Numerical inverse Laplace transformation using concentrated matrix exponential distributions[☆]

Gábor Horváth[a], Illés Horváth[b], Salah Al-Deen Almousa[a], Miklós Telek[a,*]

[a] *Department of Networked Systems and Services, Technical University of Budapest, Budapest, Hungary*
[b] *MTA-BME Information Systems Research Group, Budapest, Hungary*

Abstract

This paper investigates the performance of the numerical inverse Laplace transformation (ILT) method based on concentrated matrix exponential (CME) distributions, referred to as the CME method.

The CME method does not generate overshoot and undershoot (i.e., avoids Gibbs oscillation), preserves monotonicity of functions, its accuracy is gradually improving with the order, and it is numerically stable even for order 1000 when using machine precision arithmetic, while other methods get unstable already for order 100 using the same arithmetic.

For ILT based tail approximation, the paper recommends an abscissa shifting approach which improves the accuracy of most ILT methods and proposes a heuristic procedure to approximate the numerical accuracy of some ILT methods.

Keywords: numerical inverse Laplace transformation, Abate–Whitt framework, Gibbs oscillation, concentrated matrix exponential distributions.

1. Introduction

Laplace transforms are widely used in various scientific fields [1]. There are plenty of numerical inverse Laplace transformation (ILT) methods published in the literature; for relatively recent surveys, we refer to [2] and [3], while [4] and [5] are somewhat older but thorough books on the topic. [6] is a recent comparative study.

Among these methods one of the most widely applied and well characterized subset is the Abate–Whitt framework defined in [7]. This framework implicitly

[☆]This work is supported by the OTKA K-123914 and the TUDFO/51757/2019-ITM projects. The authors would like to thank the specially insightful comments of the reviewers, which resulted in important extensions of the paper, e.g. the abscissa shifting approach.

[*]Corresponding author

Email addresses: ghorvath@hit.bme.hu (Gábor Horváth), horvath.illes.antal@gmail.com (Illés Horváth), almousa@hit.bme.hu (Salah Al-Deen Almousa), telek@hit.bme.hu (Miklós Telek)

defines function families in which various optimizations can be performed in order to obtain efficient ILT methods.

The CME method belongs to the most general function family of the Abate–Whitt framework (referred to as Class III in [7]), but unlike most of the other methods of the framework, the CME method is non-overshooting, i.e., does not create oscillating waves at discontinuities. In [8] we recognized that concentrated matrix-exponential (CME) distributions can be integrated into the Abate–Whitt ILT framework with many beneficial features. In that paper, we reported the non-overshooting property of the approach. Recent improvements in the computation of CME distributions [9] allow us to extend the numerical ILT method also to high orders. Currently, CME distributions up to order 1000 are available. In this work we present the first numerical experiences about the ILT method based on high order CME distributions together with some qualitative properties of the CME methods.

The rest of the paper is organized as follows. Section 2 provides general introduction to ILT and presents the Abate–Whitt framework along with an integral based interpretation. Section 3 presents the proposed new method with the necessary background on the matrix exponential distribution. Section 4 provides a brief overview of several known numerical ILT methods, both within and outside the Abate–Whitt framework. The performance of the CME method is then compared with other procedures in Section 5. The paper is concluded in Section 6.

2. Inverse Laplace transformation and the Abate–Whitt framework

2.1. Inverse Laplace transformation

For a real or complex valued function $h(t)$ the Laplace transform is defined as

$$h^*(s) = \int_{t=0}^{\infty} e^{-st} h(t) \mathrm{d}t. \tag{1}$$

In some cases, symbolic ILT of $h^*(s)$ is feasible, but in a wide range of practically important cases numerical ILT is required to find an approximate value of h at point T (i.e., $h(T)$) based on $h^*(s)$.

The region of convergence for the integral in (1) is always of the form $\{s : \mathrm{Re}(s) > a\}$ (possibly including some points of the boundary line $\{\mathrm{Re}(s) = a\}$), or empty ($a = \infty$), or the entire complex plane ($a = -\infty$). The real constant a is known as the abscissa of absolute convergence. In case a is finite, the function $h^*(s)$ may extend analytically to a domain larger than the region of convergence (e.g. for $h(t) = e^{-t}$, (1) is convergent for $\mathrm{Re}(s) > -1$, but $h^*(s) = \frac{1}{1+s}$ extends analytically to any $s \neq -1$).

Many of the Laplace inversion methods are based on the following Theorem.

Theorem 1. *(Bromwich inversion formula [10, Theorem 24.4])* $h(t)$ *can be recovered from* $h^*(s)$ *by the contour integral*

$$h(t) = \int_{s=b-i\infty}^{b+i\infty} e^{st} h^*(s) \mathrm{d}s, \qquad t > 0, \tag{2}$$

where i is the complex unit and b is any real value larger than the abscissa of absolute convergence a.

Many ILT methods simply approximate the integral (2) with a finite sum, while others have entirely different interpretations [4, 5].

In this work, we assume that $h(t)$ is real for any real t. This assumption is well-suited for most practical applications and implies $h^*(\bar{s}) = \bar{h}^*(s)$ (where $\bar{s}$ denotes the complex conjugate of s). The $h^*(\bar{s}) = \bar{h}^*(s)$ property allows to reduce the number of function evaluations ($h^*(s)$) in several ILT methods. In this paper, we discuss these reduced versions of the algorithms, but all considered ILT methods are applicable with slight modification also when $h(t)$ is complex valued.

2.2. The Abate–Whitt framework

Among the wide range of ILT methods, there is a distinguished set of methods, called the Abate–Whitt framework [7], which we summarize below. In this framework, a finite linear combination of the transform values approximate h, via

$$h(T) \approx \sum_{k=1}^{N} \frac{\eta_k}{T} h^* \left(\frac{\beta_k}{T} \right), \quad T > 0, \tag{3}$$

where the nodes β_k ($1 \le k \le N$) and weights η_k ($1 \le k \le N$) are real or complex numbers that depend on N, but not on the transform h^* or the time argument T.

In order to have a real approximation in (3) (to approximate a real h), the list of nodes β_k and weights η_k must contain only real values and complex conjugate pairs. Using that $h^*(\bar{s}) = \bar{h}^*(s)$, for a complex conjugate pair we have

$$\frac{\eta_k}{T} h^* \left(\frac{\beta_k}{T} \right) + \frac{\bar{\eta}_k}{T} h^* \left(\frac{\bar{\beta}_k}{T} \right) = 2\mathrm{Re} \left(\frac{\eta_k}{T} h^* \left(\frac{\beta_k}{T} \right) \right). \tag{4}$$

Denoting the set of real nodes, complex nodes with positive imaginary part and complex nodes with negative imaginary part by $R = \{k : \beta_k \in \mathbb{R}\}$, $C^+ = \{k : \mathrm{Im}(\beta_k) > 0\}$, and $C^- = \{k : \mathrm{Im}(\beta_k) < 0\}$, we have

$$h(T) \approx h_n(T) := \sum_{k \in R} \frac{\eta_k}{T} h^* \left(\frac{\beta_k}{T} \right) + \sum_{k \in C^+} 2\mathrm{Re} \left(\frac{\eta_k}{T} h^* \left(\frac{\beta_k}{T} \right) \right), \tag{5}$$

which means that h^* needs to be evaluated only $n = |R| + |C^+|$ times instead of $N = |R| + |C^+| + |C^-|$ times as it is in (3). That is, in h_n, the subscript n denotes the number of transform function evaluations, which is referred to as the order of the approximation.

One needs to take care when selecting a method for a given $h(t)$ function so that all nodes β_k/T are within the region of convergence of (1). If some of the β_k/T nodes are outside the region of convergence, but avoid all poles of $h^*(s)$,

(5) can still be evaluated, but the resulting approximation may or may not be meaningful, depending on the analytic extension of $h^*(s)$.

An important special case is the following. For the class of $h(t)$ functions with $a \leq 0$ (this includes e.g. all bounded functions), any method that has $\mathrm{Re}(\beta_k) > 0$, $\forall k$ can be used. Such methods include several classical methods (such as the Euler-method and the Gaver–Stehfest method) and also the CME method of the present paper.

If some of the β_k/T nodes are outside the region of convergence of (1) for a given function $h(t)$, an abscissa shifting modification of (5) is needed which makes the corresponding method applicable:

$$h(T) \approx h_{n,\theta}(T) := \\ \exp(\theta T)\left(\sum_{k\in R}\frac{\eta_k}{T}h^*\left(\frac{\beta_k}{T}+\theta\right)+\sum_{k\in C^+}2\mathrm{Re}\left(\frac{\eta_k}{T}h^*\left(\frac{\beta_k}{T}+\theta\right)\right)\right). \quad (6)$$

(6) is based on the fact that the Laplace transform of $e^{-\theta t}h(t)$ is $h^*(s+\theta)$, the expression in the largest bracket computes the ILT of $h^*(s+\theta)$ at point T and it is multiplied by $\exp(\theta T)$ to obtain an approximation of $h(T)$.

Applying (6), one can shift the abscissa by any θ, e.g., to enforce $\min_k \mathrm{Re}\left(\frac{\beta_k}{T}+\theta\right) > a$. Shifting the abscissa has significant consequences on the quality of the approximation for large values of T; this is investigated in Section 5.5.

Originally, the main idea behind the Abate–Whitt framework was a Fourier-series approximation [7]. Below we present a different, integral-based interpretation based on the fact that (5) can be rewritten as

$$\begin{aligned} h_n(T) &= \sum_{k\in R}\frac{\eta_k}{T}h^*\left(\frac{\beta_k}{T}\right)+\sum_{k\in C^+}2\mathrm{Re}\left(\frac{\eta_k}{T}h^*\left(\frac{\beta_k}{T}\right)\right) \\ &= \frac{1}{T}\sum_{k\in R}\eta_k\int_0^\infty e^{-\frac{\beta_k}{T}t}h(t)\mathrm{d}t+\sum_{k\in C^+}\frac{2}{T}\mathrm{Re}\left(\eta_k\int_0^\infty e^{-\frac{\beta_k}{T}t}h(t)\mathrm{d}t\right) \\ &= \int_0^\infty h(t)\cdot\frac{1}{T}f_n(t/T)\mathrm{d}t = \int_0^\infty h(tT)\cdot f_n(t)\mathrm{d}t, \end{aligned} \quad (7)$$

where

$$f_n(t) = \sum_{k\in R}\eta_k e^{-\beta_k t}+\sum_{k\in C^+}2\mathrm{Re}\left(\eta_k e^{-\beta_k t}\right). \quad (8)$$

If $f_n(t)$ was the Dirac impulse function at point 1, then the integral in (7) would result in a perfect Laplace inversion. Figure 1 displays $f_n(t)$ for the most popular members of the Abate–Whitt framework (the Euler method and the Gaver–Stehfest method, defined in Section 4) and the CME method (defined in Section 3), for different orders. As a first impression, in accordance with expectations, the functions get visually "narrower" (more concentrated) for higher orders for all methods.

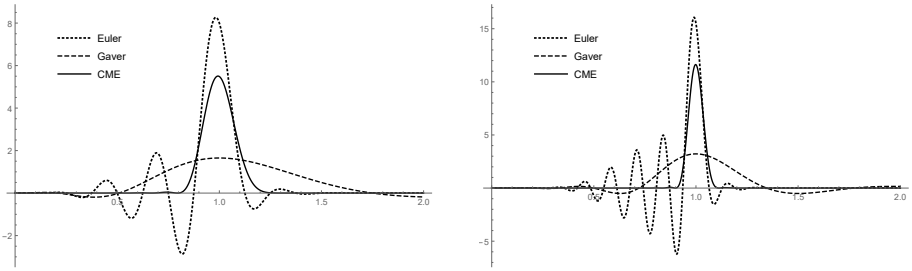

Figure 1: $f_n(t)$ approximating the Dirac impulse function for order 10 and 20

Remark 1. *The function $f_n(t)$ can also be obtained from another approach. The Laplace-transform of the Dirac impulse function at 1 is $h_D^*(s) = e^{-s}$; applying* (7) *to this function yields the approximation*

$$\begin{aligned} h_D(T) &= \sum_{k\in R} \frac{\eta_k}{T} h_D^*\left(\frac{\beta_k}{T}\right) + \sum_{k\in C^+} 2\mathrm{Re}\left(\frac{\eta_k}{T} h_D^*\left(\frac{\beta_k}{T}\right)\right) \\ &= \sum_{k\in R} \frac{\eta_k}{T} e^{-\frac{\beta_k}{T}} + \sum_{k\in C^+} 2\mathrm{Re}\left(\frac{\eta_k}{T} e^{-\frac{\beta_k}{T}}\right) = \frac{1}{T} f_n(1/T), \end{aligned} \tag{9}$$

which is a transformed version of $f_n(t)$, where the intervals $[0,1]$ and $[1,\infty]$ are flipped according to $t \to \frac{1}{T}$ and renormalized. For ILT methods outside the Abate–Whitt framework the derived $f_n(t)$ function can be computed as $f_n(t) = \frac{1}{t} h_D(1/t)$.

Since f_n approximates the unit impulse at 1, the η_k weights and the β_k nodes of the Abate–Whitt framework methods are such that f_n is *normalized*, that is,

$$\int_0^\infty f_n(t)\mathrm{d}t = 1.$$

If additionally $f_n(t)$ is non-negative for $t \geq 0$, then Theorems 2 and 3 guarantee desirable qualitative properties of the ILT.

Theorem 2. *If f_n is normalized and non-negative, then the approximation does not generate overshoot or undershoot, i.e.,*

$$\inf_{t\geq 0}(h(t)) \leq h_n(T) \leq \sup_{t\geq 0}(h(t)) \qquad \forall T > 0.$$

Proof. From (7) we have

$$h_n(T) = \int_0^\infty h(tT)\cdot f_n(t)\mathrm{d}t \leq \int_0^\infty \sup(h)\cdot f_n(t)\mathrm{d}t = \sup(h),$$

where the inequality is due to $f_n(t) \geq 0$ and the last equality is due to $\int_0^\infty f_n(t)\mathrm{d}t = 1$. The proof of the lower bound is analogous. □

Theorem 2 is particularly important for providing a condition to avoid Gibbs oscillations. Traditionally, Gibbs oscillation is the oscillating error which occurs when approximating a discontinuous function (e.g. square wave) with Fourier series [11, Chapter 1]; notoriously, the error does not vanish when adding more terms to the approximation. For ILT methods, Gibbs oscillations might occur when using a Fourier-series based method to approximate a discontinuous h function.

Avoiding overshooting is relevant in many practical applications where h has strict upper and/or lower bounds, and violating the bounds makes no physical sense, e.g., for cumulative distribution functions or probability density functions in probability theory.

Theorem 3. *If f_n is non-negative, then the ILT preserves monotonicity, that is, if $h(t)$ is increasing (resp. decreasing), then $h_n(t)$ is also increasing (resp. decreasing).*

Proof. Let $T_1 < T_2$ and assume that h is increasing. From (7) we have

$$h_n(T_1) = \int_0^\infty h(tT_1) \cdot f_n(t)\mathrm{d}t \leq \int_0^\infty h(tT_2) \cdot f_n(t)\mathrm{d}t = h_n(T_2),$$

where the inequality is due to $h(tT_1) < h(tT_2)$ and $f_n(t) \geq 0$. The decreasing case is analogous. □

Preserving monotonicity by ILT is important in practical applications where h is known to be increasing, e.g., in case of cumulative distribution functions. Both properties are relevant and useful for many other fields of application.

3. Concentrated matrix exponential distributions

3.1. Matrix exponential distributions

The class of matrix exponential distributions of order N, denoted by ME(N), contains positive random variables with probability density function (pdf) of the form

$$f_X(t) = -\alpha \mathbf{A} e^{\mathbf{A}t}\mathbf{1}, \qquad t \geq 0, \tag{10}$$

where α is a real row vector of length N, $\mathbf{A}$ is a real matrix of size $N \times N$ and $\mathbf{1}$ is a column vector of ones of size N [12]. To ensure $\int_0^\infty f_X(t)\mathrm{d}t = 1$, α and $\mathbf{A}$ are such that $\alpha\mathbf{1} = 1$ and the eigenvalues of $\mathbf{A}$ have negative real part.

Nonnegativity of $f_X(t)$ does not follow from (10), but some $(\alpha, \mathbf{A})$ pairs result in $f_X(t)$ functions that are non-negative for $t \geq 0$. Only these non-negative functions are considered to be members of the ME(N) class. Checking whether $f_X(t)$ is non-negative based on the pair $(\alpha, \mathbf{A})$ is a difficult task in general [13].

Assuming that $\mathbf{A}$ is diagonalizable with spectral decomposition

$$\mathbf{A} = \sum_{k=1}^{N} u_k \lambda_k v_k,$$

where λ_k, $k = 1, \ldots, \lambda_N$ are the eigenvalues, u_k, $k = 1, \ldots N$ are the right eigenvectors and v_k, $k = 1, \ldots N$ are the left eigenvectors of $\mathbf{A}$ with $v_k u_k = 1$, f_X can be written as

$$f_X(t) = \sum_{k=1}^{N} \underbrace{-\alpha \mathbf{A} u_k v_k \mathbf{1}}_{c_k} e^{\lambda_k t} = \sum_{k=1}^{N} c_k e^{\lambda_k t}. \tag{11}$$

Comparing (11) and (8) indicates that ME distributions with diagonalizable matrix $\mathbf{A}$ can be used in the place of $f_n(t)$ to obtain an ILT method of the Abate–Whitt framework.

3.2. Concentrated ME distributions

The *squared coefficient of variation (SCV)*

$$\mathrm{SCV}(f) := \frac{\int_{t=0}^{\infty} t^2 f(t) \mathrm{d}t}{\left(\int_{t=0}^{\infty} t f(t) \mathrm{d}t\right)^2} - 1$$

measures how concentrated a non-negative, normalized function on $\mathbb{R}^+$ is. Function f with $\mathrm{SCV}(f) = 0$ is the Dirac function and the smaller $\mathrm{SCV}(f)$ is, the better f approximates the Dirac function.

Within the class $\mathrm{ME}(N)$, the f_X with minimal SCV is not known for $N \geq 3$ analytically. Concentrated $\mathrm{ME}(N)$ distributions with low coefficient of variation have been calculated for odd values of N in [14] up to $N = 47$ and in [9] for up to $N = 2001$ using numerical optimization for the following form:

$$f_X(t) = c\, \mathrm{e}^{-\lambda t} \prod_{j=0}^{(N-1)/2} \cos^2(\omega t - \phi_j) \tag{12}$$

with positive real values of c, λ, ω and $\phi_1, \ldots, \phi_{(N-1)/2}$. According to the numerical optimization results the minimal SCV decays approximately as $N^{-2.14}$. It is worth noting that $f_X(t)$ in (12) is non-negative by construction.

3.3. The CME method

In order to apply the CME distributions for ILT, (12) can be rewritten in a form consistent with (3):

$$f_X(t) = c\, \mathrm{e}^{-\lambda t} \prod_{j=0}^{(N-1)/2} \cos^2(\omega t - \phi_j) = \sum_{k=1}^{N} \eta_k e^{-\beta_k t}, \tag{13}$$

where η_1 and β_1 are real, and $\eta_2, \ldots, \eta_N$ and $\beta_2, \ldots, \beta_N$ are $(N-1)/2$ complex conjugate pairs. The details of this transformation are provided in [14, Appendix]. Based on the number of real and complex nodes and weights, $|R| = 1$, $|C^+| = (N-1)/2$ and, using (4), the order of the CME method obtained from (13) is $n = 1 + (N-1)/2 = (N+1)/2$.

For a given order n, the η_k and β_k values providing the minimal SCV are obtained from numerical optimization. We propose to use a precompiled list of the η_k and β_k values, which are available online up to order 1000 at [15].

Since the optimization of (12) was carried out for non-negative and normalized functions, Theorems 2 and 3 ensure that the CME method is free of Gibbs oscillation and preserves monotonicity. Many of the known methods lack these properties; e.g., Figure 1 displays that the $f_n(t)$ functions for the Euler method and the Gaver–Stehfest method take negative values as well.

According to (7), the error of the order n approximation at point T depends on two factors, $f_n(t)$ and $h(t)$. Appendix A provides an asymptotic result for the approximation as $\mathrm{SCV}(f_n)$ tends to zero under some mild conditions on $h(t)$.

4. Classic methods

4.1. Methods within the Abate–Whitt framework

From the Abate–Whitt framework, for numerical comparisons we consider the Gaver–Stehfest, the Euler, and the Talbot method, which are commonly considered to be among the bests [7]. The weights η_k and nodes β_k are given in accordance with (8), e.g. from complex conjugate pairs, only the ones with $\mathrm{Re}(\beta_k) > 0$ are included.

Gaver–Stehfest method (defined only for even n)

The Gaver–Stehfest method is based on the sequence of Gaver approximants, derived by Gaver [16]. Since the convergence of the Gaver approximants was only logarithmic, it needed acceleration. A linear acceleration method was proposed by Stehfest using the Salzer acceleration scheme [17]. The accelerated version fits in the Abate–Whitt framework [7]. Hereafter we refer to the Gaver–Stehfest method as Gaver method for simplicity.

The weights η_k and nodes β_k, $k = 1, 2, \dots n$ are as follows.

$$\beta_k = k \ln(2), \quad \text{for } 1 \le k \le n,$$

$$\eta_k = (-1)^{n/2+k} \ln(2) \sum_{j=\lfloor (k+1)/2 \rfloor}^{\min(k,n/2)} \frac{j^{n/2+1}}{(n/2)!} \binom{n/2}{j} \binom{2j}{j} \binom{j}{k-j}, \quad \text{for } 1 \le k \le n,$$

where $\lfloor x \rfloor$ is the greatest integer less than or equal to x. We note that $\max_{1 \le k \le n} |\eta_k|$ increases exponentially in n, which leads to numerical error when using floating point arithmetic.

The Gaver method operates with real β_k values which is beneficial when complex arithmetic is not supported. The method exhibits Gibbs oscillation because the functions $f_n(t)$ take negative values as in Figure 1 .

Euler method (defined only for odd n)

The Euler method is an implementation of the Fourier-series method, using Euler summation to accelerate convergence [4]. The weights η_k and nodes β_k, $k = 1, 2, \ldots n$ are as follows.

$$\beta_k = \frac{(n-1)\ln(10)}{6} + \pi i(k-1), \quad 1 \le k \le n,$$
$$\eta_k = 10^{(n-1)/6}(-1)^k \xi_k, \quad 1 \le k \le n,$$

where

$$\xi_1 = \frac{1}{2},$$
$$\xi_k = 1, \quad 2 \le k \le (n+1)/2,$$
$$\xi_n = \frac{1}{2^{(n-1)/2}},$$
$$\xi_{n-k} = \xi_{n-k+1} + 2^{-(n-1)/2}\binom{(n-1)/2}{k}, \qquad \text{for } 1 \le k < (n-1)/2.$$

Similar to the Gaver method, $\max_{1\le k\le n} |\eta_k|$ increases exponentially with n, and the functions $f_n(t)$ take also negative values.

Talbot method

The Talbot method [18] is based on deforming the contour integral (2) in the Bromwich inversion. The procedure is further simplified by Abate and Valko [19] to fit in the Abate–Whitt framework according to the following nodes and weights

$$\beta_1 = \frac{2n}{5},$$
$$\beta_k = \frac{2(k-1)\pi}{5}\left(\cot\left(\frac{(k-1)\pi}{n}\right) + i\right), \quad 2 \le k \le n,$$
$$\eta_1 = \frac{1}{5}e^{\beta_1},$$
$$\eta_k = \frac{2}{5}\left[1 + i\frac{(k-1)\pi}{n}\left(1 + \left[\cot\left(\frac{(k-1)\pi}{n}\right)\right]^2\right) - i\cot\left(\frac{(k-1)\pi}{n}\right)\right]e^{\beta_k}, \quad 2 \le k \le n.$$

Similar to the Euler and the Gaver methods, $\max_{1\le k\le n} |\eta_k|$ increases exponentially with n and Gibbs oscillation might occur when using the Talbot method.

Figure 2 depicts the location of the β_k nodes of the three methods. We note that for the Gaver, Euler and CME methods, the nodes all have positive real parts, which means that all nodes are inside the range of convergence of the Laplace transform integral (1) for functions h whose abscissa of absolute convergence a is 0 or negative. For the Talbot method, some nodes have negative real parts, which might cause diverging inverse transform function in case of a bounded function, depending on the function h and the order n.

Other methods within the Abate–Whitt framework

The following methods are not considered in our numerical evaluations.

Both Zakian [20, 21, 22] and Wellekens [23] use a Fourier–series method based on Padé–approximation, but with limited applicability [7].

A method based on kernel hyperbolic approximation [5, 24] works well for orders up to order 7, but gets problematic for higher orders.

4.2. Methods outside the Abate–Whitt framework

The most common reason for a method not being a member of the Abate–Whitt framework is that the method evaluates the derivatives of h^*. The following methods share this property.

Post–Widder method

The Post-Widder method [25, Chapter 7], [4, Theorem 2] can be regarded as a version of (7) with

$$f_n(t) = \frac{n^n t^{n-1} e^{-nt}}{(n-1)!} \tag{14}$$

serving as the approximate Dirac function. $f_n(t)$ is not a pure exponential function due to the polynomial term t^n; in fact, (14) corresponds to the numerical inversion formula

$$h_n(T) := \frac{(-1)^{n-1}}{(n-1)!} \left(\frac{n}{T}\right)^n h^{*(n-1)}(n/T). \tag{15}$$

The main issue with (15) is the need for higher order differentiation. In some cases, higher order derivatives might be available analytically; otherwise, they can be approximated either by finite differences [4] or complex integrals (also known as Jagerman's method [26, 27, 28]) to result in methods that fall within the Abate–Whitt framework.

Since the function $f_n(t)$ in (14) is non-negative for $t \geq 0$ and normalized, the Post–Widder method is non-overshooting and preserves monotonicity.

In probability theory, the Erlang distribution is defined as having pdf equal to (14). The Erlang distribution is often used to approximate the Dirac distribution [29]; however, the concentrated matrix exponential distributions used in the CME method results in a much better approximation. The Erlang distribution has SCV=n^{-1}, while the CME distributions have SCV that decays approximately as $n^{-2.14}$ as shown recently in [9].

Laguerre method

This method is based on the Laguerre-series representation in [4, formulas (2.16)–(2.19)], which can be rewritten in an equivalent form as

$$h_n(T) = \sum_{j=0}^{n-1} h^{*(j)}(1/2) \cdot \sum_{k=j}^{n-1} \frac{k!}{(j!)^2(k-j)!} \underbrace{e^{-T/2} \sum_{m=0}^{k} \binom{k}{m} \frac{(-T)^m}{m!}}_{\text{Laguerre function}}. \tag{16}$$

(16) involves the derivatives of h^*, which might be handled with techniques similar to the Post–Widder method [4, 30]. The Laguerre method suffers from Gibbs oscillation.

Other methods outside the Abate–Whitt framework

Cohen [25, Section 3.1] describes a method based on the power series of $h^*(s)$ as a function of $1/s$. However, this method is not applicable when $h^*(s)$ has a pole at 0, and the approximation also gets rapidly worse for larger values of t. This method is not involved in our numerical evaluations.

4.3. Higher dimensional ILT methods

The d-dimensional Laplace transform is defined as

$$h^*(s_1, \ldots, s_d) = \int_{t_1=0}^{\infty} \ldots \int_{t_d=0}^{\infty} e^{-(s_1 t_1 + \ldots + s_d t_d)} h(t_1, \ldots, t_d) \mathrm{d}t_d \ldots \mathrm{d}t_1. \quad (17)$$

The literature on higher dimensional ILT methods is limited. Available methods typically address the 2-dimensional case and are almost exclusively 2-dimensional extensions or combinations of 1-dimensional methods. [31] provides a 2-dimensional ILT method based on finite Fourier series approximations, which is a straightforward extension of the 1-dimensional method of [32]. [33] presents two 2-dimensional ILT methods based on versions of the Gaver method and the Talbot-method. [34] provides a 2-dimensional ILT method based on the Laguerre method also using a fast Fourier transform-based algorithm. [35] presents three 2-dimensional methods, all based on known 1-dimensional methods, using Fourier-series approximations and approximations of the Bromwhich integral. A recent comparison of some 2-dimensional ILT methods is provided in [36].

Any 1-dimensional ILT method naturally extends to 2 or more dimensions simply by applying it to all variables. For the Abate–Whitt framework, the 2-dimensional version of (5) (assuming the same η_k, β_k values for the approximation in both variables) is:

$$\begin{aligned} h_n(T_1, T_2) = & \sum_{k_1 \in R} \sum_{k_2 \in R} \frac{\eta_{k_1} \eta_{k_2}}{T_1 T_2} h^* \left(\frac{\beta_{k_1}}{T_1}, \frac{\beta_{k_2}}{T_2} \right) \\ & + 2 \sum_{k_1 \in R} \frac{\eta_{k_1}}{T_1} \mathrm{Re} \left(\sum_{k_2 \in C^+} \frac{\eta_{k_2}}{T_2} h^* \left(\frac{\beta_{k_1}}{T_1}, \frac{\beta_{k_2}}{T_2} \right) \right) \\ & + 2 \sum_{k_2 \in R} \frac{\eta_{k_2}}{T_2} \mathrm{Re} \left(\sum_{k_1 \in C^+} \frac{\eta_{k_1}}{T_1} h^* \left(\frac{\beta_{k_1}}{T_1}, \frac{\beta_{k_2}}{T_2} \right) \right) \\ & + 4 \mathrm{Re} \left(\sum_{k_1 \in C^+} \sum_{k_2 \in C^+} \frac{\eta_{k_1} \eta_{k_2}}{T_1 T_2} h^* \left(\frac{\beta_{k_1}}{T_1}, \frac{\beta_{k_2}}{T_2} \right) \right). \end{aligned} \quad (18)$$

The 2-dimensional version of the CME method retains the advantages of the 1-dimensional CME method: it is non-overshooting, preserves monotonicity, is

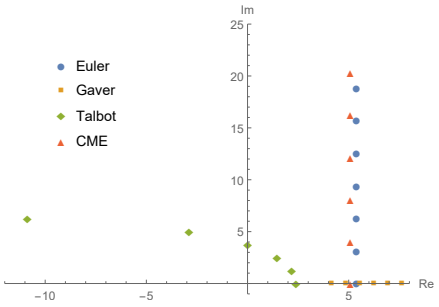

Figure 2: Location of the β_k nodes on the complex planes with order 6

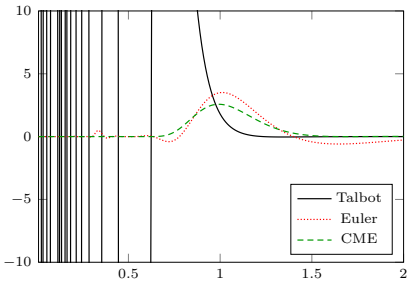

Figure 3: Numerical inverse of the Dirac impulse function with the Talbot method with order 5

accurate using machine precisions floating point arithmetic, and the quality of approximation improves when increasing the order.

Since we believe that the best general purpose 1-dimensional ILT method is the CME method, we recommend the use of its multi-dimensional extension (according to (18)) for multi-dimensional numerical ILT.

5. Numerical properties

In order to investigate the properties of the numerical ILT methods of Section 3 and 4 (Euler, Gaver, Talbot, Post–Widder (abbreviated as P–W), Laguerre and CME), we performed a set of numerical ILTs for the 6 test functions of Table 1 using our Matlab implementation, where we applied machine precision floating point arithmetic for CME and 100 digits precision arithmetic with the Matlab Symbolic Math Toolbox for other methods (although we note that even 100 digits precision is not enough for every case). For methods outside the Abate–Whitt framework, we used symbolic differentiation for the functions listed in Table 1 whenever applicable. Also, for methods not included in the comparisons here, we refer to the recent comparative study [6].

The main basis of comparison is the order of the methods; for the Abate–Whitt framework, the order is based on the value of n in (5), while for methods outside the Abate–Whitt framework, order n means that we need to evaluate the derivatives of $h^*(s)$ up to $n-1$ at a single point. As the Euler method is applicable only to odd orders, we increased the order by 1 for the Euler method whenever necessary for order based comparison.

	exp	*sin*	*heaviside*	*shifted exp*	*staircase*	*square wave*
$h(t)$	e^{-t}	$\sin t$	$\mathbb{1}(t>1)$	$\mathbb{1}(t>1)\ e^{1-t}$	$\lfloor t \rfloor$	$\lfloor t \rfloor \mod 2$
$h^*(s)$	$\frac{1}{1+s}$	$\frac{1}{s^2+1}$	$\frac{1}{s}e^{-s}$	$\frac{e^{-s}}{1+s}$	$\frac{1}{s}\frac{1}{e^s-1}$	$\frac{1}{s}\frac{1}{e^s+1}$

Table 1: Set of test functions

5.1. Order of magnitude of the weights

When using floating point arithmetic, the numerical accuracy of computing $h_n(T)$ based on (5) is closely related to the order of magnitude of the η_k weights. Applying $\log_{10}(\max_k |\eta_k|)$ as an index of potential numerical instability allows a rough assessment of the numerical precision of various methods of the Abate–Whitt framework. In case of the methods outside the Abate–Whitt framework (Section 4.2), we define the weights as the coefficients of the derivatives of h^* (in (15) and (16)) at $T = 1$.

Table 2 compares $\log_{10}(\max_k |\eta_k|)$ for the considered methods with various orders. An examination of the weights for the CME method reveals that $\log_{10}(\max_k |\eta_k|)$ does not increase beyond 7.5, thus the CME method does not suffer from significant rounding error up to order 1000 using machine precision floating point arithmetic, in contrast to other considered methods, where $\max_k |\eta_k|$ increases exponentially with the order.

	Abate–Whitt framework				non A–W	
order	Gaver	Euler	Talbot	CME	P–W	Laguerre
10	5.58	3.45	**1.29**	3.22	4.44	3.26
30	18.9	11.1	4.80	**3.76**	13.4	7.10
50	32.4	18.8	8.28	**4.28**	22.2	9.81
100	66.2	38.0	17.0	**4.94**	44.0	14.6
500	337	191	86.5	**6.56**	218	35.9
1000	676	383	173	**7.24**	435	51.9

Table 2: Value of $\log_{10}(\max |\eta_k|)$ for various methods within and outside the Abate–Whitt framework

The numerical inaccuracy caused by large $|\eta_k|$ weights can be reduced by applying higher precision floating point arithmetic at the expense of an increased computational cost. It is a significant drawback of the other considered methods that the required precision strongly depends on the order of the method, while machine precision arithmetic is sufficient up to order 1000 in case of the CME method.

5.2. Numerical ILT of the unit impulse function

According to (7), $f_n(t)$ plays a central role in the quality of the Abate–Whitt framework methods. Unfortunately, $f_n(t)$ is defined only for the Abate–Whitt framework. Based on Remark 1 we can extend the experimental assessment of the ILT methods outside the Abate–Whitt framework using the ILT of the unit impulse function with $h^*_D(s) = e^{-s}$.

Figure 4 depicts the numerical inverse of the Dirac impulse function with the considered methods, except the Talbot method, which is unstable already at order 5 (Fig. 3). In general, Figure 4 suggests that the Euler and the Gaver methods of the Abate–Whitt framework outperform the methods outside the

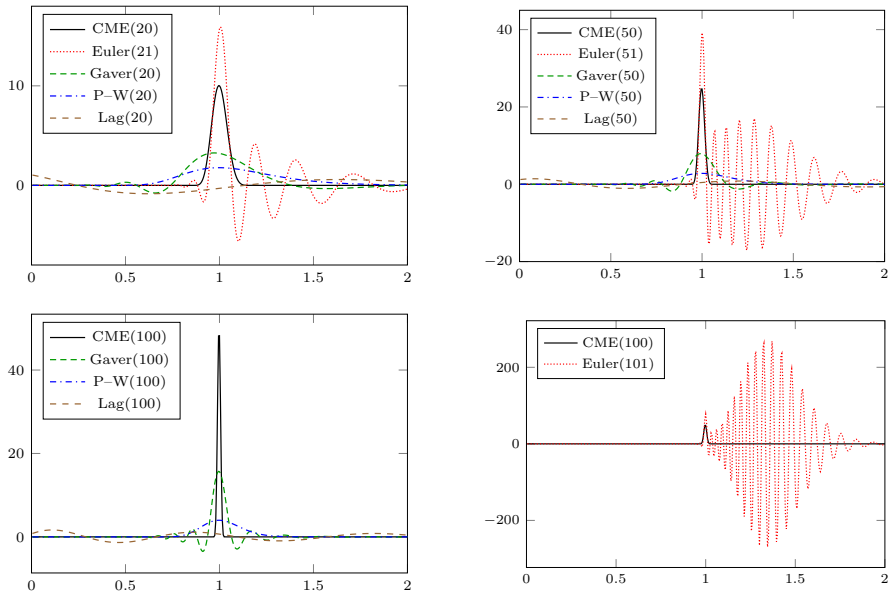

Figure 4: Numerical inverse of the Dirac impulse function

framework, with respect to approximating the peak at time 1. The narrowest peak is provided by the Euler method, but its quality is deteriorated by the strong oscillation next to the peak. The Laguerre method suffers from both problems, poor approximation of the peak and slowly decaying oscillation far from to the peak. The CME method shows better behaviour in both respects. It closely approximates the peak and has negligible oscillation next to the peak.

5.3. 1-norm of the numerical error

The original $h(t)$ functions of Table 1 and the approximating $h_n(t)$ were both evaluated at $M = 100$ equidistant points over the interval $[0, 5]$ and the empirical 1-norm was calculated by

$$\|h - h_n\|_1 = \int_0^T |h(t) - h_n(t)| \mathrm{d}t \approx \frac{1}{M} \sum_{m=1}^{M} \left| h\left(\frac{mT}{M}\right) - h_n\left(\frac{mT}{M}\right) \right|.$$

We chose the 1-norm because it is much more informative for discontinuous h functions than the ∞-norm. Table 3 presents the results. Values larger than 1000 were marked with p. inf. (practically infinite) since the original functions themselves are less than 10 for the interval $[0, 5]$. These extremely large values are the result of either numerical instability (even with 100 digits precision arithmetic) or the approximations inherent to each method.

At order 500, all known methods fail due to numerical instability with very few exceptions, which are in good agreement with the methods having lower

weights in Table 2. The precise order at which numerical instability becomes significant is further addressed in Section 5.4.

For the analytic functions e^{-t} and $\sin(t)$, most methods work well already for low orders. For e^{-t}, the Gaver, Euler, Talbot and Laguerre methods provide practically perfect results already around order 30, while the Post-Widder method and the CME method also provide good results, which improve steadily as the order is increased. For $\sin(t)$, the Gaver, Euler, and Talbot methods provide practically perfect results at around order 30, while the Post-Widder and Laguerre methods have error of $\sim 10^{-2}$. The CME method is in between, with the error decreasing from $\sim 10^{-2}$ for order 10 gradually to $\sim 10^{-6}$ for order 500.

For discontinuous functions, such as the heaviside function $\mathbb{1}(t>1)$, shifted exponential function $\mathbb{1}(t>1)e^{1-t}$, the staircase function $\lfloor t \rfloor$, and the square wave function $\lfloor t \rfloor \mod 2$, CME outperforms all known methods.

The Talbot method is practically inappropriate for many of these functions (regardless of the order). The Laguerre method also fails for the staircase and square wave functions.

The Euler and Gaver methods typically have error of $\sim 10^{-1}$ or $\sim 10^{-2}$. For both methods, Gibbs oscillations cause numerical instability above a certain order.

The Post–Widder method provides results with the error decreasing with the order, albeit very slowly: even for order 100, the error does not decrease below 10^{-2}. For the Post–Widder method, it is theoretically possible to obtain arbitrary precision by increasing the order and also increasing the precision of the floating point arithmetic used (and assuming that higher order derivatives of $h^*(s)$ can be computed with sufficient precision).

Meanwhile, the CME method is applicable at any order (up to 1000) and at a low computational cost, using only machine precision arithmetic. For all the discontinuous functions tested, it gives better results than any of the known methods at any order. The error is decreasing with the order at a rate considerably faster than for the Post–Widder method, reaching $\sim 10^{-3}$ or better for every function.

5.4. Visual comparison

In Figures 5 and 6, we investigate the dependency of the Gaver and the Euler methods on the order. The Gaver method fails to follow the oscillating feature of the original function for low order (10). It produces a smooth curve with some overshoot for medium order (50), and reaches its limit of numerical stability at order 64, despite using the Symbolic toolbox with 100 digits precision. The Euler method for low order (11) follows the oscillating feature of the original function longer; it produces a smooth curve with more dominant overshoot for medium order (51), and reaches its limit of numerical stability, despite using 100 digits precision, at order 101.

The Talbot, the Post–Widder and the Laguerre methods perform poorly in Figures 7 – 9. Among the three the Talbot methods follow the alternation of

	Gaver	Euler	Talbot	P–W	Laguerre	CME
order	$h(t) = e^{-t}$					
10	$2.03E{-}04$	$8.79E{-}05$	$\mathbf{1.66E{-}07}$	$1.77E{-}02$	$9.35E{-}07$	$1.55E{-}03$
30	$1.49E{-}11$	$2.14E{-}11$	$\mathbf{1.25E{-}18}$	$5.85E{-}03$	$2.18E{-}16$	$1.47E{-}04$
50	$2.08E{-}09$	$1.25E{-}16$	$\mathbf{1.22E{-}18}$	$3.50E{-}03$	$1.30E{-}17$	$5.16E{-}05$
100	p. inf.	$6.86E{-}18$	$\mathbf{1.22E{-}18}$	$1.75E{-}03$	$1.30E{-}17$	$1.22E{-}05$
500	p. inf.	p. inf.	$\mathbf{8.29E{-}18}$	p. inf.	$1.30E{-}17$	$4.21E{-}07$
order	$h(t) = \sin t$					
10	$1.34E{-}01$	$\mathbf{3.06E{-}04}$	$1.40E{-}03$	$2.12E{-}01$	$1.65E{-}01$	$1.68E{-}02$
30	$7.37E{-}05$	$\mathbf{8.09E{-}11}$	$2.30E{-}17$	$8.92E{-}02$	$7.61E{-}02$	$2.10E{-}03$
50	$7.29E{-}10$	$\mathbf{2.09E{-}17}$	$2.30E{-}17$	$5.62E{-}02$	$3.43E{-}02$	$7.40E{-}04$
100	p. inf.	$\mathbf{2.33E{-}26}$	$2.30E{-}17$	$2.92E{-}02$	$7.94E{-}02$	$1.80E{-}04$
500	p. inf.	p. inf.	p. inf.	p. inf.	p. inf.	$\mathbf{6.47E{-}06}$
order	$h(t) = \mathbb{1}(t > 1)$					
10	$4.47E{-}02$	$1.94E{-}02$	p. inf.	$5.71E{-}02$	$1.90E{-}01$	$\mathbf{1.26E{-}02}$
30	$1.81E{-}02$	$1.03E{-}02$	p. inf.	$3.01E{-}02$	$1.46E{-}01$	$\mathbf{3.70E{-}03}$
50	$1.15E{-}02$	$1.56E{-}02$	p. inf.	$2.29E{-}02$	$1.52E{-}01$	$\mathbf{1.50E{-}03}$
100	p. inf.	$7.98E{-}02$	p. inf.	$1.59E{-}02$	$1.26E{-}01$	$\mathbf{7.94E{-}05}$
500	p. inf.	p. inf.	p. inf.	p. inf.	p. inf.	$\mathbf{7.33E{-}08}$
order	$h(t) = \mathbb{1}(t > 1)e^{1-t}$					
10	$4.70E{-}02$	$2.03E{-}02$	p. inf.	$7.78E{-}02$	$8.02E{-}02$	$\mathbf{1.37E{-}02}$
30	$1.84E{-}02$	$1.32E{-}02$	p. inf.	$4.07E{-}02$	$5.98E{-}02$	$\mathbf{4.45E{-}03}$
50	$1.21E{-}02$	$1.80E{-}02$	p. inf.	$2.99E{-}02$	$4.93E{-}02$	$\mathbf{2.65E{-}03}$
100	p. inf.	$9.82E{-}02$	p. inf.	$1.97E{-}02$	$3.69E{-}02$	$\mathbf{8.36E{-}04}$
500	p. inf.	p. inf.	p. inf.	p. inf.	p. inf.	$\mathbf{8.69E{-}07}$
order	$h(t) = \lfloor t \rfloor$					
10	$2.18E{-}01$	$\mathbf{1.28E{-}01}$	p. inf.	$2.19E{-}01$	$8.37E+00$	$1.39E{-}01$
30	$1.69E{-}01$	$7.58E{-}02$	p. inf.	$2.02E{-}01$	p. inf.	$\mathbf{5.37E{-}02}$
50	$1.18E{-}01$	$9.73E{-}02$	p. inf.	$1.89E{-}01$	p. inf.	$\mathbf{3.28E{-}02}$
100	p. inf.	$5.24E{-}01$	p. inf.	$1.68E{-}01$	p. inf.	$\mathbf{1.58E{-}02}$
500	p. inf.	p. inf.	p. inf.	p. inf.	p. inf.	$\mathbf{5.44E{-}03}$
order	$h(t) = \lfloor t \rfloor \bmod 2$					
10	$3.64E{-}01$	$\mathbf{1.21E{-}01}$	$3.64E{-}01$	$3.88E{-}01$	$4.17E{-}01$	$1.48E{-}01$
30	$1.58E{-}01$	$8.70E{-}02$	$1.34E{-}01$	$3.11E{-}01$	$3.00E+01$	$\mathbf{5.37E{-}02}$
50	$1.12E{-}01$	$9.12E{-}02$	$2.50E{-}01$	$2.62E{-}01$	$2.05E+01$	$\mathbf{3.28E{-}02}$
100	p. inf.	$5.13E{-}01$	$7.81E{-}02$	p. inf.	p. inf.	$\mathbf{1.58E{-}02}$
500	p. inf.	p. inf.	p. inf.	p. inf.	p. inf.	$\mathbf{5.44E{-}03}$

Table 3: 1-norm error of the ILT methods for the test functions, p. inf. (practically infinite) stands for values larger than 1000

Inverse Laplace transformation of the $h(t) = \lfloor t \rfloor \mod 2$ function

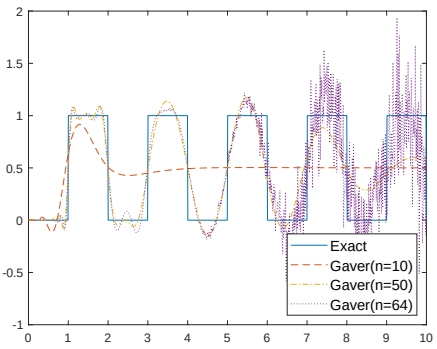

Figure 5: Gaver method

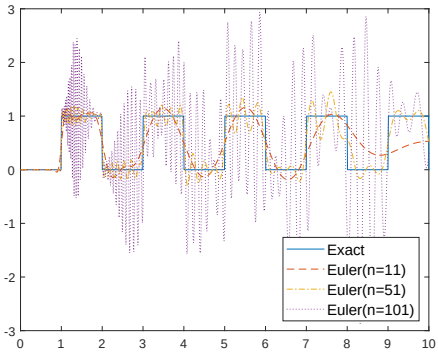

Figure 6: Euler method

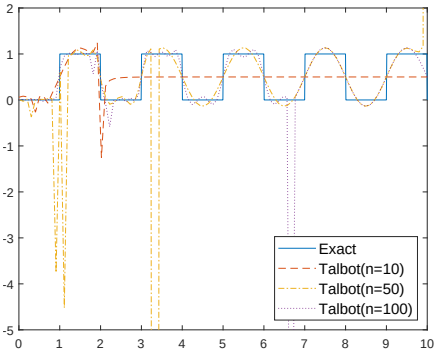

Figure 7: Talbot method

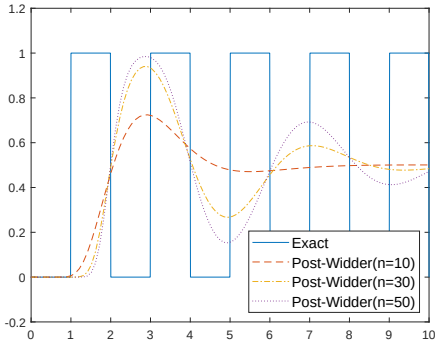

Figure 8: Post–Widder method

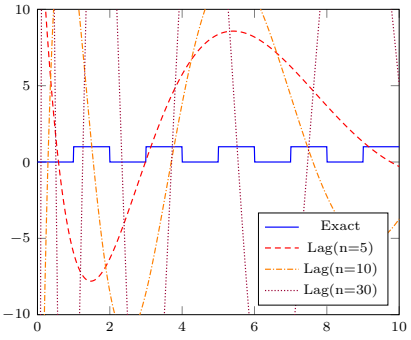

Figure 9: Laguerre method

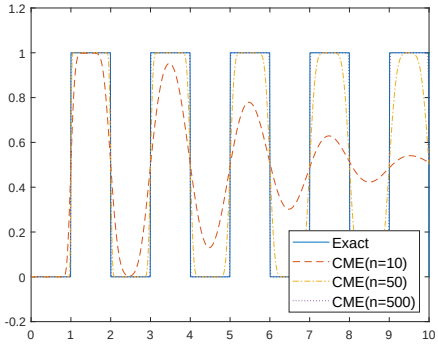

Figure 10: CME method

the square wave function best, but it exhibits randomly appearing spikes. The Post–Widder method averages out the alternation of the square wave function rather quickly, while the Laguerre method is completely off.

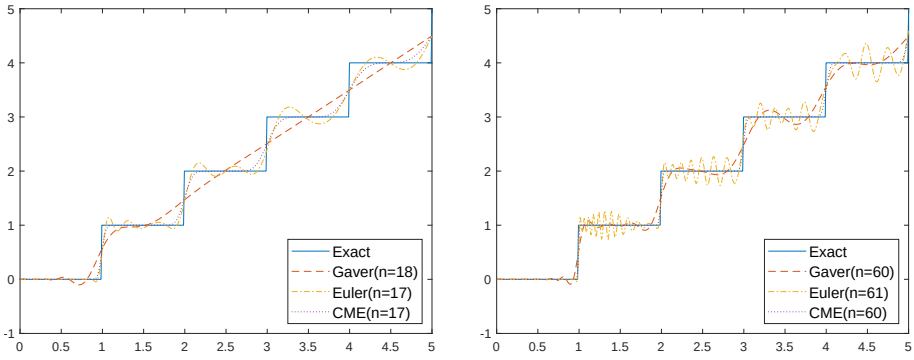

Figure 11: $h(t) = \lfloor t \rfloor$ with order 17 and 60

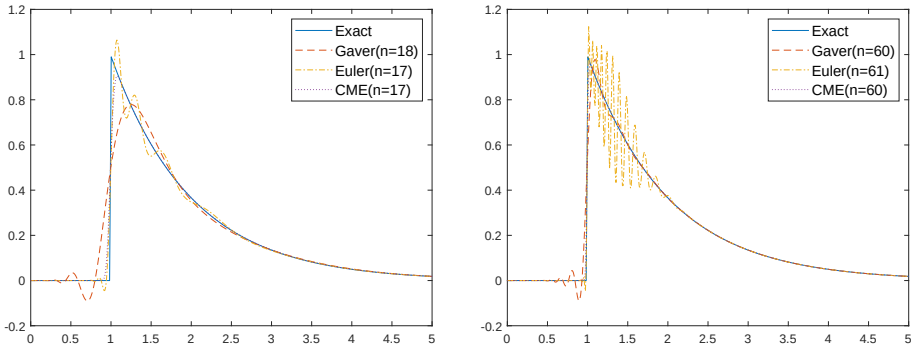

Figure 12: $h(t) = \mathbb{1}(t>1)\ e^{1-t}$ with order 17 and 60

In comparison, the ILT obtained by the CME method is depicted in Figure 10. The CME method does not produce overshoot at any order. Similar to the Euler method, the CME method follows the oscillating feature of the original function for low order (10). It produces a smooth curve for low, medium (50) and high (500) orders using machine precisions floating point arithmetic. The accuracy of the ILT increases with the order.

In Figures 11 and 12, we compare the most efficient methods for the staircase function, $h(t) = \lfloor t \rfloor$, and the shifted exponential function, $h(t) = \mathbb{1}(t>1)\ e^{1-t}$, with low and medium orders. In each case, the benefit of the non-overshooting ILT is evident. Especially, the figures with medium orders indicate the uncertainties coming from the strongly overshooting ILT of the Euler method.

The sharpest increase of the Euler and the CME methods are similar for the same orders. Approximating discontinuity, the Euler method provides a bit sharper increase/decrease than the CME method at a given order, but at the cost of significant overshoot/undershoot before and after the discontinuity.

5.5. ILT for large values of T

A useful property of the Laplace transform is that it is possible to obtain the asymptotic behavior of h based on h^*, according to the final value theorem, $\lim_{t\to\infty} h(t) = \lim_{s\to 0} sh^*(s)$.

If $h(t)$ has an exponential increase, its abscissa of absolute convergence, a, is positive and finite. To apply the Abate-Whitt framework with nodes of positive real part, it is necessary to shift the abscissa to zero, thus (6) needs to be applied with $\theta = a$, in order to ensure that $\mathrm{Re}\left(\frac{\beta_k}{T} + \theta\right) > a$ holds for $k \in \{1, \ldots, n\}$ even for large values of T. In this section we are going to show that shifting the abscissa to zero might be beneficial also when the $\mathrm{Re}\left(\frac{\beta_k}{T} + \theta\right) > a$ relation holds.

When $\lim_{t\to\infty} h(t)$ exists, most ILT methods approximate $h(T)$ well enough for large T values, where it is almost constant, thus $h(T) \approx \lim_{t\to\infty} h(t)$. The accurate approximation of the distance from the limit, i.e. $h(T) - \lim_{t\to\infty} h(t)$, is however more challenging for the ILT methods. Since adding a unit step function is possible both in time and in transform domain, we investigate the distance from the limit assuming $\lim_{t\to\infty} h(t) = 0$ in this section without loss of generality.

With respect to the decay of $h(t)$ to 0 the following three cases can be distinguished:

- superexponential decay (e.g. $e^{-(t^2+t+t^{1/2})}$), corresponding to $a = -\infty$, or
- exponential decay (e.g. $e^{-(t+t^{1/2})}$), corresponding to $-\infty < a < 0$, or
- subexponential decay (e.g. $e^{-t^{1/2}}$), corresponding to $a = 0$,

where a denotes the abscissa of absolute convergence, that is the minimal real value such that $h^*(s)$ is analytic on $\{\mathrm{Re}(z) > a\}$. There are application areas where a is known based on the underlying models (e.g. from the limit of stability of a queueing system). It can also be determined based on $h^*(s)$ using efficient numerical methods, see e.g. [37]. In the present paper, we assume that a is available.

Out of the above listed 3 convergence cases, shifting the abscissa can not be applied when the decay is superexponential, since we have $a = -\infty$. Even though superexponential functions are extremely small for large T values, the ILT based approximation of $h(T)$ for large T values is inaccurate with all ILT methods. Table 4 provides a numerical example that displays the trend of inaccuracy.

Next, we investigate the approximation accuracy for the asymptotic behavior of functions having exponential decay ($-\infty < a < 0$), where shifting the abscissa becomes relevant. As an illustrative example, let us consider the Laplace transform of the busy period in the M/M/1 queue [38]. Given the arrival rate λ and the service rate μ, the LT of the busy period is

$$h^*(s) = \frac{\lambda + \mu + s - \sqrt{(\lambda + \mu + s)^2 - 4\lambda\mu}}{2\lambda}, \tag{19}$$

method	$h(t) = \exp(-t^2)$ $h^*(s) = \frac{1}{2}e^{s^2/4}\sqrt{\pi}\mathrm{Erfc}(s/2)$			
T	5	10	20	40
precise value	1.39E-11	3.72E-44	1.92E-174	1.35E-695
CME(30)	8.74E-6	1.44E-6	5.51E-7	8.65E-8
CME(1000)	1.82E-8	7.10E-9	9.76E-10	1.93E-10
Euler(30)	-1.22E-10	1.94E-21	-1.15E-35	-1.21E-50

Table 4: Approximating a function with superexponential decay

whose inverse is

$$h(t) = \frac{e^{-(\lambda+\mu)t} I_1(2\sqrt{\lambda\mu}\, t)}{\sqrt{\lambda\mu}\, t},$$

where $I_1(z)$ is the modified Bessel function of the first kind satisfying $z^2 I_1''(z) + zI_1'(z) = (z^2+1)I_1(z)$. For this specific example the abscissa of $h^*(s)$ can be obtained symbolically, it is the (larger) solution of $(\lambda+\mu+s)^2 - 4\lambda\mu = 0$, which gives $a = 2\sqrt{\lambda\mu} - (\lambda+\mu)$. The parameters in our examples are $\lambda = 0.8, \mu = 1$, for which we have $a \approx -0.0111$.

The results are provided by the top-most block of Table 5. In the table, in those rows where $\theta = 0$ the abscissa was not shifted, while those rows where $\theta = a$ show the clear benefits of shifting the abscissa.

Table 5 contains two more examples for functions with exponential decay. When setting $\theta = a$, the function $t\exp(-t)$ is "shifted" to t, and function $\exp(-t-\sqrt{t})$ is "shifted" to $\exp(-\sqrt{t})$. In these examples the accuracy improvement due to shifting is even more prevalent.

As a result, we recommend to always shift the abscissa to zero, i.e. apply (6) **with $\theta = a$, for any ILT method when $-\infty < a < 0$.**

An intuitive explanation for the results in Table 5 is as follows. $f_n(t)$ approximates the Dirac function at $t = 1$, but the integral in (7) also assigns some weight to points away from $t = 1$ (c.f. Figure 1). If $h(tT)$ was "flat", then $f_n(t)$ properly suppresses $h(tT)$ in the intervals away from $t = 1$ and the integral is dominated by the interval close to $t = 1$. If $h(tT)$ was so large for some $t \ll 1$ or for $t \gg 1$ that $f_n(t)$ cannot suppress it, then the integral might be dominated by an interval far from $t = 1$, which causes inaccuracy. For functions with a slower rate of change (i.e. subexponential), $h(tT)$ is more flat for large values of T than for functions with a rapid change (i.e. exponential or superexponential).

Table 5 indicates the effect of "flat" $h(tT)$ also when the abscissa is shifted to zero. With abscissa shifting $h(tT) = e^{-tT-\sqrt{tT}}$ becomes $h(tT) = e^{-\sqrt{tT}}$ and $h(tT) = tTe^{-tT}$ becomes $h(tT) = tT$. For large T values $h(tT) = e^{-\sqrt{tT}}$ is sharply decreasing for $t < 1$, while the linear increase of $h(tT) = tT$ is rather "flat". As a result, the accuracy of the $h(t) = e^{-t-\sqrt{t}}$ example with $\theta = -1$

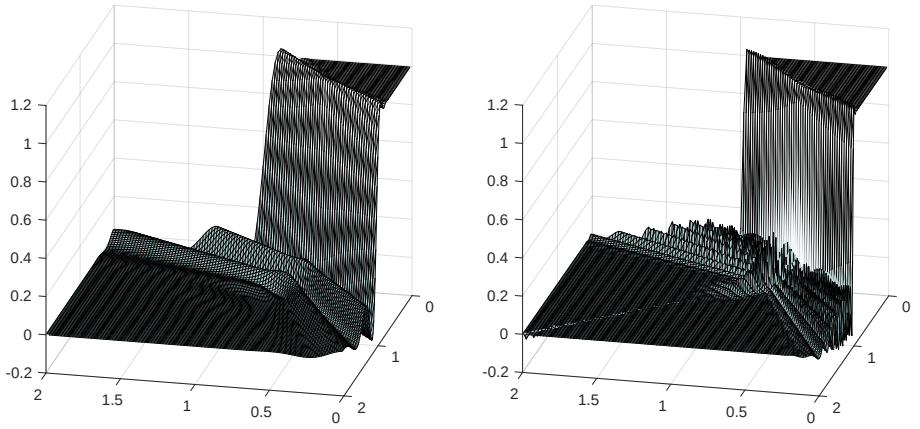

Figure 13: $\mathbb{1}(t_1 + t_2 < 1)$ with Euler method for order 11 and 41

degrades for extremely large values of T ($T = 1000$ and $T = 10000$). To cope with this potential inaccuracy, we propose a heuristic method to indicate if the approximation error gets above a given threshold in Appendix B.

Table 4 and Table 5 confirm that approximating a non-negative function, the CME method always gives non-negative results, while the Euler method might provide negative results. The Euler method typically approximates the order of magnitude of $h(T)$ better than the CME method in the regime where both methods are inaccurate.

5.6. 2-dimensional examples

Table 6 presents the test functions for 2-dimensional comparisons, which we evaluated by the Euler and the CME methods. Test functions of the form $h(t_1, t_2) = h_1(t_1)h_2(t_2)$ are avoided as they provide no additional insight compared to the 1-dimensional case.

Figure 13 displays the dependency of the Euler method on the order for the discontinuous function $\mathbb{1}(t_1 + t_2 < 1)$. The Euler method displays Gibbs-oscillations at any order, with the oscillations starting to become numerically unstable already at order 41. Figure 14 depicts the dependency of the CME method on the order. As for the 1-dimensional case, there are no Gibbs-oscillations, and the approximation improves as the order is increased.

We also compare both methods for the function $\min(t_1, t_2)$ (continuous but non-differentiable). Figure 15 presents the results of the Euler method. Since the function is smooth, the approximation for order 11 is fairly good; however, for order 61, the method gets numerically unstable. For the CME method, there are no issues. Figure 16 displays that there are no Gibbs-oscillations, and the approximation improves as the order is increased.

Computational complexity

For dimension d order n ILT in a given point, the Abate-Whitt methods evaluate the Laplace transform function at n^d (potentially complex points) and

method	θ	$h(t) = \frac{e^{-(\lambda+\mu)t} I_1(2\sqrt{\lambda\mu}\ t)}{\sqrt{\lambda\mu}\ t}$ $h^*(s) = \frac{\lambda+\mu+s-\sqrt{(\lambda+\mu+s)^2-4\lambda\mu}}{2\lambda}$ $\lambda = 0.8, \mu = 1, a = 2\sqrt{\lambda\mu} - (\lambda+\mu)$			
T		10	100	1000	10000
precise value		9.23E-3	1.09E-4	1.52E-10	1.31E-55
CME(30)	0	9.25E-3	1.11E-4	4.19E-7	8.09E-8
CME(1000)	0	9.23E-3	1.09E-4	1.18E-9	1.71E-10
Euler(30)	0	9.23E-6	1.09E-4	1.52E-10	-6.99E-20
CME(30)	-0.0111	9.23E-3	1.09E-4	1.52E-10	1.31E-55
CME(1000)	-0.0111	9.23E-3	1.09E-4	1.52E-10	1.31E-55
Euler(30)	-0.0111	9.23E-6	1.09E-4	1.52E-10	1.31E-55
method	θ	$h(t) = t\exp(-t)$ $h^*(s) = (1+s)^{-2}, a = -1$			
T		10	100	1000	10000
precise value		4.54E-3	3.72E-42	5.08E-432	1.14E-4339
CME(30)	0	4.67E-3	7.75E-7	2.26E-7	8.54E-8
CME(1000)	0	4.54E-3	4.22E-9	9.49E-10	1.64E-10
Euler(30)	0	4.54E-3	1.90E-13	-6.62E-27	-1.65E-41
CME(30)	-1	4.54E-3	3.72E-42	5.08E-432	1.14E-4339
CME(1000)	-1	4.54E-3	3.72E-42	5.08E-432	1.14E-4339
Euler(30)	-1	4.54E-3	3.72E-42	5.08E-432	1.14E-4339
method	θ	$h(t) = \exp(-t-\sqrt{t})$ $h^*(s) = \frac{1}{1+s} - \frac{e^{\frac{1}{4+4s}}\sqrt{\pi}\mathrm{Erfc}(1/(2\sqrt{1+s}))}{2(1+s)^{3/2}}, a = -1$			
T		10	100	1000	10000
precise value		1.92E-6	1.69E-48	9.37E-449	4.22E-4387
CME(30)	0	5.20E-6	8.78E-7	2.83E-7	4.40E-8
CME(1000)	0	1.93E-6	3.50E-9	5.67E-8	9.03E-11
Euler(30)	0	1.92E-6	-4.85E-16	-1.46E-29	-2.68E-44
CME(30)	-1	1.92E-6	1.76E-48	2.59E-441	1.55E-4350
CME(1000)	-1	1.92E-6	1.69E-48	8.70E-444	3.12E-4353
Euler(30)	-1	1.92E-6	1.69E-48	6.47E-449	-1.56E-4367

Table 5: Effect of shifting the abscissa

sums up the results with some weights. The only potentially time consuming ingredient of the procedure is the Laplace transform function evaluation. In our numerical experience all Laplace functions were explicit with negligible evalua-

$h(t_1,t_2)$	$\mathbb{1}(t_1+t_2<1)$	$\min(t_1,t_2)$
$h^*(s_1,s_2)$	$\frac{s_1(1-e^{-s_2})-s_2(1-e^{-s_1})}{s_1^2s_2-s_1s_2^2}$	$\frac{1}{s_1s_2(s_1+s_2)}$

Table 6: The 2-dimensional test functions

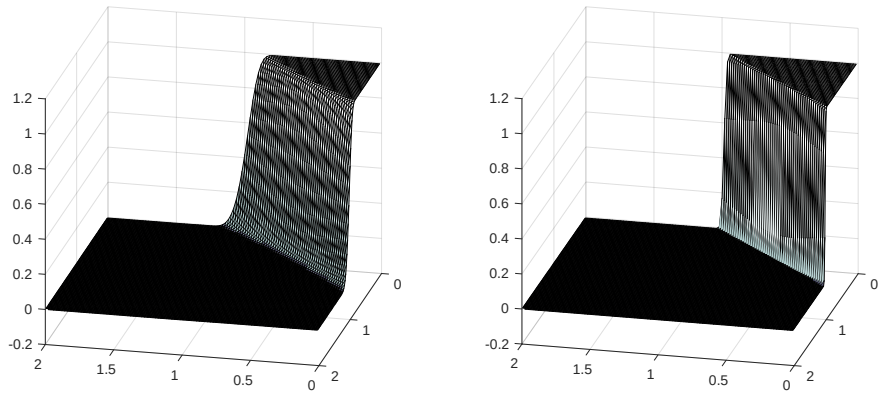

Figure 14: $\mathbb{1}(t_1+t_2<1)$ with CME method for order 10 and 50

tion time, which resulted in negligible evaluation time for the ILT methods.

6. Conclusion

This paper presents a widespread comparative analysis of the most frequently used numerical ILT methods, and concludes that the CME method, is superior to all existing methods for general purpose use with machine precision floating point arithmetic, in spite of the fact that alternative methods can be more accurate for a limited number of special functions (e.g., the sine function).

The CME method provides good results for any type of function already at relatively low order, with the error decreasing as the order is increased up to order 1000. For smooth functions, many of the existing methods also perform well, but for functions with discontinuities, the CME method outperforms all competitors with desirable properties such as the lack of Gibbs oscillation and numerically stable and precise with machine precision arithmetic.

The Matlab, Wolfram Mathematica and IPython implementations of the CME method is available online at [15].

Appendix A. The effect of SCV on the quality of the approximation

Theorem 4. *When f_n is non-negative with $\int_{t=0}^{\infty} f_n(t)\mathrm{d}t = \int_{t=0}^{\infty} t f_n(t)\mathrm{d}t = 1$ and h is bounded and Lipschitz-continuous at T, i.e.,*

$$|h(t)| \le H \quad \textit{and} \quad |h(T)-h(t)| \le L|T-t| \quad \forall t \ge 0,$$

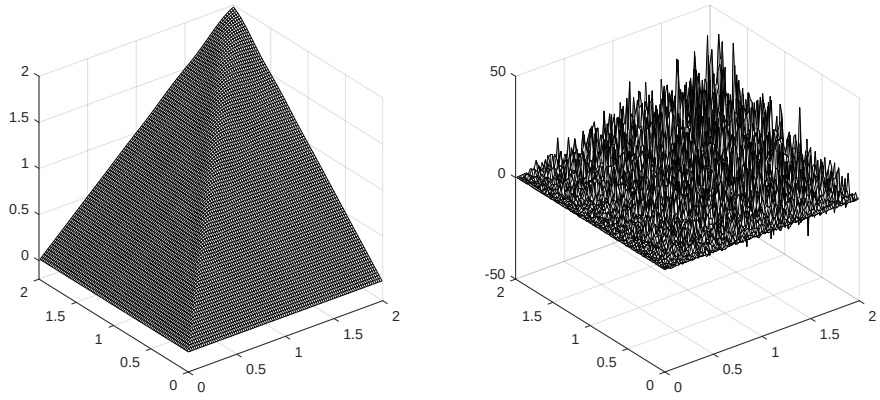

Figure 15: $\min(t_1, t_2)$ with Euler method for order 11 and 61

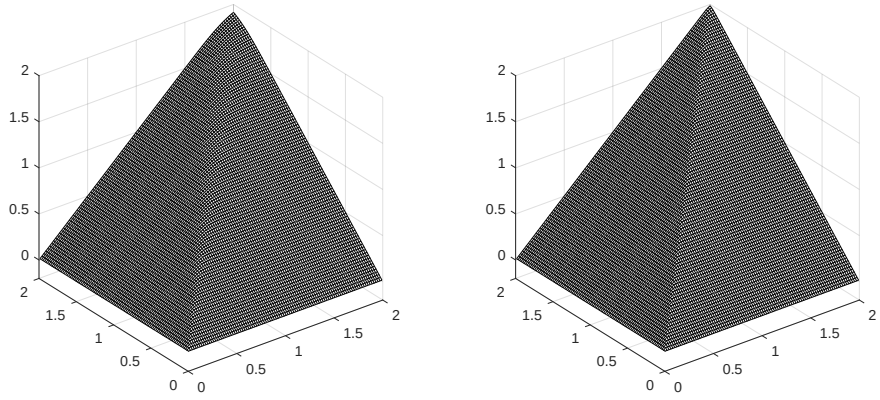

Figure 16: $\min(t_1, t_2)$ with CME method for order 10 and 60

the error of the approximation is bounded by

$$|h_n(T) - h(T)| \leq c \cdot (\mathrm{SCV}(f_n))^{1/3}, \tag{A.1}$$

where $c = 3(2HL^2T^2)^{1/3}$.

Proof.

$$\begin{aligned} |h_n(T) - h(T)| &= \left| \int_0^\infty h(t) \cdot \frac{1}{T} f_n(t/T) \mathrm{d}t - \int_0^\infty h(T) \cdot \frac{1}{T} f_n(t/T) \mathrm{d}t \right| \\ &= \left| \int_0^\infty h(Ts) \cdot f_n(s) \mathrm{d}s - \int_0^\infty h(T) \cdot f_n(s) \mathrm{d}s \right| \\ &= \left| \int_0^\infty (h(Ts) - h(T)) \cdot f_n(s) \mathrm{d}s \right| \leq A + B, \end{aligned}$$

where

$$A = \left| \int_0^{1-\varepsilon} (h(Ts) - h(T)) \cdot f_n(s) \mathrm{d}s \right| + \left| \int_{1+\varepsilon}^{\infty} (h(Ts) - h(T)) \cdot f_n(s) \mathrm{d}s \right|,$$
$$B = \left| \int_{1-\varepsilon}^{1+\varepsilon} (h(Ts) - h(T)) \cdot f_n(s) \mathrm{d}s \right|,$$
$$\varepsilon = \left(\frac{2H \cdot \mathrm{SCV}(f_n)}{LT} \right)^{1/3}.$$

We estimate A using the bound H and B using the Lipschitz coefficient L.

$$\begin{aligned} A &\leq \int_0^{1-\varepsilon} |(h(Ts) - h(T)) \cdot f_n(s)| \mathrm{d}s + \int_{1+\varepsilon}^{\infty} |(h(Ts) - h(T)) \cdot f_n(s)| \mathrm{d}s \\ &\leq 2H \left(\int_0^{1-\varepsilon} f_n(s) \mathrm{d}s + \int_{1+\varepsilon}^{\infty} f_n(s) \mathrm{d}s \right). \end{aligned}$$

The term in the bracket can be estimated by Chebyshev's inequality:

$$\int_0^{1-\varepsilon} f_n(s) \mathrm{d}s + \int_{1+\varepsilon}^{\infty} f_n(s) \mathrm{d}s \leq \frac{\mathrm{SCV}(f_n)}{\varepsilon^2},$$

where we used that $\int_{t=0}^{\infty} t f_n(t) \mathrm{d}t = 1$. For A, it gives

$$A \leq \frac{2H \cdot \mathrm{SCV}(f_n)}{\varepsilon^2}. \tag{A.2}$$

For B, we have

$$B = \left| \int_{1-\varepsilon}^{1+\varepsilon} (h(Ts) - h(T)) \cdot f_n(s) \mathrm{d}s \right| \leq \left| \int_{1-\varepsilon}^{1+\varepsilon} 2LT\varepsilon \cdot f_n(s) \mathrm{d}s \right| \leq 2LT\varepsilon. \tag{A.3}$$

Putting (A.2) and (A.3) together, we have

$$|h_n(T) - h(T)| \leq A + B \leq \frac{2H \cdot \mathrm{SCV}(f_n)}{\varepsilon^2} + 2LT\varepsilon,$$

which gives (A.1). □

An obvious consequence of Theorem 4 is the following property. If $\mathrm{SCV}(f_n) \to 0$ as $n \to \infty$, then

$$\lim_{n \to \infty} h_n(T) = h(T). \tag{A.4}$$

Finally, we remark that (A.1) is typically far from being sharp due to (A.2) being a very rough estimate.

Appendix B. Heuristic test for the error

In general, calculating the error of an ILT method (the distance between $h(t)$ and $h_n(t)$, where $h_n(t)$ is computed from $h^*(s)$ by the given ILT method), or at least a bound for $|h(t) - h_n(t)|$, is challenging. In this section we present a heuristic algorithm, for the Euler, Gaver and CME methods, to approximate the potential numerical error in the ILT as a function of $h^*(s), f_n(t)$ and T, assuming that $\lim_{t\to\infty} h(t) = \lim_{s\to 0} sh^*(s) = 0$.

For the Euler, Gaver and CME methods, the $f_n(t)$ functions have several zeros on both sides of the peak at 1 (see Figure 1). For a given method and order the zeros of $f_n(t)$ are available at least numerically (similar to the nodes β_k and weights η_k the zeros can also be pre-computed and stored as they do not depend on the function to transform). Let $\check{I}$ and $\hat{I}$ denote the number of zeros of $f_n(t)$ in $(0,1)$ and in $(1,\infty)$, and $1 > x_{-1} > x_{-2} > \ldots > x_{-\check{I}}$ and $1 < x_1 < x_2 < \ldots < x_{\hat{I}}$ the zeros of $f_n(t)$ in $(0,1)$ and in $(1,\infty)$, respectively.

The main steps of the error approximation algorithm are:

1. Compute $h_n(T)$.
2. For $i = -\check{I}, \ldots, -1$, compute
$$\tilde{c}_i = \frac{x_{-i} - x_{-i-1}}{2} h_n\left(\frac{x_{-i} + x_{-i-1}}{2} T\right) f_n\left(\frac{x_{-i} + x_{-i-1}}{2}\right). \tag{B.1}$$
3. If $|\sum_{i=-\check{I}}^{-1} \tilde{c}_i| < |\varepsilon h_n(T)|$, then the approximation $h_n(T) \approx h(T)$ is accurate with relative error in the order of ε.

The proposed heuristic algorithm is based on the following intuitive reasoning. Using the zeros of $f_n(t)$, (7) can be rewritten as

$$\begin{aligned} h_n(T) = &\underbrace{\int_{x_{-1}}^{x_1} h(tT) \cdot f_n(t)\mathrm{d}t}_{c_0} + \sum_{i=1}^{\check{I}} \underbrace{\int_{x_{-i-1}}^{x_{-i}} h(tT) \cdot f_n(t)\mathrm{d}t}_{c_{-i}} \\ &+ \sum_{i=1}^{\hat{I}} \underbrace{\int_{x_i}^{x_{i+1}} h(tT) \cdot f_n(t)\mathrm{d}t}_{c_i} = \sum_{i=-\check{I}}^{\hat{I}} c_i, \end{aligned} \tag{B.2}$$

where $x_{-\check{I}-1} = 0$ (if $f_n(t) \neq 0$) and $x_{\hat{I}+1} = \infty$ (if $\hat{I} < \infty$). For the error of the approximation, we have

$$|h(T) - h_n(T)| \leq |h(T) - c_0| + |c_0 - h_n(T)|. \tag{B.3}$$

When the order n is high, and the peak of $f_n(t)$ at 1 approximates the Dirac delta function well enough, the term c_0 will be close to $h(T)$, thus most of the error comes from the second term on the right hand side of (B.3). This term can be written as

$$|c_0 - h_n(T)| = \left| \sum_{i=-\check{I}}^{-1} c_i + \sum_{i=1}^{\hat{I}} c_i \right|.$$

T	$h_n(T)$	$h(T)$	$\sum_{i=-\check{I}}^{-1} \tilde{c}_i/h_n(T)$	$(h(T)-h_n(T))/h(T)$
10	0.04234	0.04233	5.54E-3	2.95E-4
40	1.794E-3	1.792E-3	6.77E-3	1.35E-3
60	4.337E-4	4.325E-4	8.54E-3	2.73E-3
69	2.478E-4	2.469E-4	9.90E-3	3.76E-3
70	2.334E-4	2.325E-4	1.01E-2	3.89E-3
95	5.905E-5	5.848E-5	1.82E-2	9.79E-3
96	5.613E-5	5.556E-5	1.87E-2	1.01E-2

Table B.7: Heuristic method for $h(t) = \exp(-\sqrt{t})$, CME method for order $n = 50$ and $\varepsilon = 0.01$

The algorithm also assumes that $\sum_{i=1}^{\hat{I}} c_i$ is negligible for a decaying $h(t)$, and checks the ratio of the potentially tangible error terms $\sum_{i=-\check{I}}^{-1} c_i$ with $h_n(T)$, such that $\tilde{c}_i$, a computationally inexpensive approximation of c_i, is used instead of c_i (which essentially corresponds to approximating each peak with a symmetric triangle).

Assuming that $h(t)$ is non-negative, the main difference between the Euler and Gaver methods and the CME method comes from the fact that the $f_n(t)$ functions of the Euler and Gaver methods have alternating sign between the zeros, while the $f_n(t)$ function of the CME method is non-negative. As a result, the error terms have alternating signs for the Euler and Gaver methods, and $|\sum_{i=-\check{I}}^{-1} \tilde{c}_i| < |\varepsilon h_n(T)|$ might hold even when $\sum_{i=-\check{I}}^{-1} |\tilde{c}_i| \gg |\varepsilon h_n(T)|$, while in case of the CME method, all c_i (and also all $\tilde{c}_i$) are non-negative and $\left|\sum_{i=-\check{I}}^{-1} \tilde{c}_i\right| = \sum_{i=-\check{I}}^{-1} |\tilde{c}_i|$.

Table B.7 exemplifies the application of the heuristic method for $h(t) = \exp(-\sqrt{t})$ using order 50 CME ILT, whose $f_n(t)$ has $\check{I} = 47$ numerically computable zeros in $(0, 1)$. Columns 2 and 4 of Table B.7 are computed from $h^*(s)$, while columns 3 and 5 are based on the true value of $h(t)$. Column 4 contains the estimated relative error $\sum_{i=-\check{I}}^{-1} \tilde{c}_i/h_n(T)$ computed by the heuristic procedure and column 5 contains the actual relative error $(h(T) - h_n(T))/h(T)$.

When the accuracy requirement is $\varepsilon = 0.01$, the heuristic procedure finds the ILT accurate up to $T = 69$. The actual relative error (column 5) is somewhat smaller than the estimated relative error (column 4) for all values of T in this example.

[1] K. Reddy, K. Kumar, J. Satish, S. Vaithyasubramanian, A review on applications of Laplace transformations in various fields, Journal of Advanced Research in Dynamical and Control Systems 9 (2017) 14–24.

[2] K. L. Kuhlman, Review of inverse Laplace transform algorithms for Laplace-space numerical approaches, Numerical Algorithms 63 (2) (2013) 339–355. doi:10.1007/s11075-012-9625-3.

[3] Q. Wang, H. Zhan, On different numerical inverse Laplace methods for solute transport problems, Advances in Water Resources 75 (2015) 80 – 92.

[4] J. Abate, G. L. Choudhury, W. Whitt, An introduction to numerical transform inversion and its application to probability models, in: Computational Probability, Springer US, Boston, MA, 2000, pp. 257–323. doi:10.1007/978-1-4757-4828-4.

[5] J. Valsa, L. Brancik, Approximate formulae for numerical inversion of Laplace transforms, International Journal of Numerical Modelling-electronic Networks Devices and Fields - INT J NUMER MODEL ELECTRON N 11 (1998) 153–166.

[6] N. A.-Z. Smith, L. Brancik, Comparative Study on One-Dimensional Numerical Inverse Laplace Transform Methods for Electrical Engineering, ISSN 1213-1539 18 (1) (2016) 8.

[7] J. Abate, W. Whitt, A Unified Framework for Numerically Inverting Laplace Transforms, INFORMS Journal on Computing 18 (4) (2006) 408–421.

[8] I. Horváth, Z. Talyigás, M. Telek, An optimal inverse Laplace transform method without positive and negative overshoot – an integral based interpretation, Electronic Notes in Theoretical Computer Science 337 (2018) 87 – 104.

[9] G. Horváth, I. Horváth, M. Telek, High order low variance matrix-exponential distributions, The Tenth International Conference on Matrix-Analytic Methods in Stochastic Models (MAM10).

[10] G. Doetsch, L. Debnath, Introduction to the Theory and Application of the Laplace Transformation, Springer, 1974. doi:10.1007/978-3-642-65690-3.

[11] M. Vetterli, J. Kovacevic, V. K. Goyal, Foundations of Signal Processing, Cambridge University Press, 2014.

[12] S. Asmussen, M. Bladt, Renewal theory and queueing algorithms for matrix-exponential distributions, Matrix-Analytic Methods in Stochastic Modelsdoi:10.1201/b17050-17.

[13] P. Reinecke, M. Telek, Does a given vector-matrix pair correspond to a PH distribution?, Performance Evaluation 80 (0) (2014) 40 – 51.

[14] I. Horváth, O. Sáfár, M. Telek, B. Zámbó, Concentrated Matrix Exponential Distributions, in: Computer Performance Engineering: 13th European Workshop, EPEW 2016, Chios, Greece, October 5-7, 2016, Proceedings, Springer International Publishing, Cham, 2016, pp. 18–31.

[15] inverselaplace.org, http://inverselaplace.org/, [Online; accessed 22-May-2019].

[16] D. P. Gaver, Observing stochastic processes and approximate transform inversion, Oper. Res. 14 (1966) 444–459.

[17] H. Stehfest, Algorithm 368: Numerical Inversion of Laplace Transforms [D5], Commun. ACM 13 (1) (1970) 47–49. doi:10.1145/361953.361969.

[18] A. Talbot, The Accurate Numerical Inversion of Laplace Transforms, IMA Journal of Applied Mathematics 23 (1) (1979) 97–120. doi:10.1093/imamat/23.1.97.

[19] P. Valko, J. Abate, Comparison of sequence accelerators for the Gaver method of numerical Laplace transform inversion, Comput. Math. Appl. 48 (2004) 629–636.

[20] V. Zakian, Numerical inversion of Laplace transform, Electronics Letters 5 (1969) 120 – 121. doi:10.1049/el:19690090.

[21] V. Zakian, Optimisation of numerical inversion of Laplace transforms, Electronics Letters 6 (21) (1970) 677–679. doi:10.1049/el:19700471.

[22] V. Zakian, Properties of i_{MN} approximants, in: Pad Approximants and Their Applications, Academic Press, New York, 1973, pp. 141–144.

[23] C. J. Wellekens, Generalisation of Vlach's method for the numerical inversion of the Laplace transform, Electronics Letters 6 (23) (1970) 742–744. doi:10.1049/el:19700514.

[24] L. Brancik, N. Smith, Two approaches to derive approximate formulae of NILT method with generalization, in: 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015, pp. 155–160. doi:10.1109/MIPRO.2015.7160256.

[25] A. M. Cohen, Numerical Methods for Laplace Transform Inversion, Springer, 2007. doi:10.1007/978-0-387-68855-8.

[26] D. L. Jagerman, An inversion technique for the laplace transform with Application to approximation, The Bell System Technical Journal 57 (3) (1978) 669–710. doi:10.1002/j.1538-7305.1978.tb00601.x.

[27] D. L. Jagerman, An Inversion Technique for the Laplace Transform, The Bell System Technical Journal 61 (1982) 1995–2002. doi:10.1002/j.1538-7305.1982.tb03096.x.

[28] J. Abate, W. Whitt, The Fourier-series method for inverting transforms of probability distributions, Queueing Systems 10 (1) (1992) 5–87. doi:10.1007/BF01158520.

[29] S. Asmussen, F. Avram, M. Usabel, Erlangian Approximations for Finite-Horizon Ruin Probabilities, ASTIN Bulletin 32 (2) (2002) 267281. doi:10.2143/AST.32.2.1029.

[30] J. Abate, G. L. Choudhury, W. Whitt, On the Laguerre method for numerically inverting Laplace transforms, INFORMS Journal on Computing 8 (4) (1996) 413–427. doi:10.1287/ijoc.8.4.413.

[31] M. Moorthy, Numerical inversion of two-dimensional Laplace transforms–Fourier series representation, Applied Numerical Mathematics 17 (2) (1995) 119 – 127. doi:10.1016/0168-9274(95)00015-M.

[32] K. S. Crump, Numerical inversion of Laplace transforms using a Fourier series approximation, J. ACM 23 (1) (1976) 89–96. doi:10.1145/321921.321931.

[33] P. Valko, J. Abate, Numerical inversion of 2-D Laplace transforms applied to fractional diffusion equations, Applied Numerical Mathematics 53 (1) (2005) 73 – 88. doi:10.1016/j.apnum.2004.10.002.

[34] J. Abate, G. L. Choudhury, W. Whitt, Numerical inversion of multidimensional Laplace transforms by the Laguerre method, Performance Evaluation 31 (3) (1998) 229 – 243. doi:10.1016/S0166-5316(97)00002-3.

[35] N. A.-Z. Smith, L. Brancik, On two-dimensional numerical inverse Laplace transforms with transmission line applications, 2016 Progress in Electromagnetic Research Symposium (PIERS) (2016) 227–231.

[36] N. Al-Zubaidi, Accuracy comparison of some 2D numerical inverse Laplace transform methods, in: 22nd EEICT Student Conference, 2016, pp. 282–287.

[37] P. Kowalczyk, GRPF: global complex roots and poles finding algorithm based on phase analysis, CoRR abs/1806.06522.
URL http://arxiv.org/abs/1806.06522

[38] L. Kleinrock, Queueing Systems, Wiley-Interscience, New York, NY, USA, 1975.