Journal of Mathematical Systems, Estimation, and Control
Vol. 8, No. 2, 1998, pp. 1–12

On Accurate Computation of a Class of Linear Functionals*

Sven-Åke Gustafson† Antonio R. da Silva

Abstract

We consider functions f which are defined for nonnegative arguments. A finite equidistant table of functional values is given numerically. The task is to calculate $F(f)$ where F is a given fixed functional. This task can be looked upon as a generalization of the problem of finding the integral of f over a given interval. The idea is to approximate f with a linear combination of decaying exponentials f^*, which reproduces the given table of functional values. It is assumed that $F(f^*)$ can be evaluated with ease, and several important examples are given, when this certainly is true. Several computational schemes are described and the relationships to classical numerical algorithms are pointed out.

Key words: completely monotonic, decaying exponentials, quadrature rules, interpolation, semi-infinite programming, Vandermonde matrix

AMS Subject Classifications: 65D30, 65D05

1 Formulation of the Problem

We will describe a uniform approach to the following problem: We are given an equidistant table of n values with step-size $h > 0$,

$$f_r = f(rh),\ r = 0, 1, \ldots, n-1, \tag{1.1}$$

of a function f which is defined for nonnegative arguments. Combining this information with assumptions to be introduced later, we want to calculate

*Received November 5, 1996; received in final form June 26, 1997. Summary appeared in Volume 8, Number 2, 1998. This paper was presented at the Conference on Computation and Control V, Bozeman, Montana, August 1996. The paper was accepted for publication by special editors John Lund and Kenneth Bowers.

†This author received financial support from The Federal University, Rio de Janeiro

the value of a linear functional $F(f)$. We give the following instances of such functionals:

Example 1.1 (Interpolation) *Let $T > 0$ be a fixed number and put*

$$F_1(f) = f(T). \tag{1.2}$$

Example 1.2 (Integration) *Let a and b be given real numbers such that $0 \leq a < b$ and put*

$$F_2(f) = \int_a^b f(t)\,dt. \tag{1.3}$$

Example 1.3 (One-sided Fourier integral) *Let ω be a fixed real number and put*

$$F_3(f) = \int_0^\infty e^{i\omega t} f(t)\,dt. \tag{1.4}$$

Example 1.4 (Summation and analytic continuation) *Let z be a fixed complex number and put*

$$F_4(f) = \sum_{r=0}^{\infty} f(rh) z^r. \tag{1.5}$$

Example 1.5 (Error term of the trapezoidal rule) *Let f be be such that the following expression is defined:*

$$F_5(f) = \int_0^\infty f(t)\,dt - h\left(\frac{f(0)}{2} + \sum_{r=1}^{\infty} f(rh)\right). \tag{1.6}$$

Remark 1.1 *The first three functionals occur frequently in engineering mathematics, while the remaining two may require a comment.*

Assume that the function $A(z)$ is defined by a power series with a radius of convergence $R = 1$ and that we need $A(z)$ at a point z_0 with $|z_0| > 1$. If A can be continued analytically to the point z_0, then its value there is well-defined, but how to compute it may not be obvious. However, several convergence acceleration methods are known, which use the numerical values of $f(rh)$, $r = 0, \ldots, n-1$ as input and deliver values of $f(z_0)$ with accompanying error bounds, even if the defining power series (1.5) diverges.

The functional F_5 occurs when sums are to be compared with integrals and either the sum or the integral is easily available. From (1.6) we get

$$F_5(f) = \int_0^\infty f(t)\,dt - h\left(\frac{f(0)}{2} + \sum_{r=1}^{\infty} f(rh)\right),$$

and

$$F_5(f) - \frac{h}{2}f(0) = \int_0^\infty f(t)\,dt - h\sum_{r=0}^{\infty} f(rh).$$

The classical approach to this problem is using the Euler-Maclaurin formula, but then the derivatives of f are required. We will describe a method which only requires values of the function f itself as input.

It is clear that (1.1) must be combined with *quantitative* bounds on f, if one should be able to find an estimate of $F(f)$ and an associated error bound. We have namely the simple

Lemma 1.1 *Let F be a linear functional which is defined for all functions f on the nonnegative axis. Then there are functions f satisfying (1.1) such that $F(f)$ is arbitrarily large, if $F(p) \neq 0$, where*

$$p(t) = \prod_{r=0}^{n-1}(t - rh).$$

Proof: Let f satisfy (1.1) and put

$$f_1 = f + cp,\ c \text{ a constant}$$

Then f_1 satisfies (1.1) for all c, and

$$F(f_1) = F(f) + cF(p).$$

Since $F(p] \neq 0$ was assumed, we can select c to render $|F(f_1)|$ larger than any given bound, establishing the Lemma. QED

In order to get useful results one needs to introduce assumptions on both the functionals F and the admissible functions f. We have namely, that if f^* is an approximation to f then

$$|F(f^*) - F(f)| = |F(f - f^*)|, \tag{1.7}$$

and this simple relation may be used, if $F(f^*)$ is more easily available than is $F(f)$, and we have some means of estimating the right side of (1.7). A bound of the type

$$|F(f - f^*)| \leq ||F|| \cdot ||f - f^*||,$$

is often much too conservative. Compare Remark 3.3

Often one seeks to approximate f by a polynomial f^*. This is frequently done with success for F_1 and F_2 in classical numerical analysis, even if the

potential difficulties caused by equidistant interpolation are well-known. (See e.g. [2]).

An interesting alternative is to work with sinc-expansions, where the function f is represented by its cardinal series. The reader is referred to the text-books by Stenger and Lund/Bowers to learn about this approach. See [17] and [14].

We will instead approximate f by linear combinations of decaying exponentials. This topic will be dealt with in the Sections to follow. Here we recall the fact that both the Euler-Maclaurin summation formula and the Chebyshev acceleration for power series, [7], and [9], may be looked upon as instances of exponential approximation.

2 Approximation with Linear Combinations of Decaying Exponentials

We fit sums of decaying exponentials to the table (1.1) in the following way. Let q be an integer, $\lambda_1, \ldots, \lambda_q$ $x_1, \ldots, x_q$ reals with $0 \leq \lambda_1, < \ldots, < \lambda_q$. Put

$$f^*(t) = \sum_{j=1}^{q} x_j e^{-\lambda_j t}, \tag{2.1}$$

where q and $x_j, \lambda_j, j = 1, \ldots, q$ are to satisfy the constraints

$$f(rh) = \sum_{j=1}^{q} x_j e^{-\lambda_j rh}, \ r = 0, \ldots, n-1. \tag{2.2}$$

The condition (2.2) means that

$$f^*(rh) = f(rh), \ r = 0, \ldots, n-1.$$

In (2.2) we make a change of variables and put

$$u_j = e^{-\lambda_j h}, \ j = 1, \ldots, q. \tag{2.3}$$

Then we arrive at the more familiar equations

$$\sum_{j=1}^{q} x_j u_j^r = f(rh), \ r = 0, \ldots, n-1. \tag{2.4}$$

Since we may choose the integer q, (2.4), and equivalently (2.2), is solvable. If we put $q = n$, then (2.4) becomes a Vandermonde set of equations, which as known, has a unique solutions. The system may have solutions for other combinations of q and n. A special case is $n = 2\ell, q = \ell$ when the solution, if

it exists, defines the weights and abscissæ of a generalized Gauss quadrature rule.

We next derive approximations for the functionals F_1 through F_5 in Section 1, when f is approximated by f^* in (2.1). Then we find after straight-forward computations:

$$\begin{aligned}
F_1(f^*) &= \sum_{j=1}^{q} x_j e^{-\lambda_j T}, \\
F_2(f^*) &= \sum_{j=1}^{q} \frac{x_j}{\lambda_j}(e^{-\lambda_j a} - e^{-\lambda_j b}), \\
F_3(f^*) &= \sum_{j=1}^{q} \frac{x_j}{\lambda_j - i\omega}, \\
F_4(f^*) &= \sum_{j=1}^{q} x_j \frac{1}{1 - z e^{-\lambda_j h}}, \\
F_5(f^*) &= \sum_{j=1}^{q} x_j \left(\frac{1}{\lambda_j} - \frac{h}{2} \coth(\lambda_j h/2) \right).
\end{aligned}$$

We note, that some of these expressions are remarkably simple. Thus $F_3(f^*)$ and $F_4(f^*)$ are rational expressions which are easy to tabulate, if they are considered to be functions of ω and z respectively.

3 Efficient Strategies for Determining Decay Rates and Weights

3.1 A special class of functions

It is clear, that not all functions f can be efficiently approximated by sums of decaying exponentials of the form of (2.1). We shall therefore require that f satisfies:

Assumption E: The function f is said to satisfy Assumption E on $[0, \infty]$, if there are constants $c \leq 0, B > 0$ and a Stieltje's integrator $d\alpha$ such that

$$f(t) = \int_0^\infty e^{(c-t)\tau} d\alpha(\tau), \ t \geq c, \tag{3.1}$$

$$\int_0^\infty |d\alpha(\tau)| \leq B. \tag{3.2}$$

We next illustrate that Assumption E defines a fairly large class of functions.

Example 3.1 *The exponential sum (2.1) satisfies Assumption E, with* $c = 0$. $d\alpha(\tau)$ *has the point-mass* x_j *at* $\tau = \lambda_j$ *and we may take*

$$B = \sum_{j=1}^{q} |x_j|.$$

Example 3.2 *The function*

$$f(t) = \frac{1}{3+t},$$

satisfies Assumption E since we have

$$\frac{1}{3+t} = \int_0^\infty e^{-\delta\tau} \cdot e^{-(3+t-\delta)\tau} d\tau,$$

for any $\delta > 0$. *If we now set*

$$d\alpha(\tau) = e^{-\delta\tau} d\tau,$$

we may put $c = -3 + \delta$ *and*

$$B = \int_0^\infty e^{-\delta\tau} d\tau = 1/\delta.$$

Example 3.3 *All functions satisfying Assumption E may be written as the difference between two functions, which are completely monotonic on* $[0, \infty]$.

Remark 3.1 *Functions, which satisfy Assumption E are analytic and bounded on any halfplane* $\Re(z) = c + \delta$, $\delta > 0$. *Using the expression for the inverse Laplace transform, it is straight-forward to verify that rational functions, which have poles with negative real parts and which are real-valued for real arguments satisfy Assumption E.*

3.2 Quadrature and interpolation

Definition 3.1 *Let* F *be a linear functional which is defined for functions satisfying Assumption E. Set*

$$G(\tau) = F(f), \quad \textit{when } f(t) = e^{-\tau t}. \tag{3.3}$$

Then we shall call G *the generating function of* F.

We immediately arrive at

Lemma 3.1 *Let F and G be as in Definition 3.1 and let f satisfy Assumption E. Then*

$$F(f) = \int_0^\infty e^{c\tau} G(\tau) d\alpha(\tau). \tag{3.4}$$

Proof: We find

$$\begin{aligned} \int_0^\infty e^{c\tau} G(\tau) d\alpha(\tau) &= \int_0^\infty e^{c\tau} F(e^{-t\tau}) d\alpha(\tau) = \\ F\left(\int_0^\infty e^{c\tau} e^{-t\tau} d\alpha(\tau) \right) &= F(f), \end{aligned}$$

using the expression (3.1) for $f(t)$. QED

We next derive a pair of quadrature and interpolation rules which are algebraically equivalent. We will also discuss the choice of the nodes. Combining (3.4) and (1.1) with (3.1) we get the relations

$$G(f) = \int_0^\infty e^{c\tau} G(\tau) d\alpha(\tau) \tag{3.5}$$

$$f_r = \int_0^\infty e^{(c-hr)\tau} d\alpha(\tau), \ r = 0, \ldots, n-1. \tag{3.6}$$

We next make a change of variables in the integrals (3,5) and (3.6), putting

$$e^{-h\tau} = u.$$

Then we obtain

$$G(f) = \int_0^1 u^{-c/h} \bar{G}(u) d\beta(u), \tag{3.7}$$

$$f_r = \int_0^1 u^{-c/h} u^r d\beta(u), \ r = 0, \ldots, n-1, \tag{3.8}$$

where

$$\bar{G}(u) = G(-\ln u/h), \tag{3.9}$$

and $d\beta(u)$ is of bounded variation on $[0,1]$. We next prove

Lemma 3.2 *Let $\bar{G}$ be defined by (3.9) and the generating function (3.3) of the linear functional F. Let $u_1, \ldots, u_n$ be n distinct numbers in $[0,1]$. Finally, let $x_1, \ldots, x_n$ be the solution of the linear system*

$$\sum_{j=1}^n x_j u_j^r = f_r, \ r = 0, \ldots, n-1, \tag{3.10}$$

and let $y_1, \ldots, y_n$ be the solution of the system:

$$\sum_{r=0}^{n-1} y_r u_j^r = \bar{G}(u_j),\ j = 1, \ldots, n. \tag{3.11}$$

Then

$$\sum_{r=0}^{n-1} y_r f_r = \sum_{j=1}^{n} x_j \bar{G}(u_j). \tag{3.12}$$

Proof: We find immediately

$$\sum_{r=0}^{n-1} y_r f_r = \sum_{r=0}^{n-1} y_r \sum_{j=1}^{n} x_j u_j^r = \sum_{j=1}^{n} x_j \sum_{r=0}^{n-1} y_r u_j^r = \sum_{j=1}^{n} x_j \bar{G}(u_j).$$

QED

Remark 3.2 *(3.12) offers two different, but algebraically equivalent estimates for $F(f)$ based on the table (1.1) of functional values and the generating function of (3.3).*

We also obtain

Lemma 3.3 *Use the same notations as in Lemma 3.2 and put*

$$Q(u) = \sum_{r=0}^{n-1} y_r u^r.$$

Then we get the error

$$R_n = \int_0^1 u^{-c/h} \left(\bar{G}(u) - Q(u) \right) d\beta(u), \tag{3.13}$$

when we approximate $F(f)$ with one of the two expressions in (3.12)

Proof: Using (3.8) we find

$$\sum_{r=0}^{n-1} y_r f_r = \int_0^1 u^{-c/h} \left(\sum_{r=0}^{n-1} y_r u^r \right) d\beta(u) = \int_0^1 u^{-c/h} Q(u) d\beta(u).$$

Combining this with (3.7) we arrive at the desired expression (3.13). QED

We now consider the task of selecting the nodes $u_1, \ldots, u_n$ in order to minimize bounds for the error R_n as given by (3.13). We will present the two methods A) and B) below:

Method A): Assume that $\bar{G}$ is continuous on $[0,1]$. We seek to render the expression

$$\max_{0\le u\le 1} |\bar{G}(u) - Q(u)| \tag{3.14}$$

small. The optimal value may be computed by means of the exchange algorithm by Remez. (See e.g. [1]). This requires considerable computational work and the optimal solution depends nonlinearly on $\bar{G}$.

Instead we take $u_1, \ldots, u_n$ at the zeroes of T_n^*, the shifted nth degree Chebyshev polynomial. According to [16], the value of (3.14) will be larger than the optimal by a factor which grows like $\ln n$, when $n \to \infty$. Thus we put

$$u_j = \frac{1}{2}(1 + \cos\theta_j),\ \theta_j = \frac{(j-1/2)\pi}{n}, j = 1, \ldots, n. \tag{3.15}$$

Method B): We next discuss selecting the nodes in order to render the following expression small:

$$\max_{0\le u\le 1} u^{-c/h} |\bar{G}(u) - Q(u)|. \tag{3.16}$$

Melinder [15] has generalized the result of Powell mentioned above and shown that one should take u_j at the zeroes of the shifted Jacobi polynomial of degree n corresponding to the weighting function

$$(1-u)^\alpha u^\beta,$$

where we in this case take $\alpha = -1/2,\ \beta = -(2c/h + 1/2)$. Since the three-terms recurrence relation of the Jacobi polynomials has coefficients which are known analytically, the nodes u_j can be evaluated in a stable manner. See e. g. [5].

Remark 3.3 (Construction of exponential approximation to f) *Using either Method A) or Method B) above we determine first x_j and u_j. From (2.3) we calculate λ_j and hence (2.1) gives the expression sought. Note that $F(f^*)$ may approximate $F(f)$ well, even if $f^*(t)$ deviates appreciably from $f(t)$ for $t > nh$.*

Remark 3.4 (Gauss quadrature) *If f is completely monotonic on $[0,\infty]$, then (3.10) may be replaced by (2.4) and the corresponding generalized Gauss quadrature rule can be calculated. It is well-known that numerical difficulties frequently occur, if one seeks to determine the abscissæ and weights from the numerical values of f_r in (1.1).*

4 Numerical Examples and Applications

If the table (1.1) and the functional F are given, then one only needs the generating function G to be able to compute $F(f)$ using the methods described in Section 3. We mention the examples F_1 through F_5 in Section

1. In these cases one gets the approximation formulas listed in Section 2. Next we present:

Numerical example: We consider the function

$$f(t) = \frac{1}{\sqrt{1+t}},$$

and form the table (1.1) with $h = 0.2$ and $n = 6$ Based on these data we estimate $f(T)$ for $T = 0, 0.1, \ldots 2.0$ We compare two methods:

The first one consists of constructing a polynomial of degree 5 which interpolates the given data and use this polynomial as an approximation for f. We found that in the interval $0 \leq T \leq 1$ the observed largest error had absolute value $2.3 \cdot 10^{-5}$, but for $T \geq 1$ the absolute value grew progressively with T to reach $9.4 \cdot 10^{-2}$ at $T = 2$, the largest T-value studied.

The second method was to construct an exponential approximation for $f(t)$ using Method A of Section 3, i.e., allocating the nodes u_j of (3.11) according to (3.15). The largest error observed for $0 \leq T \leq 1$ was $1.2 \cdot 10^{-7}$ and for $1 \leq T \leq 2$, it was $2.5 \cdot 10^{-5}$. The calculations were carried out by means of a Fortran program and working in single precision, in this case a relative error of at most about $1.2 \cdot 10^{-7}$. Hence we may conclude that the exponential approximation reproduced the function f within working precision in the interval $[0, 1]$, but a certain loss of accuracy was observed outside the interval. The exponential approximation was more accurate than straight-forward use of interpolating polynomials in this example. One should note that this conclusion rests heavily on the fact that the interpolation points were fixed to be equidistant. It is to be expected that if F_2 is defined as the task to integrate over parts of the interval $[0, 2]$, the conclusion would be similar.

Examples of F_3 can be found in [13] where both methods of Section 3 are discussed and applied to numerical examples.

Computation of instances of F_4 is dealt with in e.g. [3], [7], [8] and [10]. Due to the special character of F_4 it is possible to calculate the approximation $F_4(f^*)$ without first determining the weights x_i. This is true even in the case when f is completely monotonic over the positive axis and one seeks estimates based on the corresponding generalized Gauss rules. This is achieved using the ϵ-algorithm by Wynn. See [3]. In the general case of F_4 when (3.12) is used implicitly, the estimate $F_4(f^*)$ is delivered by the Chebyshev acceleration algorithm and its generalization to the Jacobi case [7], [10] and [12].

The results of preliminary numerical experiments indicate that the sums of slowly converging positive series may be estimated accurately by calculating $F_5(f)$. This could be an alternative to the Euler-Maclaurin summation formula when derivatives are not available numerically.

5 Concluding Remarks

If we employ the methods A and B of Section 3, then the estimate $F(f^*)$ becomes a linear function of the values $f_0, \ldots, f_{n-1}$, as is apparent from (3.12). This makes the sensitivity analysis simple, since the numbers $y_1, \ldots, y_n$ in (3.12) are easily available.

The analysis of the present paper may be extended to the case of a not equidistant table (1.1). If B in (3.2) is known, one may use semi-infinite programming to find upper and lower bounds for the value of $F(f)$ Considerably more computational work will be required. This matter is dealt with in [11]. An introduction to semi-infinite programming is given in [4].

References

[1] E.W. Cheney. *Introduction to Approximation Theory.* New York: McGraw-Hill, 1956.

[2] G. Dahlquist, and Å. Björck. *Numerical Methods.* Englewood Cliffs, NJ: Prentice-Hall, 1974.

[3] G. Dahlquist, S.Å. Gustafson and K. Siklo'si. Convergence acceleration from the point of view of linear programming, *BIT* **5** (1965), 1–16.

[4] K. Glashoff, and S.-Å. Gustafson. *Linear Optimization and Approximation.* New York: Springer-Verlag, 1983.

[5] G.H. Golub, J.H. Welsch. Calculation of Gauss quadrature rules, *Math. Comp.* **23** (1969), 221–230.

[6] S.-Å. Gustafson. Rapid computation of interpolation formulæ and mechanical quadrature rules, *CACM* **14** (1971), 798–801.

[7] S.-Å. Gustafson. Convergence acceleration on a general class of power series, *Computing* **21** (1978), 53–69.

[8] S.-Å. Gustafson. Two computer codes for convergence acceleration, *Computing* **21** (1978), 87–91.

[9] S.-Å. Gustafson. Numerical inversion of Laplace transforms using integration and convergence acceleration, Techn. Rep, Swedish Nuclear Fuel and Waste Management Co., Stockholm, Sweden, 1991.

[10] S.-Å. Gustafson. Rational approximation of power series from linear acceleration schemes, some numerical experiments, Working Paper No. 175, HSR, Box 2557 Ullandhaug, N-4004 Stavanger, Norway, 1993.

[11] S.-Å. Gustafson. Calculating bounds for a class of Laplace integrals, in *Parametric Optimization and Related Topics IV*, (J. Guddat, H. Th. Jongen, F. Nozicka, G. Still and F. Twilt, eds.). Frankfurt am Main: Verlag Peter Lang, 1996.

[12] S.-Å. Gustafson. Constructing accurate linear approximations to a class of exponential interpolation problems, *Proceedings of the Second World Congress of Nonlinear Analysts*. Amsterdam: Elsevier, 1996.

[13] S.-Å. Gustafson and G. Dahlquist. On the computation of slowly convergent Fourier integrals, *Methoden und Verfahren der Physik*, **6** (1972), 93–112.

[14] J. Lund and K.L. Bowers. *Sinc Methods for Quadrature and Differential Equations*, Philadelphia: SIAM, 1992.

[15] I. Melinder. Accurate approximation in weighted maximum norm by interpolation, *J. Approx. Theory*, **22** (1978), 33–45.

[16] M.J.D. Powell. On the maximum error of polynomial approximation defined by interpolation and by least squares criteria, *Comp. J.*, **9** (1966/1967), 404–407.

[17] F. Stenger. *Numerical Methods Based on Sinc and Analytic Functions*. New York: Springer-Verlag, 1993.

Stavanger College, Norway

The Federal University, Rio de Janeiro, Brazil

Communicated by John Lund