Pergamon

Computers Math. Applic. Vol. 33, No. 1/2, pp. 105–118, 1997

0898-1221/97 $17.00 + 0.00

PII: S0898-1221(96)00223-4

Moments in Quadrature Problems

W. GAUTSCHI
Department of Computer Sciences
Purdue University, West Lafayette, IN 47907-1398, U.S.A.

Abstract—An account is given of the role played by moments and modified moments in the construction of quadrature rules, specifically weighted Newton-Cotes and Gaussian rules. Fast and slow Lagrange interpolation algorithms, combined with Gaussian quadrature, as well as linear algebra methods based on moment equations, are described for generating Newton-Cotes formulae. The weaknesses and strengths of these methods are illustrated in concrete examples involving weight functions with and without singularities. New conjectures are formulated concerning the positivity of certain Newton-Cotes formulae for Jacobi weight functions and for the logistics weight, with numerical evidence being provided to support them. Finally, an inherent limitation is pointed out in the use of moment information to construct Gauss-type quadrature rules for the Hermite weight function on bounded or half-infinite intervals.

Keywords—Modified moments, Lagrange interpolation, Newton-Cotes formulae, Gaussian quadrature, Positivity.

1. INTRODUCTION

Moments and quadrature have been intimately connected ever since the classical work of Chebyshev and Stieltjes on continued fractions and the moment problem. Here we examine the role of moments in the numerical construction of quadrature rules. We confine attention to two extreme classes of quadrature rules—weighted Newton-Cotes and Gaussian—and consider ordinary as well as modified moments of the underlying weight function. For other quadrature rules intermediate between these two, the reader may consult [1], and for rules involving also derivatives and computed without the use of moment information, [2,3].

Section 2 is devoted to the numerical computation of weighted Newton-Cotes formulae. In Section 2.1, we describe an approach using a combination of Lagrange interpolation and Gauss quadrature, where both fast and slow methods of computing the elementary Lagrange interpolation polynomials are considered. Moments here enter only implicitly (if at all) through the use of the Gaussian quadrature procedure. In Section 2.2, the problem is solved via a system of linear algebraic equations—the moment equations—where the ordinary or modified moments appear explicitly in the right-hand vector of the system. All these approaches have their specific weaknesses: fast Lagrange interpolation tends to be unstable, slow interpolation time-consuming, while the moment equations are usually moderately to severely ill-conditioned, regardless of whether ordinary or modified moments are employed. If speed is of little concern—and in most applications to numerical quadrature this is probably the case—then the methods of choice are either ordinary (slow) Lagrange interpolation, or fast Lagrange interpolation at carefully arranged points, coupled with Gaussian quadrature.

The methods are illustrated in Section 3 for several weighted integrals involving a constant weight in Section 3.1, a weight function with a logarithmic and algebraic singularity in Section 3.2,

A preliminary version of this paper was presented at a minisymposium of the Cornelius Lanczos International Centenary Conference held in Raleigh, NC, December 12–17, 1993. The work is supported, in part, by the National Science Foundation under grant DMS-9305430.

and the Hermite weight function on half-infinite and finite intervals in Section 3.3. In Section 4, we apply our techniques to test a number of conjectures regarding the positivity of Newton-Cotes formulae. These involve general Jacobi weights, in combination with judiciously selected Jacobi nodes, and the logistics weight function on the real line using appropriate Laguerre nodes.

In Section 5, we recall the modified Chebyshev algorithm for generating orthogonal polynomials, and hence Gaussian quadrature rules, and, in connection with the Hermite weight function on finite and half-infinite intervals, point out an inherent limitation of this algorithm and thus of any moment-based procedure.

2. WEIGHTED NEWTON-COTES FORMULAE

The quadrature rule

$$\int_a^b f(x)w(x)\,dx = \sum_{\nu=1}^{n} w_\nu f(x_\nu) + R_n(f), \tag{2.1}$$

with $w \in L_1[a,b]$ a (usually) nonnegative weight function, is called a weighted Newton-Cotes formula if it is interpolatory, i.e., $R_n(f) = 0$ whenever $f \in \mathbb{P}_{n-1}$, the class of polynomials of degree $\le n-1$. The problem we wish to consider is this: given w, the integer $n \ge 1$, and the nodes $x_\nu = x_\nu^{(n)}$ (normally contained in $[a,b]$), compute the weights $w_\nu = w_\nu^{(n)}$—the *Cotes numbers* associated with the weight function w and the nodes x_ν. Of interest is also the stability constant

$$\sigma_n = \frac{\displaystyle\sum_{\nu=1}^{n} |w_\nu|}{\left|\displaystyle\sum_{\nu=1}^{n} w_\nu\right|} \ge 1, \tag{2.2}$$

which measures the susceptibility of the quadrature sum in (2.1) to rounding errors: the larger σ_n, the larger the error caused by cancellation. We discuss two methods of computation: one based on Lagrange interpolation combined with Gaussian quadrature, the other based on systems of moment equations.

2.1. Method Using Lagrange Interpolation

Since (2.1) is interpolatory, we have

$$w_\nu = \int_a^b \ell_\nu^{(n)}(x)w(x)\,dx, \qquad \nu = 1,2,\dots,n, \tag{2.3}$$

where $\ell_\nu^{(n)}$ are the elementary Lagrange interpolation polynomials belonging to the nodes $x_1, x_2, \dots, x_n$:

$$\ell_\nu^{(n)}(x) = \prod_{\substack{\mu=1\\ \mu\ne\nu}}^{n} \frac{x - x_\mu}{x_\nu - x_\mu}, \qquad \nu = 1,2,\dots,n. \tag{2.4}$$

The integral in (2.3) can be computed exactly (modulo roundoff) by an $\left\lfloor \frac{n+1}{2} \right\rfloor$-point Gaussian quadrature rule relative to the weight function w,

$$w_\nu = \sum_{k=1}^{\lfloor (n+1)/2 \rfloor} w_k^G \ell_\nu^{(n)}\left(x_k^G\right), \qquad \nu = 1,2,\dots,n. \tag{2.5}$$

For methods and software generating the required Gauss formulae, we refer to [4]. In particular, the modified Chebyshev algorithm [4, Section 3] takes the first n (respectively, $n+1$, if n is odd) moments or modified moments of w as input to produce a symmetric tridiagonal matrix whose eigenvalues and eigenvectors yield the quantities x_k^G, w_k^G in (2.5).

With regard to the effective calculation of the quantities $\ell_\nu^{(n)}(x_k^G)$ in (2.5), one faces a tradeoff between efficiency and accuracy. A naive procedure is to calculate for each ν the product in (2.4) as written. Since there are n such products to be formed, this requires for each fixed x a number of multiplications of the order n^2. Given that $O(n)$ values of x are needed in (2.5), the total complexity of evaluating all Cotes numbers in (2.5) is $O(n^3)$, not counting the work of generating the Gaussian weights w_k^G and nodes x_k^G.

A more efficient procedure is to use the barycentric formula

$$\ell_\nu^{(n)}(x) = \frac{\lambda_\nu^{(n)}/(x - x_\nu)}{\sum_{\mu=1}^n \lambda_\mu^{(n)}/(x - x_\mu)} \qquad (x \neq x_\nu), \tag{2.6}$$

where $\lambda_\nu^{(n)}$ are the auxiliary quantities

$$\lambda_\nu^{(n)} = \prod_{\substack{\mu=1 \\ \mu\neq\nu}}^{n} \frac{1}{x_\nu - x_\mu}, \qquad \nu = 1, 2, \dots, n. \tag{2.7}$$

Each evaluation of $\ell_\nu^{(n)}(x)$ according to (2.6) is an $O(n)$ process, and since $O(n)$ such evaluations are to be made, we get an $O(n^2)$ procedure for evaluating all Cotes numbers in (2.5), provided that the quantities (2.7) can also be generated with $O(n^2)$ operations. This is indeed the case, if one uses an algorithm suggested in [5], namely

$$\begin{aligned}
&\lambda_1^{(1)} = 1;\\
&\text{for } \nu = 2, 3, \dots, n \text{ do}\\
&\left|\begin{aligned}
\lambda_\mu^{(\nu)} &= \frac{\lambda_\mu^{(\nu-1)}}{x_\mu - x_\nu}, \qquad \mu = 1, 2, \dots, \nu - 1;\\
\lambda_\nu^{(\nu)} &= -\sum_{\mu=1}^{\nu-1} \lambda_\mu^{(\nu)}.
\end{aligned}\right.
\end{aligned} \tag{2.8}$$

Here, the first set of equations in the ν-loop is a direct consequence of the definition (2.7), while the second equation follows from the identity

$$1 \equiv \sum_{\mu=1}^{\nu} \ell_\mu^{(\nu)}(x) = \sum_{\mu=1}^{\nu} \lambda_\mu^{(\nu)} \prod_{\substack{\kappa=1 \\ \kappa\neq\mu}}^{\nu} (x - x_\kappa)$$

by comparing the leading coefficients (of the power $x^{\nu-1}$) on the left and right.

While the procedure based on (2.6) and (2.8) is one order more efficient than the naive procedure described above, it is also more exposed to the detrimental influence of rounding errors. This is because it requires, in the last step of the ν-loop in (2.8), the formation of a sum, which can be subject to significant cancellation error if $\max_{1\leq\mu\leq\nu-1} |\lambda_\mu^{(\nu)}|$ is much larger than $|\lambda_\nu^{(\nu)}|$. Numerical experimentation with (2.8) in Section 3 will show that this indeed is likely to happen. No harmful arithmetic operations occur in the naive $O(n^3)$ procedure, since it uses only the benign operations of multiplication and division (disregarding the formation of differences such as $x_\mu - x_\nu$, which occur in both algorithms).

It may be worth pointing out that the cancellation effect referred to above does not depend on how the nodes are scaled or shifted, since the quantities $\lambda_\mu^{(\nu)}/\lambda_\nu^{(\nu)}$, which control cancellation, are invariant with respect to any linear transformation of the nodes. The effect, however, may depend on the order in which the nodes are arranged. In [5] it is recommended that they be arranged in the order of decreasing distance from their centerpoint.

2.2. Moment-Based Methods

Suppose one knows the first n modified moments of w relative to a system of polynomials $\pi_{\mu-1} \in \mathbb{P}_{\mu-1}$,

$$m_{\mu-1} = \int_a^b \pi_{\mu-1}(x)w(x)\,dx, \qquad \mu = 1,2,\ldots,n. \tag{2.9}$$

It is assumed, of course, that these moments exist. When $\pi_{\mu-1}(x) = x^{\mu-1}$, we are dealing with ordinary moments; important examples of modified moments are those relative to a set of orthogonal polynomials, $\pi_{\mu-1}(x) = \pi_{\mu-1}(x;v)$, where v is also a nonnegative weight function, possibly, but not necessarily, identical with w. Putting $f(x) = \pi_{\mu-1}(x)$, $\mu = 1,2,\ldots,n$, in (2.1) then yields the system of *moment equations*

$$\sum_{\nu=1}^{n} \pi_{\mu-1}(x_\nu)\,w_\nu = m_{\mu-1}, \qquad \mu = 1,2,\ldots,n, \tag{2.10}$$

a system of linear algebraic equations with the Vandermonde-like coefficient matrix

$$V_n = [\pi_{\mu-1}(x_\nu)]_{\mu,\nu=1}^{n}. \tag{2.11}$$

Also here, there are slow and fast methods for solving (2.10). Straightforward Gauss elimination is an $O(n^3)$ process, while there are known $O(n^2)$ algorithms (cf. [6,7] and references cited therein) that take advantage of the Vandermonde structure. All solution procedures, however, are subject to the ill-conditioning of the matrix V_n, which, even for modified moments, can be quite severe (cf. [8]). Again, in Section 3, we will illustrate the numerical properties of these algebraic procedures on typical examples.

3. EXAMPLES

In this section, we report on a number of examples calculated in double and quadruple precision on the Sparc 2 workstation (machine precision $\approx 1.11 \times 10^{-16}$ and 0.963×10^{-34}, respectively). In general, we will be interested in determining the maximum (over ν) relative errors in the Cotes numbers w_ν, as well as the stability constants σ_n in (2.2). The former are computed by comparing double-precision with quadruple-precision results, while the latter are computed throughout in quadruple precision. Of special interest are situations in which $\sigma_n = 1$, which indicates positivity (more precisely, nonnegativity) of the respective Newton-Cotes formulae. New examples of (conjectured) positive Newton-Cotes formulae will be given in Section 4.

In connection with moment-based methods, we will also be interested in seeing numerical values, or estimates, of the condition numbers of the respective matrices V_n (cf. (2.11)).

3.1. Classical Newton-Cotes Formulae on $[-1,1]$

These are for the weight function $w(x) \equiv 1$ on $[-1,1]$ and the equidistant nodes $x_\nu = -1 + 2(\nu-1)/(n-1)$, $\nu = 1,2,\ldots,n$. We first illustrate the $O(n^3)$ method of Section 2.1, and in Table 3.1 report the maximum relative errors of the Cotes numbers w_ν in the column headed "err w" and the stability constants σ_n in the column headed "stab." As can be seen, the accuracy is quite satisfactory, and the stability constants, as is well known, grow rapidly with n.

The results in Table 3.1 should be contrasted with those in Table 3.2 obtained by the $O(n^2)$ method of Section 2.1. Here, all accuracy is practically gone by the time n reaches 35. Instead of the stability constant we list in Table 3.2 under "err λ" the maximum relative error of the $\lambda_\nu^{(n)}$ in (2.7). As can be seen, the latter correlate well with the errors in the w_ν.

Rearranging the nodes x_ν so that they move from both ends toward the origin as ν increases (cf. the remark at the end of Section 2.1), for example, defining

Table 3.1. Accuracy and stability of classical Newton-Cotes formulae generated by the $O(n^3)$ method of Section 2.1.

n	err w	stab	n	err w	stab
5	1.9(−15)	1.00(0)	25	2.8(−14)	5.63(3)
10	5.2(−14)	1.00(0)	30	5.7(−13)	1.83(4)
15	9.3(−15)	2.03(1)	35	1.5(−14)	2.52(6)
20	6.1(−14)	6.33(1)	40	2.9(−13)	7.86(6)

Table 3.2. Accuracy of classical Newton-Cotes formulae generated by the $O(n^2)$ method of Section 2.1.

n	err w	err λ	n	err w	err λ
5	1.8(−15)	8.9(−15)	25	1.0(−6)	1.5(−6)
10	7.3(−12)	5.6(−13)	30	4.2(−3)	3.8(−4)
15	1.5(−12)	3.1(−12)	35	6.2(−2)	1.0(−1)
20	1.0(−7)	1.2(−8)	40	2.3(+1)	9.5(0)

Table 3.3. Accuracy of classical Newton-Cotes formulae generated by the $O(n^2)$ method of Section 2.1 with rearranged nodes.

n	err w	err λ	n	err w	err λ
5	1.8(−15)	5.6(−17)	25	5.2(−12)	2.9(−14)
10	2.9(−14)	7.8(−16)	30	3.2(−9)	3.5(−13)
15	2.5(−14)	2.2(−15)	35	4.5(−9)	2.0(−12)
20	2.1(−12)	1.4(−14)	40	2.1(−7)	1.5(−11)

Table 3.4. Classical Newton-Cotes formulae from ordinary moments; accuracy and condition numbers.

n	err w	cond	n	err w	cond
5	1.7(−15)	5.0(1)	25	3.8(−7)	2.9(11)
10	3.1(−13)	1.4(4)	30	2.7(−4)	8.5(13)
15	1.8(−11)	3.6(6)	35	4.1(−2)	2.4(16)
20	5.1(−9)	1.1(9)	40	2.3(+1)	6.9(18)

Table 3.5. Classical Newton-Cotes formulae from Legendre moments; accuracy and condition numbers.

n	err w	cond	n	err w	cond
5	1.4(−15)	2.7(1)	25	1.2(−7)	3.7(9)
10	7.5(−14)	1.5(3)	30	1.3(−6)	7.0(11)
15	1.5(−13)	1.4(5)	35	1.0(−5)	1.4(14)
20	3.0(−9)	2.1(7)	40	1.1(−2)	2.9(16)

$$x_\nu = \begin{cases} -1 + \frac{\nu-1}{n-1}, & \nu \text{ odd}, \\ 1 - \frac{\nu-2}{n-1}, & \nu \text{ even}, \end{cases} \tag{3.1}$$

improves the accuracy of the $O(n^2)$ method considerably. This is shown in Table 3.3.

We next illustrate the method based on moment equations, where for ordinary moments

$$m_{\mu-1} = \int_{-1}^{1} t^{\mu-1}\, dt = \begin{cases} 0, & \mu \text{ even}, \\ 2/\mu, & \mu \text{ odd}, \end{cases}$$

the matrix V_n in (2.11) is the Vandermonde matrix in the nodes x_ν. We will choose for the modified moments the Legendre moments

$$m_{\mu-1} = \int_{-1}^{1} \pi_{\mu-1}(t; w)\, dt = \begin{cases} 2, & \mu = 1, \\ 0, & \mu > 1, \end{cases}$$

where $\pi_{\mu-1}$ are the monic Legendre polynomials. In this case, V_n becomes a Vandermonde-like matrix in the terminology of [9]. In Tables 3.4 and 3.5 we list, along with the errors in the w_ν, the condition numbers of the respective matrices V_n. For ordinary moments, these are the ∞-condition numbers computed by [8, equations (3.3),(3.4)] (see also [8, Example 3.3]), while for Legendre moments, they are Frobenius-norm condition numbers computed according to [8, equations (5.16),(5.17)]. We used the LINPACK routines DGECO and DGESL, and our own quadruple-precision versions thereof, to solve the linear system (2.10), since they provide estimates for the condition number. Table 3.4 reports on the results for ordinary moments, Table 3.5 on those for Legendre moments. Modified moments are seen to give only marginal improvements (1–2 decimal orders) over ordinary moments, both in terms of accuracy and condition. It was observed that the LINPACK estimates of the condition numbers agree very well in order of magnitude with those computed analytically.

Rearranging of the nodes as in (3.1) does not significantly alter the results. Indeed, the ∞-condition number in the case of ordinary moments remains the same.

3.2. Newton-Cotes Formulae for Logarithmic/Algebraic Weight

We now consider the weight function $w(x) = x^{-1/2} \ln(1/x)$ on $[0, 1]$ and equally spaced nodes $x_\nu = (\nu - 1)/(n - 1)$, $\nu = 1, 2, \ldots, n$. The results based on slow and fast Lagrange interpolation are very similar to those in Tables 3.1 and 3.2 for classical Newton-Cotes formulae, except that the stability constant grows somewhat faster, from 1.26 when $n = 5$, to 5.10×10^8 when $n = 40$. The Cotes numbers seem to exhibit an interesting sign pattern: for each $n \geq 3$, the first two weights are positive, while from then on they alternate in sign. A similar improvement as in Table 3.3 is observed also in this example, if the nodes are rearranged analogously to (3.1).

Ordinary moments are easily computed rationally from $m_{\mu-1} = 1/(\mu-1/2)^2$, $\mu = 1, 2, \ldots$, while modified moments relative to the shifted Legendre polynomials are also expressible rationally in terms of μ (cf. [10]). The respective moment equations, as expected, become gradually ill-conditioned, more quickly so for ordinary moments. This is shown in Table 3.6. Here again, the LINPACK estimates of the condition numbers agree with those computed analytically within 1–2 decimal orders. Both overestimate the actual loss of relative accuracy by several orders of magnitude.

We also experimented with the x_ν being the Chebyshev nodes on $[0, 1]$, but observed only a slight improvement in the case of ordinary moments, and none for modified moments. The stability constants, on the other hand, as computed by the $O(n^3)$ method, are all very close to 1.

3.3. Newton-Cotes Formulae for the Hermite Weight on Finite or Half-Infinite Intervals

Our interest here is in the weight function $w(x) = e^{-x^2}$ on $[0, c]$, where $0 < c \leq \infty$. We first consider nodes that are equally spaced (with endpoints included) on $[0, c]$ when $c < \infty$,

Table 3.6. Newton-Cotes formulae for logarithmic/algebraic weight function from moment-based methods; accuracy and condition numbers.

	ordinary moments		modified moments	
n	err w	cond	err w	cond
5	6.7(−15)	1.4(3)	9.5(−16)	2.9(2)
10	6.4(−12)	4.2(7)	1.7(−14)	4.0(5)
15	1.6(−7)	1.4(12)	9.2(−12)	5.7(8)
20	3.1(−4)	4.4(16)	9.6(−10)	9.0(11)
30	1.0(0)	4.7(25)	2.1(−4)	3.7(18)
40	1.4(+1)	5.0(34)	1.5(0)	2.4(25)

and on $[0,5]$ when $c=\infty$. The choice of "5" is arbitrary but motivated by that fact that $w(x) < 1.5 \times 10^{-11}$ if $x \geq 5$.

The ordinary moments of w are expressible in terms of the incomplete gamma function

$$\gamma(a,x) = \int_0^x e^{-t} t^{a-1}\, dt$$

as

$$m_{\mu-1} = \int_0^c t^{\mu-1} e^{-t^2}\, dt = \frac{1}{2}\gamma\left(\frac{1}{2}\mu, c^2\right), \qquad 0 < c \leq \infty, \tag{3.2}$$

where for $c=\infty$ we have $\gamma(\mu/2,\infty) = \Gamma(\mu/2)$. They can be computed accurately by our routine in [11]. For the modified moments we take

$$m_{\mu-1} = \int_0^c \pi_{\mu-1}(t;w)\, e^{-t^2} dt, \qquad 0 < c \leq \infty, \tag{3.3}$$

where $\pi_{\mu-1}(\,\cdot\,;w)$ is the (monic) orthogonal polynomial of degree $\mu-1$ relative to the weight function $w(t) = e^{-t^2}$ on $[0,c]$. Thus,

$$m_{\mu-1} = \begin{cases} \dfrac{1}{2}\gamma\left(\dfrac{1}{2}, c^2\right) & \text{if } \mu = 1, \\ 0 & \text{otherwise,} \end{cases} \qquad 0 < c \leq \infty. \tag{3.4}$$

To make use of these polynomials in forming the matrix V_n of (2.11), and in generating the Gauss formulae required in (2.5), one needs the (nonstandard) recurrence relation for these polynomials. Its coefficients can be computed by the discretization procedure of Section 4.3 in [4].

Our computations for $c = 1, 2, 5, \infty$ produced results which, on the whole, are rather similar in quality to those for classical Newton-Cotes formulae (cf. Section 3.1). An unexpected breakdown, nevertheless, was observed in the case $c=\infty$ for $n = 40$, with nodes on $[0,5]$ rearranged in order of decreasing distance from the midpoint: for one of the Gauss nodes x_k^G, the sum in the denominator of (2.6) became exactly zero to machine (double) precision. Quadruple-precision computation revealed serious cancellation, the absolute value of the sum being about 17 decimal orders smaller than the largest term in modulus! This is just another instance suggesting caution in the use of fast Lagrange interpolation.

It seems interesting to experiment with Chebyshev nodes of the first and second kind, both on $[0,c]$, $0 < c < \infty$, to see whether or not the respective Newton-Cotes formulae are positive. One would expect, indeed, that both are positive when c is small, since in the limit as $c \to 0$ they approach the Fejér formulae, which are known to be positive (cf. [12,13]). By means of our $O(n^3)$ method (in quadruple precision) we found that for Chebyshev nodes of the first kind, we have positivity when $c=1$, at least for $1 \leq n \leq 40$. When $c=2$, we found positivity for all $n \leq 40$, except $n=3$, in which case the stability constant σ_3 (cf. (2.2)) turned out to be larger than 1. It can be verified analytically, in this case, that the Cotes number for the largest of the three nodes, x_3, is indeed

$$w_3 = \frac{1}{2\sqrt{3}} \left\{ 1 - \frac{4}{\sqrt{3}} - e^{-4} + (\sqrt{3}-1)\sqrt{\pi}\ \mathrm{erf}\ 2 \right\} = -0.010467\ldots.$$

It seems reasonable to conjecture the existence of a $\bar{c}$ with $1 < \bar{c} < 2$ such that for all $0 < c < \bar{c}$ the Newton-Cotes formulae for $w(x) = e^{-x^2}$ on $[0, c]$, based on Chebyshev points of the first kind, are positive. For Chebyshev points of the second kind, we conjecture the same for some $\bar{\bar{c}}$ with $2 < \bar{\bar{c}} < 3$, having observed positivity for $c = 2$ and $1 \le n \le 40$, but not for $c = 3$. What was also found to be interesting is that for $c > \bar{c}$ (respectively, $c > \bar{\bar{c}}$), the stability constants σ_n essentially decrease as n increases and in fact seem to approach 1 as $n \to \infty$. This surely is in marked contrast to classical Newton-Cotes formulae!

4. POSITIVITY CONJECTURES

We use the accurate $O(n^3)$ procedure of Section 2.1 for generating Newton-Cotes formulae to numerically test their positivity (for all n). We examine two weight functions w—the Jacobi weight on $[-1, 1]$ and the logistics weight on $(-\infty, \infty)$. For the former we test a wide-ranging conjecture of Milovanović [14] and supply ample evidence for its support. For the latter we advance a conjecture of our own. To reduce the amount of computation time, all computations were done in double (rather than quadruple) precision, but the accuracy was monitored by doing the calculations also in single precision and observing the degree of deterioration of single-precision accuracy.

4.1. Jacobi Weight Functions

The positivity of Newton-Cotes formulae for the Jacobi weight function $w^{(\alpha,\beta)}(x) = (1-x)^\alpha (1+x)^\beta$ on $[-1, 1]$, $\alpha > -1$, $\beta > -1$, where the nodes are Jacobi abscissas belonging to parameters other than α, β, has been investigated by a number of researchers. The earliest examples are the positive quadrature rules of Fejér [12] for $w = w^{(0,0)}$ having as abscissas the Chebyshev points of the first and second kind. Subsequent work for $w = w^{(0,0)}$ dealt with ultraspherical and more general Jacobi abscissas, either for all n [15–17], or for selected fixed n [18,19]. Ultraspherical abscissas were considered in combination with the Chebyshev weight function of the first kind in [20], and for ultraspherical weight functions in [21].

Askey, for $w = w^{(0,0)}$, in addition to proving positivity of all Cotes numbers for a variety of Jacobi abscissas, in [17] held out the possibility that positivity may hold for all Jacobi abscissas with parameters α, β satisfying $-1 < \alpha \le \beta \le 3/2$. Our computations seem to confirm this conjecture except for the upper left-hand corner of the region where we noted nonpositivity for even values of n. We thus revise the conjecture, claiming positivity only in the region $\{\alpha \le \beta < \alpha + 2,\ -1 < \alpha < -1/2\} \cup \{-1/2 \le \alpha \le \beta \le 3/2\}$, and of course, by symmetry, in the companion region reflected along $\alpha = \beta$.

In all the work above, the quadrature nodes in each rule are zeros of one and the same (Jacobi) polynomial. It appears that more satisfactory results are attainable if one takes as abscissas the zeros of *two* (related) polynomials. Fejér's $(2n-1)$-point formula for $w = w^{(0,0)}$ with the zeros of U_{2n-1} as abscissas gives us a clue on how this might be done. Note that $U_{2n-1} = 2T_n U_{n-1}$. Thus, the nodes in this case are the zeros of $P_n^{(-1/2,-1/2)}$ and $P_{n-1}^{(1/2,1/2)}$. This interpretation suggests the following sweeping generalization.

CONJECTURE 4.1. *(See [14]). For any $\alpha > -1$, $\beta > -1$, let*

$$X_n^0 = \{x \in (-1,1) : P_n^{(\alpha,\beta)}(x) = 0\}, \qquad X_n^1 = \{x \in (-1,1) : P_{n-1}^{(\alpha+1,\beta+1)}(x) = 0\}. \tag{4.1}$$

Then the $(2n-1)$-point Newton-Cotes formula

$$\begin{aligned} \int_{-1}^{1} (1-x)^{\alpha+1/2}(1-x)^{\beta+1/2} f(x)\,dx &= \sum_{x_k \in X_n^0 \cup X_n^1} w_k^{(n)} f(x_k) + R_{2n-1}(f), \\ R_{2n-1}(f) &= 0, \qquad \text{all } f \in \mathbb{P}_{2n-2}, \end{aligned} \tag{4.2}$$

has nonnegative coefficients $w_k^{(n)}$ for all $n = 1, 2, \ldots$.

The Fejér formula mentioned above is the special case $\alpha = \beta = -1/2$ of this conjecture.

The conjecture has been checked numerically by computing (in double precision) the stability constant σ_n (cf. (2.2)) and testing whether or not it equals exactly 1. We found that it always does, for $\alpha = -0.75(0.25)4.00$, $\beta = \alpha(0.25)4.00$, and $n = 5(5)40$. (It suffices, by symmetry, to consider $\beta \geq \alpha$.) We also examined in more detail $\alpha = -0.9(0.1)1.0$, $\beta = \alpha(0.1)1.0$, $n = 1(1)40$, and confirmed the conjecture in these cases as well. We used the procedures `recur` and `gauss` of [4] to generate the necessary Jacobi polynomials and their zeros.

The evidence in support of Conjecture 4.1 is thus fairly convincing.

4.2. Logistics Weight Function

Here we consider

$$w(x) = \frac{e^{-x}}{(1+e^{-x})^2} = \frac{1}{4\cosh^2(x/2)}, \qquad x \in \mathbb{R}. \tag{4.3}$$

The corresponding orthogonal polynomials can be generated rather easily from their known recurrence relation (cf. [4, footnote 3, p. 45]). Since $w(x) \sim e^{-|x|}$ as $x \to \pm\infty$, it seems natural to take as quadrature abscissas the zeros of Laguerre polynomials and their negative counterparts, including 0, if n is odd. Thus, we let

$$X_n = \begin{cases} \{\pm x :\ L_{n/2}(x) = 0\}, & n \text{ even}, \\ \{x = 0\} \cup \{\pm x :\ L_{\lfloor n/2 \rfloor}(x) = 0\}, & n \text{ odd}. \end{cases} \tag{4.4}$$

Then we propose the following conjecture.

CONJECTURE 4.2. *For any even $n = 2, 4, \ldots$, the n-point Newton-Cotes formula*

$$\begin{aligned} \int_{-\infty}^{\infty} f(x) \frac{e^{-x}}{(1+e^{-x})^2}\,dx &= \sum_{x_k \in X_n} w_k^{(n)} f(x_k) + R_n(f), \\ R_n(f) &= 0, \qquad \text{all } f \in \mathbb{P}_{n-1}, \end{aligned} \tag{4.5}$$

has all coefficients nonnegative, $w_k^{(n)} \geq 0$.

When n is odd, the conjecture is false.

We have verified in quadruple precision that $\sigma_n = 1$, n even, for all $n = 2(2)80$, lending strong support to the validity of the conjecture. Quadruple precision was necessary to counteract significant deterioration of accuracy for large n. In this way, it was possible to still verify $\sigma_n = 1$ to at least 20 decimal digits.

It may be worth observing that the method of moment systems in this example quickly deteriorates because of severe ill-conditioning of the matrices involved, both in the case of ordinary moments and, slightly less so, in the case of modified moments (relative to the orthogonal polynomials for w). The former can be computed from

$$m_k = \begin{cases} 2\int_0^\infty t^k \dfrac{e^{-t}}{(1+e^{-t})^2}\,dt, & k \text{ even}, \\ 0, & k \text{ odd}, \end{cases} \tag{4.6}$$

by making the change of variables $e^{-t} = x$, using equation 4.272.14 in [22], simplifying the result, and expressing it in terms of the Riemann zeta function by using [23, equation 23.2.16]. The result, for k even, is

$$m_k = 1 \quad (k=0), \qquad m_k = \left(1 - 2^{1-k}\right)\,(2\pi)^k\,|B_k| \quad (k > 0,\ \text{even}), \tag{4.7}$$

where B_k is the Bernoulli number. The modified moments are 1 for $k = 0$, and 0 otherwise.

5. GAUSS FORMULAE

As we have mentioned earlier, moments and modified moments enter also in the construction of Gauss-type quadrature formulae

$$\begin{aligned} \int_a^b f(x)\,w(x)\,dx &= \sum_{\nu=1}^{n} w_\nu\, f(x_\nu) + R_n(f), \\ R_n(f) &= 0, \qquad \text{all } f \in \mathbb{P}_{2n-1}, \end{aligned} \tag{5.1}$$

although there are other techniques—and they are sometimes more effective—that do not rely on moment information (cf. [4]).

It is generally assumed that the modified moments of w,

$$m_{\mu-1} = \int_a^b p_{\mu-1}(x)\,w(x)\,dx, \qquad \mu = 1, 2, \ldots, \tag{5.2}$$

are formed with polynomials $\{p_k\}$ satisfying a three-term recurrence relation

$$\begin{aligned} p_0(x) = 1, &\qquad p_{-1}(x) = 0, \\ p_{k+1}(x) = (x - a_k)\,p_k(x) - b_k\,p_{k-1}(x), &\qquad k = 1, 2, \ldots, \end{aligned} \tag{5.3}$$

with known coefficients $a_k \in \mathbb{R}$, $b_k \geq 0$. If $a_k = b_k = 0$, all k, this will produce ordinary moments, while modified moments use a set of orthogonal polynomials $p_k(\cdot) = \pi_k(\,\cdot\,; v)$ relative to some positive weight function v. The coefficients $a_k = a_k(v)$, $b_k = b_k(v) > 0$ then depend on v.

There is a well-known algorithm—the *modified Chebyshev algorithm* (cf. [4, Section 3])—that takes as input the first $2n$ moments (5.2) (ordinary or modified) and the first $2n-1$ coefficients a_k, b_k, $k = 0, 1, \ldots, 2n-2$, and produces the first n recursion coefficients $\alpha_k = a_k(w)$, $\beta_k = b_k(w)$ for the orthogonal polynomials $p_k(\cdot) = \pi_k(\,\cdot\,; w)$ relative to the weight function w. These in turn, by well-known eigenvalue techniques, allow us to compute the Gauss nodes x_ν and weights w_ν in (5.1).

The major problem with moment-related approaches is the possible ill-conditioning of the underlying nonlinear moment map

$$G_n : \mathbb{R}^{2n} \to \mathbb{R}^{2n} \qquad m \longmapsto \gamma, \tag{5.4}$$

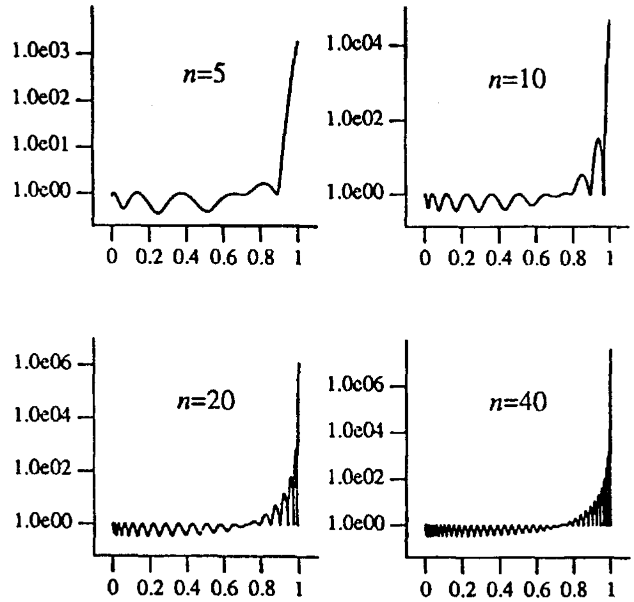

Figure 1. The polynomial g_n for $w(x) = x^{-1/2}\ln(1/x)$ on $[0, 1]$.

where $m^\mathsf{T} = [m_0, m_1, \ldots, m_{2n-1}]$, $\gamma^\mathsf{T} = [x_1, \ldots, x_n;\ w_1, \ldots, w_n]$. In the case of ordinary moments, if w is supported on the positive real line and normalized to make $m_0 = 1$, it is known, e.g., that [24, Section 5.2]

$$\operatorname{cond}_\infty G_n \geq \frac{1}{2} \max_{1\leq\nu\leq n} \left[\frac{\pi_n(-1;w)}{\pi_n'(x_\nu;w)}\right]^2 \tag{5.5}$$

for some suitably (in terms of the ∞-norm) defined condition number. The quotient on the right of (5.5) typically tends to ∞ exponentially fast as $n \to \infty$, since the point -1 is well outside the support interval of w. For this reason, ordinary moments are unsuitable for generating Gauss formulae, unless n is quite small.

A better chance (but no guarantee!) of succeeding can be had by using modified moments, provided the auxiliary weight function v in the respective polynomials $p_k(\cdot) = \pi_k(\,\cdot\,; v)$ is chosen to reflect the peculiarities inherent in the given weight function w. For the condition of the underlying moment map (5.4) one then has [25, Theorem 3.1] (in terms of the Frobenius norm)

$$\operatorname{cond}_F G_n = \left\{\int_a^b g_n(x;w)v(x)\,dx\right\}^{1/2}, \tag{5.6}$$

where g_n is a polynomial of degree $4n-2$, which is positive on $\mathbb{R}$ and can be defined in terms of the elementary Hermite interpolation polynomials associated with the Gauss nodes x_ν and in terms of the Gauss weights w_ν. Therefore, g_n depends only on the given weight function w, whereas the influence of the chosen weight function v for the modified moments comes into play in the second factor of the integrand in (5.6). The extent of ill-conditioning is thus crucially determined by the magnitude of g_n on the support of v.

In Figure 1 we show graphs of g_n, $n = 5, 10, 20, 40$, for the logarithmic/algebraic weight function w of Section 3.2. It can be seen in this case that g_n is less than 1 over much of the interval $[0, 1]$. If, as proposed in Section 3.2, one chooses Legendre moments, that is, $v \equiv 1$, then the integral in (5.6) remains rather small, even for large values of n. Indeed, the condition numbers for the n-values of Figure 1 turn out to be 5.73, 14.4, 38.6 and 107., respectively. The modified Chebyshev algorithm, accordingly, works very well in this case.

The situation is rather different for the Hermite weight w on $[0, c]$ (cf. Section 3.3), unless c is relatively small. For $c = 5$, for example, the polynomials g_n behave as shown in Figure 2. Using

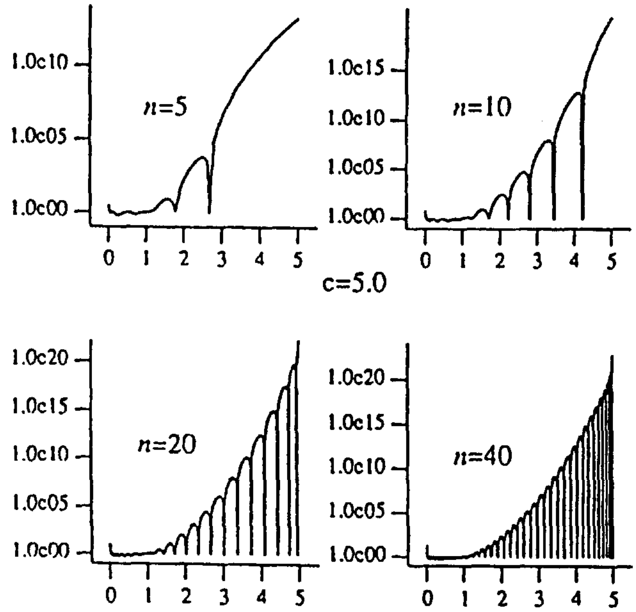

Figure 2. The polynomial g_n for $w(x) = e^{-x^2}$ on $[0, 5]$.

Table 4.1. The condition of G_n for Hermite weights on $[0, c]$.

n	$c = 1$	$c = 2$	$c = 5$	$c = \infty$
5	7.9(−1)	1.7(0)	8.3(1)	8.3(1)
10	7.8(−1)	1.6(0)	2.0(4)	8.7(4)
20	7.7(−1)	1.4(0)	4.7(4)	1.3(11)
40	7.6(−1)	1.3(0)	3.8(4)	3.5(23)

Table 4.2. Accuracy of modified Chebyshev algorithm for $w(x) = e^{-x^2}$ on $[0, \infty]$.

n	err α	err β
5	2.8(−13)	2.7(−13)
10	1.0(−10)	1.2(−10)
15	1.2(−8)	1.5(−8)
20	7.8(−7)	1.0(−6)
25	3.4(−5)	4.7(−5)
30	1.1(−3)	1.6(−3)
35	2.4(−2)	3.8(−2)
40	6.8(−2)	1.4(−1)

again Legendre moments on $[0, c]$, one now obtains condition numbers 3.52×10^{12}, 1.14×10^{19}, 1.36×10^{20} and 8.57×10^{19} for the four values of n, which are unacceptably large.

One might argue that the choice of Legendre moments is a poor choice in this case, since $v(x) \equiv 1$ does not mimic the behavior of $w(x) = e^{-x^2}$ on $0 \le x \le 5$. That, of course, is a valid point. Surprisingly, however, the difficulty persists for large c, even if we make better choices. Indeed, the best choice of all, $v = w$, gives rise to condition numbers shown in Table 4.1. While for $0 < c \le 5$, these optimal condition numbers are still acceptable, they are no longer so if $c = \infty$ or c much larger than 5. To illustrate this, we have run the modified Chebyshev algorithm for $c = \infty$ with the true modified moments ($v = w$) randomly perturbed at the level of machine (double) precision and obtained for the computed recursion coefficients α_k, β_k the relative errors shown in Table 4.2. We can see that the modified Chebyshev algorithm deteriorates rather rapidly and looses all (double-precision) accuracy by the time n reaches 40.

The lesson to be learned from this example is that the approach via modified moments (even the best ones!) can be inherently limited. It is therefore no surprise that the computation of the orthogonal polynomials for these laterally supported Hermite weights must use different techniques to succeed, for example, appropriate discretization [26, Section 6] or "domain decomposition" [27].

REFERENCES

1. T.N.L. Patterson, An algorithm for generating interpolatory quadrature rules of the highest degree of precision with preassigned nodes for general weight functions, *ACM Trans. Math. Software* **15**, 123–136; 137–143 (1989).
2. J. Kautsky and S. Elhay, Calculation of the weights of interpolatory quadratures, *Numer. Math.* **40**, 407–422 (1982).
3. S. Elhay and J. Kautsky, Algorithm 655—IQPACK: FORTRAN subroutines for the weights of interpolatory quadratures, *ACM Trans. Math. Software* **13**, 399–415 (1987).
4. W. Gautschi, Algorithm 726—ORTHPOL: A package of routines for generating orthogonal polynomials and Gauss-type quadrature rules, *ACM Trans. Math. Software* **20**, 21–62 (1994).
5. W. Werner, Polynomial interpolation: Lagrange versus Newton, *Math. Comp.* **43**, 205–217 (1984).
6. D. Calvetti and L. Reichel, Fast inversion of Vandermonde-like matrices involving orthogonal polynomials, *BIT* **33**, 473–484 (1993).
7. T. Kailath and V. Olshevsky, Displacement structure approach to polynomial Vandermonde and related matrices, *Linear Algebra Appl.* (toappear).
8. W. Gautschi, How (un)stable are Vandermonde systems?, In *Asymptotic and Computational Analysis*, (Edited by R. Wong), Dekker, New York, pp. 193–210, (1990).
9. W. Gautschi, The condition of Vandermonde-like matrices involving orthogonal polynomials, *Linear Algebra Appl.* **52/53**, 293–300 (1983).
10. W. Gautschi, On the preceding paper 'A Legendre polynomial integral' by James L. Blue, *Math. Comp.* **33**, 742–743 (1979).
11. W. Gautschi, Algorithm 542—Incomplete gamma functions, *ACM Trans. Math. Software* **5**, 482–489 (1979).
12. L. Fejér, Mechanische Quadraturen mit positiven Cotesschen Zahlen, *Math. Z.* **37**, 287–309 (1933).
13. W. Gautschi, Numerical quadrature in the presence of a singularity, *SIAM J. Numer. Anal.* **4**, 357–362 (1967).
14. G.V. Milovanović, personal communication, December 1993.
15. R. Askey and J. Fitch, Positivity of the Cotes numbers for some ultraspherical abscissas, *SIAM J. Numer. Anal.* **5**, 199–201 (1968).
16. R. Askey, Positivity of the Cotes numbers for some Jacobi abscissas, *Numer. Math.* **19**, 46–58 (1972).
17. R. Askey, Positivity of the Cotes numbers for some Jacobi abscissas II, *J. Inst. Math. Appl.* **24**, 95–98 (1979).
18. G. Sottas, On the positivity of quadrature formulas with Jacobi abscissas, *Computing* **29**, 83–88 (1982).
19. G. Sottas, Positivity domain of ultraspherical type quadrature formulas with Jacobi abscissas: Numerical investigations, In *Numerical Integration III*, (Edited by H. Brass and G. Hämmerlin), Internat. Ser. Numer. Math., Vol. 85, Birkhäuser, Basel, pp. 285–294, 1988.
20. C.A. Micchelli, Some positive Cotes numbers for the Chebyshev weight function, *Aequationes Math.* **21**, 105–109 (1980).
21. M. Kütz, On the positivity of certain Cotes numbers, *Aequationes Math.* **24**, 110–118 (1982).
22. I.S. Gradshteyn and I.M. Ryzhik, *Table of Integrals, Series, and Products*, Academic Press, Orlando, (1980).
23. M. Abramowitz and I.A. Stegun, Editors, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, Nat. Bur. Standards Appl. Math. Ser., Vol. 55, U.S. Government Printing Office, Washington, DC, 1964.

24. W. Gautschi, Questions of numerical condition related to polynomials, In *Studies in Numerical Analysis*, (Edited by G.H. Golub), Studies in Mathematics, Vol. 24, The Mathematical Association of America, 1984.
25. W. Gautschi, On generating orthogonal polynomials, *SIAM J. Sci. Statist. Comput.* **3**, 289–317 (1982).
26. W. Gautschi, Computational problems and applications of orthogonal polynomials, In *Orthogonal Polynomials and Their Applications*, (Edited by C. Brezinski, L. Gori and A. Ronveaux), IMACS Annals Comput. Appl. Math., Vol. 9, Baltzer, Basel, pp. 61–71, 1991.
27. R.C.Y. Chin, A domain decomposition method for generating orthogonal polynomials for a Gaussian weight on a finite interval, *J. Comput. Phys.* **99**, 321–336 (1992).