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

\[ \int_{-1}^{+1} f(x) \, dx \approx \sum_{k=1}^{N} \alpha_k f(x_k) + \sum_{k=1}^{N+1} \beta_k f(\xi_k), \]

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

\[ \int_{-1}^{+1} P_N(x) \phi_{N+1}(x) x^k \, dx = 0, \quad k = 0, 1, \ldots, N, \]

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
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

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

and

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

where \( m = \lfloor (N + 1)/2 \rfloor \).

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

\[
\int_{-1}^{+1} P_N(x)\phi_{N+1}(x)T_k(x) \, dx = 0, \quad k = 0, 1, \cdots, N.
\]

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

\[
b_{m-1} = \tau_1 - 1,
\]

\[
b_{m-k} = \sum_{i=1}^{k-1} b_{m-k+i} \tau_i + \tau_k, \quad k = 2, 3, \cdots, m,
\]

where

\[
\tau_k = -\int_{-1}^{+1} P_N(x)T_{N+2k}(x) \, dx \int_{-1}^{+1} P_N(x)T_N(x) \, dx.
\]

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

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

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

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

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

\[
J = \frac{N}{2(l + 1)} I_{N+1,l+1} - \frac{N}{2(l - 1)} I_{N+1,l-1},
\]

where

\[
I_{N,l} = \int_{-1}^{+1} P_N(x)T_l(x) \, dx.
\]
On the other hand, using a property of the Legendre polynomials, (8) can be transformed into
\[ J = \frac{1}{N + 1} \int_{-1}^{1} (1 - x^2)T_i(x) \, d(P_N(x)), \]
which can be expressed as
\[ J = \frac{2 + 1}{2(N + 1)} I_{N,i+1} + \frac{2 - 1}{2(N + 1)} I_{N,i-1}. \]

Since \( \tau_k = I_{N,N+2k}/I_{N,N} \), the recurrence formula
\[ \tau_{k+1} = \frac{[(N + 2k - 1)(N + 2k) - (N + 1)N(N + 2k + 2)]}{[(N + 2k + 3)(N + 2k + 2) - (N + 1)N(N + 2k)]} \tau_k, \]
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
\[ \alpha_k = \frac{C_N}{P_N'(x_k)\phi_{N+1}(x_k)} + \frac{2}{NP_{N-1}(x_k)P_N'(x_k)}, \quad k = 1, 2, \ldots, N, \]
\[ \beta_k = \frac{C_N}{\phi_{N+1}(\xi_k)P_N'(\xi_k)}, \quad k = 1, 2, \ldots, N + 1, \]
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
\[ \int_{-1}^{1} f(x) \, dx \approx \sum_{k=0}^{N+1} \alpha_k f(x_k) + \sum_{k=0}^{N+1} \beta_k f(\xi_k), \]
where the \( x_k \)'s are abscissas of the Lobatto quadrature formula. Consequently, \( x_0 = -1, x_{N+1} = +1 \) and \( x_1, x_2, \ldots, x_N \) 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
\[ \int_{-1}^{1} (1 - x^2)P_N^{(1,1)}(x)\phi_{N+1}(x)T_k(x) \, dx = 0, \quad k = 0, 1, 2, \ldots, N. \]

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
\[ \tau_k = -\int_{-1}^{1} (1 - x^2)P_N^{(1,1)}T_{N+2k} \, dx \Big/ \int_{-1}^{1} (1 - x^2)P_N^{(1,1)}T_N \, dx. \]
Using the relation
\[ \int_{-1}^{1} (1 - x^2)P_{N+1}^{(1,1)}T_i \, dx = \frac{1}{N+2} [ (l+2)I_{N+1,i+1} - (l-2)I_{N+1,i-1} ], \]
where \( I_{N,i} \) 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
\[ \tau_1 = \frac{3(N + 2)}{(2N + 5)}. \]

The expressions for the weights are
\[ \alpha_k = \frac{C_N}{2P_N'(x_k)\phi_{N+1}(x_k)} + \frac{2}{(N+1)(N+2)[P_{N+1}(x_k)]^2}, \]
\[ \text{for } k = 1, 2, \cdots, N, \]
\[ \alpha_0 = \alpha_{N+1} = \frac{2}{(N+2)(N+1)} - \frac{C_N}{2(N+1)\phi_{N+1}(1)}, \]
\[ \beta_k = \frac{N+2}{(N+1)} \frac{C_N}{2[\phi_N(\xi_k) - \xi_k P_{N+1}(\xi_k)]\phi_{N+1}(\xi_k)}, \quad k = 1, 2, \cdots, N + 1, \]
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

SUBROUTINE KRONRC(N, A, hi, W2, EPS, IER)

THIS SUBROUTINE CALCULATES THE ABSCISSAS A AND WEIGHTS W1
OF THE (2*N+1)-POINT QUADRATURE FORMULA WHICH IS OBTAINED
FROM THE N-POINT GAUSSIAN RULE BY OPTIMAL ADDITION OF
N+1 POINTS. THE OPTIMALLY ADDED POINTS ARE CALLED KRONRC
ABSCISSAS. ABSCISSAS AND WEIGHTS ARE CALCULATED FOR
INTEGRATION ON THE INTERVAL (-1,1). SINCE THIS QUADRATURE
FORMULA IS SYMMETRICAL WITH RESPECT TO THE ORIGIN, ONLY
THE NONNEGATIVE ABSCISSAS ARE CALCULATED. WEIGHTS CORRESPONDING TO SYMMETRICAL ABSCISSAS ARE EQUAL.
IN ADDITION, THE WEIGHTS W2 OF THE GAUSSIAN RULE ARE
CALCULATED.

REAL A,AK,AN,B,C,TAU,H,W2,XX

DIMENSION A(2*N+1),W1(2*N+1),W2(2*N+1),TAU(2*N+1)

INPUTPARAMETERS
N CRITERION OF THE GAUSSIAN QUADRATURE FORMULA TO WHICH
ABSCISSAS MUST BE ADDED.
EPS REQUESTED ABSOLUTE ACCURACY OF THE ABSCISSAS. THE
ITERATIVE PROCESS TERMINATES IF THE ABSOLUTE
DIFFERENCE BETWEEN TWO SUCCESSIVE APPROXIMATIONS
IS LESS THAN EPS.

CLTPLTPARAMETERS
A VECTOR OF DIMENSION N+1 WHICH CONTAINS THE NONNEGATIVE
ABSCISSAS. A(1) IS THE LARGEST ABSCISSA, A(2*K) IS A GAUSSIAN ABSCISSA, A(2*K-1) IS A KRONRC ABSCISSA.
W1 VECTOR OF DIMENSION N+1 WHICH CONTAINS THE WEIGHTS
CORRESPONDING TO THE ABSCISSAS A.
W2 VECTOR OF DIMENSION N+1, CONTAINING THE GAUSSIAN
WEIGHTS. W2(2*K-1) =C AND W2(2*K) IS THE GAUSSIAN
WEIGHT CORRESPONDING TO A(2*K).
IER ERROR CODE
IF IER=0 ALL ABSCISSAS ARE FOUND WITHIN THE
REQUESTED ACCURACY.
IF IER=1 ONE OF THE ABSCISSAS IS NOT FOUND AFTER
50 ITERATION STEPS AND THE COMPUTATION IS TERMINATED.

RECLIRC SUBPROGRAMS
ABWE1 CALCULATES THE KRONRC ABSCISSAS AND CORRESPONDING
WEIGHTS.
ABWE2 CALCULATES THE GAUSSIAN ABSCISSAS AND THE CORRESPONDING
WEIGHTS.
C
IER = C
AP = A+1
M = (A+1)/2
INDEKS = 1
IF(2*M.EQ.N) INDEKS=0
C = 2.CC
AN = C.CC
CC 1 K=1,N
AN = AN +1.DC
AN = AN +1.DC
1 C = C+AN/(AN+0.5CC)
CC 2 K=1,N+1
2 W2(K) = C.CD+C

C CALCULATION OF THE CHERYSHEV COEFFICIENTS CF THE CRITIC-
C CALCULATION OF THE CHEBYSHEV COEFFICIENTS CF THE CRITIC-
C CALCULATION OF THE CHEBYSHEV COEFFICIENTS CF THE CRITIC-
C CALCULATION OF THE CHEBYSHEV COEFFICIENTS CF THE CRITIC-
C CALCULATION OF THE CHEBYSHEV COEFFICIENTS CF THE CRITIC-
CALL ABWE1(A(N+1),B,M,EFS,W1,N,IER)
RETURN
END

SUBROUTINE ABWE1(X,A,N,EPS,W,M,IER)
REAL *8 A0,A1,B0,B1,B2,CCEF,CO,C1,C2,DELTA,F,FD,W,X,YY
CIPERSICN A12C1)
COMMON COEF,INDEKS
ITER = C
KA = C
IF(X.EC.C.CDC) KAI=1
1 ITER = ITER+1
C START ITERATIVE PROCESS FOR THE COMPLETITION OF A KRONCDE
ABSCISSA.
C TEST THE NUMBER OF ITERATION STEPS
2 IF(ITER.LT.5C) GOTO 2
IER = 1
RETURN

B1 = C.CC
B2 = A(N+1)
YY = 4.CC*X*X-2.0C0
C1 = C.CC
IF(INDEKS.EQ.1) GOTO 3
AI = N+N+1
C2 = AI*A(N+1)
DIF = 2.DC
GOTO 4
3 AI = N+1
C2 = C.CC
CIF = l.CC
4 DO K=1,N
AI = AI-DIF
I = N-K+1
BC = El
B1 = B2
CC = Cl
C1 = C2
B2 = YY*BC-B1+AI*AI(I)
5 I = l.INDEKS
D2 = YY*Cl-D1+AI*AI(I)
IF(INDEKS.EQ.1) GOTO 6
F = X*(B2-B1)
FD = C2+D1
GOTO 7
6 F = C.50*(B2-B0)
FD = 4.CC*X*C2
7 DELTA = F/FD
X = X-DELTA
IF(KA.EC.1) GOTO 8
C TEST CONVERGENCE.
IF(INDEKS.EQ.1) GOTO 1
1 KA = 1
C2 = GTC 1
C COMPLETATION OF THE WEIGHT.
DC = I.C0
Cl = X
C2 = C.CC+0
D0 = K=2*N1
A1 = A1+0
C2 = C1
C1 = C2
w = CCEF/(FD*W)
RETURN ENC

SLERCLINE ABEW2(X,A,,$A$,N,EPS,W1,W2,N1,IER)
REAL A,AN,CCEF,CELTA,PC,P1,P2,PD0,PC1,PC2,W1,W2,X,YY
DIMENSION AI(2C1)
COMPA COEF,INDeks
ITER = C
KA = C
IF(X-EC.C*DCC) KA=1

C START ITERATIVE PROCESS FOR THE COMPUTATION OF A GAUSSIAN
ABSCISSA.

1 ITER = ITER+1
C TEST ON THE NUMBER OF ITERATION STEPS.
IF(ITER.LT.50) GCTC 2
IER = 1
RETURN
2 PC = 1.CC
P1 = X
PCC = C.DCC
PCI = 1.CC+0
AL = C.CD+C
CC 3 K=2,A1
AI = AL+1.DO
P2 = ((AI+AL+1.CC)*X*P1-A1*PO)/(AI+1.CO)
PC2 = ((AI+AL+1.C+O)*P1+X*PCI)-Al*PD0)/(/AI+1.CO)
PO = P1
P1 = P2
PCC = PCI
3 PCI = PC2
DELTA = P2/PC2
4C X = X-DELTA
IF(KA.EC.1) GCTC 4
C TEST ON CONVERGENCE.
IF(CAES(DELTA).GT.EPS) GCTC 1
KA = 1
GCTC 1
4 AN = A1
C COMPUTATION OF THE GAUSSIAN WEIGHT.
W2 = 2.DO/(AN*PD2*PC)
P1 = C.DCC
P2 = A(A+1)
YY = 4.CDC*X*X-2.CC
DC 5 K=1.A
I = N-K+1
PC = P1
P1 = P2
5 P2 = YY*P1-P0*A(I)
IF(INDEKS.EQ.1) GCTC 6
C COMPUTATION OF THE OTHER HEIGHT.
W1 = CCEF/(PD2*X*(P2-P1))*W2
GCTC 7
6 W1 = 2.CO*COEF/(PC2*(P2-P0))*W2
7 RETURN
ENC