Numer Algor (2013) 63:339–355
DOI 10.1007/s11075-012-9625-3

ORIGINAL PAPER

Review of inverse Laplace transform algorithms for Laplace-space numerical approaches

Kristopher L. Kuhlman

Received: 20 April 2012 / Accepted: 11 July 2012 / Published online: 2 August 2012

Abstract A boundary element method (BEM) simulation is used to compare the efficiency of numerical inverse Laplace transform strategies, considering general requirements of Laplace-space numerical approaches. The two-dimensional BEM solution is used to solve the Laplace-transformed diffusion equation, producing a time-domain solution after a numerical Laplace transform inversion. Motivated by the needs of numerical methods posed in Laplace-transformed space, we compare five inverse Laplace transform algorithms and discuss implementation techniques to minimize the number of Laplace-space function evaluations. We investigate the ability to calculate a sequence of time domain values using the fewest Laplace-space model evaluations. We find Fourier-series based inversion algorithms work for common time behaviors, are the most robust with respect to free parameters, and allow for straightforward image function evaluation re-use across at least a log cycle of time.

Keywords Numerical Laplace transform inversion · Boundary element method · Diffusion · Helmholtz equation · Laplace-space numerical methods · Groundwater modeling

1 Introduction

Simulation methods that are posed in Laplace-transformed space, then numerically inverted back to the time domain (i.e., Laplace-space methods),

K. L. Kuhlman (✉)
Repository Performance Department, Sandia National Laboratories,
4100 National Parks Highway, Carlsbad, New Mexico, USA
e-mail: klkuhlm@sandia.gov

are a viable alternative to the more standard use of finite differences in time. We use the the two-dimensional boundary element method (BEM) as an example of this type of approach, to solve the Laplace-transformed diffusion equation (i.e., the Yukawa or modified Helmholtz equation). We investigate five numerical inverse Laplace transform methods and implementation approaches, namely the methods of Schapery [29], Stehfest [30], Talbot [33], Weeks [34], and de Hoog et al. [13]. Naively implemented Laplace-space simulations can be more computationally expensive than finite differences in time, but they have the advantage of allowing evaluation at any time, without evolving from an initial condition, and image function calculations are trivially parallelized across Laplace parameters [10]. When Laplace-space numerical models are used in parameter estimation, hundreds or thousands of forward simulations may be required—making forward model efficiency critical. Although parameter estimation may be done directly in Laplace space [6], choosing an efficient inversion strategy is important in most applications.

The Laplace transform has a long history of use to derive analytical solutions to diffusion and wave problems (e.g., see Duffy's citation list [15, pp. 191–220]). Often the analytical inverse transform is too difficult to find or evaluate in closed form. A researcher then resorts to approximate analytical methods (e.g., [16, 31]) or numerical inversion (e.g., [24, 25]). Numerical methods can similarly benefit from the Laplace transform, converting the time-dependence of a differential equation to parameter dependence. Laplace-space finite-element approaches have seen application to groundwater flow and solute transport (e.g., [26, 32]), and Laplace-space BEM has also been used in groundwater applications (e.g., [20, Section 10.3], [22, Section 10.1]). The Laplace transform analytic element method [19] is a transient extension of the analytic element method. These different Laplace-space approaches may have diverse spatial solution strategies, but they have a common requirement of effective Laplace transform numerical inversion algorithms. We couple a BEM model in the Laplace domain with a numerical Laplace transform inversion routine, but our conclusions should be valid for both gridded and mesh-free Laplace-space numerical methods. Any Laplace-space numerical approach begins with determination of optimal Laplace parameter values. Then each image function evaluation is computed from the simulation. The final step involves approximating the time-domain solution from the vector of image function values using the algorithm of choice.

Bellman et al. [7] was an early review book on numerical Laplace transform inversion for linear and non-linear problems, but without the benefit of the many algorithms that have since been developed. Davies and Martin [12] performed a thorough survey, assessing numerical Laplace transform inversion algorithm accuracy for techniques available in 1979, using simple functions for their benchmarks. Duffy [14] reviewed the numerical inversion characteristics for more pathological time behaviors using the Fourier series, Talbot, and Weeks inversion methods. The review book by Cohen [9] summarizes historical reviews and discusses commonly used inversion methods and their variations. More details and examples can be found in these reference regarding the convergence behavior of the five inversion algorithms discussed here.

While these published numerical inverse Laplace transform algorithm reviews are thorough and useful, they focus on computing a single time-domain solution as accurately as possible. These reviews did not try to minimize Laplace-space function evaluations, since their functions were simple closed-form expressions, not simulations. We investigate Laplace transform inversions algorithms that can compute a sequence of time domain values using the fewest Laplace-space model evaluations, a desirable property for use in Laplace-space numerical methods. Using numerical Laplace transform inversion in a simulation approach, rather than a time-marching method, allows the researcher to readily switch between fast and accurate by changing the number of approximation terms in the inversion.

In the next section we define the mathematical formulation of the governing equation and Laplace transform. In the third section we introduce the five inverse Laplace transform algorithms. In the final section we compare results using five different inversion algorithms to invert the BEM modified Helmholtz solution on the same domain with four different boundary conditions, leading to recommendations for Laplace-space numerical approaches.

2 Governing equation and Laplace transform

The BEM model generally simulates *substance* flow (e.g., energy or groundwater), which can be related to a potential ϕ (e.g., temperature or hydraulic head). The medium property α is diffusivity [L^2/T], the ratio of the conductance in the substance flux and potential gradient relation (e.g., Fourier's or Darcy's law) to the substance capacity per unit mass (e.g., heat capacity or storativity). The BEM (e.g., [8, 20, 22]) can be used to solve the diffusion equation

$$\nabla^2 \phi = \frac{1}{\alpha} \frac{\partial \phi}{\partial t}, \tag{1}$$

where α is real constant in space and time. We consider (1) in a domain subject to a combination of Dirichlet $\phi\left(\Gamma_{\mathrm{u}}(s), t\right) = f_{\mathrm{u}}(s, t)$ and Neumann $\hat{n} \cdot \nabla \phi \left(\Gamma_{\mathrm{q}}(s), t\right) = f_{\mathrm{q}}(s, t)$ boundary conditions along the perimeter of the 2D domain $\Gamma = \Gamma_{\mathrm{u}} \cup \Gamma_{\mathrm{q}}$, where $\hat{n}$ is the boundary unit normal, and s is a boundary arc-length parameter. Without loss of generality, we only consider homogeneous initial conditions.

The Laplace transform is

$$\mathcal{L}\{f(t)\} \equiv \bar{f}(p) = \int_0^{\infty} f(t) e^{-pt} \, \mathrm{d}t, \tag{2}$$

where p is the generally complex-valued Laplace parameter, and the overbar denotes a transformed variable. The transformed diffusion equation with zero initial conditions is the homogeneous Yukawa or modified Helmholtz equation,

$$\nabla^2\overline{\phi} - q^2\overline{\phi} = 0, \tag{3}$$

where $q^2 = p/\alpha$. Equation (3) arises in several groundwater applications, including transient, leaky, and linearized unsaturated flow [5]. The transformed boundary conditions are $\bar{\phi}\,(\Gamma_{\mathrm{u}}(s)) = f_{\mathrm{u}}(s)\,\bar{f}_{\mathrm{t}}(p)$ and $\hat{n} \cdot \nabla\bar{\phi}(\Gamma_{\mathrm{q}}(s)) = f_{\mathrm{q}}(s)\,\bar{f}_{\mathrm{t}}(p)$, where the temporal and spatial behaviors have been decomposed as in separation of variables. Arbitrary time behavior can be developed through convolution in t (Duhamel's theorem), which is multiplication of image functions in Laplace space. Here, $\bar{f}_{\mathrm{t}}(p)$ represents the Laplace transform of the time behavior applied to the boundary conditions. The Laplace transformation makes it possible to solve transient diffusion (a parabolic equation) using the BEM, which is well-suited for elliptical-type equations.

The inverse Laplace transform is defined as the Bromwich contour integral,

$$\mathcal{L}^{-1}\left\{\bar{f}(p)\right\} = f(t) = \frac{1}{2\pi i}\int_{\sigma - i\infty}^{\sigma + i\infty} \bar{f}(p)e^{pt}\,\mathrm{d}p, \tag{4}$$

where the abscissa of convergence $\sigma > 0$ is a real constant chosen to put the contour to the right of all singularities in $\bar{f}(p)$. In Laplace-space numerical approaches, (3) is solved by a suitable numerical method, therefore only samples of $\bar{f}(p)$ are available; this precludes an analytical inversion. Five numerical inverse Laplace transform algorithms are discussed in the following section.

3 Numerical inverse Laplace transform methods

Equation (4) is an integral equation for unknown $f(t)$ given $\bar{f}(p)$; its numerical solution is broadly split into two categories. Methods are either based on quadrature or functional expansion using analytically invertible basis functions. Davies [11, Chap. 19] relates most major classes of inverse Laplace transform methods using a unified theoretical foundation; we adopt a simplified form of their general notation. The Fourier series and Talbot methods are quadrature-based, directly approximating (4). Weeks' and Piessen's methods are $\bar{f}(p)$ expansions using complex-valued basis functions, while the Gaver-Stehfest and Schapery methods use real-valued functions to accomplish this.

The numerical inverse Laplace transform is in general an ill-posed problem (e.g., [2]). No single approach is optimal for all circumstances and temporal behaviors. This difficulty has lead to the diversity of viable numerical approaches in the literature (e.g., [9]).

3.1 Gaver-Stehfest method

The Post-Widder formula [2, 36] is an approximation to (4) that only requires $\bar{f}(p)$ for real p to represent (2) as an asymptotic Taylor series expansion.

The formula requires high-order analytic image function derivatives, and is impractical for numerical computation. Stehfest proposed a discrete version of the Post-Widder formula using finite differences and Salzer summation [30],

$$f(t,N)=\frac{\ln 2}{t}\sum_{k=1}^{N}V_k\bar{f}\left(k\frac{\ln 2}{t}\right). \tag{5}$$

The V_k coefficients only depend on the number of expansion terms, N (which must be even), which are

$$V_k=(-1)^{k+N/2}\sum_{j=\lfloor(k+1)/2\rfloor}^{\min(k,N/2)}\frac{j^{\frac{N}{2}}(2j)!}{(\frac{N}{2}-j)!\,j!\,(j-1)!\,(k-j)!\,(2j-k)!}. \tag{6}$$

These become very large and alternate in sign for increasing k. The sum (5) begins to suffer from cancellation for $N\geq$ the number of decimal digits of precision (e.g., double precision = 16). For $\bar{f}_{\mathrm{t}}(p)$ that are non-oscillatory and continuous, $N\leq 18$ is usually sufficient [30]. If programmed using arbitrary precision (e.g. Mathematica or a multi-precision library [4, 17]), the method can be made accurate for most cases [1]. Unfortunately, p is explicitly a function of t; for each new t, a new $\bar{f}(p)$ vector is needed. In Laplace-space numerical approaches, each vector element is constructed using a simulation, therefore this can be a large penalty.

The method is quite easy to program; the V_j can be computed once and saved as constants. This method has been popular due to its simplicity and adequacy for exponentially decaying $\bar{f}_{\mathrm{t}}(p)$.

3.2 Schapery's method

We can expand the deviation of $f(t)$ from steady-state f_{s} using exponential basis functions [29],

$$f(t,N)=f_{\mathrm{s}}+\sum_{i=1}^{N}a_ie^{-p_it}, \tag{7}$$

where a_i is a vector of unknown constants. Applying (2) to (7) gives

$$\bar{f}(p_j,N)=\frac{f_{\mathrm{s}}}{p_j}+\sum_{i=1}^{N}\frac{a_i}{p_i+p_j}\qquad j=1,2,\ldots,M. \tag{8}$$

The p_j are selected (a geometric series is recommended [22]) to cover the important fluctuations in $\bar{f}(p)$. After setting $p_i=p_j$ the a_i coefficients can be determined as the solution to $P_{ij}a_i = \left(\bar{f}(p_j) - f_{\mathrm{s}}/p_j\right)$. The symmetric matrix to decompose is

$$P_{ij} = \begin{bmatrix} (2p_1)^{-1} & (p_1+p_2)^{-1} & \ldots & (p_1+p_N)^{-1} \\ (p_2+p_1)^{-1} & (2p_2)^{-1} & \ldots & (p_2+p_N)^{-1} \\ \vdots & \vdots & \ddots & \vdots \\ (p_N+p_1)^{-1} & (p_N+p_2)^{-1} & \ldots & (2p_N)^{-1} \end{bmatrix},$$

which only depends on p_j; it can be decomposed independently of $\bar{f}(p)$ and f_{s}.

This method is not difficult to implement when existing matrix decomposition libraries are available, and only requires real computation. The method has been used for inverting BEM results [22], but has two main drawbacks. First, in its formulation above, it requires a steady-state solution, but (7) could be posed without f_{s}. Secondly, no theory is presented for optimally picking p_j; some trial and error is required [22].

3.3 Möbius transformation methods

We can use the Möbius transformation to conformally map the half-plane right of σ to the unit disc, making the Laplace domain more amenable to approximation using orthonormal polynomials (e.g., Chebyshev [28], [21, Section 28] or Laguerre [23, 34], [21, Section 30]). If σ was chosen properly, $\bar{f}(p)$ is guaranteed to be analytic inside the unit circle. The most-used inverse Laplace transform method from this class is Weeks' method, which uses a complex power series to expand $\bar{f}(p)$ inside the unit circle. Upon inverse Laplace transformation, the power series becomes a Laguerre polynomial series.

Weeks method is

$$f(t, N+1) = e^{(\kappa - b/2)t} \sum_{n=0}^{N} a_n \mathrm{L}_n(bt), \tag{9}$$

where $\mathrm{L}_n(z)$ is an n-order Laguerre polynomial and κ and b are free parameters. Weeks suggested $\kappa = \sigma + 1/t_{\max}$ and $b = N/t_{\max}$, where $t_{\max}$ is the maximum transformed time. The parameters b and κ are chosen to optimize convergence; some schemes are given [35] for finding optimum parameter values for a given $\bar{f}_{\mathrm{t}}(p)$, but search techniques require hundreds of $\bar{f}(p)$ evaluations. A more general form of (9) can also be used, which allows for more general asymptotic behavior of the image function [11, Section 19.5]. Weeks assumed $p\bar{f}(p)$ is analytic at infinity. The Laplace transform of (9) is known, but to make it easier to represent with polynomials, $\bar{f}(p)$ is mapped inside the unit circle via $z = (p - \kappa - 2b)/(p - \kappa + 2b)$. The coefficients a_n are determined by the midpoint rule,

$$a_n = \frac{1}{2M} \sum_{j=-M}^{M-1} \Psi\left[\exp\left(i\theta_{j-\frac{1}{2}}\right)\right] \exp\left(-in\theta_{j-\frac{1}{2}}\right) \tag{10}$$

where $\theta_j = j\pi/M$ and the conformally-mapped image function is

$$\Psi(z) = \frac{b}{1-z}\bar{f}\left(\kappa - \frac{b}{2} + \frac{b}{1-z}\right). \tag{11}$$

The argument of $\bar{f}(z)$ in (11) is the inverse mapping of $z \mapsto p$, it shows p does not functionally depend on t, but Weeks' rules-of-thumb for b and κ depend on $t_{\max}$.

There are other related methods which use different orthonormal polynomials to represent $\bar{f}(p)$ inside the unit circle. Chebyshev polynomials (known as Piessen's method [28]) can be used to expand the $\bar{f}(z)$ on the interval $[-1, 1]$. The Weeks method is moderately easy to program, requiring the use of Clenshaw recurrence formula to accurately implement Laguerre polynomials. Piessen's method is similar to implement, with a similar recurrence formula for Chebyshev polynomials.

3.4 Talbot method

We can deform the Bromwich contour into a parabola around the negative real axis if $\bar{f}(p)$ is analytic in the region between the Bromwich and the deformed Talbot contours [33]. Numerically, $\bar{f}(p)$ must not overflow as $p \to -\infty$ (e.g., in the BEM implementation, the Green's function is the second-kind modified Bessel function, which grows exponentially as $p \to -\infty$). Oscillatory $\bar{f}_{\mathrm{t}}(p)$ often have pairs of poles near the imaginary p axis; these poles must remain to the left of the deformed contour.

The Talbot method makes the Bromwich contour integral converge rapidly, since p becomes large and negative, making the e^{pt} term in (4) very small. A one-parameter "fixed" Talbot method was implemented [1]; the Bromwich contour is parametrized as $p(\theta) = r\theta(\cot(\theta) + i)$, where $0 \leq \theta \leq \pi$, and as a rule of thumb $r = 2M/(5t_{\max})$. The fixed Talbot method is

$$f(t, N) = \frac{r}{N}\left[\frac{\bar{f}(r)}{2}e^{rt} + \sum_{k=1}^{N-1} \Re\left\{e^{tp(\theta_k)}\bar{f}\left[p(\theta_k)\right]\left[1 + i\zeta(\theta_k)\right]\right\}\right], \tag{12}$$

where $\zeta(\theta_k) = \theta_k + [\theta_k \cot(\theta_k) - 1]\cot(\theta_k)$ and $\theta_k = k\pi/N$ [1]. Although $\bar{f}(p)$ doesn't depend on t, the free parameter r depends on $t_{\max}$.

Step change $\bar{f}_{\mathrm{t}}(p)$ for non-zero time become very large as $p \to -\infty$, since $\mathcal{L}[H(t-\tau)] = e^{-\tau p}/p$, where $H(t-\tau)$ is the Heaviside step function centered on time τ. This can lead to precision loss, and stability or convergence issues with the underlying numerical model, although Mathematica's arbitrary precision capabilities have been used to get around this problem [1].

The fixed Talbot method is very simple to program; Abate and Valkó provide a ten-line Mathematica implementation [1].

3.5 Fourier series method

We can manipulate (4) into a Fourier transform; first it is expanded into real and imaginary parts ($p = \gamma + i\omega$),

$$f(t) = \frac{e^{\gamma t}}{2\pi i} \int_{-\infty}^{\infty} [\cos(\omega t) + i \sin(\omega t)] \left\{ \Re\left[\bar{f}(p)\right] + i\Im\left[\bar{f}(p)\right] \right\} i \, \mathrm{d}\omega.$$

Multiplying out the terms, keeping only the real part due to $f(t)$ symmetry, and halving the integration range due to symmetry again, leaves

$$f(t) = \frac{e^{\gamma t}}{\pi} \int_{0}^{\infty} \Re\left[\bar{f}(p)\right] \cos(\omega t) - \Im\left[\bar{f}(p)\right] \sin(\omega t) \, \mathrm{d}\omega. \tag{13}$$

When $f(t)$ is real, (13) can be represented using the complex form or just its real or imaginary parts. Although these three representations are equivalent, when evaluating (13) with the trapezoid rule, the full complex form gives the smallest discretization error [11]. The trapezoid rule approximation to (13) is essentially a discrete Fourier transform,

$$f(t, 2N+1) = \frac{e^{\gamma t}}{T} {\sum_{k=0}^{2N}}' \Re\left[\bar{f}\left(\gamma_0 + \frac{i\pi k}{T}\right) \exp\left(\frac{i\pi t}{T}\right)\right], \tag{14}$$

where $\gamma_0 = \sigma - \log(\epsilon)/T$, ϵ is the desired relative accuracy (typically 10^{-8} in double precision), T is a scaling parameter (often $2t_{\max}$), and the prime indicates the $k = 0$ summation term is halved. The p in (14) do not depend on t, but the free parameter T depends on $t_{\max}$.

The non-accelerated Fourier series inverse algorithm (14) is almost useless because it requires thousands of $\bar{f}(p)$ evaluations [3, Section 9.8]. Practical approaches accelerate the convergence of the sum in (14). Although this is sometimes called a fast-Fourier transform (FFT) method (e.g., [9, Chap 4.4]), rarely do the number of $\bar{f}(p)$ evaluations in an accelerated approach justify an FFT approach. The method implemented uses non-linear double acceleration with Padé approximation and an analytic expression for the remainder in the series [13]. Although there are several other ways to accelerate the Fourier series approach [9], this method is popular and straightforward. Non-linear acceleration techniques drastically reduce the required number of function evaluations, but can lead to numerical dispersion [18, 26]. For diffusion, dispersion associated with non-linear acceleration is not noticeable. Schapery's, Talbot's, and Weeks' methods are not accelerated in a non-linear manner, and therefore may lead to less numerical dispersion, which may be more important in wave systems with sharp fronts.

The creation of the Padé approximation [13] is relatively straightforward in programming languages that facilitate matrix manipulations (e.g., modern Fortran, Matlab, or NumPy [27]). There is no dependence on matrix decomposition routines.

Table 1 Algorithmic summary

Method	Limitations on $\bar{f}(p)$ and $f(t)$	$p(t)$?	$p(t_{\max})$?	p
Stehfest	No oscillations, no discontinuities in $f(t)$	Yes	No	Real
Schapery	Smoothly varying $f(t)$, f_s exists	No	No	Real
Weeks	None	No	Yes	Complex
Fixed Talbot	No high-frequency $f(t)$, $\bar{f}(p \to -\infty)$ exists	No	Yes	Complex
Fourier series	None	No	Yes	Complex

3.6 Algorithm properties summary

Table 1 summarizes aspects of the five inverse methods. The third column indicates whether p is explicitly a function of t, the fourth column indicates if the rules-of-thumb used for the optimum parameters depend on $t_{\max}$, and the fifth column indicates whether the transform requires complex p and $\bar{f}(p)$.

For all methods considered here, computational effort to compute $f(t)$ from the vector of $\bar{f}(p)$ values was insignificant compared to the effort required to compute the BEM solution used to fill the $\bar{f}(p)$ vector. This suggests a more complicated method, which allows re-use of $\bar{f}(p)$ across more values of t and converges in less evaluations of $\bar{f}(p)$, would be efficient for Laplace-domain numerical methods. If existing libraries or simulations only support real arguments, then the Stehfest, Schapery, or Piessen's methods must be used. Complex p methods will pay a slight penalty in computational overhead compared to real-only p routines. Computing with arbitrary or higher-than-double precision (e.g., [1]) will incur a much larger penalty than the change from real to complex double precision. Generally, complex p methods have better convergence properties than real-only methods. Expansion of $\bar{f}(p)$ along the real p axis is separation of non-orthogonal exponentials, while expansion along the imaginary p axis is separation of oscillatory functions [21, Section 29].

4 Numerical comparison

Four test problems were solved using the BEM for values of p required by each algorithm's rules of thumb. The test problem domain is a 3×2 rectangle, with homogeneous initial conditions and specified potential at two ends $\bar{\phi}(x=0) = -2\bar{f}_{\mathrm{t}}(p)$, and $\bar{\phi}(x=3) = 2\bar{f}_{\mathrm{t}}(p)$, and zero normal flux along the other sides $\partial\bar{\phi}/\partial y(y=\{0,2\}) = 0$. All plots show the solution computed at a point closer to the $x=0$ boundary ($x=1/3$), midway between the insulated boundaries ($y=1$).

The first problem computes $\bar{f}(p)$ using the optimum p at each t (like most inverse Laplace transform surveys), according to the rules-of-thumb for each method. While this is most accurate, it is very inefficient—especially when many values of t are required. In the following sections, all methods except Stehfest use the same $\bar{f}(p)$ to invert all t. A method's sensitivity to non-optimal free parameters is important in practical use for Laplace-space numerical

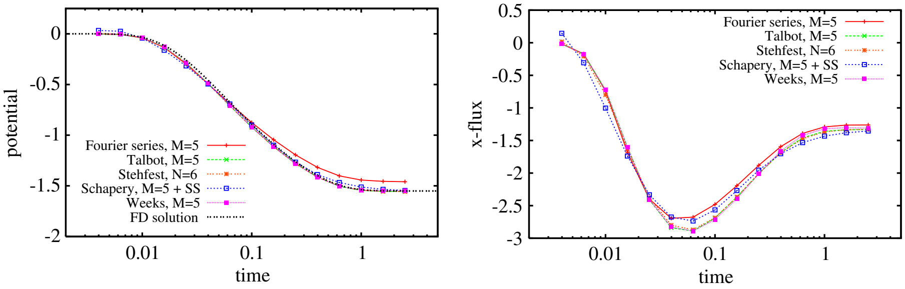

Fig. 1 Plots of potential and flux through time with five methods for $\bar{f}_t(p) = 1/p$, using optimum p at each t. $15 \times 5 = 75$ total $\bar{f}(p)$ evaluations are used by each method

approaches. By inverting more than one time with the same set of Laplace-space function evaluations, large gains in efficiency can be made. The t range used in the plots spans three orders of magnitude; it was chosen to show the evolution of potential and substance flux from initial conditions to steady state.

4.1 Steady boundary conditions, optimum p

The first problem has steady-state boundary conditions. The transient behavior is solely due to evolution from the zero initial condition, $\bar{f}_t(p) = \mathcal{L}[H(t)] = 1/p$; $\bar{f}(p)$ has a pole at the origin. All methods performed equally well with this simple test problem, although the Fourier series method deviates from the finite difference solutions at larger time. Figure 1 shows the inverted potential and flux using as few evaluations of $\bar{f}(p)$ possible, without major deviations from the finite difference benchmark solution. Some trial and error was needed to use the Schapery method (i.e., further optimization may be possible).

As shown in Fig. 2, all the methods performed very well for nine $\bar{f}(p)$ terms per t but at least 135 $\bar{f}(p)$ evaluations are needed total for each method. Schapery's method does the worst in this case, but this may be improved with further optimization of p_j values. The finite-difference approach took at

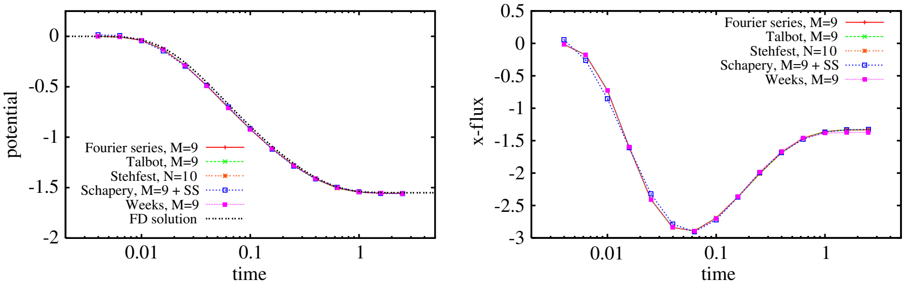

Fig. 2 Plots of potential and flux through time with five methods for $\bar{f}_t(p) = 1/p$, using optimum p at each t. $15 \times 9 = 135$ total $\bar{f}(p)$ evaluations are used by each method. Fourier series, Talbot, Stehfest, and Weeks curves are nearly coincident

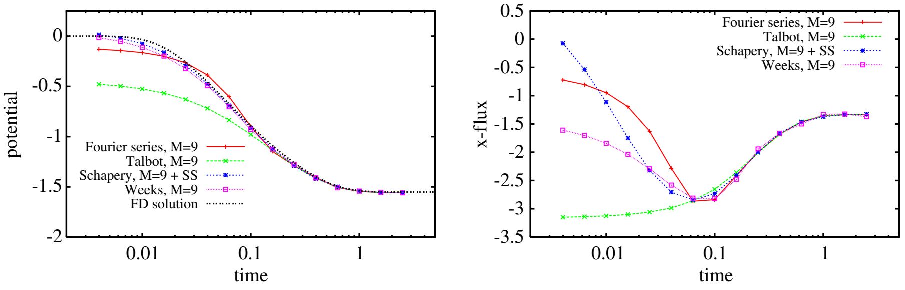

Fig. 3 Plots of potential and flux through time with four methods for $\bar{f}_{\mathrm{t}}(p) = 1/p$, same p used across all t. Nine total $\bar{f}(p)$ evaluations are used by each method

least an order of magnitude less computational effort for the given accuracy. Making Laplace-space numerical methods useful alternatives to traditional time-marching approaches, requires improvements to this inefficiency.

4.2 Steady boundary conditions, same p

All methods had more difficulty obtaining accurate results for a wide t range using only one vector of $\bar{f}(p)$ (no Stehfest method, since p explicitly depends on t). Only the last log-cycle of times is inverted accurately when using nine $\bar{f}(p)$ (Fig. 3). All the methods—except possibly Schapery's—have a more difficult time with the flux at early time (especially the fixed Talbot method). The apparent success of Schapery's method can be attributed to the expansion of the deviation from steady-state, which in this case decays exponentially with time.

Figure 4 shows that when increasing to 51 $\bar{f}(p)$ terms, most convergence problems disappear, except at small times. Grouping t values by log-cycles and inverting them together using the same $\bar{f}(p)$ is more economical than using the optimal p for each t and is still relatively accurate. The results shown in Fig. 4 are nearly as accurate as those shown in Fig. 2, but required 1/3 the $\bar{f}(p)$ model evaluations.

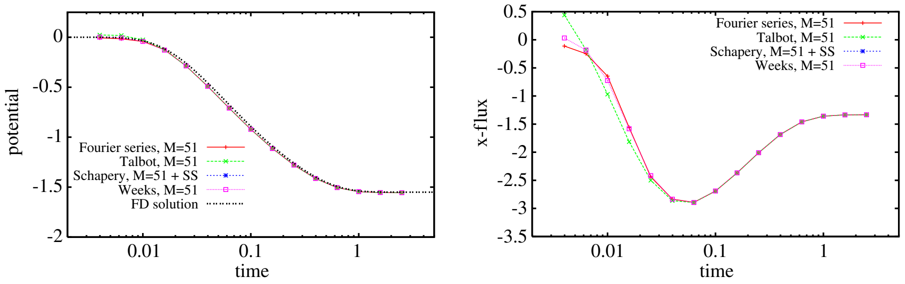

Fig. 4 Plots of potential and flux through time with four methods for $\bar{f}_{\mathrm{t}}(p) = 1/p$, same p used across all t. Fifty-one total $\bar{f}(p)$ evaluations are used by each method. Schapery and Weeks curves are nearly coincident

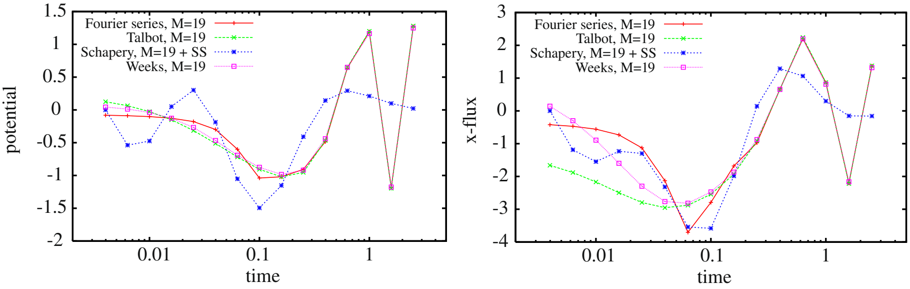

Fig. 5 Plots of potential and flux through time with four methods for $\bar{f}_{\mathrm{t}}(p) = p/(p^2 + 16)$, same p used across all t. Nineteen total $\bar{f}(p)$ evaluations are used by each method

4.3 Sinusoidal boundary conditions, same p

This problem uses temporally sinusoidal boundary conditions, $\bar{f}_{\mathrm{t}}(p) = \mathcal{L}(\cos 4t) = \frac{p}{p^2+16}$. This boundary condition violates some assumptions of the inverse transform algorithms (i.e., no steady-state solution, oscillatory in time), but the behavior is still relatively simple and smooth, with singularities at $p = \pm 4i$.

Figure 5 shows the Schapery method fails since there is no f_{s}, but the other methods do well for 19 terms across one t log cycle. Figure 6 shows all methods besides Schapery do well for 51 terms, across at least two t log cycles. A modified version of (8), substituting $\frac{p}{p^2+16}$ for f_{s}/p_j could extend Schapery's approach to this case, but this solution was not considered here because of its problem specificity.

4.4 Step-change boundary condition for $\tau > 0$, same p

Finally, the same domain was simulated but with step-change boundary conditions at $\tau = 0.08$, or $\bar{f}_{\mathrm{t}}(p) = \mathcal{L}(H(t - 0.08)) = e^{-0.08p}/p$, with singularities at the origin and $p = -\infty$. This function, and those derived from it (e.g., a pulse or a square wave) are difficult functions to invert accurately, because $f(t)$ is discontinuous. Figures 7 and 8 show the Talbot method does not work for

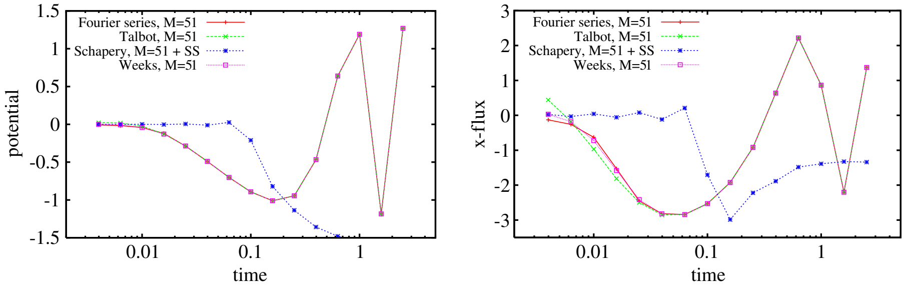

Fig. 6 Plots of potential and flux through time with four methods for $\bar{f}_{\mathrm{t}}(p) = p/(p^2 + 16)$, same p used across all t. Fifty-one total $\bar{f}(p)$ evaluations are used by each method. Fourier series, Talbot, and Weeks curves are nearly coincident for potential

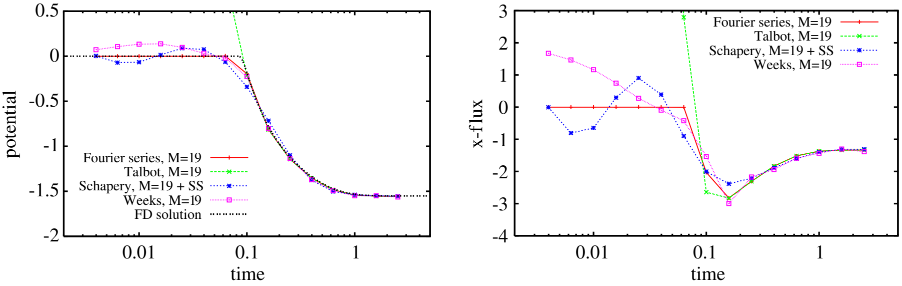

Fig. 7 Plots of potential and flux through time with four methods for $\bar{f}_t(p) = \exp(-0.08p)/p$, same p used across all t. Nineteen total $\bar{f}(p)$ evaluations are used by each method. Weeks' solution is undefined for $t < 0.08$

$t < \tau$ in double precision, since $\bar{f}_t(p)$ grows exponentially as $p \to -\infty$. The Weeks and Schapery methods do worse than the Fourier series approach (even with $N = 51$), but their parameters can be optimized further to improve these methods.

Although this step boundary condition could be implemented more accurately by shifting the results from the first example by $t = 0.08$, other step-derived time behaviors including a pulse or square wave cannot be simplified in this way.

4.5 Numerical results summary

Table 2 summarizes results from numerical testing with these four simple boundary condition time behaviors. The second column indicates what limit there is on the number of terms in the approximation and therefore the accuracy of the method. The size of p required by the Weeks and Fourier series methods grow much slower than those required by the fixed Talbot method. The third column indicates what parameters are needed to be tuned by the implementer to increase convergence of the method, and whether a good choice is critical to the success of the method—an automatic method should

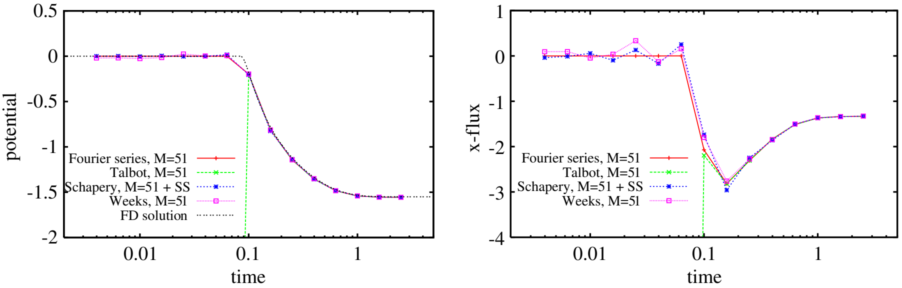

Fig. 8 Plots of potential and flux through time with four methods for $\bar{f}_t(p) = \exp(-0.08p)/p$, same p used across all t. Fifty-one total $\bar{f}(p)$ evaluations are used by each method. Weeks' solution is undefined for $t < 0.08$

Table 2 Numerical summary

Method	Number of terms	Free parameters	Implementation
Stehfest	$N \leq$ decimal precision	None	Easiest
Schapery	Depends on choice of p_{j}	p_{j} via trial & error	Moderate
Weeks	$p \rightarrow i\infty$ slowly as N grows	κ & b (very sensitive to b)	Moderate
Fixed Talbot	$p \rightarrow -\infty$ quickly as N grows	$r = \dfrac{2M}{5t_{\text{max}}}$ (automatic)	Easy
Fourier series	$p \rightarrow i\infty$ slowly as N grows	$T = 2t_{\text{max}}$ (automatic)	Most difficult

not require searching or optimizing parameters to obtain a robust solution. We define robustness as the ability of a solution to remain useful, even when not at optimality. We prioritize a solution that is good enough and stable over one that is excellent but catastrophically sensitive to parameter choice. The fourth column indicates the ease of implementation in modern Fortran, Matlab, or NumPy. The methods could also be implemented in a variable-precision environment like Mathematica or mpmath [17], but this would further require the BEM model be implemented in such an environment.

The modest success of the Schapery method is a bit surprising, given its simplicity and use of real p. The results of the previous section were the product of many iterations of trial and error, this effort was not included in the implementation effort. A better rule or parametrization of p_j might make this method more widely useful.

The sensitivity of Weeks' method to the parameter choices was also surprising; similarly, the method could have been improved after some optimization [35], but Weeks' rule of thumb was used for the parameters. One of the noted advantages of Weeks' method is the need to only compute optimal p once, then any time can accurately be inverted [14, 18, 35]. When using the simple rules-of-thumb for the the free parameters, this was not found to be the case. The generalized form of Weeks' method can include information about behavior of $\bar{f}(p) \rightarrow \infty$ (related to behavior as $t \rightarrow 0$), but this requires problem-specific knowledge.

The Fourier series method is more robust with respect to non-optimal p values, even though Duffy [14] cites this as a reason to use Weeks' method over the Fourier series approach.

5 Conclusions

Laplace-space numerical approaches to solve the diffusion equation have several viable alternative inverse Laplace transform algorithms to choose from. Historically, most Laplace-space solutions to the diffusion equation have used real-only methods (i.e., Gaver-Stehfest or Schapery). More robust methods require complex arithmetic and $\bar{f}(p)$ evaluations, but have the benefits of:

1. handling a broader class of time behaviors (Fourier series method);
2. still being relatively simple to implement (fixed Talbot method);

3. only utilizing double-precision complex data types, which are handled natively by modern Fortran, Matlab, or NumPy, and by common extensions in C++ (Fourier series and Weeks' methods).

Several practical recommendations are made regarding Laplace-space numerical modeling:

1. If many observations are needed across several time log cycles, large gains in efficiency can come from inverting groups of times with a single $\bar{f}(p)$ vector (e.g., grouped by log cycle). This complicates the implementation, but leads to much faster simulations.
2. If calculating $\bar{f}(p)$ is very expensive, and some numerical dispersion is allowable (not solving a wave problem with sharp fronts), then the Fourier series method approach is most economical, and is automatic and robust regarding free-parameter selection.
3. If only a single $\bar{f}_{\mathrm{t}}(p)$ is needed, then it may be worthwhile to optimize free parameters needed by Weeks' or Piessen's methods, or incorporate information about asymptotic $\bar{f}(p)$ behavior. Selection of optimum b is far from automatic, and the Weeks method is not robust for non-optimal free parameters values.
4. If implementation time is a large factor, the fixed Talbot is quite simple to code and was automatic (no need to select optimum parameters). The fixed Talbot may not work for non-zero step-time behavior without extended precision.
5. If complex-valued function evaluations are not feasible (e.g., only real matrix or special function libraries are available), the Schapery or Piessen's methods are capable of using the same p values to invert different times, which the Gaver-Stehfest method cannot.
6. When appropriate, the strategy used by Schapery to expand the deviation from a reference state could be incorporated as a strategy to improve other algorithms.

Acknowledgements The author thanks Professors Cho Lik Chan and Barry Ganapol from the Aerospace and Mechanical Engineering Department at the University of Arizona for initial inspiration and direction on this manuscript.

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

1. Abate, J., Valkó, P.: Multi-precision Laplace transform inversion. Int. J. Numer. Methods Eng. **60**, 979–993 (2004). doi:10.1002/nme.995
2. Al-Shuaibi, A.: Inversion of the Laplace transform via Post-Widder formula. Integral Transforms Spec. Funct. **11**(3), 225–232 (2001). doi:10.1080/10652460108819314
3. Antia, H.: Numerical Methods for Scientists and Engineers, 2nd edn. Birkhäuser, Cambridge, MA (2002)

4. Bailey, D.H., Hida, Y., Li, X.S., Thompson, B.: ARPREC: An Arbitrary Precision Computation Package. Tech. Rep. LBNL-53651, Lawrence Berkley National Lab (2002)
5. Bakker, M., Kuhlman, K.: Computational issues and applications of line-elements to model subsurface flow governed by the modified Helmholtz equation. Adv. Water Resour. (2011). doi:10.1016/j.advwatres.2011.02.008
6. Barnhart, K.S., Illangasekare, T.H.: Automatic transport model data assimilation in Laplace space. Water Resour. Res. **48**, W01510, 12 (2012). doi:10.1029/2011WR010955
7. Bellman, R., Kalaba, R.E., Lockett, J.A.: Numerical Inversion of the Laplace Transform: Applications to Biology, Economics, Engineering, and Physics. Elsevier, Amsterdam, The Netherlands; New York (1966)
8. Brebbia, C., Telles, J., Wrobel, L.: Boundary Element Techniques: Theory and Practice in Engineering. Springer, Berlin Heidelberg New York (1984)
9. Cohen, A.M.: Numerical Methods for Laplace Transform Inversion. Springer, Berlin Heidelberg New York (2007)
10. Davies, A., Crann, D.: Parallel Laplace transform methods for boundary element solutions to diffusion-type problems. J. Boundary Elements **BETEQ 2001**(1), 231–238 (2002)
11. Davies, B.: Integral Transforms and their Applications, 3rd edn. Springer, Berlin Heidelberg New York (2005)
12. Davies, B., Martin, B.: Numerical inversion of the Laplace transform: a survey and comparison of methods. J. Comput. Phys. **33**, 1–32 (1979). doi:10.1016/0021-9991(79)90025-1
13. de Hoog, F., Knight, J., Stokes, A.: An improved method for numerical inversion of Laplace transforms. SIAM J. Sci. Statist. Comput. **3**, 357–366 (1982). doi:10.1137/0903022
14. Duffy, D.G.: On the numerical inversion of Laplace transforms: comparison of three new methods on characteristic problems from applications. ACM Trans. Math. Softw. **19**(3), 333–359 (1993). doi:10.1145/155743.155788
15. Duffy, D.G.: Transform Methods for Solving Partial Differential Equations. CRC Press, Boca Raton, FL (2004)
16. Hantush, M.: Modification of the theory of leaky aquifers. J. Geophys. Res. **65**(11), 3713–3725 (1960). doi:10.1029/JZ065i011p03713
17. Johansson, F.: mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 0.17). http://code.google.com/p/mpmath/ (2011). Accessed Dec 2011
18. Kano, P.O., Brio, M., Moloney, J.V.: Application of the Weeks method for the numerical inversion of the Laplace transform to the matrix exponential. Communications in Mathematical Sciences **3**(3), 335–372 (2005)
19. Kuhlman, K.L., Neuman, S.P.: Laplace-transform analytic-element method for transient porous-media flow. J. Eng. Math. **64**(2), 113–130 (2009). doi:10.1007/s10665-008-9251-1
20. Kythe, P.K.: An Introduction to Boundary Element Methods. CRC Press, Boca Raton, FL (1995)
21. Lanczos, C.: Applied Analysis. Dover, New York (1988)
22. Liggett, J.A., Liu, P.L.: The Boundary Integral Equation Method for Porous Media Flow. Unwin, London, UK (1982)
23. Lyness, J., Giunta, G.: A modification of the Weeks method for numerical inversion of the Laplace transform. Math. Comput. **47**(175), 313–322 (1986). doi:10.2307/2008097
24. Malama, B., Kuhlman, K., Revil, A.: Theory of transient streaming potentials associated with axial-symmetric flow in unconfined aquifers. Geophys. J. Int. **179**(2), 990–1003 (2009). doi:10.1111/j.1365-246X.2009.04336.x
25. Mishra, P.K., Neuman, S.P.: Improved forward and inverse analyses of saturated-unsaturated flow toward a well in a compressible unconfined aquifer. Water Resour. Res. **46**(7), W07508, 16 (2010). doi:10.1029/2009WR008899
26. Morales-Casique, E., Neuman, S.P.: Laplace-transform finite element solution of nonlocal and localized stochastic moment equations of transport. Commun. Comput. Phys. **6**(1), 131–161 (2009)
27. Oliphant, T.E.: Python for scientific computing. Comput. Sci. Eng. **9**(3), 10–20 (2007). doi:10.1109/MCSE.2007.58
28. Piessens, R.: A new numerical method for the inversion of the Laplace transform. J. Inst. Math. Appl. **10**, 185–192 (1972). doi:10.1093/imamat/10.2.185

29. Schapery, R.: Approximate methods of transform inversion for visco-elastic stress analysis. In: Proceedings of the Fourth US National Congress on Applied Mechanics, vol. 2, pp. 1075–1085 (1962)
30. Stehfest, H.: Algorithm 368: numerical inversion of Laplace transforms. Commun. ACM **13**(1), 47–49 (1970). doi:10.1145/361953.361969
31. Sternberg, Y.: Flow to wells in the presence of radial discontinuities. Ground Water **7**(6), 17–20 (1969). doi:10.1111/j.1745-6584.1969.tb01666.x
32. Sudicky, E., McLaren, R.: The Laplace transform Galerkin technique for large-scale simulation of mass transport in discretely fractured porous formations. Water Resour. Res. **28**(2), 499–514 (1992). doi:10.1029/91WR02560
33. Talbot, A.: The accurate numerical inversion of Laplace transforms. IMA J. Appl. Math. **23**(1), 97 (1979). doi:10.1093/imamat/23.1.97
34. Weeks, W.: Numerical inversion of Laplace transforms using Laguerre functions. J. ACM **13**(3), 419–429 (1966). doi:10.1145/321341.321351
35. Weideman, J.: Algorithms for parameter selection in the Weeks method for inverting the Laplace transform. SIAM J. Sci. Comput. **21**(1), 111–128 (1999). doi:10.1137/S1064827596312432
36. Widder, D.: The Laplace Transform. Princeton University Press, Princeton, NJ (1941)