(t)} = -\omega^2\frac{d\tilde{\lambda}(t)}{t^2 - \omega^2}, \tag{4.3}$$

so that the algorithm (3.14) is applicable with $x = \omega$. The quantity $h(x;d\tilde{\lambda})$ required in this algorithm is given by (3.19) and (3.18), where the ξ_ν have to be replaced by $\tilde{\xi}_\nu$ and where $h(x;d\lambda) = h(x;dt)$ is computable explicitly. There follows

$$h(x;d\tilde{\lambda}) = h(x;dt) = \log\left|\frac{x+1}{x-1}\right| \quad \text{if } m = 2,$$

$$h(x;d\tilde{\lambda}) = \sum_{\nu=1}^{m-2}\frac{c_\nu}{1 + \tilde{\xi}_\nu x}\log\left|\frac{(x+1)(\tilde{\xi}_\nu + 1)}{(x-1)(\tilde{\xi}_\nu - 1)}\right| \quad \text{if } m > 2. \tag{4.4}$$

Table II.

n	m	Integral	Relative Error	Error Constant
1	2	0.7608957E+01	0.4115E+00	0.2810E+01
2	4	0.1282002E+02	0.8449E-02	0.3754E-01
3	6	0.1292781E+02	0.1120E-03	0.3170E-03
4	8	0.1292914E+02	0.9031E-05	0.1432E-05
5	10	0.1292918E+02	0.6228E-05	0.4018E-08
1	1	0.7608957E+01	0.4115E+00	0.2810E+01
2	2	0.1255069E+02	0.2928E-01	0.3384E-01
3	3	0.1290194E+02	0.2112E-02	0.2810E-03
4	4	0.1292901E+02	0.1936E-04	0.1310E-05
5	5	0.1292916E+02	0.7334E-05	0.3707E-08
6	6	0.1292919E+02	0.5048E-05	0.7184E-11
1	4	0.9417684E+01	0.2716E+00	0.3617E+01
2	4	0.1282002E+02	0.8449E-02	0.3754E-01
3	4	0.1292480E+02	0.3450E-03	0.2997E-03
4	4	0.1292901E+02	0.1936E-04	0.1310E-05
5	4	0.1292917E+02	0.6596E-05	0.3601E-08
6	4	0.1292918E+02	0.6080E-05	0.6781E-11
1	2	0.7608957E+01	0.4115E+00	0.2810E+01
2	2	0.1255069E+02	0.2928E-01	0.3384E-01
3	2	0.1290019E+02	0.2248E-02	0.2627E-03
4	2	0.1292699E+02	0.1750E-03	0.1144E-05
5	2	0.1292900E+02	0.2024E-04	0.3139E-08
6	2	0.1292914E+02	0.9326E-05	0.5907E-11
7	2	0.1292916E+02	0.7850E-05	0.8083E-14

This is implemented in the driver program `driver2`.

Selected results for $\omega = 1.001$, with the other parameters being the same as in `driver1`, are shown in Table II.

The results are very similar in quality to those in `driver1`. Both routines were also run in double precision for $\omega = 1.1,\ 1.05,\ 1.01,\ 1.005,\ 1.001$, and $m = 2n$, and produced satisfactory results except for the case $\omega = 1.001$ with the first driver, where the discretization procedure for several values of $n \leq 10$ did not converge, not even with `ncapm` = 1000. The first driver, however, was much slower than the second by factors varying from about 60 to 9000.

Example 4.2: $I_2(\omega) = \int_0^1 (t^{-1/2}\Gamma(1 + t)/(t + \omega))dt$, $\omega > 0$. Here, $d\lambda(t) = t^{-1/2}dt$, and the integrand has poles at $-\omega$ and at the negative natural numbers. Accordingly, we take

$$\xi_\nu = \frac{1}{\nu}, \quad \nu = 1, 2, \ldots, m - 1; \quad \xi_m = \frac{1}{\omega}.$$

Unless ω is very small, the discretization method of Section 3.1 is expected to work well. It is implemented in the driver program `driver3`, the core of

Table III.

n	m	Integral	Relative Error	Error Constant
1	2	0.2485091E+01	0.2665E-01	0.3101E-01
2	4	0.2550594E+01	0.9960E-03	0.1217E-03
3	6	0.2553083E+01	0.2123E-04	0.2113E-06
4	8	0.2553135E+01	0.6903E-06	0.2062E-09
5	10	0.2553138E+01	0.1502E-06	0.1289E-12

which is almost identical with the one displayed in Example 4.1 except for the obvious change in the generation of the ξ_ν and the fact that we need the recursion coefficients of the Jacobi polynomials (transformed to the interval $[0,\ 1]$) with parameters $\alpha = 0$, $\beta = -1/2$.[2] Thus, it starts out with the following:

```
      call recur(ncapm,6,0.,-.5,a,b,ierr)
      do 10 k=1,ncapm
        a(k)=.5*(1.+a(k))
        b(k)=.25*b(k)
   10 continue
      b(1)=sqrt(8.)*b(1)
```

The routine `driver3` for $\omega = 0.5$, $m = 2n$, yielded the results shown in Table III.

Similar results were obtained for other values of $\omega \geq 0.5$, including integer and near-integer values, as well as for $m = n$ and $m = 2$ (even for $m = 0$, since the poles are not very close to the interval $[0,1]$).

For ω very small, the pole $-\omega$ should be treated separately, by letting $\tilde{\omega}_m(t) = \Pi_{\nu=1}^{m-1}(1 + \xi_\nu t)$, and noting that

$$\omega_m(t) = \left(1 + \frac{1}{\omega}t\right)\tilde{\omega}_m(t);$$

hence

$$\frac{d\lambda(t)}{\omega_m(t)} = \omega\frac{d\tilde{\lambda}(t)}{t + \omega}, \quad d\tilde{\lambda}(t) = \frac{d\lambda(t)}{\tilde{\omega}_m(t)}.$$

The algorithm in (3.12) is now applicable with $x = -\omega$ and $h(x;d\tilde{\lambda})$ for $x < 0$ given by (3.19). It thus remains to compute $h(x;d\lambda)$ for $x < 0$ and $d\lambda(t) = t^{-1/2}dt$ on $[0,1]$. This is elementary; we have

[2]In Example 3.2 of Gautschi [1993a], these parameters were erroneously taken to be $\alpha = -1/2$, $\beta = 0$. The numerical results reported there thus refer to a different integrand, one in which $t^{-1/2}$ is replaced by $(1 - t)^{-1/2}$, and not to $I_2(\omega)$ as claimed.

$$h(x;d\lambda) = \int_0^1 \frac{t^{-1/2}dt}{x - t} = -\int_0^1 \frac{t^{-1/2}dt}{|x| + t},$$

and the transformation $t = s^2$ gives

$$-\int_0^1 \frac{s^{-1} \cdot 2sds}{|x| + s^2} = -\frac{2}{|x|}\int_0^1 \frac{ds}{1 + \frac{s^2}{|x|}}$$

$$= -\frac{2}{|x|}\int_0^{1/\sqrt{|x|}} \frac{\sqrt{|x|}dt}{1 + t^2} = -\frac{2}{\sqrt{|x|}} \tan^{-1}\left(\frac{1}{\sqrt{|x|}}\right).$$

Thus,

$$h(x;d\tilde{\lambda}) = -\frac{2}{\sqrt{|x|}} \tan^{-1}\left(\frac{1}{\sqrt{|x|}}\right) \quad \text{if } m = 1,$$

$$h(x;d\tilde{\lambda}) = \sum_{\nu=1}^{m-1} \frac{2c_\nu}{1 + \xi_\nu x}\left\{-\frac{1}{\sqrt{|x|}} \tan^{-1}\left(\frac{1}{\sqrt{|x|}}\right) + \sqrt{\xi_\nu} \tan^{-1}\sqrt{\xi_\nu}\right\} \quad \text{if } m > 1, \quad (4.5)$$

where c_ν is given by (3.18).

The procedure is implemented in the driver program `driver4`, which computes n-point Gauss approximations, $n = 1, 2, \ldots, n_{\max}$, to $I_2(\omega)$ for $\omega = 0.001$.

As was already noted in Gautschi [1993a, p. 124], the constants c_ν in (4.5) can assume large values of different signs, which cause large cancellation errors in the summation of (4.5). It is expedient, therefore, to execute this part of the computation in double precision. The results are then as shown in Table IV.

Note that in the case $m = 2$ the sum in (4.5) consists of only one term, and cancellation cannot occur.

Example 4.3. Here we examine the Generalized Fermi-Dirac integral (cf. Gautschi [1993b])

$$F_k(\eta, \theta) = \int_0^\infty \frac{t^k \sqrt{1 + \frac{1}{2}\theta t}}{e^{-\eta+t} + 1}dt, \quad \eta \in \mathbb{R}, \theta \geq 0.$$

Table IV.

n	m	Integral	Relative Error	Error Constant
1	2	0.9650458E+02	0.2059E-02	0.2025E-03
2	4	0.9669596E+02	0.7991E-04	0.8408E-06
3	6	0.9670349E+02	0.2037E-05	0.1472E-08
4	8	0.9670369E+02	0.1392E-07	0.1441E-11
5	10	0.9670364E+02	0.4595E-06	0.9012E-15
1	1	0.9628445E+02	0.4335E-02	0.3138E-03
2	2	0.9670098E+02	0.2799E-04	0.1220E-05
3	3	0.9670364E+02	0.5383E-06	0.2111E-08
4	4	0.9670369E+02	0.1392E-07	0.2058E-11
5	5	0.9670370E+02	0.9281E-07	0.1284E-14
1	1	0.9628445E+02	0.4335E-02	0.3138E-03
2	1	0.9669199E+02	0.1210E-03	0.1789E-05
3	1	0.9670336E+02	0.3379E-05	0.3823E-08
4	1	0.9670370E+02	0.9281E-07	0.4319E-11

The values of k that are of physical interest are the half-integers $k = 1/2$, $k = 3/2$, and $k = 5/2$.

For numerical purposes, the integral is conveniently written in the form

$$F_k(\eta, \theta) = \int_0^\infty \frac{\sqrt{1 + \frac{1}{2}\theta t}}{e^{-\eta} + e^{-t}} t^k e^{-t} dt, \tag{4.6}$$

which suggests the generalized Laguerre measure $d\lambda(t) = t^k e^{-t} dt$ as the measure of integration. The poles are all simple and occur in conjugate complex pairs on the line $\mathrm{Im}\zeta = \eta$ at odd integer multiples of π away from the real axis. None of these poles is difficult. We thus take m even and put

$$\zeta_\mu = -\frac{1}{\eta + \mu i\pi}, \qquad \zeta_{\mu+1} = -\frac{1}{\eta - \mu i\pi}, \qquad \mu(\text{odd}) = 1, 3, \ldots, m-1 \tag{4.7}$$

(cf. Case 2 of Section 2.2), i.e.,

$$\xi_\nu = \frac{-\eta}{\eta^2 + (2\nu - 1)^2\pi^2}, \qquad \eta_\nu = \frac{(2\nu - 1)\pi}{\eta^2 + (2\nu - 1)^2\pi^2}, \qquad \nu = 1, 2, \ldots, m/2. \tag{4.8}$$

The discretization method of Section 3.1 is then the method of choice, where natural values of m are $m = 2n$, $m = 2\lfloor (n+1)/2 \rfloor$, and $m = 2$. This is implemented in the driver `driver5`. We ran the program for $\theta = 10^{-4}$ and $\eta = \pm 1$. In all cases `ncapm` = 100 (200 in double precision) was

Table V.

n	m	Integral	Relative Error	Error Constant
1	2	0.2754090E+00	0.5199E-01	0.2782E+00
2	4	0.2901222E+00	0.1343E-02	0.5457E-01
3	6	0.2905045E+00	0.2740E-04	0.9089E-02
4	8	0.2905124E+00	0.2197E-06	0.1394E-02
5	10	0.2905125E+00	0.3958E-06	0.2033E-03

satisfactory. The results for $k = 0.5$, $\eta = -1$ are shown in Table V; the others are similar.

Example 4.4. Here we examine the Generalized Bose-Einstein integral (cf. Gautschi [1993b])

$$G_k(\eta, \theta) = \int_0^\infty \frac{t^k \sqrt{1 + \frac{1}{2}\theta t}}{e^{-\eta+t} - 1} dt, \quad \eta < 0, \theta \geq 0,$$

with the same values of k as in Example 4.3.

More convenient, for numerical work, is the form

$$G_k(\eta, \theta) = \int_0^\infty \frac{t \sqrt{1 + \frac{1}{2}\theta t}}{e^{-\eta} - e^{-t}} t^{k-1} e^{-t} dt, \tag{4.9}$$

where a factor t was split off to make the integrand regular at $t = 0$ even if $\eta = 0$. As a consequence, the appropriate measure is now $d\lambda(t) = t^{k-1}e^{-t}dt$.

The poles are again conjugate complex on $\mathrm{Im}\zeta = \eta$, at a distance of multiples of 2π from the real axis, thus even more benign than those in Example 4.3. But in addition there is the real pole at η. Thus, m (if $\neq 0$) here is to be taken odd, and we put

$$\zeta_\mu = -\frac{1}{\eta + (\mu + 1)i\pi}, \qquad \zeta_{\mu+1} = -\frac{1}{\eta - (\mu + 1)i\pi},$$

$$\mu(\text{odd}) = 1, 3, \ldots, m - 2, \tag{4.10}$$

$$\zeta_m = -\frac{1}{\eta},$$

hence

Table VI.

n	m	Integral	Relative Error	Error Constant
1	1	0.2843094E+00	0.2512E+00	0.1600E+00
2	3	0.3767363E+00	0.7828E-02	0.1738E-01
3	5	0.3796441E+00	0.1706E-03	0.2003E-02
4	7	0.3797078E+00	0.2694E-05	0.2344E-03
5	9	0.3797086E+00	0.6533E-06	0.2762E-04
6	11	0.3797089E+00	0.2542E-07	0.3265E-05

$$\xi_\nu = \frac{-\eta}{\eta^2 + 4\nu^2\pi^2}, \quad \eta_\nu = \frac{2\nu\pi}{\eta^2 + 4\nu^2\pi^2}, \quad \nu = 1, 2, \ldots, (m-1)/2 \tag{4.11}$$

(cf. Case 5 of Section 2.2). Unless $|\eta|$ is very small, the discretization method of Section 3.1 is again an appropriate choice. It is implemented in the driver program `driver6`.

Some care is required in programing the denominator in the integral of (4.9). It is best written as $e^{-t}(e^{t-\eta} - 1)$, where the exponential in parentheses is computed by Taylor expansion if, say, $|t - \eta| \le 1$. We ran `driver6` again for $\theta = 10^{-4}$, $\eta = -1$, and for the same values of k as before. The results in Table VI for $k = 0.5$ and $m = 2n - 1$ are typical.

For small values of $|\eta|$, the pole at η should be treated separately as described in Section 3.4. We have

$$d\hat{\lambda}(t) = -\eta \frac{d\tilde{\lambda}(t)}{t - \eta}, \quad d\tilde{\lambda}(t) = \frac{d\lambda(t)}{\tilde{\omega}_m(t)}, \tag{4.12}$$

with the reciprocal of $\tilde{\omega}_m$ as given in (3.21). Since $|\eta|$ is very small, we can approximate ξ_ν and η_ν in (4.11) by

$$\xi_\nu \approx 0, \quad \eta_\nu \approx \frac{1}{2\nu\pi}, \quad \nu = 1,2, \ldots, (m-1)/2, \tag{4.13}$$

thus making $\tilde{\omega}_m$ independent of η. The first relation in (4.12) calls for the algorithm (3.12) with $x = \eta$. The quantity $h(x;d\tilde{\lambda})$ required in this algorithm is given by (3.23), which, by an elementary computation based on (3.21), (3.22), and the approximations (4.13), can be cast in the form

$$h(x;d\tilde{\lambda}) = h(x;d\lambda) \quad \text{if } m = 1,$$

$$h(x;d\tilde{\lambda}) = \sum_{\nu=1}^{(m-1)/2} \frac{2\nu\pi p_\nu}{x^2 + 4\nu^2\pi^2}(2\nu\pi h(x;d\lambda)$$

$$-\mathrm{Re}[(2\nu\pi - ix)h(2\nu\pi i;d\lambda)]) \quad \text{if } m \geq 3, \tag{4.14}$$

where

$$p_\nu = \prod_{\substack{\mu=1 \\ \mu\neq\nu}}^{(m-1)/2} \frac{\mu^2}{\mu^2 - \nu^2}. \tag{4.15}$$

The quantity $h(2\nu\pi i;d\lambda)$, for $\nu \geq 1$, can be computed by the algorithm (3.20), which converges rapidly. For $h(x;d\lambda)$, $x < 0$, we have

$$h(x;d\lambda) = \int_0^\infty \frac{t^{k-1}e^{-t}}{x - t}dt = -\int_0^\infty \frac{t^{k-1}e^{-t}}{|x| + t}dt.$$

Making the change of variables $t \mapsto |x|t$ yields

$$h(x;d\lambda) = -|x|^{k-1}\int_0^\infty \frac{t^{k-1}e^{-|x|t}}{1 + t}dt.$$

The integral on the right is expressible in terms of the incomplete gamma function as $\Gamma(k)e^{|x|}\Gamma(1 - k, |x|) = \Gamma(k)e^{|x|}\{\Gamma(1 - k) - \gamma(1 - k, |x|)\}$ (e.g., see Gradshteyn and Ryzhik [1965, Eq. 3.383.10]), which, upon using

$$\Gamma(k)\Gamma(1 - k) = \frac{\pi}{\sin(\pi k)}$$

and the power series expansion of

$$\gamma(1 - k, |x|),$$

yields

$$h(x;d\lambda) = -e^{|x|}\left(\frac{\pi}{\sin(\pi k)}|x|^{k-1} - \Gamma(k)\sum_{j=0}^\infty (-1)^j \frac{|x|^j}{(j + 1 - k)j!}\right). \tag{4.16}$$

Here, the series on the right converges rapidly when $|x|$ is small.

The driver `driver7` uses these procedures to generate n-point rational Gauss quadrature approximations, $n = 1, 2, \ldots, n_{\max}$ to $G_k(\eta, \theta)$.

The roles of `ncapm` and `kapmax` should be noted in this code. The former is an upper bound on the discretization parameter N in the discretization method to compute the recursion coefficients $\tilde{\alpha}_k$, $\tilde{\beta}_k$ for the measure $d\tilde{\lambda}$.

Table VII.

n	m	Integral	Relative Error	Error Constant
2	3	0.2201195E+01	0.7196E-02	0.3995E-04
4	7	0.2217145E+01	0.2423E-05	0.4785E-06
6	11	0.2217149E+01	0.2728E-06	0.6306E-08
2	1	0.2224269E+01	0.3211E-02	0.5409E-04
4	3	0.2217148E+01	0.9180E-06	0.7365E-06
6	5	0.2217150E+01	0.1653E-06	0.1113E-07
2	1	0.2224269E+01	0.3211E-02	0.5409E-04
4	1	0.2217136E+01	0.6402E-05	0.1703E-05
6	1	0.2217150E+01	0.1573E-06	0.7118E-07

Since all complex poles are benign, a relatively small value of ncapm should suffice. The latter, kapmax, bounds the parameter κ in the iterative procedure (3.20) to compute $h(2\nu\pi i; d\lambda)$. This parameter, too, need only be moderately large.

Table VII shows results for $\eta(= x) = -0.001$, $\theta = 10^{-4}$, $k = 0.5$. With eps chosen to be 10^2 times the machine precision, and nmax = 10, the choices ncapm = 50 and kapmax = 20 (100, respectively 50 in double precision) proved to be adequate.

The results are seen to depend little on the choice of m, because the pole at η is the dominant feature in this example. Without the special treatment of the pole η, when $\eta = -0.001$, the discretization method would not converge with ncapm set at 1000.

Example 4.5. Here we examine a radiation transfer integral,

$$H_m(c) = 2\int_0^1 P_m(x)[\sin(2\pi x)]^2 e^{-c/x}dx, \quad c > 0,$$

where P_m is the Legendre polynomial of degree m.[3] The case of large m is of particular interest in applications.

Although the approach presented here is not the best possible, it nicely illustrates a case where the pole in question has high multiplicity.

We make the change of variables

$$x = \frac{1}{1 + t/c}, \quad 0 \le t \le \infty,$$

which yields

[3]This example was kindly brought to the author's attention by Dr. Martin Gander.

$$H_m(c) = \frac{2e^{-c}}{(2\pi)^2 c}\int_0^\infty P_m\left(\frac{1}{1+t/c}\right)\left[\frac{2\pi}{1+t/c}\sin\frac{2\pi}{1+t/c}\right]^2 e^{-t}dt.$$

This suggests taking

$$d\lambda(t) = e^{-t}dt \quad \text{on } [0, \infty]$$

and

$$s(t) = \left[\frac{2\pi}{1+t/c}\sin\frac{2\pi}{1+t/c}\right]^2$$

in (2.1′). The respective n-point rational Gauss formula with

$$n = 1 + \left\lfloor \frac{1}{2}m \right\rfloor$$

is then exact on $\mathbb{S}_{2n} = \mathbb{Q}_m \oplus \mathbb{P}_{2n-1-m}$, where

$$\mathbb{Q}_m = \text{span}\{g : g(t) = (1+t/c)^{-s},\ s = 1, 2, \ldots, m\}.$$

Indeed, by the definition of n, we have

$$2n - 1 - m = \begin{cases} 1 & \text{if } m \text{ is even,} \\ 0 & \text{if } m \text{ is odd,} \end{cases}$$

so that $\mathbb{P}_{2n-1-m} \supset \mathbb{P}_0$. On the other hand, $P_m(1/(1+t/c)) \in \mathbb{Q}_m \oplus \mathbb{P}_0$.

For $m = 50$ and $c = 2$, for example, one obtains (in double precision)

$$H_{50}(2) = \begin{cases} 0.29351229600590E-07 & \text{for } n = 26, \\ 0.29351229563622E-07 & \text{for } n = 27. \end{cases}$$

While both values should be exactly the same, heavy cancellation in the quadrature sum, caused by the highly oscillatory behavior of P_{50}, resulted in a loss of about six decimal digits.

5. CONCLUDING REMARKS

Often the poles of the integrand function depend on a parameter, as for instance in Examples 4.1, 4.2, and 4.3. The process of computing the integrals by rational Gauss formulae may then be computer intensive, since the Gauss formula has to be regenerated for every new value of the parameter. It stands to reason, however, that there is no need to match the poles exactly, but that suitable approximations might suffice. Thus, we may

Table VIII. Rational Gauss Quadrature with Approximate Matching of Poles

η_0	m	n_0	$n_{\max}$	η_0	m	n_0	$n_{\max}$
				0.0	2	16	40
–1.0	2	15	30	1.0		17	49
–5.0		11	14	5.0		15	61
–10.0		9	10	10.0		21	93
				0.0	4	11	38
–1.0	4	11	28	1.0		12	47
–5.0		10	13	5.0		10	55
–10.0		8	8	10.0		12	92
				0.0	8	9	32
–1.0	8	8	27	1.0		9	45
–5.0		8	12	5.0		8	49
–10.0		7	7	10.0		9	88

use the same quadrature rule for all values of the parameter in an interval centered about some fixed parameter, the quadrature rule matching only poles corresponding to this fixed value of the parameter. In this way the discretization method—the most computer-intensive part of the procedure—has to be applied only once. We have done so already for some "difficult" poles in Examples 4.1 and 4.4, but the idea is applicable more generally.

We illustrate this in the case of the generalized Fermi-Dirac integral of Example 4.3. Here we may generate the rational Gauss formulae for some fixed value η_0 of the parameter η (i.e., let $\eta = \eta_0$ in (4.7)), and apply them to $F_k(\eta, \theta)$ in (4.6) for all η in the interval, say, $|\eta - \eta_0| \le 1/2$. To describe how well this works, let m be the number of poles matched and ε a prescribed error tolerance. Then for each $\eta = \eta_0 \pm \nu/10$, $\nu = 0, 1, \ldots, 5$, we determine the smallest integer $n \ge m/2$ such that the n-point quadrature rule approximates the integral to within a relative error $\le \varepsilon$. We let n_0 be the integer n for $\eta = \eta_0$ and $n_{\max}$ the maximum n observed for the 11 values of η in $|\eta - \eta_0| \le 1/2$. In Table VIII we show n_0 and $n_{\max}$ for selected values of η_0 and m in the case $k = 0.5$ and $\theta = 10^{-4}$. The choice of ε was generally 10^3 times the machine precision (double precision, IEEE Standard), but had to be increased to 10^6 times the machine precision for $\eta_0 = 5$ and $\eta_0 = 10$ in order to achieve convergence in the routine `dabmod` using the Stieltjes procedure with `ncapm` = 800. For higher accuracy requirements, overflow problems prevent Stieltjes' procedure to converge. (Similar convergence problems are observed with the Lanczos algorithm, presumably because of underflow of some of the weights $\omega_k^{(N)}$ in (3.7) for N very large.) It appears, therefore, that positive values of η_0 are more resistent to convergence and that the length of the η-interval in which the same quadrature rule is used may have to be shortened.

Analogous procedures are applicable to Bose-Einstein integrals for $\eta_0 < 0$, with similar results.

Table IX.

n	m	Integral	Relative Error	Relative Error	
1	2	0.3611843E+00	0.5910E-01	0.9834E-01	theta=1.
2	4	0.3832193E+00	0.1694E-02	0.3657E-02	
3	6	0.3838555E+00	0.3727E-04	0.9254E-04	
4	8	0.3838697E+00	0.1595E-06	0.1165E-05	
1	2	0.7691087E+00	0.6228E-01	0.1247E+00	theta=10.
2	4	0.8186551E+00	0.1870E-02	0.2277E-02	
3	6	0.8201542E+00	0.4188E-04	0.9285E-03	
4	8	0.8201885E+00	0.2179E-07	0.4591E-03	
1	2	0.2266731E+01	0.6191E-01	0.1249E+00	theta=100.
2	4	0.2411823E+01	0.1865E-02	0.1220E-02	
3	6	0.2416227E+01	0.4195E-04	0.3054E-02	
4	8	0.2416332E+01	0.1560E-05	0.1791E-02	
5	10	0.2416334E+01	0.2054E-05	0.1123E-02	
6	12	0.2416331E+01	0.7710E-06	0.7565E-03	

In Examples 4.3 and 4.4, following Sagar [1991], we have chosen for θ the rather small value 10^{-4}. As θ increases, convergence, even of the rational Gauss formulae, slows down on account of the square root singularity at $t = -2/\theta$. However, fast convergence can be restored, even for large values of θ, if we incorporate the function $s(t) = \sqrt{1 + (1/2)\theta t}$ into a weight function as indicated in Remark 2.1. This necessitates only one small change in the routine `abmod` (indicated in an appropriate comment statement) that amounts to replacing $\omega_k^{(N)}$ in (3.9) by $s(\tau_k^{(N)})\omega_k^{(N)}$. The effect of this change can be significant, as is shown in Table IX for the Fermi-Dirac integral with $k = 1/2$, $\eta = -1$, $m = 2n$, and $\theta = 1,\ 10,\ 100$, the last column referring to results obtained without this change.

Similar techniques, with similar success, are applicable to Bose-Einstein integrals.

ACKNOWLEDGMENTS

This work was carried out during visits to the Department of Mathematics and Applied Mathematics at the University of Padua and to the Institute of Scientific Computing at the Federal Institute of Technology (ETH) in Zurich. The author thanks the respective hosts, Profs. R. Zanovello and W. Gander, for their generous and cordial hospitality. He is also indebted to F.W.J. Olver for pointing out the relevance of the incomplete gamma function in Eq. (4.16).

REFERENCES

GALANT, D. 1992. Algebraic methods for modified orthogonal polynomials. *Math. Comput. 59*, 200 (Oct.), 541–546.

GAUTSCHI, W. 1981. Minimal solutions of three-term recurrence relations and orthogonal polynomials. *Math. Comput. 36*, 154 (Apr.), 547–554.

GAUTSCHI, W. 1982. An algorithmic implementation of the generalized Christoffel theorem. In *Numerical Integration*, G. Hämmerlin, Ed. International Series of Numerical Mathematics, vol. 57. Birkhäuser-Verlag, Basel, Switzerland, 89–106.

GAUTSCHI, W. 1993a. Gauss-type quadrature rules for rational functions. In *Numerical Integration IV*, H. Brass and G. Hämmerlin, Eds. International Series of Numerical Mathematics, vol. 112. Birkhäuser-Verlag, Basel, Switzerland, 111–130.

GAUTSCHI, W. 1993b. On the computation of generalized Fermi-Dirac and Bose-Einstein integrals. *Comput. Phys. Commun. 74*, 233–238.

GAUTSCHI, W. 1994. Algorithm 726: ORTHPOL—A package of routines for generating orthogonal polynomials and Gauss-type quadrature rules. *ACM Trans. Math. Softw. 20*, 1 (Mar. 1994), 21–62.

GRADSHTEYN, I. S. AND RYZHIK, I. M. 1965. *Table of Integrals, Series, and Products*. 4th ed. Academic Press, Inc., New York, NY.

ROVBA, E. A. 1996. Quadrature formulae of interpolatory rational type. *Dokl. Akad. Nauk Belarusi 40*, 3 (May-Jun.), 42–46. In Russian.

SAGAR, R. P. 1991. A Gaussian quadrature for the calculation of generalized Fermi-Dirac integrals. *Comput. Phys. Commun. 66*, 271–275.

VAN ASSCHE, W. AND VANHERWEGEN, I. 1993. Quadrature formulas based on rational interpolation. *Math. Comput. 61*, 204 (Oct.), 765–783.

Received: July 1997; accepted: March 1999