Numerical Laplace Transform Inversion and Selected Applications

Patrick O. Kano, Ph.D.

March 5, 2010

Applied Energetics

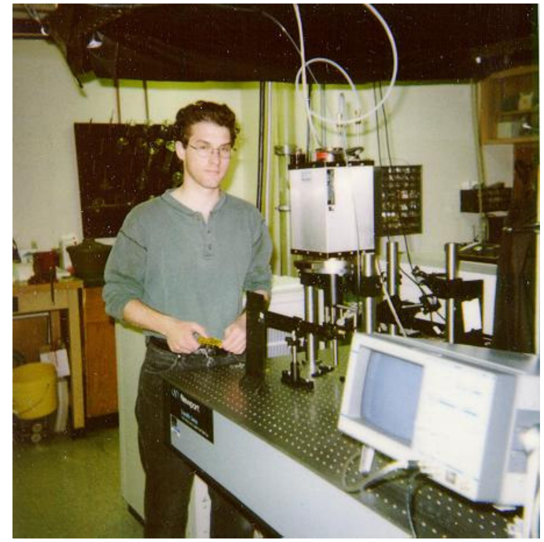
Outline

- The talk is organized as follows:
  1. Basic definitions and analytic inversion
  2. Issues in numerical Laplace transform inversion
  3. Introduce three of the most commonly known numerical inversion procedures
     1. Talbot's Method
     2. Weeks' Method
     3. Post's Formula
  4. Illustrate through applications
     1. Pulse propagation in dispersive materials
     2. Calculation of the matrix exponential
  5. Future directions and conclusions

Overview

Contributions

Laplace Transform Definition



A Laplace transform is a tool to make a difficult problem into a simpler one.

Difficult Time Dependant Problem → Solve Simpler Laplace Space Problem → Time Dependant Solution

$$E\left[f(t)\right]=0 \xrightarrow{L} F\left(s\right) \xrightarrow{L^{-1}} f(t)$$

?

- Laplace transform solution methods are a standard of mathematics, physics, and engineering undergraduate education.
- Textbook examples however utilize known Laplace transform pairs.

Laplace Transform Definitions

$$F(s) = L\big(f(t)\big)(s) = \int_0^{\infty} e^{-st} f(t)\,dt \qquad \text{for all } s \text{ where } \sigma < \mathrm{Re}(s)$$

- A sufficient existence condition is that *f(t)* be
  - piecewise continuous for nonnegative values of *t*
  - of exponential order

There exist nonnegative σ and M, such that $|f(t)| \le Me^{\sigma t}$ for all $0 \le t$

- Intuitively, the Laplace transform can be viewed as the continuous analog to a power series.

No Transform $f(t) = e^{t^2}$

$$\sum_{n=0}^{\infty} a_n x^n \xrightarrow[\substack{\sum \to \int \\ n \to t \\ a_n \to f(t) \\ x = e^{-s}}]{} \int_0^{\infty} f(t) e^{-st}\,dt$$

Laplace Transform Inversion

- How does one return from the Laplace space representation to the time domain?

Richard Bellman

We can alleviate some of the suspense at the very beginning by cheerfully confessing that there is no *single* answer to this question.

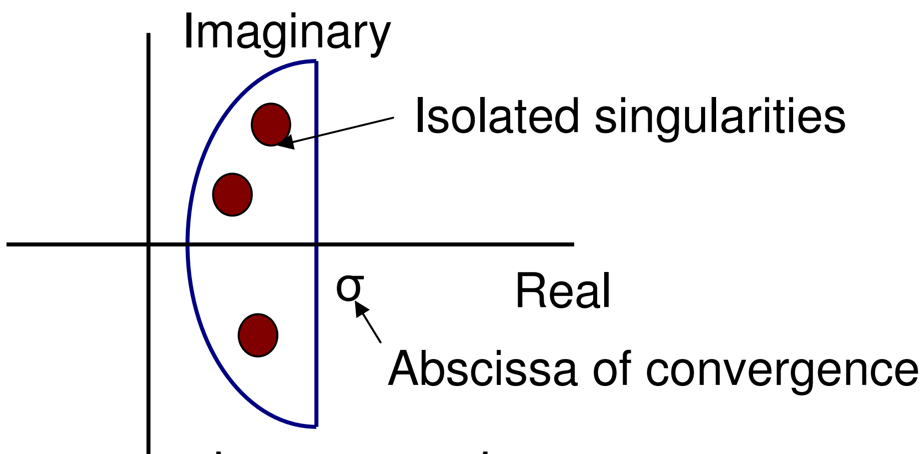
Instead, there are many particular methods geared to appropriate situations.

This is the usual situation in mathematics and science and, hardly necessary to add, a very fortunate situation for the brotherhood.

Analytic Inversion

- The analytic inversion of the Laplace transform is a well-known application of the theory of complex variables.
- For isolated singularities, the Bromwich contour is the standard approach.

$$f(t) = \frac{1}{2\pi i} \int_{\sigma - i\infty}^{\sigma + i\infty} F(s) e^{st} ds$$



- Laplace transform inversion is not a unique operation.
- In practice, one can assume that the analytic inverse is well-defined.

if two function $f_1(t)$ and $f_2(t)$ have the same Laplace transform

$F(s) \equiv L(f_1) = L(f_2)$

then, for all $\tau > 0$, $\int_0^{\tau} f_1(t) - f_2(t) dt = 0$

Lerch's theorem

Numerical Inversion Issues

- The numerical inversion of the Laplace transform is an inherently ill-posed problem.

Inherent sensitivity due to the multiplication by a exponential function of time.

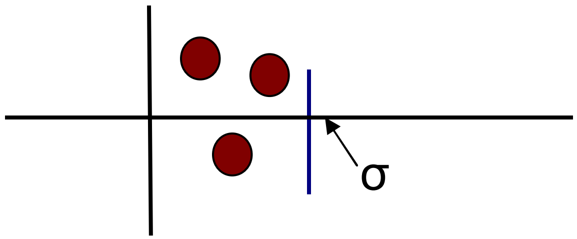
$$f(t) = \frac{1}{2\pi i} \int_{\Lambda} F(s) e^{st} ds$$

Algorithmic and finite precision errors can lead to exponential divergence of numerical solutions.

- To combat these numerical issues one may use 2 tactics
  1. Fixed-point high precision variables.
  2. Use of multiple algorithms, each with efficacy for certain classes of functions.

Fixed-Point High Precision Variables

- High precision variables are required for most inversion methods.



Double Precision: 10^{-308}-10^{308}

$10^{308} \sim e^{709}$ ← Large for even modest times.

- This requirement is important consequences:
  - Numerical LT methods are typically slower than other time-propagation methods.
  - Implementation requires either
    - An environment where high precision variables are innate.
    - Additional high precision variable software packages.

- **Mathematica**
- **ARPREC**
  - An **Ar**bitrary **Pre**cision **C**omputation Package
  - Lawrence Berkley National Laboratory
  - D. Bailey, Y. Hida, X. Li, B. Thompson

GMP «Arithmetic without limitations»

- **GMP**
  - **G**NU **M**ultiple **P**recision Arithmetic Library
- **MP**
  - Matlab Based Toolbox
  - Ben Barrows
  - Matlab file exchange

Numerical Inversion

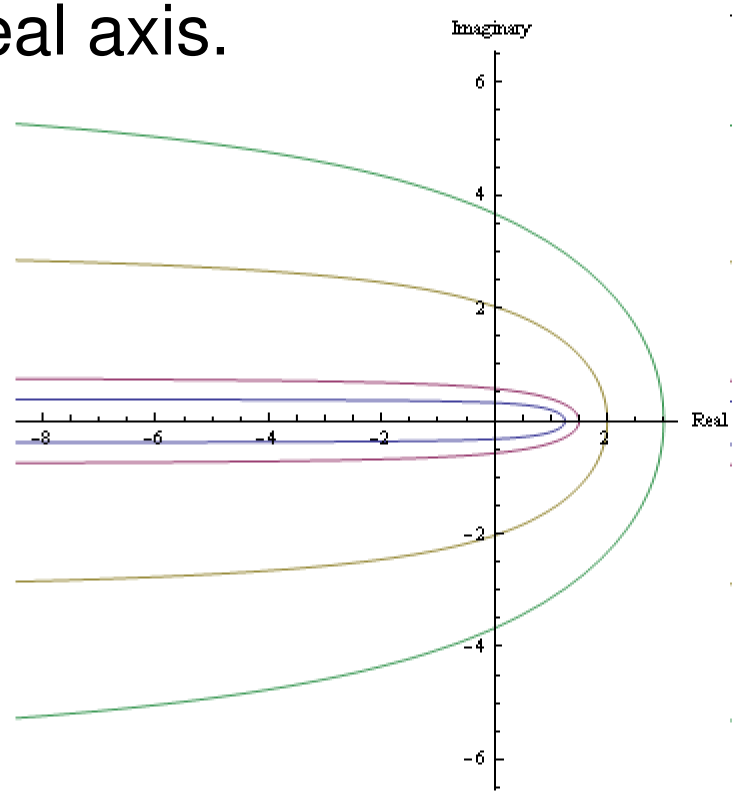
Time→						
Euler 1707-1783	Laplace 1749-1827	Heaviside 1850-1925	Post 1930	Weeks 1966	Fourier Series 1968	Talbot 1979

- Post's Formula
  - Alternative to integration; arises from Laplace's method
  - Post (1930), Gaver (1966), Valko-Abate (2004)
- The Weeks method
  - Laguerre polynomial expansion
  - Ward (1954), Weeks (1966), Weideman (1999)
- Fourier series expansion
  - Fourier related method
  - Koizumi (1935), Dubner-Abate (1968), DeHoog-Knight-Stokes (1982), D'Amore (1999)
- Talbot's method
  - Deformed contour method
  - Talbot (1979), Weideman & Trefethen (2007)

Talbot's Method (1979)

- Talbot's method is based on a deformation of the Bromwich contour.
- The idea is to replace the contour with one which opens towards the negative real axis.

$$s(\theta) = \sigma + \lambda\theta(i\nu + \cot\theta)$$
$$\lambda, \nu, \sigma \text{ are real}$$
$$-\pi < \theta < \pi$$



- Talbot's method requires that

$$F(s) \to 0 \text{ as } |s| \to \infty$$
$$\operatorname{Im}(s_j) < K \text{ for all singularities } s_j$$

Talbot's Method

- The method is easily implemented in Mathematica.

$$s(\theta) = \sigma + \lambda\theta(i\nu + \cot\theta)$$

$$\sigma = 0, \nu = 1, -\pi < \theta < \pi$$

$$\frac{ds}{d\theta} = i\lambda - \lambda[\theta + (\theta\cot\theta - 1)\cot\theta]$$

$$f(t) = \frac{1}{2\pi i}\int_{-\pi}^{\pi} e^{st} F(s)\frac{ds}{d\theta}\,d\theta$$

Precision	Run Time
10	0.047
20	0.141
40	0.391
80	1.625

The Talbot method answers are accurate up-to the computation precision for time t=1.

$$F(s) = e^{-2\sqrt{s}} \leftrightarrow f(t) = \frac{e^{-1/t}}{\sqrt{\pi t^3}}$$

$$\sigma = 0, \lambda = \tfrac{1}{2}, \nu = 1$$

```
Timeval = 1;
Rval = 1/2;

Flap[s_]=Exp[-2*Sqrt[s]];
Tfunexact[t_] =Exp[-1/t]/Sqrt[Pi*t*t*t];
Valexact = N[Tfunexact[Timeval],1000]

STalbot[r_,a_]=r*a*Cot[a]+I*r*a;
dsda[r_,a_]=I*r*(1+I*(a+Cot[a]*(a*Cot[a]-1)));
TimeDfun[r_,t_]:=1/(2*Pi*I)*NIntegrate[Exp[STalbot[r,a]*t]*Flap[STalbot[r,a]]*dsda[r,a],{a,-Pi,Pi},WorkingPrecision→20];
{Timeval,Approxval}=Timing[TimeDfun[Rval,Timeval]]
RelError = Abs[Approxval-Valexact]/Valexact
```

Talbot's Method

The primarily difficulty lies in the selection of appropriate values for the contours parameters.

$$F(s) = \frac{s}{s^2 + 2} \leftrightarrow f(t) = \cos\left(t\sqrt{2}\right)$$

$$\sigma = 0, \lambda = \tfrac{1}{2}, \nu = 1$$

Mathematica's adaptive integration fails for the same parameter values.

- Attempts have been made to automate the selection:
  - "Algorithm 682: Talbot's method of the Laplace inversion problems", Murli & Rizzardi, 1990. [FORTRAN]

- This is an active area of research.
  - *Optimizing Talbot's contours for the inversion of the Laplace transform*, A. Weideman, 2006
  - *Parabolic and Hyperbolic contours for computing the Bromwich integral*, A. Weideman & L.N. Trefethen, 2007

Post's Formula (1930)

- Emil Post's inversion procedure provides an alternative to Bromwich contour integration

$$f(t) = \lim_{q \to \infty} \frac{(-1)^q}{q!} \left(\frac{q}{t} \right)^{q+1} \left(\frac{d^q}{ds^q} F(s) \right)_{s=q/t}$$

- There are two features of Post's formula which are particularly attractive
  - It contains no parameters, save the order of the derivative and the precision of the computations.
  - The inversion is performed using
    - Only real values for *s*
    - Without priority knowledge of poles

- Post's formula manifests the same inherent ill-posedness from which all numerical inversion procedures suffer.
  - Errors are amplified → multiplicative factor grows quickly with the order of the derivative *q*
  - The method converges slowly
  - One needs an expression or approximation for the higher order derivatives of *F(s)*

Derivation

- Post's formula can be derived using Laplace's method

if $\left.\frac{dg}{d\tau}\right|_{\tau_0} = 0$ and if $\left.\frac{d^2g}{d\tau^2}\right|_{\tau_0} \neq 0$, then

as $\mathrm{k} \to \infty$, the integral

$$\mathrm{I(k)} = \int_0^\infty e^{-kg(\tau)} h(\tau) d\tau$$

has the asymptotic behaviour

$$\mathrm{I}(\mathrm{k}, \tau_0) \sim \sqrt{\frac{2\pi}{\mathrm{k}\left.\frac{d^2g}{d\tau^2}\right|_{\tau_0}}} e^{-kg(\tau_0)} h(\tau_0)$$

$$F(s) = \int_0^\infty e^{-s\tau} f(\tau) d\tau$$

Take the k^{th} derivative with respect to s

$$F^k(s) = (-1)^k \int_0^\infty \tau^k e^{-s\tau} f(\tau) d\tau$$

Rearrange and evaluate at $s = k/t$

$$(-1)^k F^k(k/t) = \int_0^\infty e^{k(\ln\tau - \tau/t)} f(\tau) d\tau$$

Approximate

Assign $\tau_0 = t$, $h(\tau) = f(\tau)$, $g(\tau) = \tau/t - \ln\tau$, $\left.\frac{\mathrm{d}^2\mathrm{g}}{\mathrm{d}\tau^2}\right|_{\tau_0} = \frac{1}{t^2}$

$$(-1)^k F^k(k/t) \sim \sqrt{\frac{2\pi t^2}{k}} e^{-k(1-\ln t)} f(t)$$

$$\frac{k}{t^{k+1}}(-1)^k F^k(k/t) \sim \sqrt{2\pi k}\, e^{-k} f(t)$$

With Stirling's formula

$$\mathrm{k}! = \sqrt{2\pi k}\, e^{-k} k^k \left(1 + \frac{1}{12k} + \frac{1}{288k^2} + \dots\right)$$

$$f(t) \sim \frac{(-1)^k}{k!}\left(\frac{k}{t}\right)^{k+1} F^k(k/t)\left(1 + \frac{1}{12k} + \frac{1}{288k^2} + \dots\right)$$

Derivatives

- Finite differences an obvious method by which to approximate the derivatives of a reasonably behaved function.

$$f_q(t) = (-1)^q q \frac{\ln(2)}{t} \binom{2q}{q} \Delta^q F\left(q \frac{\ln(2)}{t}\right)$$

$$\Delta F(nx) = F((n+1)x) - F(nx)$$

$$f_q(t) = q \frac{\ln(2)}{t} \binom{2q}{q} \sum_{j=0}^{q} (-1)^j \binom{q}{j} F\left((q+j)\frac{\ln(2)}{t}\right)$$

$$f(t) = \lim_{q \to \infty} f_q(t)$$

Gaver-Post Formula
1966

- The Gaver functionals can be computed by a recursive algorithm:

$$G_0^{(n)} = n \frac{\ln 2}{t} F\left(n \frac{\ln 2}{t}\right) \quad 1 \le \mathrm{n}$$

$$\mathrm{G}_{\mathrm{p}}^{(\mathrm{n})} = \left(1 + \frac{n}{p}\right) G_{p-1}^{(n)} - \left(\frac{n}{p}\right) G_{p-1}^{(n+1)} \quad 1 \le \mathrm{p},\ \mathrm{p} \le \mathrm{n} \quad \longrightarrow \quad f_{\mathrm{q}}(t) = G_q^q$$

Derivatives

- Post's formula does not require a finite difference approximation.
- For a particular function form, e.g. composition of two functions, a tailored method may be more robust.

Faa di Bruno's formula

$$\frac{d^q}{dx^q} f(g(x)) = \sum_{p=0}^{q} \frac{d^p f}{dx^p}(g(x)) B_{q,p}\left(\frac{dg}{dx}, \frac{d^2 g}{dx^2}, \ldots, \frac{d^{q-p+1} g}{dx^{q-p+1}}\right)$$

$$B_{q,p} = \sum_{m=1}^{q-p+1} \binom{q-1}{m-1} \frac{d^m g}{dx^m} B_{q-m,p-1}$$

$$B_{0,0} = 1$$

$$B_{q,0} = 0 \text{ for } 1 \le q$$

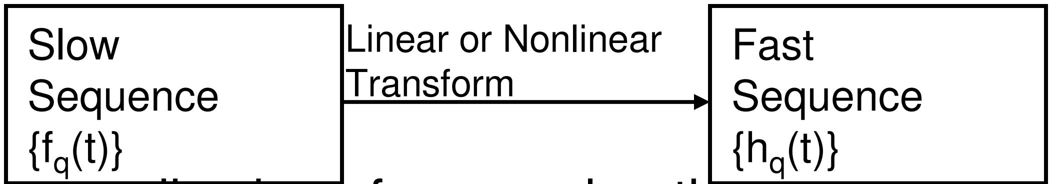
$$B_{q,1} = \frac{d^m g}{dx^m}$$

$$B_{q,q} = (g(x))^q$$

Bell Polynomials of the Second Kind
www.mathworks.com/matlabcentral/fileexchange/14483

Acceleration

- Sequence acceleration methods be used to greatly increase accuracy



- The proper application of an acceleration convergence method requires some additional knowledge about the series.

$$a_q = \frac{\left| f(t) - f_{q+1}(t) \right|}{\left| f(t) - f_q(t) \right|}$$

$$\lim_{q\to\infty} a_q = \begin{cases} 1 & \text{Logarithmic Convergence} \\ 0 < c < 1 & \text{Linear Convergence} \end{cases}$$

- Post's formula is logarithmically convergent

Gaver (1966)

$$f_q(t) \sim f(t) + \frac{c_1(t)}{q} + \frac{c_1(t)}{q^2} + \cdots$$

$$\lim_{q\to\infty} \frac{\left| f_{q+1}(t) - f(t) \right|}{\left| f_q(t) - f(t) \right|} = \lim_{q\to\infty} \frac{q}{q+1} = 1$$

Acceleration

- Wynn-rho algorithm is well suited to logarithmically convergent sequences.
- Studies have shown that it is useful for the Post formula:

Post Inversion Formula and Sequence Acceleration
UA VIGRE Project 2009
J. Cain & B. Berman

The algorithm yields an approximation for the function

$f(t) = \lim_{q\to\infty} f_q$, for even Q, by $\rho^0_{\text{Q-2}}$

$$
\begin{array}{llllll}
\rho^0_{-1}=0 & \rho^0_0 = f_0 & \rho^0_1 & \rho^0_2 & \cdots & \rho^0_{\text{Q-2}} \\
\rho^1_{-1}=0 & \rho^1_0 = f_1 & \rho^1_1 & \rho^1_2 & \cdots & \rho^1_{\text{Q-2}} \\
\rho^2_{-1}=0 & \rho^2_0 = f_2 & \rho^2_1 & \rho^2_2 & & \\
\vdots & \vdots & \vdots & \vdots & & \\
\vdots & \vdots & \vdots & \rho^{Q-3}_2 & & \\
\vdots & \vdots & \rho^{Q-2}_1 & & & \\
\rho^{Q-1}_{-1}=0 & \rho^{Q-1}_0 = f_{Q-1} & & & &
\end{array}
$$

$$\rho^q_{k+1} = \frac{k+1}{\rho^{q+1}_k - \rho^q_k} + \rho^{q+1}_{k-1}$$

$$k = 0,\ldots,Q-3$$

$$q = 0,\ldots,Q-1-k$$

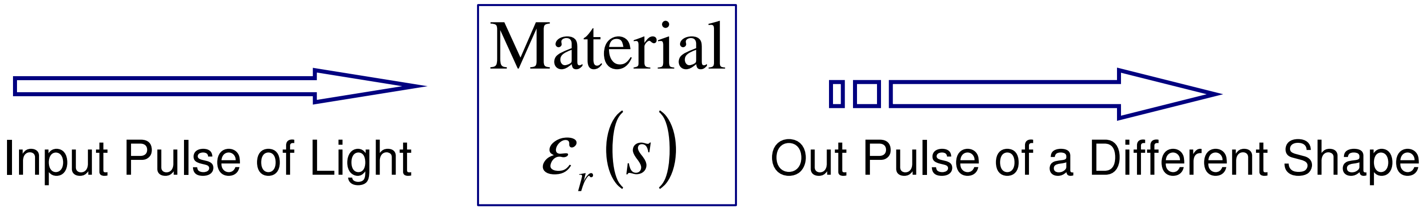
"AlternatingSigns"	method for summands with alternating signs
"EulerMaclaurin"	Euler|Maclaurin summation method
"WynnEpsilon"	Wynn epsilon extrapolation method

- *NSum* in Mathematica implements these acceleration methods.

Application of Post's Formula

NSF Grant ITR-0325097
An Integrated Simulation Environment for High-Resolution Computational Methods in Electromagnetics with Biomedical Applications
Moysey Brio, et. al.

- Rapid computation of the
  - distribution of an initial optical pulse
  - in a fixed dielectric medium
  - with a nontrivial material dispersion relation.



Biological materials often have a dielectric constant which is a complex function of wavelength.

Create databases of pre-computed tables which can be used by devices which must operate in real-time.

Cole-Type Dispersion Relation

- Many real world materials can be described by a Cole-type dispersion model.

$$\varepsilon_r(s) = \varepsilon_\infty + \sum_n \frac{\delta\varepsilon_n}{\left[1+(s\tau_n)^{1-a_n}\right]^{b_n}} + \frac{\sigma}{s\varepsilon_0}$$

$$\{a_n \in [0,1)\}$$

$$\{b_n \in (0,1]\}$$

$$\{\delta\varepsilon_n \in [0,\infty)\}$$

$$\{\tau_n \in (0,\infty)\}$$

$$\varepsilon_\infty \in (0,\infty)$$

$$\sigma \in (0,\infty)$$

Brain White Matter

$\varepsilon_\infty = 4.0$
$\sigma = 0.02$
$\{b\} = 1$

$\delta\varepsilon_1 = 32.0$
$\tau_1(ps) = 7.96$
$a_1 = 0.10$
$\delta\varepsilon_2 = 100.0$
$\tau_2(ns) = 7.96$
$a_2 = 0.10$
$\delta\varepsilon_3 = 4.0\cdot 10^4$
$\tau_3(ns) = 53.05$
$a_3 = 0.30$
$\delta\varepsilon_4 = 3.5\cdot 10^7$
$\tau_4(ms) = 7.958$
$a_4 = 0.02$

Fractional *a* Coefficients

- A standard method used in computational optics is to incorporate the dispersion relation by means of an associated difference equation.
- For fractional coefficients, it is not clear how to translate into an associated equation.

Maxwell's Equations

- Maxwell's equations are the starting point for this analysis.

$$\nabla \cdot \vec{D}(\vec{x},t)=0$$

$$\nabla \cdot \vec{H}(\vec{x},t)=0$$

$$\nabla \times \vec{H}(\vec{x},t)=\frac{\partial \vec{D}}{\partial t}(\vec{x},t)$$

$$\nabla \times \vec{E}(\vec{x},t)=-\mu_0 \frac{\partial \vec{H}}{\partial t}(\vec{x},t)$$

General assumption between electric field strength $\vec{E}$ and the displacement $\vec{D}$

$$\vec{D}(\vec{x},t)=\varepsilon_0 \vec{E}(\vec{x},t)+\varepsilon_0 \int_0^t \Phi(t-\tau)\vec{E}(\vec{x},\tau)d\tau$$ ← Temporal Convolution

- In the Laplace space, the convolution and derivatives become multiplications.

$$L\left(\int_0^t f(\tau)g(t-\tau)d\tau\right)=F(s)G(s) \longrightarrow L\left(\int_0^t \vec{E}(\vec{x},\tau)\Phi(t-\tau)d\tau\right)=\phi(s)\vec{E}(\vec{x},s)$$

Maxwell's Equations

- Maxwell's equations now have a simpler form.

$$\vec{D}(\vec{x},s)=\varepsilon_0(1+\phi(s))\vec{E}(\vec{x},s)$$

$$\nabla\cdot\vec{E}(\vec{x},s)=0$$

$$\nabla\cdot\vec{H}(\vec{x},s)=0$$

$$\nabla\times\vec{H}(\vec{x},s)=s\varepsilon_0(1+\phi(s))\vec{E}(\vec{x},s)-\varepsilon_0\vec{E}(\vec{x},t=0)$$

$$\nabla\times\vec{E}(\vec{x},s)=-\mu_0\left(s\vec{H}(\vec{x},s)-\vec{H}(\vec{x},t=0)\right)$$

- Eliminating the magnetic field *H* from the problem,

$$\nabla\times(\nabla\times\vec{H})=\nabla(\nabla\cdot\vec{H})-\nabla^2\vec{H}$$

$$\nabla\times\vec{H}(\vec{x},t=0)=\varepsilon_0\frac{\partial\vec{E}}{\partial t}(\vec{x},t=0)$$

- One obtains the wave equation in Laplace space

$$\boxed{\nabla^2\vec{E}(\vec{x},s)-s^2\mu_0\varepsilon_0(1+\phi(s))\vec{E}(\vec{x},s)=-s\mu_0\varepsilon_0\vec{E}(\vec{x},t=0)-\mu_0\varepsilon_0\frac{\partial\vec{E}}{\partial t}(\vec{x},t=0)}$$

Maxwell's Equations

- One can more succinctly state this last equation as

$$\nabla^2 \vec{E}(\vec{x},s) - \frac{s^2}{c^2}\varepsilon_r(s)\vec{E}(\vec{x},s) = -\frac{1}{c^2}\vec{V}(\vec{x},s)$$

$$\vec{V}(\vec{x},s) = s\vec{E}(\vec{x},t=0) + \frac{\partial \vec{E}}{\partial t}(\vec{x},t=0)$$

$$\boxed{\varepsilon_r(s) = 1 + \phi(s)} \qquad \frac{1}{c^2} = \varepsilon_0\mu_0$$

- Applying a Fourier transform yields the desired solution in the joint space

$$\boxed{\vec{E}(\vec{k},s) = \frac{\vec{V}(\vec{k},s)}{s^2\varepsilon_r(s) + c^2\left|\vec{k}\right|^2}}$$

Database Coefficients in the Joint Space

- The solution in a dielectric medium can be characterized by one coefficient α and its time derivative.

$$\vec{E}(\vec{k},s) = \beta\left(\left|\vec{k}\right|,s\right)\vec{E}(\vec{k},t=0) + \alpha\left(\left|\vec{k}\right|,s\right)\frac{\partial}{\partial t}\vec{E}(\vec{k},t=0)$$

$$\alpha\left(\left|\vec{k}\right|,s\right) = \frac{1}{s^2\varepsilon_r(s) + c^2\left|\vec{k}\right|^2} \qquad \beta\left(\left|\vec{k}\right|,s\right) = s\alpha\left(\left|\vec{k}\right|,s\right)$$

For a given dispersion relation $\varepsilon_r(s)$, the coefficients are pre-computed and stored in a matrix of k vs time.

- Compute high order derivatives of $\alpha(k,s)$ and $\beta(k,s)$
- beta derivatives are trivially obtain from the alpha derivatives.

$$D^q\beta(k,s) = sD^q\alpha(k,s) + qD^{q-1}\alpha(k,s)$$

The crux of the problem is the arbitrary precision calculation of the q-th derivative of α .

Derivative Approaches

1. Standard Gaver-Wynn-Rho
   - Finite Differences + Wynn-Rho Acceleration
   - A brute force application entails a computation for each k and *s.*

$$\alpha_q(k,t) = q\frac{\ln(2)}{t}\binom{2q}{q}\sum_{j=0}^{q}(-1)^j\binom{n}{j}\alpha\left(k, s=(q+j)\frac{\ln(2)}{t}\right)$$

2. Gaver-Post
   - Finite Differences + Wynn-Rho Acceleration
   - The arbitrary precision computation of the dispersion relation $\varepsilon_r(s)$ is time consuming.
   - Dispersion relation is independent of *k*
   - More efficient to store $\varepsilon_r(s)$ and call for each *k* evaluation of α.

3. Bell-Post
   - Analytic Derivatives + Wynn-Rho Acceleration
   - Store $\varepsilon_r(s)$ and its derivatives.
   - Use Faa di Bruno's formula for the q^{th} derivative of the computation of two functions.

Bell-Post Method

- The problem of determining the time dependence of α(k,t) and thus the electric field is reduced to evaluating the susceptibility function and its arbitrary order derivatives.

$$\alpha(k,s)=\frac{1}{s^2\varepsilon_r(s)+c^2k^2}$$

$$\frac{d^q}{dx^q}f(g(x))=\sum_{p=0}^{q}\frac{d^p f}{dx^p}(g(x))B_{q,p}\left(\frac{dg}{dx},\frac{d^2g}{dx^2},\ldots,\frac{d^{q-p+1}g}{dx^{q-p+1}}\right)$$

$$f(s)=\frac{1}{s}$$

$$g(s)=s^2\varepsilon_r(s)+c^2k^2$$

$$D^q\alpha(k,s)=\sum_{p=0}^{q}\frac{p!(-1)^p}{\left[s^2\varepsilon_r(s)+c^2k^2\right]^{p+1}}B_{q,p}\left(Dg(s),D^2g(s),\ldots,D^{q-p+1}g(s)\right)$$

Leibniz Rule

$$D^n g(s)=s^2D^n\varepsilon_r(s)+2snD^{n-1}\varepsilon_r(s)+n(n-1)D^{n-2}\varepsilon_r(s)$$

Cole-Type Dispersion Relation

- For white brain matter the derivatives of $\varepsilon_r(s)$ can be found by using the Faa di Bruno formula.

$$\varepsilon_r(s) = \varepsilon_\infty + \sum_n \frac{\delta\varepsilon_n}{\left[1+(s\tau_n)^{1-a_n}\right]^{b_n}} + \frac{\sigma}{s\varepsilon_0}$$

$$\{a_n \in [0,1)\}$$

$$\{b_n \in (0,1]\}$$

$$\{\delta\varepsilon_n \in [0,\infty)\}$$

$$\{\tau_n \in (0,\infty)\}$$

$$\varepsilon_\infty \in (0,\infty)$$

$$\sigma \in (0,\infty)$$

$$\frac{d^q \varepsilon_r(s)}{ds^q} = \frac{\sigma}{\varepsilon_0}\frac{(-1)^q q!}{s^{q+1}} + \sum_n \delta\varepsilon_n \frac{d^q}{ds^q} f_n\left(g_n(s)\right)$$

$$g(s) = (\tau s)^{1-a}$$

$$f(s) = (1+s)^{-b}$$

$$\frac{d^q f(g(s))}{ds^q} = \sum_{k=0}^{q} f^k(g(s)) B_{q,k}\left(g^1, g^2, \ldots, g^{q-k+1}(s)\right)$$

$$g^{(p)}(s) = \left[\tau^{1-a} \prod_{j=0}^{p-1} 1-a-j\right] s^{1-a-p}$$

$$f^{(k)}(g) = \left[(-1)^k \prod_{j=0}^{k-1} b+j\right](1+g)^{-(b+k)}$$

Mathematic Implementation Flow Diagram

Inputs

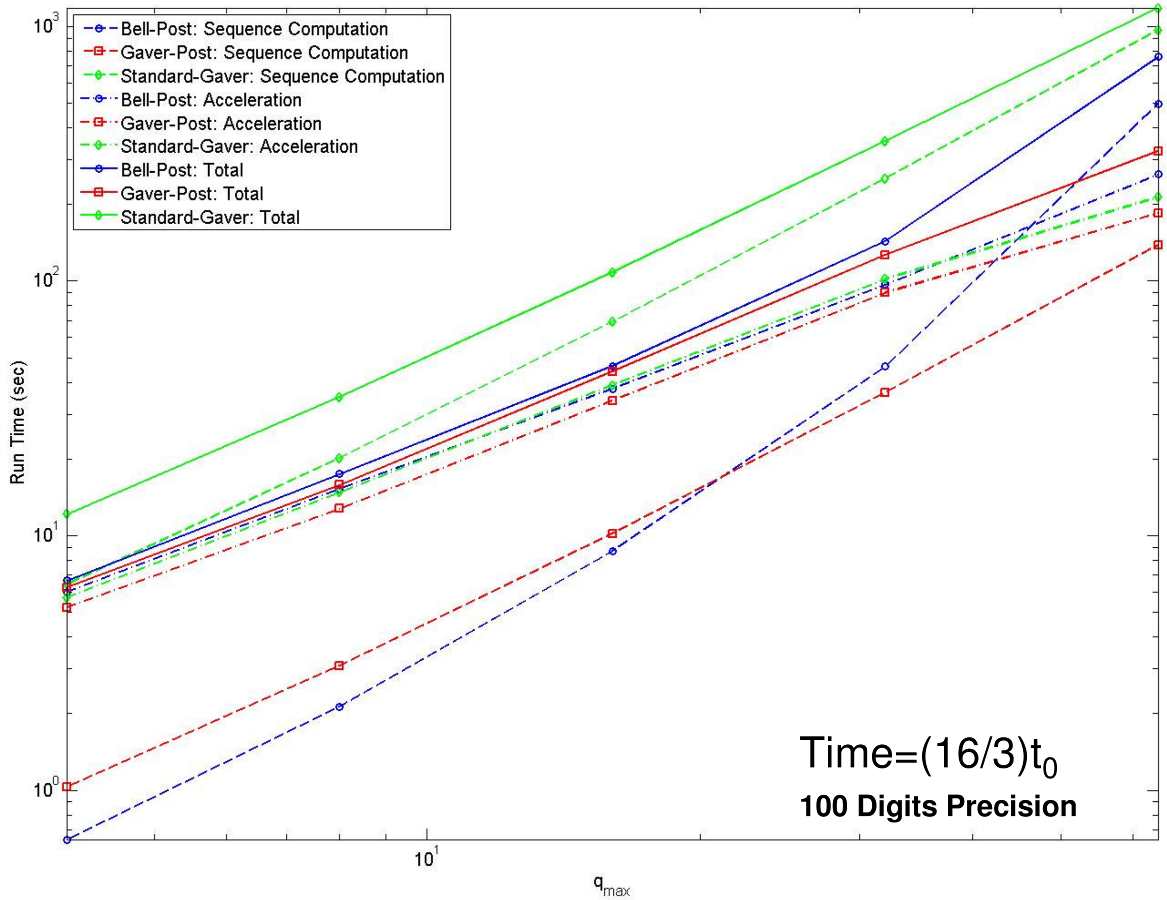
1. Choose $[q_{min}, q_{max}]$
2. Inversion time t
3. Take an explicit expression for $\varepsilon_r(s)$ and its derivatives
4. A set of wavenumber k

Repeat for each q:

- Evaluate $\varepsilon_r(s)$ and its derivatives at $s=q/t$
- Compute $s^2\varepsilon_r(s)$ and its derivatives via Leibniz's rule
- Compute the Bell polynomials from the recursion relation
- For k, compute $s^2\varepsilon_r(s) + c^2k^2$
- Compute the q^{th} and $(q-1)^{th}$ derivatives of $\alpha(k,s)$
- Compute the q^{th} derivative of $\beta(k,s)$
- Approximate the inversion coefficients via Post's formula

Apply Wynn-rho acceleration

Brain White Matter: Run Time

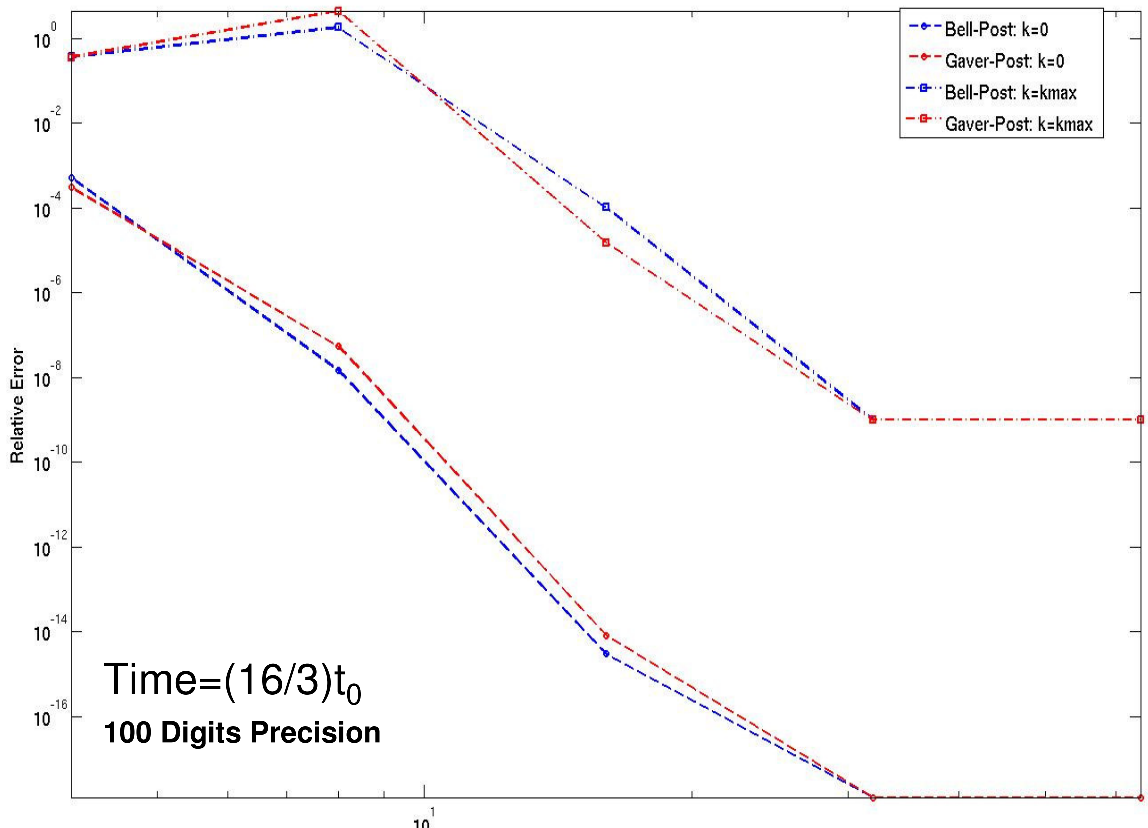


$$T = Aq^p$$

Case	Log A	p
Bell-Post	-0.592	1.670
Gaver-Post	-0.189	1.438
Brute Gaver	0.148	1.655

- The Bell-Post and Gaver-Post methods are faster than a standard Gaver.
- The acceleration dominates over the sequence computation times.
- The time follows a polynomial growth with q-max.

Brain White Matter: Accuracy



- The Bell-Post and Gaver-Post methods have comparable accuracy
  - At higher precision
  - and Post formula derivative orders.

The Weeks Method (1966)

- The Weeks' method is one of the most well known algorithms for the numerical inversion of a scalar Laplace space function.
- It popularity is due, in part, to the fact that it returns an explicit expression for the time domain function.
- The Weeks method assumes that
  - a smooth time domain function of bounded exponential growth
  - can be expressed as the limit of an expansion in scalar Laguerre polynomials.

$$f(t) = \lim_{N\to\infty} f_N(t)$$

$$f_N(t) = e^{\sigma t}\sum_{n=0}^{N-1} a_n e^{-bt} L_n(2bt)$$

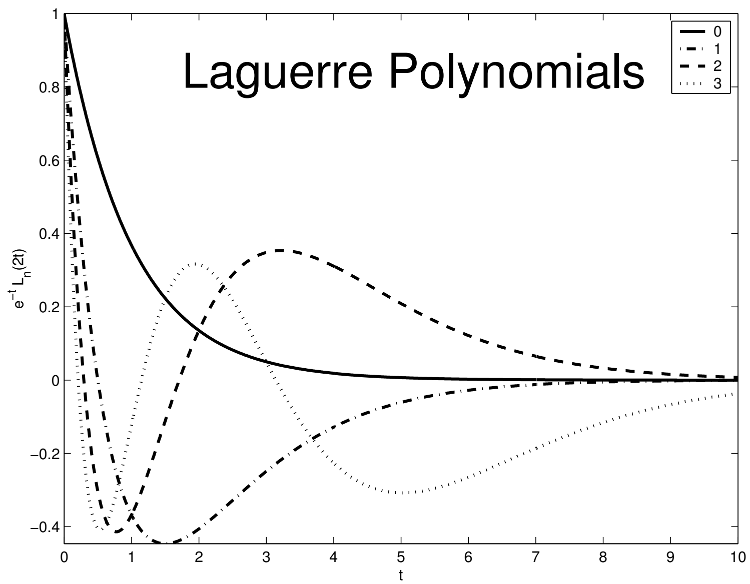
$$L_n(x) = \frac{e^x}{n!}\frac{d^n}{dx^n}\left(e^{-x}x^n\right)$$

The coefficients $\{a_n\}$

1. contain the information particular to the Laplace space function
2. may be complex scalar, vectors, or matrices
3. time independent

The Weeks Method

- Two free scaling parameters *σ* and *b*, must be selected according to the constraints that
  - $b>0$ → ensures that the Laguerre polynomials are well behaved for large t
  - $\sigma>\sigma_0$-abscissa of convergence



The Weeks Method

- The computation of the coefficients begins with a Bromwich integration in the complex plane.

$$s = \sigma + iy, \sigma > \sigma_0, -\infty < y < \infty$$

$$f(t) = \frac{e^{\sigma t}}{2\pi} \int_{-\infty}^{\infty} e^{iyt} F(\sigma + iy) dy$$

- Assume the expansion

$$f(t) = e^{\sigma t} \sum_{n=0}^{\infty} a_n e^{-bt} L_n(2bt)$$

- Equate the two expressions

$$\sum_{n=0}^{\infty} a_n e^{-bt} L_n(2bt) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{iyt} F(\sigma + iy) dy$$

Key Weeks Method Facts

- It is known that the weighted Laguerre coefficients have the Fourier representation.

$$e^{-bt}L_n(2bt) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{iyt}\frac{(iy-b)^n}{(iy+b)^{n+1}}\,dy$$

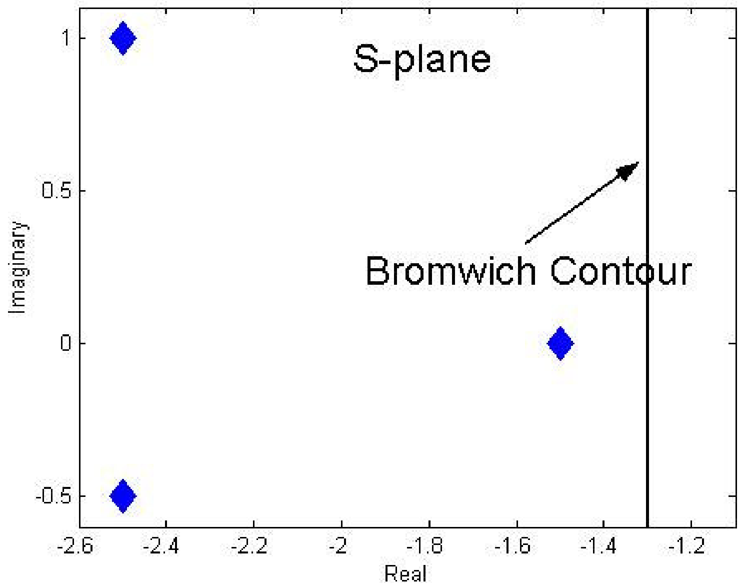
- Performing
  - the appropriate substitution,
  - assuming it is possible to interchange the sum and integral
  - equating integrands leaves

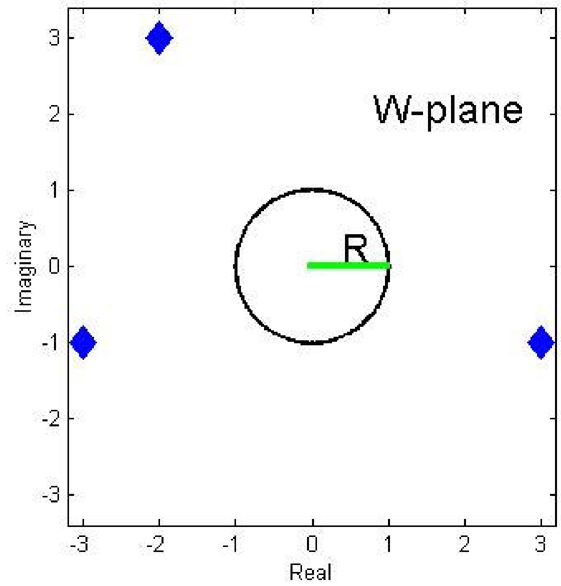
$$\sum_{n=0}^{\infty} a_n \frac{(iy-b)^n}{(iy+b)^{n+1}} = F(\sigma+iy)$$

Moebius Transformation

- One may apply a transformation from complex variable *s* to a new complex variable *w*

$$w = \frac{s-\sigma-b}{s-\sigma+b} \quad \text{or with } s=\sigma+iy\text{ , } w=\frac{iy-b}{iy+b}$$





Isolated singularities of *F(s)* in the s-half-plane are mapped to the exterior of the unit circle in the w-plane.

W-plane Representation

- With the change of variables, one obtains
  - a power series in *w*
  - whose radius of convergence is greater than 1.

$$\sum_{n=0}^{\infty} a_n w^n = \frac{2b}{1-w} F\left(\sigma - b\,\frac{w+1}{w-1}\right)$$

- The function is analytic on the unit circle.

$$a_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{-in\theta}\,\frac{2b}{1-e^{i\theta}}\,F\left(\sigma - b\,\frac{e^{i\theta}+1}{e^{i\theta}-1}\right)d\theta$$

- Numerically, the evaluation of the integral can be computed very accurately using the midpoint rule

$$a_n \approx \frac{e^{-in\pi/2M}}{2M}\sum_{m=-M}^{M} e^{-in\theta_m}\,\frac{2b}{1-e^{i\theta_{m+1/2}}}\,F\left(\sigma - b\,\frac{e^{i\theta_{m+1/2}}+1}{e^{i\theta_{m+1/2}}-1}\right) \qquad \theta_m = \frac{m\pi}{M}$$

Matrix Exponential Application

- An application of the Weeks method is to the calculation of the matrix exponential.

$$\frac{d\vec{x}}{dt} = A\vec{x} \text{ , } \vec{x}(t=0) = \vec{x}_0 \rightarrow \vec{x}(t) = e^{At}\vec{x}_0$$

What does it mean “the exponential of a matrix”?

$$e^A = I + A + \frac{A^2}{2!} + \frac{A^3}{3!} + \cdots$$

Why don’t we just calculate this?

“Nineteen Dubious Ways to Compute the Exponential of a Matrix”, SIAM Review 20, C. B. Moler & C. F. Van Loan, **1978**.

“Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later”, SIAM Review 45, C. B. Moler & C. F. Van Loan, **2003**.

- Inverse Laplace Transform (#12)

$$e^{At} = \frac{1}{2\pi i} \int_{Bromwich} [sI - A]^{-1} e^{st} ds$$

Apply the Weeks method

Resolvent Matrix of *A*

Matrix Exponential Application

- Matlab: Pade' approximation with scaling and squaring (#3)

$$e^{A} = \left(e^{A/2n} \right)^{2n}$$

$$e^{B} \approx \frac{N_{pq}(B)}{D_{pq}(B)}$$

expm

$$N_{pq}(B) = \sum_{j=0}^{p} \frac{(p+q-j)!\, p!}{(p+q)!\, j!\,(p-j)!} B^{j}$$

$$D_{pq}(B) = \sum_{j=0}^{q} \frac{(p+q-j)!\, q!}{(p+q)!\, j!\,(q-j)!} (-B)^{j}$$

- Matlab demos
  - *expmdemo1*: Pade' + Scaling + Squaring in an m-file
  - *expmdemo2*: Taylor Series
  - *expmdemo3*: Similarity Transformation

Beam Propagation Equation

- Nonparaxial scalar beam propagation equation

$$\frac{\partial u}{\partial z} = i\beta u - i\sqrt{\left(\frac{\partial^2}{\partial x^2} + k_0^2 n^2(x, y, z)\right)} u$$

u = a component of the electric field

- Discretisation in space yields a set of ODEs

$$\frac{\partial u}{\partial z} = i\beta u - i\sqrt{\left(D + k_0^2 n^2(x, y, z)\right)} u$$

- The Laplace transform in *z* yields

$$\hat{u}(s) = \left[sI - i\beta\left(I - \sqrt{I + A}\right)\right]^{-1} \vec{u}(0)$$

$$A = \frac{D + k_0^2 \overline{n}^2 - \beta^2 I}{\beta^2}$$

Beam Propagation Equation

- The Laplace space function is of a matrix exponential

$$\hat{u}(s) = [sI - M]^{-1}\vec{u}(0) \quad M = i\beta\left(I - \sqrt{I + A}\right)$$

$$e^{Mt} = \frac{1}{2\pi i}\int_{Bromwich} [sI - M]^{-1} e^{st} ds$$

Expensive Matrix Inversions

- The issue is how to pick the Laguerre polynomial parameters *σ* and *b.*
  - *Weeks' original suggestions*

$$\sigma = \max\left(0, \sigma_0 + \frac{1}{t}\right) \quad b = \frac{2N}{t} \quad N = \#\text{expansion coefficient}$$

  - *Error-Estimate Motivated Approach*
    - *Weideman Method*
      - *minimization of the error estimate as a function of* σ and *b*
    - *Min-Max Method*
      - *Maximum the radius of convergence as a function of* σ and *b*

Error Estimates

- A straight forward error estimate yields three contributions
  - Discretisation error: Finite integral sampling ← Midpoint rule on circular contour
  - Truncations error: Finite number of Laguerre polynomials
  - Round-off error: Finite computation precision

$$E_{total} \le e^{\sigma t}\left(\sqrt{\sum_{n=N}^{\infty}\left\|a_n\right\|_F^2}+\varepsilon\sqrt{\sum_{n=0}^{N-1}\left\|a_n\right\|_F^2}\right)$$

Weideman Method

Truncation   Round-Off

$$\left\|a_n\right\|_F^2 \le \frac{1}{2\pi}\int_{|w|=r}\frac{K(r)}{|w|^{n+1}}dw = \frac{K(r)}{r^n}$$

$K(r) =$ norm of the resolvent matrix

r = w - plane circular contour radius

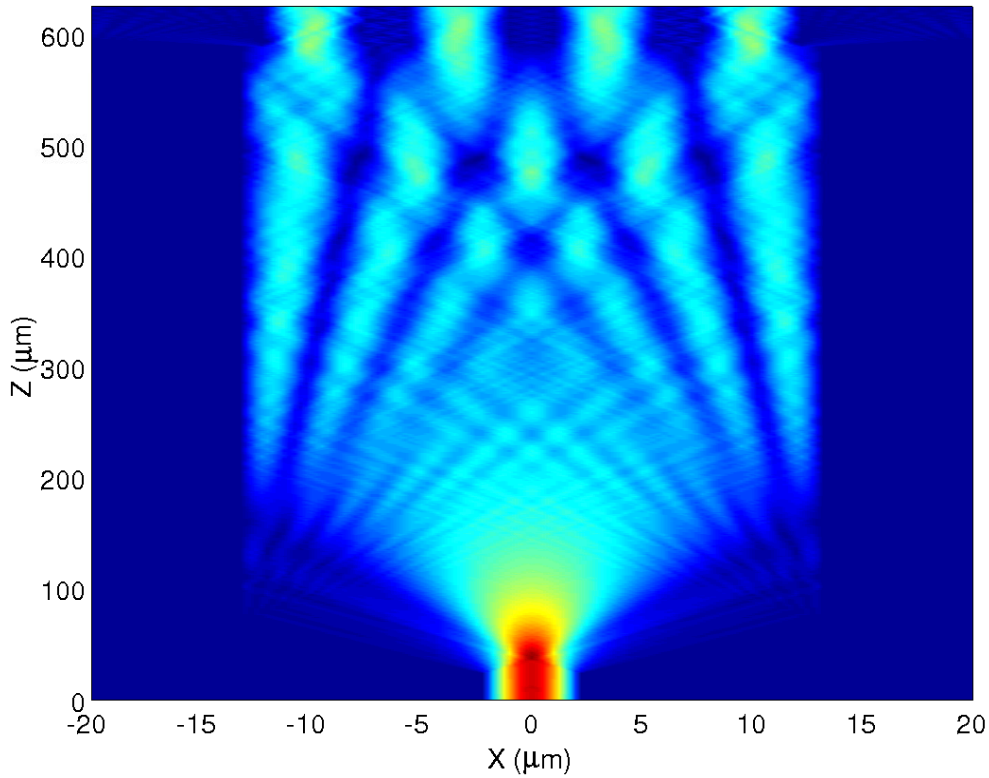
$$T_E \le \frac{K(r)}{r^{N-1}(r-1)}$$

Min-Max Method

Beam Propagation Equation Example

- Multi-mode interference coupler

	σ	b	Maximum Solution Absolute Error
Weeks	**1**	**32**	**12.28**
Min-Max	**20**	**10**	**0.00119**
Weideman	**11.84**	**16.79**	**0.000425**

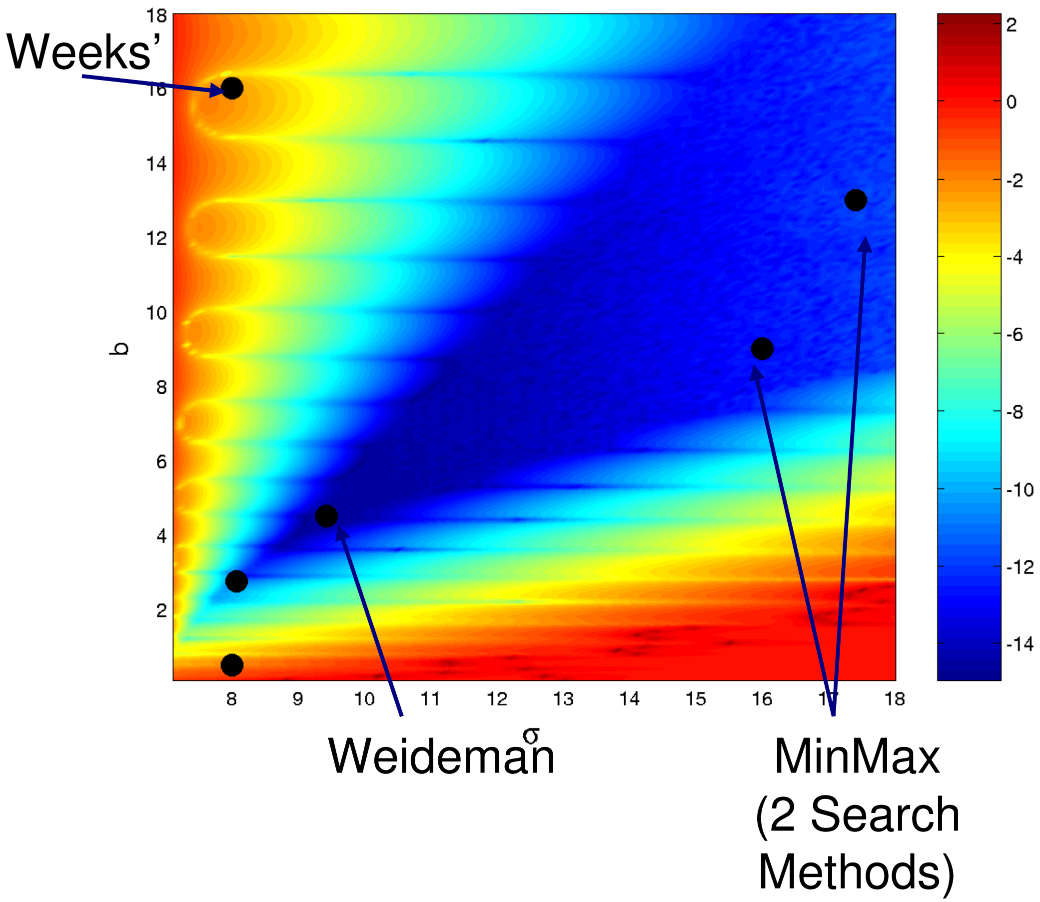


By proper selection of the parameters, it is possible to perform the calculations in double precision.

Pathological Matrices

- An application is the exponential of special matrices.

www.math.arizona.edu/~brio/WEEKS_METHOD_PAGE/pkanoWeeksMethod.html



gallery('pei',6)

2	1	1	1	1	1
1	2	1	1	1	1
1	1	2	1	1	1
1	1	1	2	1	1
1	1	1	1	2	1
1	1	1	1	1	2

6x6 Pei Matrix
Maximum Element Relative Error
32 Coefficients

Eigenvalues
1
N+1 (7)

Future Directions

- December 2009 – NSF proposal submitted

NLAP-CL: Robust Parallel Numerical Laplace Transform Inversion via a C-CUDA Library and Application to Optical Pulse Propagation

- Mosey Brio – University of Arizona
- Patrick Kano – Applied Energetics, Inc.
- Paul Dostert – Coker College

Extend NSF supported Post's Formula Work to 2D and 3D

Mathematica is too slow.

(CUDA)
NVIDIA's Compute Unified Device Architecture

Tables of coefficients for multiple dispersive materials

NLAP requires multiple simple arithmetic computations in high precision.

MATLAB C-MEX Files

Accelerating MATLAB with CUDA™ Using MEX Files

MATLAB NLAP Front End

Summary & Conclusions

- The purpose the of the presentation was to provide some insight and illustrate applications of numerical Laplace transform inversion.

- Standard methods
  - Talbot's Method
  - Post's Formula
  - The Weeks method

- Illustrated two examples
  - Calculation of the matrix exponential
  - Optical pulse propagation in dispersive media

- Numerical Laplace transform inversion is a topic
  - multitude of nuances to provide avenues for further research
  - popularity increase as computing power improves
  - potential for practical application in diverse fields

- Great utility and intellectual merit to the development of a general numerical Laplace inversion toolbox.

Sources

- Numerical Inversion of the Laplace Transform, Bellman, Kalaba, Lockett, 1966.
- Peter Valko's NLAP website: www.pe.tamu.edu/valko/public%5Fhtml/NIL/
- "Numerical inversion of Laplace transforms using Laguerre functions", W. Weeks, Journal of the ACM, vol. 13, no. 3, pp.419-429, July 1966.
- "The accurate numerical inversion of Laplace transforms", J. Inst. Math. Appl., vol. 23, 1979.
- "Application of Weeks method for the numerical inversion of the Laplace transform the matrix exponential", P. Kano, M. Brio, J. Moloney, Comm. Math. Sci., 2005.
- "Application of Post's formula to optical pulse propagation in dispersive media", P. Kano, M. Brio, Computers and Mathematics with Applications, 2009.

BACKUPS

- BACKUP

Laguerre Polynomials

- An unstable approach to obtain the time domain function is to generate the Laguerre polynomials is to use the recurrence relation

$$(n+1)L_{n+1}(x) = (2n+1-x)L_n(x) - nL_{n-1}(x)$$

$$L_0(x) = 1$$

$$L_1(x) = 1 - x$$

$$f(t) = e^{(\sigma-b)t}\sum_{n=0}^{\infty} a_n L_n(2bt)$$

- A backward Clenshaw algorithm is a stable method.

Analytic Inversion

$$F(s) = \frac{1}{s^2}$$

$$g(s) = \frac{1}{s^2} e^{st}$$

$$s = 0 \left[\text{order } 2\right]$$

$$\int_{\sigma - i\infty}^{\sigma + i\infty} \frac{1}{s^2} e^{st} ds = 2\pi i r$$

$$r = \frac{1}{(m-1)!} \frac{d^{m-1}}{ds^{m-1}} \left[(s - s_0)^m g(s) \right]_{s = s_0}$$

$$r = \frac{d}{ds} \left[s^2 \frac{1}{s^2} e^{st} \right]_{s=0} = te^{0t} = t$$

$$\frac{1}{2\pi i} \int_{\sigma - i\infty}^{\sigma + i\infty} \frac{1}{s^2} e^{st} ds = t$$

$$f(t) = L^{-1}\left(\frac{1}{s^2} \right) = t$$

Application of Post's Formula

- Post's formula was implemented to
  - invert the Fourier-Laplace space coefficients
  - which arise from the solution of the optical dispersive wave equation.

- We considered three implementations
  - Standard Gaver-Wynn-Rho
  - Gaver-Post
  - Bell-Post

Brain White Matter

Relative Error, Bell-Post, $k=k_{max}$

q	$\frac{2}{3}\tau_0$	$\frac{4}{3}\tau_0$	$\frac{8}{3}\tau_0$	$\frac{16}{3}\tau_0$
Precision = 25				
$1 \to 4$	$3.682 \cdot 10^{-2}$	$5.952 \cdot 10^{-2}$	0.7637	0.3708
$1 \to 8$	$9.891 \cdot 10^{-2}$	$6.692 \cdot 10^{-3}$	1.004	0.2810
$1 \to 16$	$7.612 \cdot 10^{-4}$	$8.735 \cdot 10^{-4}$	$9.471 \cdot 10^{-2}$	$8.872 \cdot 10^{-2}$
$1 \to 32$	$6.217 \cdot 10^{-5}$	$1.151 \cdot 10^{-4}$	$7.445 \cdot 10^{-3}$	$2.629 \cdot 10^{-2}$
$1 \to 64$	$4.13 \cdot 10^{-3}$	$0. \cdot 10^{-3}$	$0. \cdot 10^{-3}$	$0. \cdot 10^{-3}$
Precision = 50				
$1 \to 4$	$3.682 \cdot 10^{-2}$	$5.952 \cdot 10^{-2}$	0.7637	0.3708
$1 \to 8$	$1.972 \cdot 10^{-7}$	$2.225 \cdot 10^{-5}$	$1.470 \cdot 10^{-3}$	1.846
$1 \to 16$	$7.303 \cdot 10^{-12}$	$9.037 \cdot 10^{-13}$	$7.625 \cdot 10^{-11}$	$1.034 \cdot 10^{-4}$
$1 \to 32$	$7.479 \cdot 10^{-15}$	$5.339 \cdot 10^{-15}$	$3.661 \cdot 10^{-12}$	$9.873 \cdot 10^{-10}$
$1 \to 64$	$2.228 \cdot 10^{-15}$	$5.451 \cdot 10^{-15}$	$3.655 \cdot 10^{-12}$	$9.873 \cdot 10^{-10}$
Precision = 100				
$1 \to 4$	$3.682 \cdot 10^{-2}$	$5.952 \cdot 10^{-2}$	0.7637	0.3708
$1 \to 8$	$1.972 \cdot 10^{-7}$	$2.225 \cdot 10^{-5}$	$1.470 \cdot 10^{-3}$	1.846
$1 \to 16$	$7.303 \cdot 10^{-12}$	$9.037 \cdot 10^{-13}$	$7.625 \cdot 10^{-11}$	$1.034 \cdot 10^{-4}$
$1 \to 32$	$1.456 \cdot 10^{-17}$	$5.413 \cdot 10^{-15}$	$3.655 \cdot 10^{-12}$	$9.873 \cdot 10^{-10}$
$1 \to 64$	$1.457 \cdot 10^{-17}$	$5.413 \cdot 10^{-15}$	$3.655 \cdot 10^{-12}$	$9.873 \cdot 10^{-10}$

Relative Error, Gaver-Post, $k=k_{max}$

q	$\frac{2}{3}\tau_0$	$\frac{4}{3}\tau_0$	$\frac{8}{3}\tau_0$	$\frac{16}{3}\tau_0$
Precision = 25				
$1 \to 4$	$4.332 \cdot 10^{-2}$	$8.065 \cdot 10^{-2}$	0.8107	0.3633
$1 \to 8$	0.1536	$8.984 \cdot 10^{-3}$	0.8861	0.3146
$1 \to 16$	$1.052 \cdot 10^{-3}$	$1.155 \cdot 10^{-3}$	0.1300	$9.411 \cdot 10^{-2}$
$1 \to 32$	*CLA*	*CLA*	*CLA*	*CLA*
$1 \to 64$	*CLA*	*CLA*	*CLA*	*CLA*
Precision = 50				
$1 \to 4$	$4.332 \cdot 10^{-2}$	$8.065 \cdot 10^{-2}$	0.8107	0.3633
$1 \to 8$	$6.159 \cdot 10^{-6}$	$5.123 \cdot 10^{-5}$	$8.098 \cdot 10^{-3}$	4.509
$1 \to 16$	$2.521 \cdot 10^{-11}$	$1.047 \cdot 10^{-12}$	$5.269 \cdot 10^{-9}$	$1.537 \cdot 10^{-5}$
$1 \to 32$				$1. \cdot 10^{-9}$
$1 \to 64$	*CLA*	*CLA*	*CLA*	*CLA*
Precision = 100				
$1 \to 4$	$4.332 \cdot 10^{-2}$	$8.065 \cdot 10^{-2}$	0.8107	0.3633
$1 \to 8$	$6.159 \cdot 10^{-6}$	$5.123 \cdot 10^{-5}$	$8.098 \cdot 10^{-3}$	4.509
$1 \to 16$	$2.521 \cdot 10^{-11}$	$1.047 \cdot 10^{-12}$	$5.269 \cdot 10^{-9}$	$1.537 \cdot 10^{-5}$
$1 \to 32$	$1.456 \cdot 10^{-17}$	$5.413 \cdot 10^{-15}$	$3.655 \cdot 10^{-12}$	$9.873 \cdot 10^{-10}$
$1 \to 64$	$1.457 \cdot 10^{-17}$	$5.413 \cdot 10^{-15}$	$3.655 \cdot 10^{-12}$	$9.873 \cdot 10^{-10}$

At higher precision and Post formula approximation order, the Gaver-Post has an accuracy/unit run time comparable or better than the Bell-Post method.

The Bell-Post method performs well at modest values for the precision and order of the Post formula approximation.