MATHEMATICS OF COMPUTATION, VOLUME 28, NUMBER 125, JANUARY, 1974

# A Note on the Optimal Addition of Abscissas to Quadrature Formulas of Gauss and Lobatto Type

By Robert Piessens and Maria Branders

**Abstract.** An improved method for the optimal addition of abscissas to quadrature formulas of Gauss and Lobatto type is given.

**1. Introduction.** We consider the quadrature formula

$$\int_{-1}^{+1} f(x)\,dx \simeq \sum_{k=1}^{N} \alpha_k f(x_k) + \sum_{k=1}^{N+1} \beta_k f(\xi_k), \tag{1}$$

where the $x_k$'s are the abscissas of the $N$-point Gaussian quadrature formula. We want to determine the additional abscissas $\xi_k$ and the weights $\alpha_k$ and $\beta_k$ so that the degree of exactness of (1) is maximal. This problem has already been discussed by Kronrod [1] and Patterson [2] and it is well known that the abscissas $\xi_k$ must be the zeros of the polynomial $\phi_{N+1}(x)$ which satisfies

$$\int_{-1}^{+1} P_N(x)\phi_{N+1}(x)x^k\,dx = 0, \qquad k = 0, 1, \cdots, N, \tag{2}$$

where $P_N(x)$ is the Legendre polynomial of degree $N$. Thus, $\phi_{N+1}(x)$ must be an orthogonal polynomial with respect to the weight function $P_N(x)$. Then, the weights $\alpha_k$ and $\beta_k$ can be determined so that the degree of exactness of (1) is $3N + 1$ if $N$ is even and $3N + 2$ if $N$ is odd.

Szegö [3] proved that the zeros of $\phi_{N+1}(x)$ and $P_N(x)$ are distinct and alternate on the interval $[-1, +1]$. Kronrod [1] gave a simple method for the computation of the coefficients of $\phi_{N+1}(x)$. This method requires the solution of a triangular system of linear equations, which is, unfortunately, very ill-conditioned. Patterson [2] expanded $\phi_{N+1}(x)$ in terms of Legendre polynomials. The coefficients of this expansion satisfy a linear system of equations which is well-conditioned, although its construction requires a certain amount of computing time.

The present note proposes the expansion of $\phi_{N+1}(x)$ in a series of Chebyshev polynomials. We also give explicit formulas for the weights $\alpha_k$ and $\beta_k$. Finally, we consider the optimal addition of abscissas to Lobatto rules. As compared with Patterson's method, our method has three advantages:

(i) It leads to a considerable saving in computing time since the formulas are much simpler.

(ii) The loss of significant figures through cancellation and round-off is slightly reduced, as we verified experimentally. This is in agreement with some theoretical results given by Gautschi [4].

(iii) It is applicable for every value of $N$, while Patterson's method fails in the

Received May 15, 1972, revised March 2, 1973.
*AMS (MOS) subject classifications* (1970). Primary 65D30; Secondary 33A65.

Lobatto case for $N = 7, 9, 17, 22, 27, 35, 36, 37, 40, \cdots$, since some of the denominators in his recurrence formulae become zero.

**2. Optimal Addition of Abscissas to Gaussian Quadrature Formulas.** It is evident that $\phi_{N+1}(x)$ is an odd or even function depending on whether $N$ is even or odd. Thus, $\phi_{N+1}(x)$ can be expressed as

$$\phi_{N+1}(x) = \sum_{k=0}^{m}{}' b_k T_{2k}(x), \quad \text{if } N \text{ is odd,} \tag{3}$$

and

$$\phi_{N+1}(x) = \sum_{k=0}^{m} b_k T_{2k+1}(x), \quad \text{if } N \text{ is even,} \tag{4}$$

where $m = [(N + 1)/2]$.

It is clear that the polynomial $\phi_{N+1}(x)$ is only defined to within an arbitrary multiplicative constant. For the sake of convenience, we assume $b_m = 1$.

From (2), we derive the condition

$$\int_{-1}^{+1} P_N(x)\phi_{N+1}(x)T_k(x)\,dx = 0, \qquad k = 0, 1, \cdots, N. \tag{5}$$

In order to calculate the coefficients $b_k$, $k = 0, 1, \cdots, m - 1$, (3) or (4) is substituted in (5). This leads to the system of equations

$$\begin{aligned} b_{m-1} &= \tau_1 - 1, \\ b_{m-k} &= \sum_{j=1}^{k-1} b_{m-k+j}\tau_j + \tau_k, \qquad k = 2, 3, \cdots, m, \end{aligned} \tag{6}$$

where

$$\tau_k = -\int_{-1}^{+1} P_N(x)T_{N+2k}(x)\,dx \Big/ \int_{-1}^{+1} P_N(x)T_N(x)\,dx. \tag{7}$$

In order to derive a recurrence formula for $\tau_k$, we consider the integral

$$J = \int_{-1}^{+1} [xP_N(x) - P_{N+1}(x)]T_l(x)\,dx. \tag{8}$$

Using a well-known property of the Chebyshev polynomials, we obtain

$$J = \frac{1}{2}\int_{-1}^{+1} [xP_N - P_{N+1}]\,d\left(\frac{T_{l+1}}{l+1} - \frac{T_{l-1}}{l-1}\right), \tag{9}$$

and, by integrating by parts, this integral can be expressed as

$$J = \frac{N}{2(l+1)} I_{N,l+1} - \frac{N}{2(l-1)} I_{N,l-1}, \tag{10}$$

where

$$I_{N,l} = \int_{-1}^{+1} P_N(x)T_l(x)\,dx. \tag{11}$$

On the other hand, using a property of the Legendre polynomials, (8) can be transformed into

$$J = \frac{1}{N+1} \int_{-1}^{+1} (1 - x^2) T_l(x)\, d(P_N(x)),$$

which can be expressed as

$$J = \frac{2+l}{2(N+1)} I_{N,l+1} + \frac{2-l}{2(N+1)} I_{N,l-1}. \tag{12}$$

Since $\tau_k = I_{N,N+2k}/I_{N,N}$, the recurrence formula

$$\tau_{k+1} = \frac{[(N+2k-1)(N+2k) - (N+1)N](N+2k+2)}{[(N+2k+3)(N+2k+2) - (N+1)N](N+2k)} \tau_k, \tag{13}$$

where $\tau_1 = (N+2)/(2N+3)$ can be easily derived from (10) and (12).

System (6) is easier to construct than the corresponding system of Patterson [2], inasmuch as his method requires a set of recursions of variable lengths, while in our method only one recursion is needed. Moreover, further economy is achieved in solving the equation $\phi_{N+1}(x) = 0$, since, using a modification of Clenshaw's algorithm of summation, an odd or even Chebyshev series can be evaluated more efficiently than an odd or even Legendre series [5, p. 10]. Indeed, the computing time can be halved.

Explicit formulas for the weights are

$$\alpha_k = \frac{C_N}{P_N'(x_k)\phi_{N+1}(x_k)} + \frac{2}{NP_{N-1}(x_k)P_N'(x_k)}, \qquad k = 1, 2, \cdots, N, \tag{14}$$

$$\beta_k = \frac{C_N}{\phi_{N+1}'(\xi_k)P_N(\xi_k)}, \qquad k = 1, 2, \cdots, N+1, \tag{15}$$

where $C_N = 2^{2N+1}(N!)^2/(2N+1)!$.

**3. Optimal Addition of Abscissas to Lobatto Quadrature Formulas.** We now consider the quadrature formula

$$\int_{-1}^{+1} f(x)\, dx \simeq \sum_{k=0}^{N+1} \alpha_k f(x_k) + \sum_{k=1}^{N+1} \beta_k f(\xi_k), \tag{16}$$

where the $x_k$'s are abscissas of the Lobatto quadrature formula. Consequently, $x_0 = -1$, $x_{N+1} = +1$ and $x_1, x_2, \cdots, x_N$ are the zeros of the Jacobi polynomial $P_N^{(1,1)}(x)$. It is our purpose to determine the free abscissas $\xi_k$ and the weights $\alpha_k$ and $\beta_k$ so that the degree of exactness of (16) is maximal. Then, $\xi_k$ must be a zero of the polynomial $\phi_{N+1}(x)$ which satisfies

$$\int_{-1}^{+1} (1 - x^2) P_N^{(1,1)}(x) \phi_{N+1}(x) T_k(x)\, dx = 0, \qquad k = 0, 1, 2, \cdots, N. \tag{17}$$

Again, we express $\phi_{N+1}(x)$ in terms of Chebyshev polynomials as in (3) or (4), according to the parity of $N$. The coefficients $b_k$ can be found by solving the system (6) where

$$\tau_k = -\int_{-1}^{+1} (1 - x^2) P_N^{(1,1)} T_{N+2k}\, dx \Big/ \int_{-1}^{+1} (1 - x^2) P_N^{(1,1)} T_N\, dx. \tag{18}$$

Using the relation

$$\int_{-1}^{+1} (1 - x^2)P_N^{(1,1)} T_l \, dx = \frac{1}{N+2} [(l+2)I_{N+1,l+1} - (l-2)I_{N+1,l-1}],$$

where $I_{N,l}$ is defined by (11), the recurrence formula

$$(19) \quad \tau_{k+1} = \frac{[(N+2k-1)(N+2k-2) - (N+1)(N+2)](N+2k+2)}{[(N+2k+3)(N+2k+4) - (N+1)(N+2)](N+2k)} \tau_k$$

can be derived from (13).

The starting value for (19) is

$$\tau_1 = 3(N+2)/(2N+5).$$

The expressions for the weights are

$$(20) \quad \alpha_k = \frac{C_N}{2P_N'(x_k)\phi_{N+1}(x_k)} + \frac{2}{(N+1)(N+2)[P_{N+1}(x_k)]^2}, \quad \text{for } k = 1, 2, \cdots, N,$$

$$(21) \quad \alpha_0 = \alpha_{N+1} = \frac{2}{(N+2)(N+1)} - \frac{C_N}{2(N+1)\phi_{N+1}(1)},$$

$$(22) \quad \beta_k = \frac{N+2}{2(N+1)} \frac{C_N}{[P_N(\xi_k) - \xi_k P_{N+1}(\xi_k)]\phi_{N+1}'(\xi_k)}, \qquad k = 1, 2, \cdots, N+1,$$

where $C_N = 2^{2N+3}[(N+1)!]^2/(2N+3)!$.

**Appendix. Computer program.** In this appendix, we describe a FORTRAN program for the construction of the quadrature formula (1). A listing of this program is reproduced in the supplement at the end of this issue. A program for the construction of the quadrature formula (11) may be obtained from the authors.

The program consists of three subroutines: the main subroutine KRONRO and two auxiliary subroutines ABWE1 and ABWE2, which are called by KRONRO.

In KRONRO the coefficients of the polynomial $\phi_{N+1}(x)$ are calculated.

In ABWE1 the abscissas $x_k$ and weights $\alpha_k$ are calculated.

In ABWE2 the abscissas $\xi_k$ and weights $\beta_k$ are calculated.

The abscissas are calculated using Newton-Raphson's method. Starting values for this iterative process are provided by [6]

$$x_k \simeq \left(1 - \frac{1}{8N^2} + \frac{1}{8N^3}\right) \cos\left(\frac{2k - 1/2}{2N+1} \pi\right)$$

and

$$\xi_k \simeq \left(1 - \frac{1}{8N^2} + \frac{1}{8N^3}\right) \cos\left(\frac{2k - 3/2}{2N+1} \pi\right).$$

The program has been tested on the computer IBM 370/155 of the Computing Centre of the University of Leuven, for $N = 2(1)50(10)200$. The computations were carried out in double precision (approximately 16 significant figures). For $N = 200$, the maximal absolute error of the abscissas is $8.6 \times 10^{-16}$ and of the weights $3.3 \times 10^{-15}$.

For $N = 50$, the computing time is 1.7 sec., for $N = 100$, 6.4 sec. and for $N = 200$, 24.7 sec.

**Acknowledgement.** This research was supported in part by the Fonds voor Kollektief Fundamenteel Onderzoek under Grant No. 10.174.

Applied Mathematics Division
Katholieke Universiteit Leuven
Heverlee, B-3030, Belgium

1. A. S. KRONROD, *Nodes and Weights for Quadrature Formulas,* "Nauka", Moscow, 1964; English transl., Consultants Bureau, New York, 1965. MR **32** #597; #598.

2. T. N. L. PATTERSON, "The optimum addition of points to quadrature formulae," *Math. Comp.*, v. 22, 1968, pp. 847–856. MR **39** #3701.

3. G. SZEGÖ, "Über gewisse orthogonale Polynome, die zu einer oszillierenden Belegungsfunktion gehören," *Math. Ann.*, v. 110, 1935, pp. 501–513.

4. W. GAUTSCHI, "On the construction of Gaussian quadrature rules from modified moments," *Math. Comp.*, v. 24, 1970, pp. 245–260. MR **44** #2341a.

5. C. W. CLENSHAW, *Chebyshev Series for Mathematical Functions,* National Physical Laboratory Mathematical Tables, vol. 5, Her Majesty's Stationary Office, London, 1962. MR **26** #362.

6. F. G. TRICOMI, *Vorlesungen über Orthogonalreihen*, Die Grundlehren der math. Wissenschaften in Einzeldarstellungen mit besonderer Berücksichtigung der Anwendungsgebiete, Band 76, Springer-Verlag, Berlin, 1955. MR **17,** 30.

SUPPLEMENT TO

# A Note on the Optimal Addition of Abscissas to Quadrature Formulas of Gauss and Lobatto Type

by

ROBERT PIESSENS & MARIA BRANDERS

pp. 135-139, this issue

```
1          SUBROUTINE KRONRD(N,A,W1,W2,EPS,IER)
     C
     C  THIS SUBROUTINE CALCULATES THE ABSCISSAS A AND WEIGHTS W1
     C  OF THE (2*N+1)-POINT QUADRATURE FORMULA WHICH IS OBTAINED
     C  FROM THE N-POINT GAUSSIAN RULE BY OPTIMAL ADDITION OF
     C  N+1 POINTS. THE OPTIMALLY ADDED POINTS ARE CALLED KRONROD
     C  ABSCISSAS. ABSCISSAS AND WEIGHTS ARE CALCULATED FOR
     C  INTEGRATION ON THE INTERVAL (-1,1). SINCE THIS QUADRATURE
     C  FORMULA IS SYMMETRICAL WITH RESPECT TO THE ORIGINE,ONLY
     C  THE NONNEGATIVE ABSCISSAS ARE CALCULATED. WEIGHTS CORRES-
     C  PONDING TO SYMMETRICAL ABSCISSAS ARE EQUAL.
     C  IN ADDITION, THE WEIGHTS W2 OF THE GAUSSIAN RULE ARE
     C  CALCULATED.
     C
2          REAL*8 A,AK,AN,B,C,TAU,W1,W2,XX
3          DIMENSION A(201),B(201),TAU(201),W1(201),W2(201)
4          COMMON C,INDEKS
     C
     C  INPUTPARAMETERS
     C     N    ORDER OF THE GAUSSIAN QUADRATURE FORMULA TO WHICH
     C          ABSCISSAS MUST BE ADDED.
     C     EPS  REQUESTED ABSOLUTE ACCURACY OF THE ABSCISSAS. THE
     C          ITERATIVE PROCESS TERMINATES IF THE ABSOLUTE
     C          DIFFERENCE BETWEEN TWO SUCCESSIVE APPROXIMATIONS
     C          IS LESS THAN EPS.
     C
     C  OUTPUTPARAMETERS
     C     A    VECTOR OF DIMENSION N+1 WHICH CONTAINS THE NONNEGA-
     C          TIVE ABSCISSAS. A(1) IS THE LARGEST ABSCISSA.A(2*K)
     C          IS A GAUSSIAN ABSCISSA.A(2*K-1) IS A KRONROD ABSCISSA.
     C     W1   VECTOR OF DIMENSION N+1 WHICH CONTAINS THE WEIGHTS
     C          CORRESPONDING TO THE ABSCISSAS A.
     C     W2   VECTOR OF DIMENSION N+1, CONTAINING THE GAUSSIAN
     C          WEIGHTS. W2(2*K-1) =0 AND W2(2*K) IS THE GAUSSIAN
     C          WEIGHT CORRESPONDING TO A(2*K).
     C     IER  ERROR CODE
     C          IF IER=0  ALL ABSCISSAS ARE FOUND TO WITHIN THE
     C          REQUESTED ACCURACY.
     C          IF IER=1 ONE OF THE ABSCISSAS IS NOT FOUND AFTER
     C          50 ITERATION STEPS AND THE COMPUTATION IS TERMINATED.
     C
     C  REQUIRED SUBPROGRAMS
     C     ABWE1  CALCULATES THE KRONROD ABSCISSAS AND CORRES-
     C            PONDING WEIGHTS.
     C     ABWE2  CALCULATES THE GAUSSIAN ABSCISSAS AND THE COR-
     C            RESPONDING WEIGHTS.
```

```
C
      IER = 0
      NP = N+1
      M = (N+1)/2
      INDEKS = 1
      IF(2*M.EQ.N) INDEKS=0
      D = 2.0D0
      AN = 0.0D0
      DO 1 K=1,N
      AN = AN +1.D0
    1 D = D*AN/(AN+0.5D0)
      DO 2 K=1,NP
    2 W2(K) = 0.0D+0

      N2 = N+N+1
      M1 = M-1
C  CALCULATION OF THE CHEBYSHEV COEFFICIENTS OF THE ORTHO-
C  GONAL POLYNOMIAL.
      TAU(1) = (AN+2.D0)/(AN+AN+3.D0)
      B(M) = TAU(1)-1.D0
      IF(N.LT.3) GOTO 4
      AK = AN
      DO 3 L=1,M1
      AK = AK +2.0D0
      TAU(L+1) = ((AK-1.D0)*AK-AN*(AN+1.D0))*(AK+2.D0)*TAU(L)/
     1 (AK*((AK+3.D0)*(AK+2.D0)-AN*(AN+1.D0)))
      ML = M-L
      B(ML) = TAU(L+1)
      DO 3 LL=1,L
      MM = ML+LL
    3 B(ML) = B(ML)+TAU(LL)*B(MM)
    4 B(M+1) = 1.0D0
C  CALCULATION OF APPROXIMATE VALUES FOR THE ABSCISSAS
      BB = SIN(1.570796/(SNGL(AN+AN)+1.))
      X = SQRT(1.-BB*BB)
      S = 2.*BB*X
      C = SQRT(1.-S*S)
      COEF = 1.-(1.-1./AN)/(8.*AN*AN)
      XX = COEF*X
      DO 5 K=1,N,2
C  CALCULATION OF THE K-TH ABSCISSA (=KRONROD ABSCISSA) AND
C  THE CORRESPONDING WEIGHT.
      CALL ABWE1(XX,B,M,EPS,W1(K),N,IER)
      IF(IER.EQ.1) RETURN
      A(K) = XX
      Y = X
      X = Y*C-BB*S
      BB = Y*S+BB*C
      XX = COEF*X
      IF(K.EQ.N) XX = 0.D0
C  CALCULATION OF THE (K+1)-TH ABSCISSA (=GAUSSIAN ABSCISSA)
C  AND THE CORRESPONDING WEIGHTS.
      CALL ABWE2(XX,B,M,EPS,W1(K+1),W2(K+1),N,IER)
      IF(IER.EQ.1) RETURN
      A(K+1) = XX
      Y = X
      X = Y*C-BB*S
      BB = Y*S+BB*C
    5 XX = COEF*X
      IF(INDEKS.EQ.1) GOTO 6
      A(N+1) = 0.D0
```

```
      CALL ABWE1(A(N+1),B,M,EPS,W1(N+1),N,IER)
    6 RETURN
      END

      SUBROUTINE ABWE1(X,A,N,EPS,W,N1,IER)
      REAL*8 A,AI,B0,B1,B2,COEF,D0,D1,D2,DELTA,F,FD,W,X,YY
      DIMENSION A(201)
      COMMON COEF,INDEKS
      ITER = 0
      KA = 0
      IF(X.EQ.0.D0) KA=1
    1 ITER = ITER+1
C  START ITERATIVE PROCESS FOR THE COMPUTATION OF A KRONROD
C  ABSCISSA.
C  TEST ON THE NUMBER OF ITERATION STEPS
      IF(ITER.LT.50) GOTO 2
      IER = 1
      RETURN
    2 B1 = 0.D0
      B2 = A(N+1)
      YY = 4.D0*X*X-2.0D0
      D1 = 0.D0
      IF(INDEKS.EQ.1) GOTO 3
      AI = N+N+1
      D2 = AI*A(N+1)
      DIF = 2.D0
      GOTO 4
    3 AI = N+1
      D2 = 0.D0
      DIF = 1.D0
    4 DO 5 K=1,N
      AI = AI-DIF
      I = N-K+1
      B0 = B1
      B1 = B2
      D0 = D1
      D1 = D2
      B2 = YY*B1-B0+A(I)
      I = I+INDEKS
    5 D2 = YY*D1-D0+AI*A(I)
      IF(INDEKS.EQ.1) GOTO 6
      F = X*(B2-B1)
      FD = D2+D1
      GOTO 7
    6 F = 0.5D0*(B2-B0)
      FD = 4.D0*X*D2
    7 DELTA = F/FD
      X = X-DELTA
      IF(KA.EQ.1) GOTO 8
C  TEST ON CONVERGENCE.
      IF(DABS(DELTA).GT.EPS) GOTO 1
      KA = 1
      GOTO 1
C  COMPUTATION OF THE WEIGHT.
    8 D0 = 1.D0
      D1 = X
      AI = 0.D+0
      DO 9 K=2,N1
      AI = AI+1.D+0
      D2 = ((AI+AI+1.D+0)*X*D1-AI*D0)/(AI+1.D+0)
      D0 = D1
```

```
    9   D1 = D2
        W = COEF/(FD*D2)
        RETURN
        END

        SUBROUTINE ABWE2(X,A,N,EPS,W1,W2,N1,IER)
        REAL*8 A,AN,COEF,DELTA,P0,P1,P2,PD0,PD1,PD2,W1,W2,X,YY
        DIMENSION A(201)
        COMMON COEF,INDEKS
        ITER = 0
        KA = 0
        IF(X.EQ.0.0D0) KA=1
C  START ITERATIVE PROCESS FOR THE COMPUTATION OF A GAUSSIAN
C  ABSCISSA.
    1   ITER = ITER+1
C  TEST ON THE NUMBER OF ITERATION STEPS.
        IF(ITER.LT.50) GOTO 2
        IER = 1
        RETURN
    2   P0 = 1.D0
        P1 = X
        PD0 = 0.D0
        PD1 = 1.0D+0
        AI = 0.0D+0
        DO 3 K=2,N1
        AI = AI+1.D0
        P2 = ((AI+AI+1.D0)*X*P1-AI*P0)/(AI+1.D0)
        PD2 = ((AI+AI+1.D+0)*(P1+X*PD1)-AI*PD0)/(AI+1.D0)
        P0 = P1
        P1 = P2
        PD0 = PD1
    3   PD1 = PD2
        DELTA = P2/PD2
        X = X-DELTA
        IF(KA.EQ.1) GOTO 4
C  TEST ON CONVERGENCE.
        IF(DABS(DELTA).GT.EPS) GOTO 1
        KA = 1
        GOTO 1
    4   AN = N1
C  COMPUTATION OF THE GAUSSIAN WEIGHT.
        W2 = 2.D0/(AN*PD2*P0)
        P1 = 0.0D0
        P2 = A(N+1)
        YY = 4.0D0*X*X-2.D0
        DO 5 K=1,N
        I = N-K+1
        P0 = P1
        P1 = P2
    5   P2 = YY*P1-P0+A(I)
        IF(INDEKS.EQ.1) GOTO 6
C  COMPUTATION OF THE OTHER WEIGHT.
        W1 = COEF/(PD2*X*(P2-P1))+W2
        GOTO 7
    6   W1 = 2.D0*COEF/(PD2*(P2-P0))+W2
    7   RETURN
        END
```