SIAM J. NUMER. ANAL.
Vol. 57, No. 5, pp. 2200–2216

GAUSS–HERMITE QUADRATURE FOR THE BROMWICH INTEGRAL*

J. A. C. WEIDEMAN†

Abstract. Several popular methods for the numerical inversion of the Laplace transform are based on quadrature approximation of the Bromwich integral. A key ingredient is contour deformation, and a variety of contours have been proposed. The choice of quadrature rule, however, has always been the rapidly convergent trapezoidal rule. In this paper, the classical Gauss–Hermite rule is proposed as an alternative to the trapezoidal rule. By means of a numerical saddle point approach, contour parameters are computed for the case of a parabolic contour where the singularities of the transform are assumed to be located on the nonpositive real axis. In various numerical experiments, it is shown that the Gauss–Hermite method proposed here converges more rapidly than the best methods based on the trapezoidal rule.

Key words. inverse Laplace transform, quadrature, matrix exponential

AMS subject classifications. 65R10, 65D32

DOI. 10.1137/18M1196273

1. Introduction.

Many methods for the numerical inversion of the Laplace transform are based on the quadrature approximation of the Bromwich integral [3, 9]

$$f(t) = \frac{1}{2\pi i}\int_\Gamma e^{zt}F(z)\,dz. \tag{1.1}$$

The given image function $F(z)$ is assumed analytic in the half-plane $\operatorname{Re} z > \sigma_0$ with σ_0 the convergence abscissa. The original function $f(t)$ is to be evaluated at one or more values of the time variable $t > 0$. The contour Γ is typically taken to be the line $\operatorname{Re} z = \sigma$ with $\sigma > \sigma_0$. By application of Cauchy's theorem, however, it is possible to deform this contour into one that is better suited for computation. Such contour deformation techniques are powerful tools for both the analytical and the numerical calculation of (1.1).

In the case of numerical computation the contour deformation is used as follows. On the Bromwich line $\operatorname{Re} z = \sigma$ the integral (1.1) is not ideally suited for quadrature because of the typical slow decay of $|F(z)|$ as $\operatorname{Im} z \to \pm\infty$, as well as the oscillatory nature of the factor e^{zt}. By deforming this line into a Hankel contour,[1] which is a contour that starts at infinity in the third quadrant, winds around all singularities of $F(z)$ and ends at infinity in the second quadrant, the decay properties of the factor e^{zt} in the left half-plane can be exploited. Because of the strong damping of the oscillations, the computation of the integral becomes more efficient, i.e., fewer quadrature nodes and hence function evaluations are required to reach a prescribed accuracy.

In theory, the contour deformation is valid as long as the deformed contour encloses all singularites of $F(z)$ and provided $|F(z)| \to 0$ uniformly in $\operatorname{Re} z \le \sigma_0$ as

*Received by the editors June 22, 2018; accepted for publication (in revised form) July 12, 2019; published electronically September 12, 2019.
https://doi.org/10.1137/18M1196273
Funding: The work of the author was supported by the National Research Foundation of South Africa grant 103321 and by the Center of Excellence in Mathematics and Statistics.
†Applied Mathematics, Stellenbosch University, Stellenbosch, 7600, South Africa (weideman@sun.ac.za).
[1] A few Hankel contours are displayed in Figure 6.

$|z| \to \infty$; see [20]. In order to be a viable numerical strategy, however, two key implementation details require attention:

(a) the choice of contour Γ, and

(b) the choice of quadrature rule.

The most famous method in this class is the one of Talbot [20], who proposed (a) a cotangent contour (which evolved from a steepest descent argument [19]) and (b) the trapezoidal rule (which is known to integrate rapidly decaying functions on infinite domains very accurately [22]). As pointed out in [22], however, Talbot's method was anticipated by about two decades by Butcher [2], who suggested the use of (a) a parabolic contour and (b) the trapezoidal rule. Other contours that have since been considered are hyperbolas [13, 14, 27], contours defined by rational functions [4, 20], and a modified cotangent contour [4] (the latter essentially a truncated version of Talbot's original contour).

Most research efforts in this area have focused on (a), namely, the choice of contour. This involves not only the type of contour but also the optimal tuning of the parameters that define each such contour. As for (b), almost all methods that we are aware of use either the trapezoidal rule or its sibling, the midpoint rule. One exception is the method of Salzer [16], with later additions by Piessens [15], who created special Gaussian quadrature formulas for (1.1). As a general inversion tool, however, the Salzer–Piessens class of methods has not acquired the same popularity as the Butcher–Talbot methods.

In this paper we consider an alternative to the trapezoidal/midpoint rule for (b), namely, Gauss–Hermite quadrature. On a parabolic contour, the exponential factor in (1.1) gives rise to decay of Gaussian type in the integrand, which is the typical situation in which the Gauss–Hermite rule can provide high accuracy.

We alluded to the fact that once a type of contour is selected, its free parameters still have to be fixed for maximum accuracy. These optimal parameters depend on the singularity locations of $F(z)$, and a one-size-fits-all approach is not feasible. Instead, we have to settle for a "one-size-fits-many" approach, by restricting ourselves to the special singularity configuration that seems to arise most often in practice, namely, singularities on the nonpositive real axis. Examples are $F(z) = (z+1)^{-1}$ and $z^{-1/2}$ (assuming the appropriate branch cut for the latter), or the matrix-valued function $F(z) = (zI + A)^{-1}$ where A is a symmetric positive definite matrix and I the identity matrix of the same size (see section 5 below). For this class of transforms optimal parameters and optimal convergence rates have been proposed in a series of papers [4, 25, 27], the results of which are summarized in Table 1. As is evident from the last entry of that table (as well as the numerical results to follow), the method introduced here provides a substantial improvement over the current collection of methods in the Butcher–Talbot family.

It should be noted that only quadrature methods in the Butcher–Talbot class are represented in Table 1. There are other, nonquadrature methods that can have higher rates of convergence. For example, a method based on best approximation to the exponential on the negative real axis was discussed in [23]. In the setting of Table 1 this method would converge as $10^{-1.9m}$, but generating it is not so simple as setting up a quadrature rule. Other pros and cons are discussed in [23].

Relatedly, all the methods listed in Table 1 achieve geometric convergence only because the contour of integration is allowed to change with m—it widens and moves into the right half-plane as the number of quadrature nodes is increased. This rules out any recursive implementation that reuses function values. It should be possible to implement the trapezoidal-based methods in such a recursive algorithm, provided

TABLE 1

Estimated order of convergence for Butcher–Talbot methods as cited in the literature. These results are restricted to a single value of t in (1.1) and to transforms $F(z)$ that are analytic everywhere off the nonpositive real axis. It is also assumed that $\overline{F(z)} = F(\overline{z})$ so that symmetry can be exploited. That is, the image function $F(z)$ need only be evaluated in either the upper (or lower) half-plane. Here m represents the number of quadrature nodes in such a half-plane, i.e., the total number of evaluations of $F(z)$. (The cotangent entries refer to various sets of parameter choices in, and modifications of, Talbot's original contour [20]*. As no convergence rate was mentioned explicitly in* [20]*, we estimated the rate recorded in the first row of this table from the empirical data given in the abstract of that paper.)*

Method (contour, rule)	Order of convergence	Reference
cotangent, trapezoidal	$10^{-0.6m}$	[20]
cotangent, trapezoidal	$10^{-0.8m}$	[25]
parabola, trapezoidal	$10^{-0.9m}$	[27]
hyperbola, trapezoidal	$10^{-1.0m}$	[27]
rational, trapezoidal	$10^{-1.1m}$	[4]
cotangent, trapezoidal	$10^{-1.2m}$	[4]
parabola, Gauss–Hermite	$10^{-1.4m}$	[this paper]

the contour is kept fixed at all levels of refinement. Whether the resulting suboptimal accuracy will be compensated for by the decrease in total function evaluations has, to the best of our knowledge, not been investigated yet. A drawback of the method proposed here is that such a recursive implementation is not even an option, because of the nonuniform spacing of the Hermite nodes.

The outline of the paper is as follows: in section 2 we review Gauss–Hermite quadrature in the classical setting of integration of a real-valued function on the real line. In particular, we recall the contour integral formula for the error in the case of analytic integrands. The general setting is extended to the special case of the Bromwich integral in section 3, where the integrand is now complex valued on the real axis. We propose contour parameters and present a first set of numerical experiments using transforms taken from tables. By analyzing the integral error formula in section 4, we provide an informal justification of the contour parameters proposed here. In further tests, this time taken from actual engineering problems, the superiority of the Gauss–Hermite method is confirmed in section 5.

2. Gauss–Hermite quadrature. Before considering the complex integral (1.1), here is a quick review of some details of Gauss–Hermite quadrature for a real-valued function $g(x)$ defined on the real line, i.e.,

$$\int_{-\infty}^{\infty} e^{-x^2} g(x)\,dx \approx \sum_{k=1}^{n} w_k g(x_k). \tag{2.1}$$

The nodes, x_k, are the roots of $H_n(x)$, the Hermite polynomial of degree n. The w_k are the weights, determined uniquely by the property that if $g(x)$ is a polynomial of degree not exceeding $2n-1$, then the approximation sign can be replaced by an equality. For smooth functions $g(x)$, the convergence can be quite fast. For example, if the function can be analytically extended into a sizeable region of the complex plane, then the convergence is rapid. This statement can be made precise with the aid of the following theorem, proofs of which can be found in [1, 6, 18].

THEOREM 2.1. *The error in the approximation* (2.1) *(left side minus right) can be expressed as*

$$R_n(f) = \frac{1}{2\pi i} \int_C \Phi_n(z) g(z)\,dz, \tag{2.2}$$

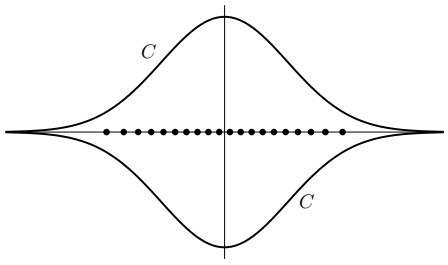

FIG. 1. *An example of the contour C of Theorem* 2.1.

where

$$\Phi_n(z) = \frac{Q_n(z)}{H_n(z)}, \qquad Q_n(z) = \int_{-\infty}^{\infty} e^{-x^2} \frac{H_n(x)}{z-x}\, dx. \tag{2.3}$$

The contour C in (2.2) starts at $z = -\infty$, goes below the zeros of $H_n(x)$ in the lower half-plane to $z = +\infty$, and then returns above the zeros in the upper half-plane with $g(z)$ assumed analytic inside of it. (A sample contour is shown in Figure 1.)

Known as the Hermite function of the second kind, the function $Q_n(z)$ in (2.3) decreases rapidly in magnitude as $|z| \to \infty$. For precise estimates, see [6, 10], where a closed form expression for $Q_n(z)$ in terms of Kummer's U-function can also be found. Likewise, $H_n(z)$ increases rapidly, so the so-called characteristic function $\Phi_n(z)$ in the error integral (2.2) can be quite small provided the contour C is not overly restricted by singularities or growth of $g(z)$. Using such arguments, Barrett [1] derived an error estimate in the case where $g(z)$ has a pair of conjugate poles. The estimate is of root-exponential form $O(e^{-c\sqrt{n}})$, where the constant c increases linearly with the distance from the pole to the real axis.

In [17, 18], a semigraphical approach for estimating the error integral (2.2) was proposed. Since we shall use this technique in the following, we give a brief outline. If $g(z)$ is a meromorphic function, then the error integral (2.2) can in principle be evaluated with the help of a keyhole contour that encircles the poles. Assuming simple poles, the contributions will be terms of the form $\varrho_0 \Phi_n(z_0)$, where $z = z_0$ is a pole and ϱ_0 is the corresponding residue. An order-of-magnitude error estimate can therefore be made by looking at level curves of $\Phi_n(z)$, as shown in Figure 2. Similar plots for other quadrature rules are displayed in [17, 18].[2]

As an example of extracting order-of-magnitude error estimates from plots such as these, consider the function $g(x) = 1/(x^2 + 4)$, with poles located at $z_0 = \pm 2i$. In Figures 2(a) and 2(b) these points are located, respectively, on the 10^{-5} and 10^{-8} level curves, roughly. The actual errors in the 8- and 16-point Gauss–Hermite rules are, respectively, about 6.2×10^{-6} and 1.1×10^{-8}.

When the function is not meromorphic but has branch point singularities, the application of Cauchy's theorem will require a modification that we shall not pursue here. On the other hand, when $g(z)$ is an entire function a saddle point analysis should be used, with the saddle points effectively playing the role of the poles; for details, see [6, 18]. It will be necessary to include such an analysis in our error optimizing strategy of section 4.

[2]Curiously, the Gauss–Hermite figures in the technical report [17] got omitted in the final published paper [18].

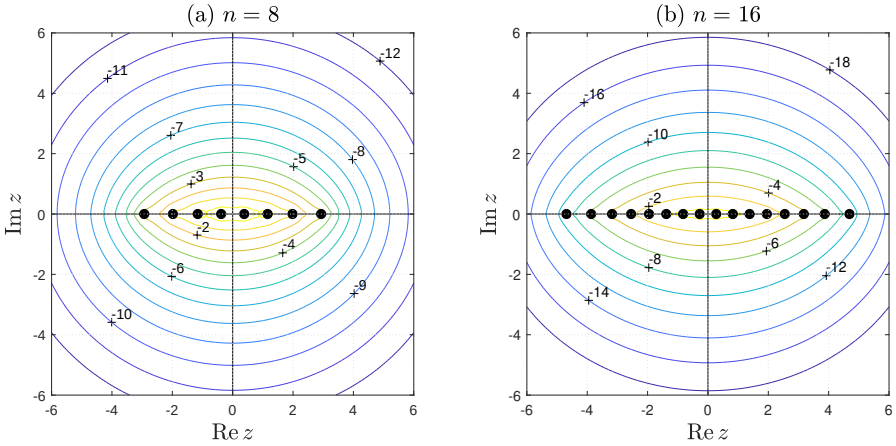

FIG. 2. *Level curves of* $\log_{10}|\Phi_n(z)|$ *with* $\Phi_n(z)$ *defined in* (2.3). *The dots are the quadrature nodes of the* 8- *and* 16-*point rules, respectively. Based on the location of the singularities of* $g(z)$, *order-of-magnitude estimates for the error in Gauss–Hermite quadrature can be deduced from these plots as explained in the text.*

TABLE 2

Absolute errors in the n-*point numerical approximation of the integral* (2.4) *for the case* $g(x) = 1/(x^2+1)$. *The two columns marked GH refer to the Gauss–Hermite rule as applied, respectively, to the left side of* (2.4) *and to the right side. The optimal value of* L *was determined by numerical sampling. The final column shows the corresponding errors in the trapezoidal rule, with optimal step size* h *likewise determined empirically. The results show the clear superiority of the Gauss–Hermite method with optimal scaling over the other two rules.*

n	GH ($L = 1$)	GH (optimal L)	Trap (optimal h)
16	1.6×10^{-4}	2.6×10^{-7} (0.63)	2.9×10^{-6} (0.39)
32	1.6×10^{-6}	1.9×10^{-11} (0.58)	8.4×10^{-10} (0.26)
48	4.6×10^{-8}	6.2×10^{-15} (0.55)	9.4×10^{-13} (0.20)

Evidently, singularities further from the real axis correspond to higher accuracy, a fact that can be exploited as follows. If one rescales the independent variable via $x = Lt$, with L a constant, then

$$\int_{-\infty}^{\infty} e^{-x^2} g(x)\,dx = L\int_{-\infty}^{\infty} e^{-L^2t^2} g(Lt)\,dt = L\int_{-\infty}^{\infty} e^{-t^2} e^{t^2} e^{-L^2t^2} g(Lt)\,dt, \tag{2.4}$$

and the rule (2.1) can now be applied to the function $G(t) = e^{t^2(1-L^2)} g(Lt)$. For the right L, the singularities of the function $G(z)$ may be farther from the real axis than those of $g(z)$ and improved accuracy can be expected. On the other hand, the residues may also be larger because of the additional exponential factor, so there is a balance.

A general theory for the optimal choice of L does not seem to exist. Instead, we offer numerical results in Table 2 where we have determined L empirically. The effect of the optimal scaling factor is clear: it improves the classical rule (fixed $L = 1$) by several orders of magnitude, and it also improves on the trapezoidal rule with optimal step size by a substantial amount. It is the latter improvement that provided the inspiration for the present paper.

For the example of Table 2, a theoretical estimate for the optimal step size in the trapezoidal rule in the limit of large n can be easily derived; see, for example, [22].

The corresponding theoretical estimate for the optimal Gauss–Hermite parameter L is not known, but the analysis of section 4 could provide a first step in this direction.

3. Inversion formula for the Laplace transform. As contour of integration in (1.1), we consider the parabola

$$\Gamma: \quad z = \mu(1+i\phi)^2, \qquad -\infty < \phi < \infty, \quad \mu > 0. \tag{3.1}$$

The contour intersects the real axis at $z = \mu$, and the imaginary axis at $z = \pm 2\mu i$. A few such contours can be seen in Figure 6. With ϕ extended to a complex variable, the conformal map (3.1) transplants the inside of the parabola in the z-plane to the upper half of the complex ϕ-plane, with the negative real axis mapping to the line $\operatorname{Im}\phi = +i$. The outside of the parabola maps to the lower half of the complex ϕ-plane. For parameter tuning, this means one has to analyze the quadrature error in a strip of analyticity that is nonsymmetric about the real axis. This was done for the trapezoidal rule in [27].

By the change of variables (3.1), the contour integral (1.1) is transformed to the real line

$$\int_\Gamma e^{zt}F(z)\,dz = \int_{-\infty}^{\infty} e^{z(\phi)t}F(z(\phi))z'(\phi)\,d\phi.$$

The scaling strategy (2.4) is introduced by defining r via $\phi = Lr$, leading to

$$f(t) = \frac{L}{2\pi i}\int_{-\infty}^{\infty} e^{-r^2}e^{r^2+z(Lr)t}F(z(Lr))z'(Lr)\,dr = \int_{-\infty}^{\infty} e^{-r^2}g(r)\,dr, \tag{3.2}$$

where

$$g(r) = \frac{L}{2\pi i}e^{r^2+z(Lr)t}F(z(Lr))z'(Lr). \tag{3.3}$$

Gauss–Hermite quadrature can now be applied to the integral on the far right of (3.2). If the symmetry property mentioned in the caption of Table 1 is satisfied, the approximation can be computed as

$$f(t) \approx 2\,\mathrm{Re}\Big\{\sum_{k=1}^{m} w_k g(r_k)\Big\}, \tag{3.4}$$

where the r_k are the positive (resp., nonnegative) roots of H_n and $m = n/2$ (resp., $m = (n+1)/2$) when n is even (resp., odd). The w_k are the corresponding Gauss–Hermite weights, but when n is odd the weight corresponding to the node at $r = 0$ should be halved. A MATLAB implementation of the formula (3.4) can be found in [24].

For a given n, the accuracy of the method (3.4) depends quite strongly on good choices of μ and L. An obvious relationship between these parameters is $\mu L^2 t = 1$, because that will make the exponential factor in (3.3) vanish. This leaves one with the "correct" exponential behavior in (3.2), and only a single parameter to tune. Focusing only on the exponential part disregards the singularities in (3.3), however, and as discussed below (2.4), it might be advantageous to allow some exponential growth in the factor $g(r)$ in (3.2) in exchange for a bigger region of analyticity.

The actual dependence of the error on μ and L can be seen in Figure 3, where we display absolute errors for the simple test case $F(z) = 1/z$, $f(1) = 1$. It is clear that accuracy in the 10^{-11}–10^{-13} range can be achieved with the $n = 16$ rule (i.e., 8 function evaluations in total), provided μ and L are near optimal.

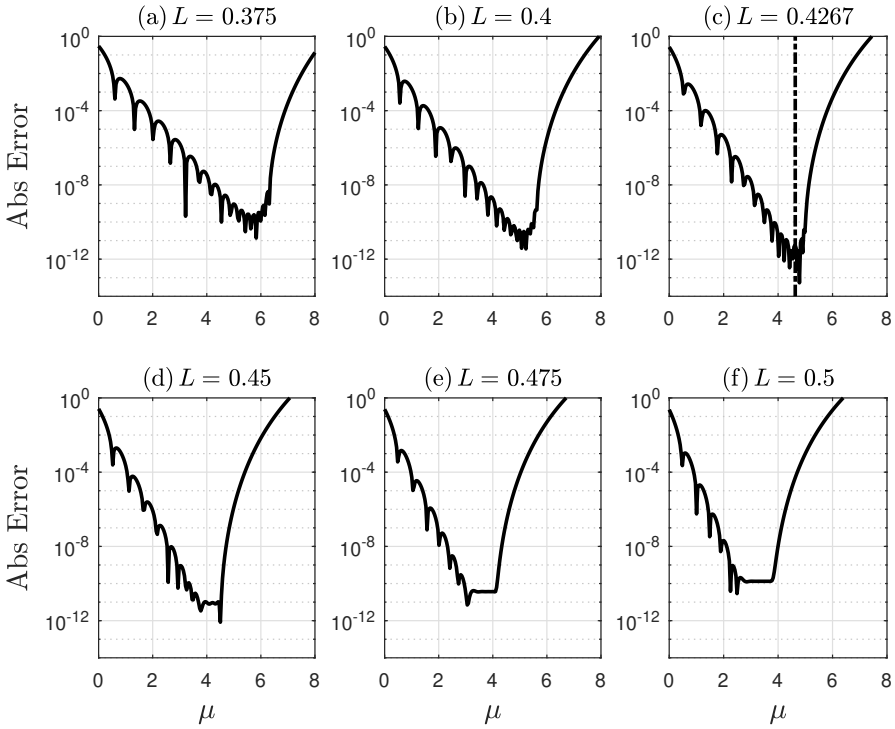

FIG. 3. *Absolute errors in the method* (3.4) *as functions of* μ *for increasing values of* L *for the* $n = 16$ *case. The model problem is* $F(z) = 1/z$, $t = 1$. *The parameter choices in subplot* (c) *(with the value of* μ *fixed at the vertical dash-dot line) correspond to the suggested parameters in the* $n = 16$ *row of Table* 3.

TABLE 3

Suggested parameters in the method (3.4) *for the special case* $F(z) = 1/z$, $t = 1$. *The fourth and fifth columns list, respectively, the estimated error and the optimal location of the saddle point as explained in section* 4. *(These numbers are not used in the actual implementation.) The software on the author's web page* [24] *contains additional data values and to higher precision.*

n	μ	L	$\log_{10}$(err. est.)	saddle point
4	1.4545	0.7450	-2.5204	$4.2932 - 2.4955i$
8	2.5217	0.5736	-5.3300	$5.9702 - 3.8447i$
12	3.5772	0.4840	-8.1250	$7.2770 - 4.8744i$
16	4.6299	0.4267	-10.9125	$8.3868 - 5.7573i$
20	5.6801	0.3860	-13.6954	$9.3678 - 6.5362i$

By using the method described in the next section, we propose the parameters in Table 3. The row corresponding to $n = 16$ in the table corresponds to the near-optimal choice of parameters corresponding to the vertical dash-dot line in Figure 3(c). The reference to saddle points in the table will be clarified in the next section.

At first, the parameters in Table 3 seem to be of limited use, as they are presented here for a single transform $F(z)$ at a specific value of t. Fixing $t = 1$ is no restriction, however, as a change of variable $zt = \zeta$ in (1.1) leads to

$$f(t) = \frac{1}{2\pi i t} \int_{\Gamma'} e^{\zeta} F(\zeta/t)\, d\zeta. \tag{3.5}$$

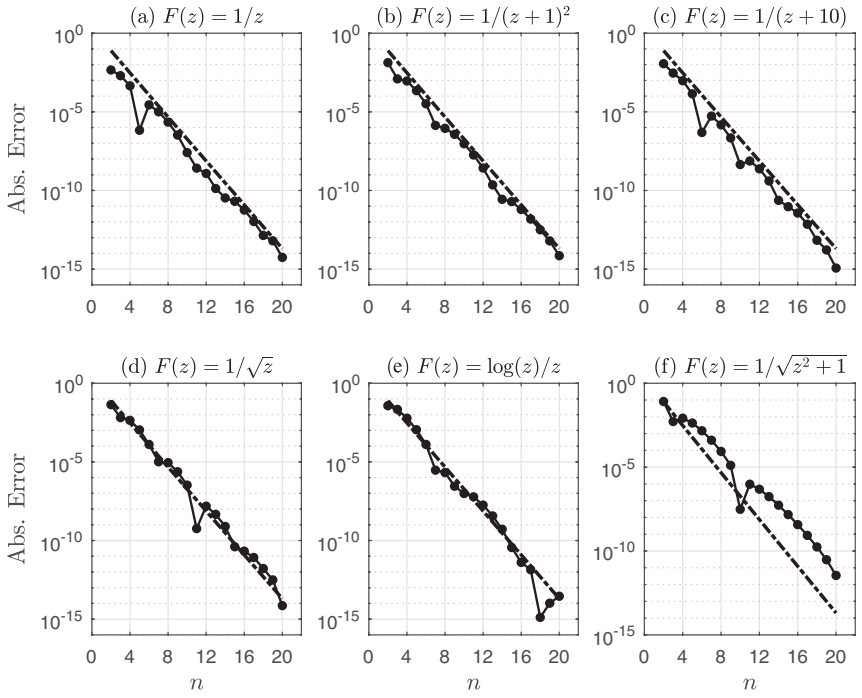

FIG. 4. *The dotted curves show absolute errors on a semilog scale in the method* (3.4) *when applied to various* $F(z)$ *to compute* $f(1)$. *The dash-dot line is the same in all cases and represents the proposed convergence rate as computed in section* 4 *and given in the fourth column of Table* 3. *Although the contour parameters (some of which are shown in Table* 3*) were tuned to be optimal for the transform in* (a), *the dash-dot curve is seen to be a good estimate of the actual convergence rate in all cases. The exception is transform* (f), *whose singularities consist of two branch points at* $z = \pm i$ *and a branch cut connecting them. Unlike the others, this transform is not analytic off the negative real axis, but the suggested contour parameters still produce good results even though the predicted convergence rate is inaccurate.*

In view of the fact that $z = \mu(1+i\phi)^2$, this means that in practice the only modification required is to replace the values of μ by μ/t in Table 3.

The consideration of $F(z) = 1/z$ is not a severe restriction either. It has been observed that contour parameters tuned optimally for this transform also work well for many others [4, 25, 27], particularly in those cases where the singularities of $F(z)$ are close to the negative real axis. Indeed, the model $F(z) = 1/z$ was originally used by Talbot to derive his cotangent contour [19].

The method (3.4), with parameters as listed in Table 3, was used to invert several transforms and absolute errors are shown in Figure 4. The examples include transforms with poles at various locations on the negative real axis, as well as transforms with branch cuts on the negative real axis. The results are about the same for all cases shown. To show that the proposed parameters are capable of producing good results even when singularities are off the negative real axis, we have included $F(z) = 1/\sqrt{z^2+1}$ among our test examples. The branch points (at $z = \pm i$) are fairly close to the real axis, so the results are still good. When moving these branch points further out on the imaginary axis, however, the accuracy deteriorates.

By fitting a least squares curve to the fourth column of that table (i.e., to the dash-dot curve in Figure 4), we conjecture that the achievable convergence rate for

this method is

$$R_n = O(e^{-1.6n}), \qquad n \to \infty. \tag{3.6}$$

This is the rate recorded in the last entry of Table 1 above, and suggests more rapid convergence than any of the other methods listed in that table. Numerical verification will follow in section 5.

The pure exponential convergence in (3.6) is an apparent contradiction to the root-exponential convergence rate mentioned below Theorem 1. The explanation is that singularities on the negative real axis of the the z-plane map to the upper half of the r-plane; in the lower half of the r-plane the integrand is free of any singularities. Hence, by adjusting parameters appropriately (as will be discussed in the next section), a region of analyticity that grows with n is obtained. The root-exponential convergence mentioned in section 2 is a consequence of a fixed region of analyticity.

In order to compute the optimal parameters μ and L listed in Table 3, one strategy would be to minimize numerically the actual error (since the true value is known). This is not so easy, because the objective function is highly oscillatory as can be seen in Figure 3. Neither is such a strategy very illuminating and it does not lend itself to further theoretical analysis. Instead, we offer the approach of the next section, which draws from the ideas of [4, 18, 27].

4. Parameter selection. In order to estimate the parameters μ and L in the method (3.4), we return to the graphical error representations of section 2. There we displayed contour curves of the modulus of the characteristic function only; now we also include the integrand, $F(z) = 1/z$. Such contour plots in the complex r-plane are shown in Figure 5 for the case $n = 16$. The key difference between these contour plots and the ones in Figure 2 is that there is no symmetry across the upper and lower half-planes. This stems from the fact that the integrand (3.3) is not a real function, so there is no up-down symmetry across the real r-axis. The integrand has very different properties in the upper and lower halves of the r-plane. As explained in the first paragraph of section 3, the interior of the parabolic contour, where the singularites are located, maps to the upper half of the r-plane and the exterior of the parabola, which is free of singularities, maps to the lower half-plane.

Following [18], we conjecture that the error is dominated by the pole and saddle point contributions seen in Figure 5. The only pole of $F(z) = 1/z$ is at the origin of the z-plane, which is mapped to $r_0 = i/L$ in the r-plane. The corresponding residue is $\varrho_0 = 2\,e^{-1/L^2}$, and the pole contribition to the error therefore amounts to

$$P_n(\mu, L) = |\varrho_0 \Phi_n(r_0)|. \tag{4.1}$$

Also visible in Figure 5 are saddle points in the lower half-plane, located where two level curves intersect. These points are defined by the roots of $\frac{d}{dr}\big(g(r)Q_n(r)/H_n(r)\big)$, and by logarithmic differentiation it follows that

$$\frac{g'(r)}{g(r)} + \frac{Q_n'(r)}{Q_n(r)} - \frac{H_n'(r)}{H_n(r)} = 0. \tag{4.2}$$

With $g(r)$ defined by (3.3), the first term is a rational function in r (parametrically dependent on μ and L). The second term involves a ratio of two Kummer U-functions (see [10, Table 1] and [5, eq. (13.3.22)]), and the third term is the ratio of two Hermite polynomials, $H_n(r)$ and $H'(r) = 2nH_{n-1}(r)$. Given n, μ, and L, it is therefore possible to solve (4.2) for the saddle point by any complex univariate root finding method,

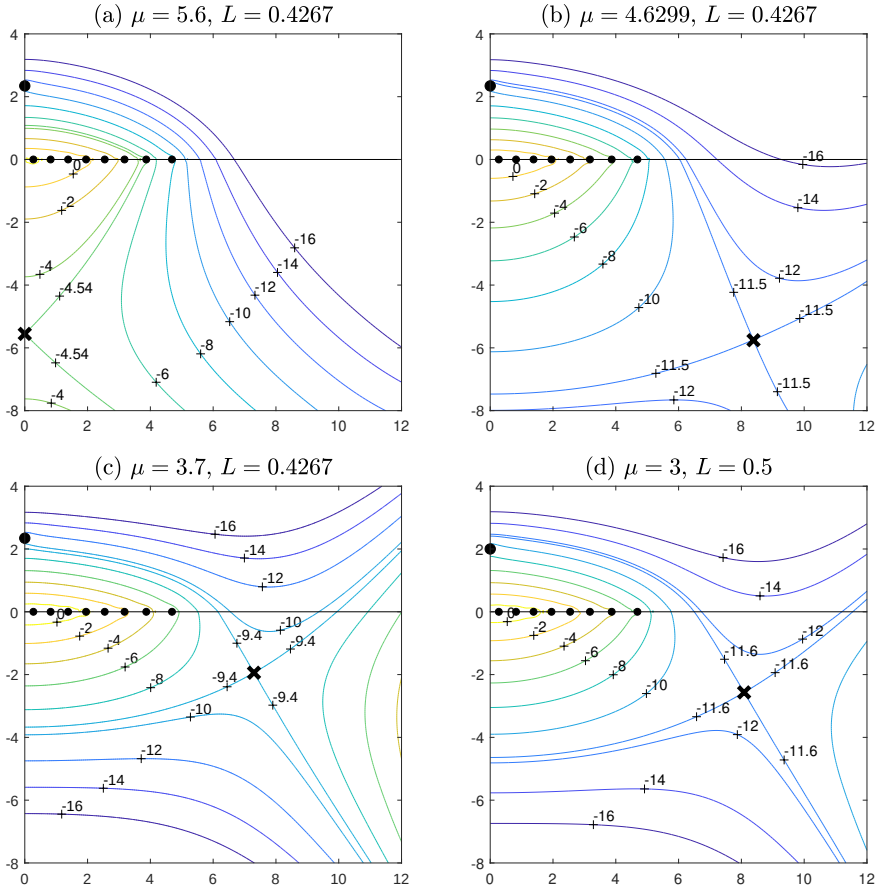

FIG. 5. *Level curves of* $\log_{10}\left(|\Phi_n(r)g(r)|/(2\pi)\right)$ *in the complex* r*-plane for the case* $F(z) = 1/z$, $n = 16$, *and four choices of* μ *and* L. *Because of symmetry, only the right half-plane is shown in each case. The pole* $r_0 = i/L$ *is represented by the bigger dot on the positive imaginary axis and the saddle point by the cross in the lower half-plane. The smaller dots on the real axis are the Gauss–Hermite nodes. In* (b), *the estimated pole and saddle point contributions are of equal magnitude, namely,* 1.2×10^{-11}. *We conjecture* (b) *to be the optimal configuration, and the numbers are recorded in the* $n = 16$ *row of Table* 3. *All other cases shown are suboptimal: in* (a) *and* (c) *the saddle point contributions are* 8.8×10^{-5} *and* 8.5×10^{-10}, *respectively, both of which dominate the pole contribution,* 1.2×10^{-11}. *In* (d) *the pole contribution is* 8.3×10^{-10}, *which dominates the saddle point contribution,* 5.8×10^{-12}. *In the text we connect these estimated error values to the actual errors shown in Figure* 3.

assuming a good initial guess of the saddle point can be provided. To obtain such a guess, plots like those in Figure 5 are helpful. In our implementation we used the MATLAB `fsolve` function as root finder, and the function `kummerU` in the MATLAB Symbolic Toolbox for computing the U-function.

If the saddle point is located at $r = r_1$, say, then we shall estimate the magnitude of its contribution by [6, 17]

$$S_n(\mu, L) = \frac{d}{2\pi}|g(r_1)\Phi_n(r_1)| \tag{4.3}$$

with d defined by

$$d = \sqrt{2\pi}\Big(\Big|\frac{d^2}{dr^2}\log\big(g(r)\Phi_n(r)\big)\Big|_{r=r_1}\Big)^{-1/2}. \tag{4.4}$$

Once r_1 has been computed from (4.2), the expressions (4.3) and (4.4) can also be evaluated in terms of rational functions, U-functions, and Hermite polynomials.

As explained in the caption of Figure 5, for some choices of μ and L the pole contribution (4.1) dominates the saddle point contribution (4.3), and vice versa. We therefore propose to compute μ and L by solving the minmax problem

$$\max\left\{\log\big(P_n(\mu,L)\big),\log\big(S_n(\mu,L)\big)\right\}=\min \tag{4.5}$$

for which we used the `fminsearch` function in MATLAB. This is how the numbers in Table 3 were computed. The number in the fourth column of that table is the achievable minimum value of this expression.

Following up on the last sentence in the caption of Figure 5, we now connect the contours shown in that figure to the actual errors shown in Figure 3. First, look at the sequence (a)→(b)→(c) in Figure 5, in which L is kept fixed while the value of μ is reduced. The corresponding actual errors can all be read off from Figure 3(c), from right to left, in the immediate neighborhood of the vertical dash-dot line.

When μ is too large, as in Figure 5(a), the saddle point is located on the negative imaginary r-axis and dominates the pole contribution. The total error is therefore suboptimal, and its value can read off from the error curve just to the right of the dash-dot line in Figure 3(c). By reducing μ, the saddle point moves down the imaginary r-axis, but before μ reaches its optimum value the saddle point splits into two, one moving into the fourth quadrant (shown in Figure 5) and a mirror image into the third (not shown). The saddle contributions continue to diminish until the optimum value of μ is reached, as shown in Figure 5(b) and confirmed by the dash-dot line in Figure 3(c). Reducing μ further results in an increasing value of the saddle contribution. (It thus appears that if the saddle point on the imaginary axis dominates, then the error curve in Figure 3 is smooth as in the right "arms" of those error curves. But if the dominating saddle point is located off the real axis the error curve is highly oscillatory, as in the arms on the left.) As for Figure 5(d), its counterpart is Figure 3(f), and in particular the small horizontal section of the error curve. The fact that this part of the curve is independent of μ is consistent with a dominating pole contribution. The contours and nodes for these four cases are shown in Figure 6.

The parameter estimation strategy outlined in this section is not a rigorous analysis. At this stage we can offer no proof that a contour of integration can be defined so as to include (resp., exclude) the quadrature nodes (resp., pole), to pass through the saddle points in the direction of steepest descent, and then to connect legitimately to the point at infinity as required by Theorem 1. The main complicating factor is the fact that the characteristic function defined in Theorem 1 is not an elementary function. Even its known approximations are nontrivial [10]. The corroboration of the proposed parameters lies in how well they work in practice, however, and this is supported by the results shown in Figure 4. Further evidence is provided in the next section.

We emphasize that we do not propose that the analysis of this section be repeated for every transform encountered. Our objective was to generate a set of practical parameters (listed in Table 3) that would work well for a large class of transforms, namely, those with singularties on or near the negative real axis. The analysis could perhaps be extended to other classes, such as those transforms with singularities on the imaginary axis, but the Butcher–Talbot class of methods does not work particularly well for such transforms and therefore we have not attempted this.

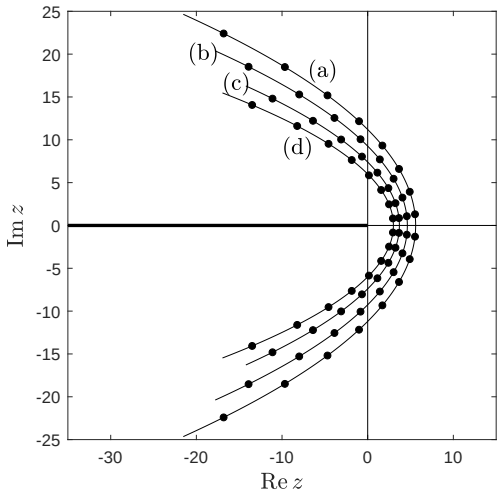

FIG. 6. *The four contours and node locations associated with the four cases displayed in Figure* 5. *In* (a)*, the contour extends too far into the right half-plane, so the error is dominated by the saddle point on the negative imaginary* r*-axis. Contour* (b) *is proposed as the optimal situation, with pole and saddle point contributions in balance. The nodes on contour* (c) *do not extend far enough into the left half-plane, so the error is dominated by endpoint effects, controlled by a saddle point relatively far out near the real* r*-axis. Contour* (d) *is too close to the origin, hence the error is dominated by the pole contribution.*

5. Comparisons. In this section we compare the method of this paper with two of the competitors listed in Table 1. The first of these is the method proposed in [23, 27], which is also implemented on the parabola (3.1) but integrates with the trapezoidal rule. The second is the modified Talbot method of [4, 23], which, to our knowledge, is the most rapidly convergent method in the Butcher–Talbot family for transforms in this class (analytic everywhere off the negative real axis). In all test problems the superior rate of convergence of the Gauss–Hermite method is evident.

Example 1. The following transform arises in a study of viscoplastic rods [8]:

$$F(z) = \frac{(100z-1)\sinh(\frac{1}{2}\sqrt{z})}{z(z\sinh(\sqrt{z})+\sqrt{z}\cosh(\sqrt{z}))}. \tag{5.1}$$

The apparent square root singularity can be removed by expanding the hyperbolic functions, and the only singularities are an infinity of poles on the negative real axis that stretches to $-\infty$. In Figure 7 the Gauss–Hermite method (2.1) with parameter values listed in Table 3 is compared to the trapezoidal rule implemented on a parabola (with parameters proposed in [23, 27]), and to the modified Talbot method (with parameters as in [4, 23]).

Example 2. Another transform from [8] arises in a viscous fluid mechanics problem:

$$F(z) = \frac{1}{z}\exp\left(-0.5\sqrt{\frac{z(1+z)}{1+0.4z}}\right). \tag{5.2}$$

This time there is a nonremovable square root singularity but with appropriate definition of the branch cut the transform becomes analytic off the negative real axis.

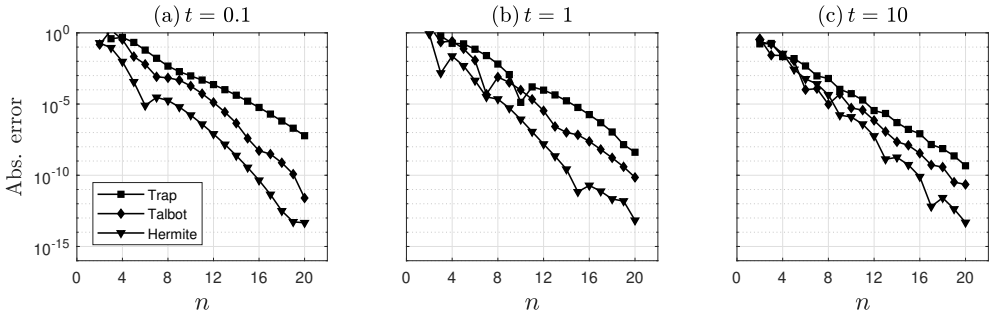

FIG. 7. *Absolute errors in the Gauss–Hermite method* (3.4)*, the improved Talbot method of* [4, 23]*, and the trapezoidal rule implemented on a parabola* [23, 27]*. The empirical results shown are consistent with the ranking of the three methods in order of convergence rate, as summarized in Table* 1*. The transform is* (5.1)*.*

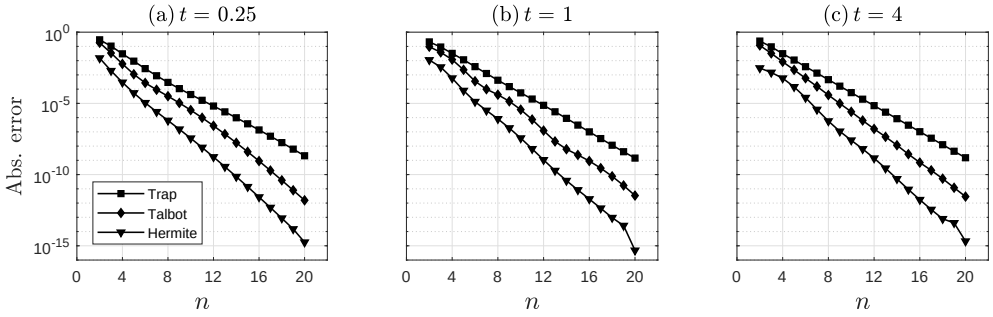

FIG. 8. *Same as Figure* 7*, but for the transform* (5.2)*.*

The results for this problem are shown in Figure 8, which again are consistent with the ranking in Table 1.

Examples 1 and 2 involve scalar transforms $F(z)$. When solving PDEs, vector-valued transforms are encountered such as the following one, and here it becomes clear why one would want to minimize the number of function evaluations.

Example 3. In [23], the heat equation,

$$u_t = 0.02\,\nabla^2 u, \quad (x, y) \in [-1, 1] \times [-1, 1], \tag{5.3}$$

was considered, subject to $u = 0$ on the boundary and $u(x, y, 0) = (1 - x^2)(1 - y^2)e^x$. When discretized in the usual manner with the five-point difference stencil on a regular grid with spacing $h = 0.02$, the PDE (5.3) is approximated by a semidiscrete system of ODEs $\boldsymbol{u}_t = A\boldsymbol{u}$, where $-A$ is a symmetric positive definite matrix of size 9801×9801. Laplace transformation yields

$$\boldsymbol{u}(t) = \frac{1}{2\pi i} \int_\Gamma e^{zt} F(z) \boldsymbol{u}_0 \, dz, \qquad F(z) = (zI - A)^{-1}, \tag{5.4}$$

where I is the identity matrix of the same size as A. The eigenvalues of A, which are located on the negative real axis, are now the singularities that should be considered when choosing Γ.

Equation (5.4) is nothing but a contour integral expression for $\boldsymbol{u}(t) = e^{At}\boldsymbol{u}_0$. In practice, the explicit computation of the matrix exponential by a general code such as

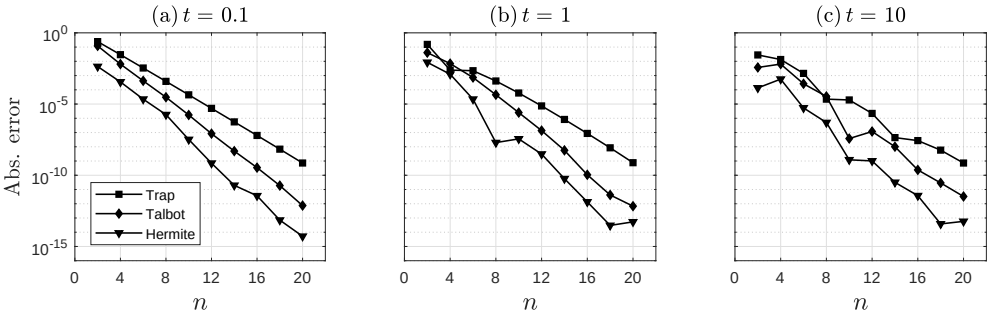

FIG. 9. *Same as Figure* 7*, but for the transform* (5.4)*. The displayed value is the error computed at* $(x, y) = (0, 0)$.

`expm` in MATLAB is not recommended, however, as the runtime can be excessive for large A. The advantage of the formulation (5.4) is that its implementation requires only the solution of a shifted linear system of the form $(zI - A)\boldsymbol{v} = \boldsymbol{u}_0$ at each quadrature node, which is far less expensive. Nevertheless, when A is large this can still be costly, which provided the original motivation for our quest of section 4, namely, a good set of parameters that will minimize the number of nodes. Of course, for this particular problem with constant coefficients the eigenvalue decomposition of the matrix is explicitly known, and this was used in computing the reference values for the results presented below.

The results for problem (5.4) are presented in Figure 9. The error at the origin of the domain is plotted as a function of n. Again the superior convergence rate of the Gauss–Hermite method is evident.

Example 4. Related to the previous example is the connection to rational approximation of the exponential as outlined in [23]. Consider

$$e^{\lambda} = \frac{1}{2\pi i} \int_{\Gamma} \frac{e^z}{z - \lambda}\, dz, \tag{5.5}$$

which can be associated with (5.4) by thinking of λ as an eigenvalue of A. When the integral is approximated by a quadrature sum, a rational approximation of type $(n - 1, n)$ to e^{λ} results; see [23]. This expression does not only hold for real λ, but can be extended into the complex plane. The accuracy in the complex plane can be an indication of the accuracy to be expected when problems such as the semidiscrete ODE of Example 3 are solved, in the cases where eigenvalues are located off the real axis, perhaps in the presence of dispersion or convection.

In Figure 10 we present the absolute errors in this rational approximation, for the three methods compared here. We refer to [23] for similar contour plots of the error, which also include the method of best approximation. That method provides the best approximation as its name indicates. But the gap between it and the quadrature methods has shrunk now with the addition of the Gauss–Hermite method to that group.

6. Conclusions. We have introduced a new member to the Butcher–Talbot family of methods for numerical Laplace transform inversion. Its novelty lies in the fact that it replaces the commonly used trapezoidal rule with Gauss–Hermite quadrature. By deforming the contour in the Bromwich integral to a parabolic curve, the decay in the integrand is of Gaussian type and therefore suitable for Gauss–Hermite

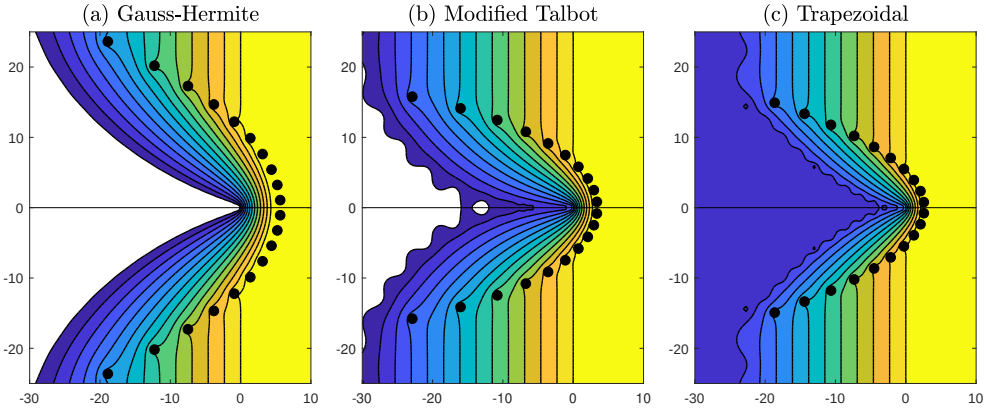

FIG. 10. *Absolute errors in the approximation of* (5.5) *in the complex* λ*-plane, for the case* $n = 20$. *The dots are the quadrature nodes on a contour (not displayed) with parameters tuned optimally as given in the present paper for the Gauss–Hermite method, and in* [4, 23, 27] *for the other two methods. The error levels are* 10^{-13}, $10^{-12}, \ldots, 10^{0}$ *from left to right, in each figure. (Where color is available, dark blue corresponds to* 10^{-13} *accuracy and yellow to* 10^{0}.*) All regions with errors less than* 10^{-13} *have been left blank. The* 10^{-13} *is roughly the error level predicted in Table* 3 *for the proposed set of parameters, and the Gauss–Hermite method delivers this accuracy on the negative real axis as well as in a fairly wide region surrounding it. The modified Talbot method reaches this accuracy on a much smaller region, and the trapezoidal rule on the parabola does not achieve it at all.*

quadrature. Using a heuristic, semigraphical approach, the error for the special case $F(z) = 1/z$ was estimated numerically and this was used to compute optimal contour parameters. In numerical tests the parameters thus computed were demonstrated to work well, not only for this special transform but for many transforms whose singularities are restricted to a narrow region of the negative real axis. In further numerical tests we have compared the new method with two others in the Butcher–Talbot family, namely, the trapezoidal rule applied on the parabola, and the modified Talbot method. In all tests considered the Gauss–Hermite method converged the fastest.

Two practical implementation matters have not been addressed in this paper. First, the numerical inversion of the Laplace transform can be unstable, and it is no different for the present method. Going beyond the 20-point rule (which is the last data set in Table 3) the error may start increasing again. (Evidence of this can be seen in certain frames of Figures 4, 7, and 9.) This is because of the factor e^{zt} in the Bromwich integral, which becomes large as the optimal contour moves too far into the right half-plane. This causes the terms in (3.4) to become huge and roundoff error contaminates the summation. As discussed in [4, 26], the error introduced by this can be modeled as $O(\epsilon\, e^{\mu t})$, where ϵ is machine epsilon. A practical code would monitor this quantity as n increases, and when it gets larger than the predicted error (2.2), it means the optimal accuracy for this rule has been achieved. In most practical situations accuracy requirements are not that stringent, however, and relatively small values of n can be used for which roundoff error is negligible.

Second, because the optimal μ parameter proposed in Table 3 depends on the value of t (recall the discussion below (3.2)), it means that $F(z)$ gets evaluated on different contours for different t. When $F(z)$ is inexpensive to evaluate, this is not an issue since one can use a separate optimal contour for each t. In more complicated problems one would like to minimize the number of evaluations of $F(z)$ by reusing values computed on the same contour and nodes and then some loss of accuracy is

inevitable. In [27] it was shown how to pick parameters that are close to optimal for an interval of t values, but this has yet to be done for the present method.

A final word about the parabola as the contour of choice: not only does this seem the right choice for Gauss–Hermite quadrature, it is also useful in the practical setting of solving PDEs. In Example 3 of section 5 the PDE was purely diffusive and the operator A normal. When convection is added, however, the operator becomes nonnormal. As was demonstrated in [12, 26], it is useful then to think of the spectrum as occupying a parabolic region surrounding the negative real axis. In this situation it is convenient to use a parabolic contour to enclose this region.

The two main rivals for integration on the real line have always been the trapezoidal rule and Gauss–Hermite quadrature. Historically, the trapezoidal rule was preferred because of the fact that its weights and nodes are explicitly available, as opposed to those of the Gauss–Hermite rule which have to be computed by iteration.[3] Because of recent software advances, such as the algorithm of [21] or the code in [7], this has become less of a deciding factor. The results presented here should support the Gauss–Hermite rule as a serious contender for inclusion in future software implementations for the numerical inversion of the Laplace transform.

Acknowledgments. Part of this work was completed while the author was visiting the Numerical Analysis Group in the Mathematical Sciences Institute at Oxford University in February 2017. Discussions with Nick Trefethen and his group contributed greatly to the results of this paper. Nick Hale read a draft and suggested many improvements.

REFERENCES

[1] W. Barrett, *Convergence properties of Gaussian quadrature formulae*, Comput. J., 3 (1961), pp. 272–277.

[2] J. C. Butcher, *On the numerical inversion of Laplace and Mellin transforms*, Conference on Data Processing and Automatic Computing Machines, Salisbury, Australia, Weapons Research Establishment, Salisbury, Australia, 1957, 117.

[3] A. M. Cohen, *Numerical Methods for Laplace Transform Inversion*, Numerical Methods and Algorithms 5, Springer, New York, 2007.

[4] B. Dingfelder and J. A. C. Weideman, *An improved Talbot method for numerical Laplace transform inversion*, Numer. Algorithms, 68 (2015), pp. 167–183.

[5] *NIST Digital Library of Mathematical Functions*, F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller and B. V. Saunders, eds., release 1.0.17, 2017, http://dlmf.nist.gov/.

[6] J. D. Donaldson and D. Elliott, *A unified approach to quadrature rules with asymptotic estimates of their remainders*, SIAM J. Numer. Anal., 9 (1972), pp. 573–602.

[7] T. A. Driscoll, N. Hale, and L. N. Trefethen, *Chebfun Guide*, Pafnuty Publications, Oxford, 2014, http://www.chebfun.org/docs/guide/.

[8] D. G. Duffy, *On the numerical inversion of Laplace transforms: Comparison of three new methods on characteristic problems from applications*, ACM Trans. Math. Software, 19 (1993), pp. 333–359.

[9] D. G. Duffy, *Transform Methods for Solving Partial Differential Equations*, 2nd ed., Chapman and Hall/CRC, Boca Raton, FL, 2004.

[10] D. Elliott and P. D. Tuan, *Asymptotic estimates of Fourier coefficients*, SIAM J. Math. Anal., 5 (1974), pp. 1–10.

[11] A. Gil, J. Segura, and N. M. Temme, *Asymptotic approximations to the nodes and weights of Gauss-Hermite and Gauss-Laguerre quadratures*, Stud. Appl. Math., 140 (2018), pp. 298–332.

[3] For the n–point Gauss–Hermite rule (as well as Gauss–Laguerre), there now exist explicit asymptotic formulas for the nodes and weights that are accurate to machine precision when $n \geq 100$; see [11].

[12] K. J. in 't Hout and J. A. C. Weideman, *A contour integral method for the Black–Scholes and Heston equations*, SIAM J. Sci. Comput., 33 (2011), pp. 763–785.
[13] M. López-Fernández and C. Palencia, *On the numerical inversion of the Laplace transform of certain holomorphic mappings*, Appl. Numer. Math., 51 (2004), pp. 289–303.
[14] M. López-Fernández, C. Palencia, and A. Schädle, *A spectral order method for inverting sectorial Laplace transforms*, SIAM J. Numer. Anal., 44 (2006), pp. 1332–1350.
[15] R. Piessens, *Gaussian quadrature formulas for the numerical integration of Bromwich's integral and the inversion of the Laplace transform*, J. Engrg. Math., 5 (1971), pp. 1–9, https://doi.org/10.1007/BF01535429.
[16] H. E. Salzer, *Orthogonal polynomials arising in the numerical evaluation of inverse Laplace transforms*, Math. Tables Aids Comput., 9 (1955), pp. 164–177.
[17] H. Takahasi and M. Mori, *Error estimation in the numerical integration of analytic functions*, Rep. Comput. Centre Univ. Tokyo, 3 (1970), pp. 41–108.
[18] H. Takahasi and M. Mori, *Estimation of errors in the numerical quadrature of analytic functions*, Appl. Anal., 1 (1971), pp. 201–229.
[19] A. Talbot, *The Accurate Numerical Inversion of Laplace Transforms*, Technical report, TR/61, Brunel University, London, 1976.
[20] A. Talbot, *The accurate numerical inversion of Laplace transforms*, J. Inst. Math. Appl., 23 (1979), pp. 97–120.
[21] A. Townsend, T. Trogdon, and S. Olver, *Fast computation of Gauss quadrature nodes and weights on the whole real line*, IMA J. Numer. Anal., 36 (2016), pp. 337–358.
[22] L. N. Trefethen and J. A. C. Weideman, *The exponentially convergent trapezoidal rule*, SIAM Rev., 56 (2014), pp. 385–458.
[23] L. N. Trefethen, J. A. C. Weideman, and T. Schmelzer, *Talbot quadratures and rational approximations*, BIT, 46 (2006), pp. 653–670.
[24] J. A. C. Weideman, Useful Matlab Files, http://appliedmaths.sun.ac.za/~weideman/research/matl.html.
[25] J. A. C. Weideman, *Optimizing Talbot's contours for the inversion of the Laplace transform*, SIAM J. Numer. Anal., 44 (2006), pp. 2342–2362.
[26] J. A. C. Weideman, *Improved contour integral methods for parabolic PDEs*, IMA J. Numer. Anal., 30 (2010), pp. 334–350.
[27] J. A. C. Weideman and L. N. Trefethen, *Parabolic and hyperbolic contours for computing the Bromwich integral*, Math. Comp., 76 (2007), pp. 1341–1356.