Fractional calculus: from physics to mathematical models (and numerical simulations)

Roberto Garrappa

Dipartimento di Matematica - Università degli Studi di Bari - Italia
roberto.garrappa@uniba.it

Calcolo Scientifico e Modelli Matematici
Alla Ricerca delle Cose Nascoste Attraverso le Cose Manifeste
Roma, 6-8 Aprile 2022

Some questions

Fractional calculus: from physics to mathematical models (and numerical simulations)

- Why using fractional calculus ?
- Is fractional calculus really useful and/or necessary ?
- What kind of fractional-order operator should be used ?

From classical/theoretical to problem-oriented motivations !

- What difficulties in computation ?

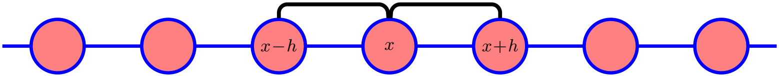
Financial support:

- EU Cost Action 15225 - Fractional Systems
- Istituto Nazionale di Alta Matematica (INdAM-GNCS)
- University of Bari

Classical motivations: Local/Global Jump Random Walk

Local Jump: the particle can jump to x only if it was previously at $x+h$ or $x-h$

- Same probability $\frac{1}{2}$ to jump towards left or right



Probability of the particle to be at x at time $t+\tau$

$$u(x,t+\tau) = \frac{1}{2}u(x-h,t) + \frac{1}{2}u(x+h,t)$$

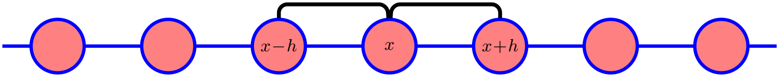
Subtract $u(x,t)$ to both sides, assume $h^2 = 2\tau$ and divide by τ

$$\frac{u(x,t+\tau)-u(x,t)}{\tau} = \frac{u(x-h,t)-2u(x,t)+u(x+h,t)}{h^2} \quad \xrightarrow{\tau,h\to 0} \quad u_t - \Delta u = 0$$

Classical motivations: Local/Global Jump Random Walk

Local Jump: the particle can jump to x only if it was previously at $x+h$ or $x-h$

- Same probability $\frac{1}{2}$ to jump towards left or right



Probability of the particle to be at x at time $t+\tau$

$$u(x, t+\tau) = \frac{1}{2} u(x-h, t) + \frac{1}{2} u(x+h, t)$$

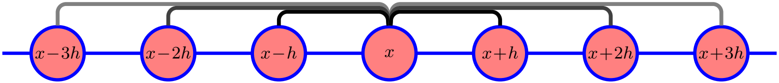
Subtract $u(x,t)$ to both sides, assume $h^2 = 2\tau$ and divide by τ

$$\frac{u(x,t+\tau) - u(x,t)}{\tau} = \frac{u(x-h,t) - 2u(x,t) + u(x+h,t)}{h^2} \quad \xRightarrow{\tau,h\to 0} \quad u_t - \Delta u = 0$$

Classical motivations: Local/Global Jump Random Walk

Global Jump: the particle can jump to x from any position $x + kh$ or $x - kh$

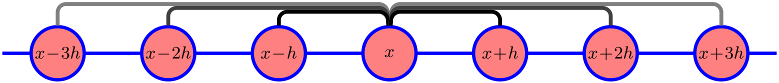
- Probability to jump k positions depends on the distance $\sim \dfrac{c_s}{|k|^{1+2s}}$



Classical motivations: Local/Global Jump Random Walk

Global Jump: the particle can jump to x from any position $x + kh$ or $x - kh$

- Probability to jump k positions depends on the distance $\sim \frac{c_s}{|k|^{1+2s}}$



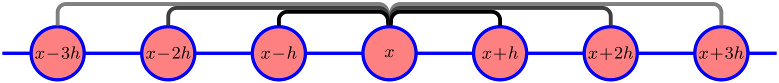
Probability of the particle to be at x at time $t + \tau$

$$u(x, t + \tau) = c_s \sum_{k \in \mathbb{Z}^\star} \frac{1}{|k|^{1+2s}} u(x + kh, t)$$

Classical motivations: Local/Global Jump Random Walk

Global Jump: the particle can jump to x from any position $x+kh$ or $x-kh$

- Probability to jump k positions depends on the distance $\sim \dfrac{c_s}{|k|^{1+2s}}$



Probability of the particle to be at x at time $t+\tau$

$$\frac{u(x,t+\tau)-u(x,t)}{\tau} = c_s h \sum_{k\in\mathbb{Z}^\star} \frac{u(x+\xi,t)-u(x,t)}{|hk|^{1+2s}}$$

Subtract $u(x,t)$, chose $c_s \sum_{k\in\mathbb{Z}^\star} \dfrac{1}{|k|^{1+2s}} = 1$, consider $\tau = h^{2s}$ and let $\tau \to 0$

$$\frac{\mathrm{d}}{\mathrm{d}t} u(x,t) = C_s \int_{\mathbb{R}} \frac{u(x+\xi,t)-u(x,t)}{|\xi|^{1+2s}} \mathrm{d}\xi \quad \Longleftrightarrow \quad u_t + (-\Delta)^s u = 0$$

Classical motivations: Viscoelasticity models

Describe relationship between strain and stress

Elastic bodies | Viscous materials

Hook

$$\sigma(t) = E\epsilon(t)$$

Newton

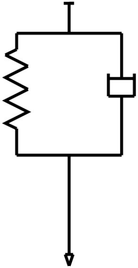
$$\sigma(t) = \eta\dot{\epsilon}(t)$$

Alternative: $\sigma(t) = v \dfrac{\mathrm{d}^\alpha}{\mathrm{d}t^\alpha}\epsilon(t) \quad 0 < \alpha < 1$ Combine Hook and Newton

Classical motivations: Viscoelasticity models

Describe relationship between strain and stress

Elastic bodies | Viscous materials | Viscoelastic (intermediate)



Hook: $\sigma(t) = E\epsilon(t)$

Newton: $\sigma(t) = \eta\dot{\epsilon}(t)$

Maxwell: $\sigma(t) + \frac{\eta}{E}\dot{\sigma}(t) = \eta\dot{\epsilon}(t)$

Voigt: $\sigma(t) = E\epsilon(t) + \eta\dot{\epsilon}(t)$

Alternative: $\sigma(t) = v\frac{\mathrm{d}^\alpha}{\mathrm{d}t^\alpha}\epsilon(t) \quad 0 < \alpha < 1$ Combine Hook and Newton

Classical motivations: Viscoelasticity models

Describe relationship between strain and stress

Elastic bodies | Viscous materials | Viscoelastic (intermediate)

Hook

$\sigma(t) = E\epsilon(t)$

Newton

$\sigma(t) = \eta\dot{\epsilon}(t)$

Alternative: $\sigma(t) = v\dfrac{\mathrm{d}^\alpha}{\mathrm{d}t^\alpha}\epsilon(t) \quad 0 < \alpha < 1$ Combine Hook and Newton

Some questions

Classical motivations are mathematically reasonable but too much theoretical !

Are they supported from experimental data ?

How satisfactory is the fitting of real data ?

Is it possible a different (data and problem oriented) approach ?

A case study: computational electromagnetism

Maxwell's equations:

Standard Maxwell's equations with polarization:

$$\begin{cases} \nabla \times \mathbf{H} = \epsilon_0 \dfrac{\partial}{\partial t}\mathbf{E} + \dfrac{\partial}{\partial t}\mathbf{P} & \text{Ampere's law} \\ \nabla \times \mathbf{E} = -\mu_0 \dfrac{\partial \mathbf{H}}{\partial t} & \text{Faraday's law} \end{cases}$$

E: electric field   **H**: magnetic field   **P**: polarization

Real world applications

- design of data and energy storage devices
- design of antennas
- medical diagnostic (MRI), cancer therapy, ...

Maxwell's equations:

Standard Maxwell's equations with polarization:

$$\begin{cases} \nabla \times \mathbf{H} = \epsilon_0 \dfrac{\partial}{\partial t}\mathbf{E} + \dfrac{\partial}{\partial t}\mathbf{P} & \text{Ampere's law} \\ \nabla \times \mathbf{E} = -\mu_0 \dfrac{\partial \mathbf{H}}{\partial t} & \text{Faraday's law} \end{cases}$$

E: electric field　　**H**: magnetic field　　**P**: polarization

The complex susceptibility

The polarization **P** depends on the electric field **E** (constitutive law)

$$\hat{\mathbf{P}} = \varepsilon_0 \, \hat{\chi}(\omega) \, \hat{\mathbf{E}}$$

- $\hat{\chi}(\omega)$ is a specific feature of the matter (or system)

Simplified notation (just for easy of presentation)

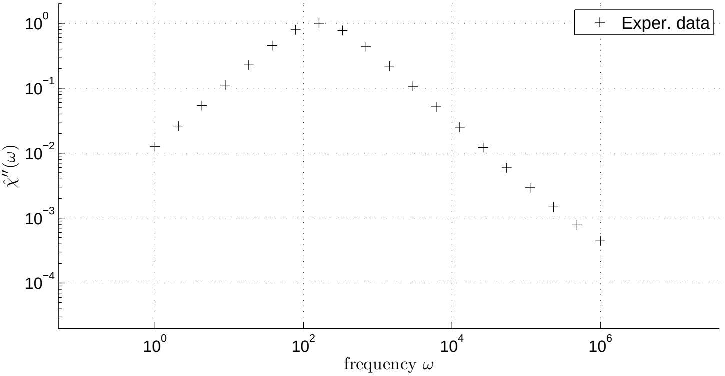
Determining the complex susceptibility $\hat{\chi}(\omega)$

How to derive $\hat{\chi}(\omega) = \hat{\chi}'(\omega) - \mathrm{i}\hat{\chi}''(\omega)$?

Determining the complex susceptibility $\hat{\chi}(\omega)$

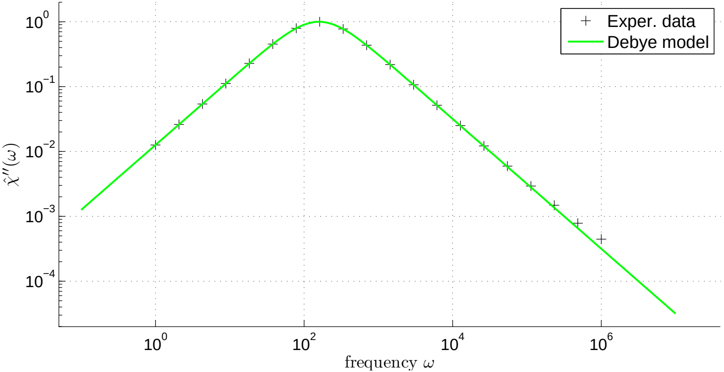
How to derive $\hat{\chi}(\omega) = \hat{\chi}'(\omega) - \mathrm{i}\hat{\chi}''(\omega)$?

Experimental data (in the frequency domain): $\hat{P} = \hat{\chi}(\omega)\hat{E}$



Match experimental data into a mathematical model

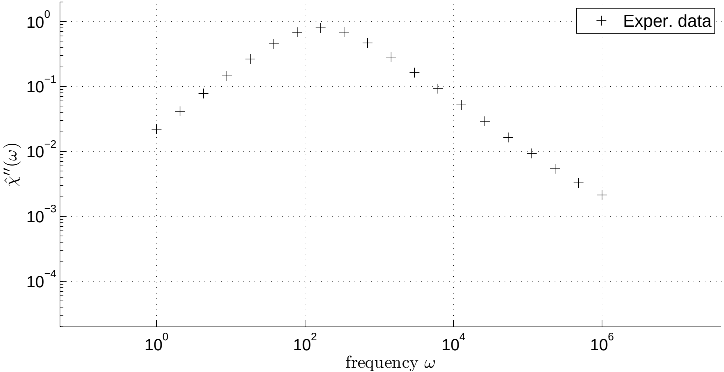
Determining the complex susceptibility $\hat{\chi}(\omega)$



The Debye model: $\hat{\chi}(\omega) = \dfrac{1}{\mathrm{i}\omega\tau + 1}$ (standard materials)

Ordinary differential equation: $\tau \dfrac{d}{dt} P(t) + P(t) = E(t)$

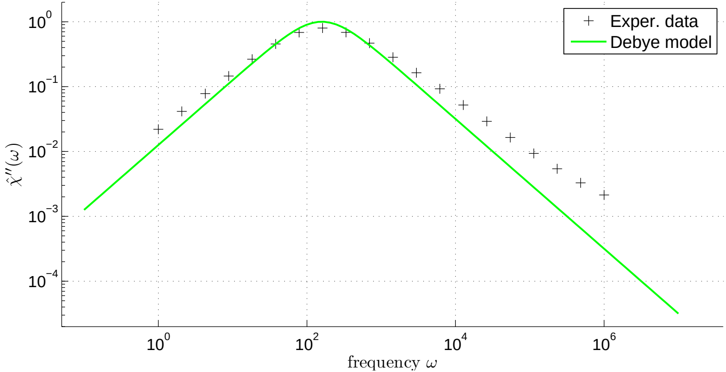
Determining the complex susceptibility $\hat{\chi}(\omega)$



Materials with anomalous dielectric properties:

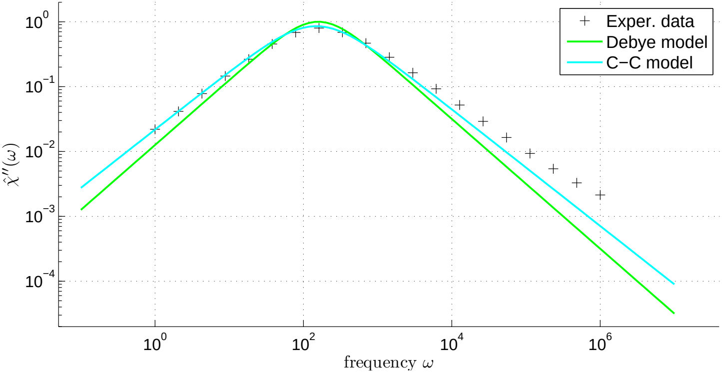
- amorphous polymers
- complex systems (biological tissues)

Determining the complex susceptibility $\hat{\chi}(\omega)$



Debye model not satisfactory

Determining the complex susceptibility $\hat{\chi}(\omega)$

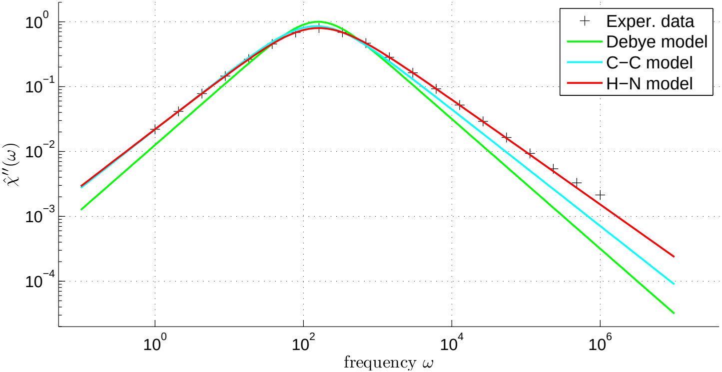


The Cole-Cole model: $$\hat{\chi}(\omega) = \frac{1}{(\mathrm{i}\omega\tau)^{\alpha} + 1} \qquad 0 < \alpha < 1$$

Fractional differential equation: $$\tau^{\alpha} \frac{d^{\alpha}}{dt^{\alpha}} P(t) + P(t) = E(t)$$

Only partially satisfactory

Determining the complex susceptibility $\hat{\chi}(\omega)$

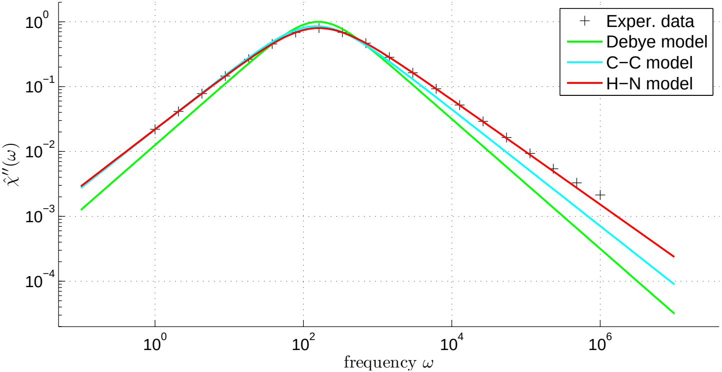


The Havriliak-Negami model: $$\hat{\chi}(\omega) = \frac{1}{\left((\mathrm{i}\omega\tau)^{\alpha} + 1\right)^{\gamma}} \qquad 0 < \alpha, \gamma < 1$$

Better matching thanks to three parameters α, γ and τ

S. Havriliak and S. Negami "A complex plane representation of dielectric and mechanical relaxation processes in some polymers". In: *Polymer* (1967)

Determining the complex susceptibility $\hat{\chi}(\omega)$



The Havriliak-Negami model: $\hat{\chi}(\omega) = \dfrac{1}{\big((\mathrm{i}\omega\tau)^{\alpha} + 1\big)^{\gamma}}$ $\quad 0 < \alpha, \alpha\gamma < 1$

Fractional pseudo-differential equation: $\left(\tau^{\alpha} \dfrac{d^{\alpha}}{dt^{\alpha}} + 1\right)^{\gamma} P(t) = E(t)$?

Dealing with the Havriliak-Negami model

$$\hat{\mathbf{P}}(\omega) = \frac{1}{((\mathrm{i}\omega\tau)^{\alpha} + 1)^{\gamma}} \hat{\mathbf{E}}(\omega)$$

Some contributions on numerical simulation of Havriliak-Nagami model:

- C.S.Antonopoulos, N.V.Kantartzis, I.T.Rekanos "FDTD Method for Wave Propagation in Havriliak-Negami Media Based on Fractional Derivative Approximation". In: *IEEE Trans Magn.* 53(6) (2017)
- P.Bia et al. "A novel FDTD formulation based on fractional derivatives for dispersive Havriliak–Negami media". In: *Signal Processing*, 107 (2015)
- M.F.Causley, P.G.Petropoulos, "On the Time-Domain Response of Havriliak-Negami Dielectrics". In: *IEEE Trans. Antennas Propag.*, 61(6) (2013)
- M.F.Causley, P.G.Petropoulos, and S. Jiang "Incorporating the Havriliak-Negami dielectric model in the FD-TD method". In: *J. Comput. Phys.*, 230 (2011)
- Garrappa R., "Grünwald–Letnikov operators for Havriliak–Negami fractional relaxation". In: *CNSNS* 38 (2016)

Main problems:

- Define time-domain operator for HN
- Discretize the operator for simulations

A fractional pseudo-differential operator in time domain

$$\hat{\mathbf{P}}(\omega) = \frac{1}{((\mathrm{i}\omega\tau)^{\alpha}+1)^{\gamma}}\hat{\mathbf{E}}(\omega) \qquad \Longleftrightarrow \qquad \left(\tau^{\alpha}\frac{d^{\alpha}}{dt^{\alpha}}+1\right)^{\gamma}\mathbf{P}(t) = \mathbf{E}(t) \quad ???$$

A fractional pseudo-differential operator in time domain

$$\hat{\mathbf{P}}(\omega) = \frac{1}{((\mathrm{i}\omega\tau)^{\alpha}+1)^{\gamma}}\hat{\mathbf{E}}(\omega) \qquad \Longleftrightarrow \qquad \left(\tau^{\alpha}\frac{d^{\alpha}}{dt^{\alpha}}+1\right)^{\gamma}\mathbf{P}(t) = \mathbf{E}(t) \quad ???$$

- Combination of fractional operators [Nigmatullin, Ryabov, 1997]

$$\left(\tau^{\alpha}\frac{d^{\alpha}}{dt^{\alpha}}+1\right)^{\gamma}\mathbf{P}(t) = \tau^{\alpha\gamma}\exp\left(-\frac{t}{\alpha\tau^{\alpha}}{}_0D_t^{1-\alpha}\right)\cdot {}_0D_t^{\alpha\gamma}\cdot \exp\left(\frac{t}{\alpha\tau^{\alpha}}{}_0D_t^{1-\alpha}\right)\mathbf{P}(t)$$

- Expansion in infinite series [Novikov et al. 2005] [Bia et al. 2015]

$$\left(\tau^{\alpha}\frac{d^{\alpha}}{dt^{\alpha}}+1\right)^{\gamma}\mathbf{P}(t) = \sum_{k=0}^{\infty}\binom{\gamma}{k}\tau^{\alpha(\gamma-k)}{}_0D_t^{\alpha(\gamma-k)}\mathbf{P}(t)$$

- Prabhakar derivative [Garra et al. 2014, Giusti et al. 2020]

$$\left(\tau^{\alpha}\frac{d^{\alpha}}{dt^{\alpha}}+1\right)^{\gamma}\mathbf{P}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\int_0^t\left(\frac{t'}{\tau}\right)^{-\alpha\gamma}E_{\alpha,1-\alpha\gamma}^{-\gamma}\left(-\left(\frac{t'}{\tau}\right)^{\alpha}\right)\mathbf{P}(t-t')\,\mathrm{d}t'$$

Integral formulation

$$\hat{\mathbf{P}}(\omega) = \hat{\chi}(\omega)\hat{\mathbf{E}}(\omega) \qquad \Longleftrightarrow \qquad \mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'$$

- $\chi(t)$ inverse of $\hat{\chi}(\omega)$ and depends on the model
- $\chi(t)$ is weakly singular

$$\chi(t) = \frac{1}{\tau^{\alpha\gamma}} t^{\alpha\gamma-1} E^{\gamma}_{\alpha,\alpha\gamma}\left(-\left(\frac{t}{\tau}\right)^{\alpha}\right)$$

α	γ	Model
$(0,1)$	1	Cole-Cole
1	$(0,1)$	Cole-Davidson
$(0,1)$	$(0,1)$	Havriliak-Negami

The Prabhakar function

$$E^{\gamma}_{\alpha,\beta}(z) = \frac{1}{\Gamma(\gamma)} \sum_{k=0}^{\infty} \frac{\Gamma(\gamma+k) z^k}{k!\Gamma(\alpha k+\beta)}$$

It is better do not use special functions !

Integral formulation

$$\hat{\mathbf{P}}(\omega) = \hat{\chi}(\omega)\hat{\mathbf{E}}(\omega) \qquad \Longleftrightarrow \qquad \mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'$$

- $\chi(t)$ inverse of $\hat{\chi}(\omega)$ and depends on the model
- $\chi(t)$ is weakly singular : it involves non-locality

Non-locality: no step-by-step is possible

$$\mathbf{P}(t_n) = \mathbf{P}(t_{n-1}) + h\Phi\big(t_n, \mathbf{E}(t_n)\big)$$

Long-term convolution

$$\mathbf{P}(t_n) = \sum_{j=0}^{n} \omega_{n-j}\mathbf{E}(t_j)$$

From continuous to discrete convolution

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \Longrightarrow \qquad \mathbf{P}(t_n) = \sum_{j=0}^{n} \omega_{n-j}\mathbf{E}(t_j)$$

- Product integration rules (polynomial approximation): use of $\chi(t)$
- Lubich's convolution quadrature rules (ρ, σ multistep method): use of $\hat{\chi}(t)$

$$\sum_{n=0}^{\infty} \omega_n \xi^n = \hat{\chi}\left(\frac{\delta(\xi)}{h}\right) \qquad \delta(\xi) = \frac{\sigma(1/\xi)}{\rho(1/\xi)}$$

Numerical integration of Cauchy integral

$$\omega_n = \frac{1}{n!}\,\frac{\mathrm{d}^n}{\mathrm{d}\xi^n}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\bigg|_{\xi=0} = \frac{1}{2\pi\mathrm{i}}\int_{\mathcal{C}} \xi^{-n-1}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\mathrm{d}\xi$$

$\hat{\chi}(s)$: Fourier/Laplace transform of $\chi(t)$

From continuous to discrete convolution

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \Longrightarrow \qquad \mathbf{P}(t_n) = \sum_{j=0}^{n} \omega_{n-j}\mathbf{E}(t_j)$$

- Product integration rules (polynomial approximation): use of $\chi(t)$
- Lubich's convolution quadrature rules (ρ, σ multistep method): use of $\hat{\chi}(t)$

$$\sum_{n=0}^{\infty} \omega_n \xi^n = \hat{\chi}\left(\frac{\delta(\xi)}{h}\right) \qquad \delta(\xi) = \frac{\sigma(1/\xi)}{\rho(1/\xi)}$$

Numerical integration of Cauchy integral

$$\omega_n = \frac{1}{n!}\,\frac{\mathrm{d}^n}{\mathrm{d}\xi^n}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\bigg|_{\xi=0} = \frac{1}{2\pi \mathrm{i}}\int_{\mathcal{C}} \xi^{-n-1}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\mathrm{d}\xi$$

$\hat{\chi}(s)$: Fourier/Laplace transform of $\chi(t)$

From continuous to discrete convolution

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \Longrightarrow \qquad \mathbf{P}(t_n) = \sum_{j=0}^{n} \omega_{n-j}\mathbf{E}(t_j)$$

Circular contour + trapezoidal rule + FFT to compute convolution weights

$$\omega_n = \frac{1}{2\pi \mathrm{i}} \int_{|\xi|=\rho} \xi^{-n-1}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\mathrm{d}\xi \approx \frac{\rho^{-n}}{N}\sum_{\ell=0}^{N-1}\hat{\chi}\left(\frac{\delta(\xi_\ell)}{h}\right)\mathrm{e}^{-2\pi \mathrm{i} n\ell/N}, \quad \xi_\ell = \rho \mathrm{e}^{2\pi i\ell/N}$$

FFT to compute the convolution sum

Computational cost: $\mathcal{O}\big(N\log_2 N\big) + \mathcal{O}\big(N(\log_2 N)^2\big)$ Storage: $\mathcal{O}(N)$

Acceptable in normal situations but not in computational electromagnetism

$$\frac{h}{\Delta x}c \approx 1 \qquad \Longrightarrow \qquad N \approx \frac{T}{\Delta x}c \quad \text{too large !}$$

From continuous to discrete convolution

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \Longrightarrow \qquad \mathbf{P}(t_n) = \sum_{j=0}^{n} \omega_{n-j}\mathbf{E}(t_j)$$

Circular contour + trapezoidal rule + FFT to compute convolution weights

$$\omega_n = \frac{1}{2\pi\mathrm{i}} \int\limits_{|\xi|=\rho} \xi^{-n-1}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\mathrm{d}\xi \approx \frac{\rho^{-n}}{N}\sum_{\ell=0}^{N-1}\hat{\chi}\left(\frac{\delta(\xi_\ell)}{h}\right)\mathrm{e}^{-2\pi\mathrm{i}n\ell/N}, \quad \xi_\ell = \rho\mathrm{e}^{2\pi i\ell/N}$$

FFT to compute the convolution sum

Computational cost: $\mathcal{O}\big(N\log_2 N\big) + \mathcal{O}\big(N(\log_2 N)^2\big)$ Storage: $\mathcal{O}\big(N\big)$

Acceptable in normal situations but not in computational electromagnetism

$$\frac{h}{\Delta x}c \approx 1 \qquad \Longrightarrow \qquad N \approx \frac{T}{\Delta x}c \quad \text{too large !}$$

From continuous to discrete convolution

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \Longrightarrow \qquad \mathbf{P}(t_n) = \sum_{j=0}^{n} \omega_{n-j}\mathbf{E}(t_j)$$

Circular contour + trapezoidal rule + FFT to compute convolution weights

$$\omega_n = \frac{1}{2\pi \mathrm{i}} \int_{|\xi|=\rho} \xi^{-n-1}\hat{\chi}\left(\frac{\delta(\xi)}{h}\right)\mathrm{d}\xi \approx \frac{\rho^{-n}}{N}\sum_{\ell=0}^{N-1}\hat{\chi}\left(\frac{\delta(\xi_\ell)}{h}\right)\mathrm{e}^{-2\pi \mathrm{i} n\ell/N}, \quad \xi_\ell = \rho \mathrm{e}^{2\pi i \ell/N}$$

FFT to compute the convolution sum

Computational cost: $\mathcal{O}\big(N\log_2 N\big) + \mathcal{O}\big(N(\log_2 N)^2\big)$ Storage: $\mathcal{O}\big(N\big)$

Acceptable in normal situations but not in computational electromagnetism

$$\frac{h}{\Delta x}c \approx 1 \qquad \Longrightarrow \qquad N \approx \frac{T}{\Delta x}c \quad \text{too large !}$$

A more efficient strategy

Idea: approximate the non-local problem with a series of local problems !

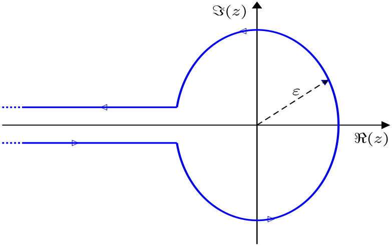
$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \chi(t) = \frac{1}{2\pi\mathrm{i}}\int_{\mathcal{C}} \mathrm{e}^{st}\hat{\chi}(s)\mathrm{d}s$$

A more efficient strategy

Idea: approximate the non-local problem with a series of local problems !

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \chi(t) = \frac{1}{2\pi\mathrm{i}}\int_{\mathcal{C}} \mathrm{e}^{st}\hat{\chi}(s)\mathrm{d}s$$

Integrate formula for the inverse LT on an Hankel contour $\mathcal{C}$ and let $\varepsilon \to 0$



Obtain a spectral representation of the kernel $\chi(t)$

$$\chi(t) = \frac{1}{\tau}\int_0^{\infty} \mathrm{e}^{-rt/\tau} K(r)\mathrm{d}r$$

A more efficient strategy

A spectral representation of the kernel $\chi(t) = \frac{1}{\tau}\int_0^{\infty} \mathrm{e}^{-rt/\tau} K(r) \mathrm{d}r$

$K(r)$	α	γ	Model
$\dfrac{r^{\alpha}\sin\alpha\pi}{\pi(r^{2\alpha}+2r^{\alpha}\cos\alpha\pi+1)}$	$(0,1)$	1	Cole-Cole
$\dfrac{\sin\gamma\pi}{\pi(r-1)^{\gamma}}$ $r>1$	1	$(0,1)$	Cole-Davidson
$\dfrac{\sin\left[\gamma\theta_{\alpha}(r)\right]}{\pi(r^{2\alpha}+2r^{\alpha}\cos\alpha\pi+1)^{\gamma/2}}$	$(0,1)$	$(0,1)$	Havriliak-Negami

$\theta_{\alpha}(r) = \frac{\pi}{2} - \arctan\left[\frac{1+r^{\alpha}\cos\pi\alpha}{r^{\alpha}\sin\pi\alpha}\right] \in [0,\pi]$

Why the spectral representation is useful?

$\chi(t)$ can now be approximated by a quadrature rule

A more efficient strategy

A spectral representation of the kernel $\chi(t) = \dfrac{1}{\tau}\displaystyle\int_0^{\infty} \mathrm{e}^{-rt/\tau} K(r)\mathrm{d}r$

$K(r)$	α	γ	Model
$\dfrac{r^\alpha \sin \alpha\pi}{\pi(r^{2\alpha} + 2r^\alpha \cos \alpha\pi + 1)}$	$(0,1)$	1	Cole-Cole
$\dfrac{\sin \gamma\pi}{\pi(r-1)^\gamma} \;\; r > 1$	1	$(0,1)$	Cole-Davidson
$\dfrac{\sin\left[\gamma\theta_\alpha(r)\right]}{\pi(r^{2\alpha} + 2r^\alpha \cos \alpha\pi + 1)^{\gamma/2}}$	$(0,1)$	$(0,1)$	Havriliak-Negami

$\theta_\alpha(r) = \frac{\pi}{2} - \arctan\left[\frac{1+r^\alpha \cos \pi\alpha}{r^\alpha \sin \pi\alpha}\right] \in [0, \pi]$

Why the spectral representation is useful?

$\chi(t)$ can now be approximated by a quadrature rule

A more efficient strategy

Approximation of the the spectral representation

$$\chi(t) = \frac{1}{\tau} \int_0^\infty \mathrm{e}^{-rt/\tau} K(r) \mathrm{d}r$$

Truncation and change of variable

$$\chi^{[R]}(t) = \frac{1}{\tau} \int_0^R \mathrm{e}^{-rt/\tau} K(r) \mathrm{d}r = \frac{R}{2\tau} \int_{-1}^1 \mathrm{e}^{-t \frac{R}{2\tau}(\xi+1)} K\left(\frac{R}{2}(\xi+1)\right) \mathrm{d}\xi$$

Discretization by Q-nodes Gauss-Legendre quadrature rule on $[-1, 1]$:

$$\chi^{[R,Q]}(t) = \sum_{q=1}^{Q} W_q \mathrm{e}^{-\xi_q t}, \quad \xi_q = \frac{R}{2\tau}(\overline{\xi}_q + 1), \quad W_q = \frac{R}{2\tau} \overline{w}_q K(\xi_q)$$

Approximation of the the spectral representation

$$\chi(t) = \frac{1}{\tau} \int_0^{\infty} \mathrm{e}^{-rt/\tau} K(r) \mathrm{d}r \qquad \chi^{[R,Q]}(t) = \sum_{q=1}^{Q} W_q \mathrm{e}^{-\xi_q t}$$

Truncation error: sharp estimation to select R

$$R \geq -\frac{\tau}{\delta t} \log\Big[\frac{\delta t \cdot \mathsf{Tol}}{2 \cdot \max_r K(r)}\Big] \quad \Longrightarrow \quad \Big|\chi(t) - \chi^{[R]}(t)\Big| \leq \frac{1}{2}\mathsf{Tol} \quad \forall t \geq \delta t$$

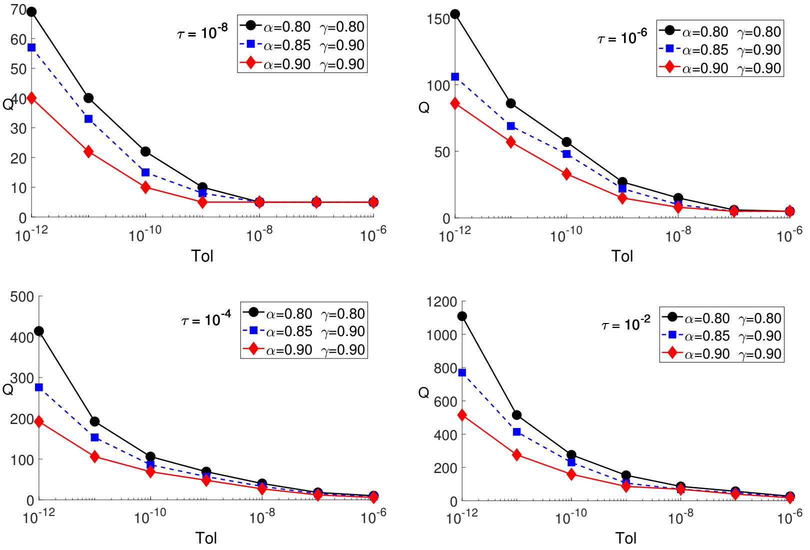
Discretization error:

$$\Big|\chi^{[R,Q]}(t) - \chi^{[R]}(t)\Big| \approx \frac{2^{2Q+1}(Q!)^4}{(2Q+1)[(2Q)!]^3} \left| \frac{\mathrm{d}^{(2Q)}}{\mathrm{d}r^{(2Q)}} \Big(\mathrm{e}^{-\bar{r}t/\tau} K(\bar{r}) \Big) \right|$$

More difficult to estimate the number of nodes Q to achieve a given tolerance

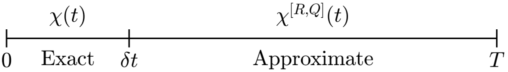
- Algorithmic approach
- Comparison with the exact value of $\chi(t)$ (available)
- Computation at scalar level (once for the whole computation): fast

Number of nodes Q to achieve accuracy tol ($t \in [1, 100]$)



Use of the approximated kernel

$$\mathbf{P}(t) = \underbrace{\int_0^{t-\delta t} \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Memory term}} + \underbrace{\int_{t-\delta t}^{0} \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Local term}}$$



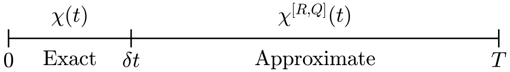
$$\chi^{[R,Q]}(t) = \sum_{q=1}^{Q} W_q \mathrm{e}^{-\xi_q t}$$

$$\mathbf{P}(t) \approx \underbrace{\sum_{q=1}^{Q} W_q \int_0^{t-\delta t} \mathrm{e}^{-\xi_q(t-t')}\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Memory term } \mathbf{M}(t)} + \underbrace{\int_{t-\delta t}^{0} \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Local term}}$$

$$\mathbf{P}(t_n) \approx \mathbf{M}(t_n) + \sum_{j=n-n_d}^{n} \omega_{n-j}\mathbf{E}(t_j) \qquad n_d = \frac{\delta t}{h} \qquad \underbrace{\mathbf{M}(t_n) = \sum_{q=1}^{Q} W_q \mathbf{M}_q(t_n)}_{\text{Memory term}}$$

Use of the approximated kernel

$$\mathbf{P}(t) = \underbrace{\int_0^{t-\delta t} \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Memory term}} + \underbrace{\int_{t-\delta t}^{0} \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Local term}}$$



$$\chi^{[R,Q]}(t) = \sum_{q=1}^{Q} W_q \mathrm{e}^{-\xi_q t}$$

$$\mathbf{P}(t) \approx \underbrace{\sum_{q=1}^{Q} W_q \int_0^{t-\delta t} \mathrm{e}^{-\xi_q(t-t')}\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Memory term } \mathbf{M}(t)} + \underbrace{\int_{t-\delta t}^{0} \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t'}_{\text{Local term}}$$

$$\mathbf{P}(t_n) \approx \mathbf{M}(t_n) + \sum_{j=n-n_d}^{n} \omega_{n-j}\mathbf{E}(t_j) \qquad n_d = \frac{\delta t}{h} \qquad \underbrace{\mathbf{M}(t_n) = \sum_{q=1}^{Q} W_q \mathbf{M}_q(t_n)}_{\text{Memory term}}$$

Approximation of the memory term

$$\mathbf{M}(t_n) = \sum_{q=1}^{Q} W_q \mathbf{M}_q(t_n) \qquad \mathbf{M}_q(t_n) = \int_0^{t_n - \delta t} \mathrm{e}^{-\xi_q (t_n - t')} \mathbf{E}(t') \, \mathrm{d}t'$$

Exponential kernel allows a step-by-step procedure

$$\mathbf{M}_q(t_n) = \mathrm{e}^{-\xi_q h} \mathbf{M}_q(t_{n-1}) + \int_{t_n - \delta t - h}^{t_n - \delta t} \mathrm{e}^{-\xi_q (t_n - t')} \mathbf{E}(t') \, \mathrm{d}t'$$

Example: the trapezoidal rule

$$\widehat{\mathbf{M}}_q(t_n) = \mathrm{e}^{-\xi_q h} \widehat{\mathbf{M}}_q(t_{n-1}) + \mathrm{e}^{-\xi_q \delta t} \frac{h}{2} \left[\mathbf{E}(t_{n-n_d}) + \mathrm{e}^{-\xi_q h} \mathbf{E}(t_{n-n_d-1}) \right],$$

- Reduced storage need: proportional to $Q \ll N$ and not to N
- Reduced computational cost: proportional to QN and not $N(\log_2 N)^2$
- Known as *Kernel compression scheme* or *Sum-of-exponentials*

Approximation of the memory term

$$\mathbf{M}(t_n) = \sum_{q=1}^{Q} W_q \mathbf{M}_q(t_n) \qquad \mathbf{M}_q(t_n) = \int_0^{t_n - \delta t} \mathrm{e}^{-\xi_q (t_n - t')} \mathbf{E}(t') \, \mathrm{d}t'$$

Exponential kernel allows a step-by-step procedure

$$\mathbf{M}_q(t_n) = \mathrm{e}^{-\xi_q h} \mathbf{M}_q(t_{n-1}) + \int_{t_n - \delta t - h}^{t_n - \delta t} \mathrm{e}^{-\xi_q (t_n - t')} \mathbf{E}(t') \, \mathrm{d}t'$$

Example: the trapezoidal rule

$$\widehat{\mathbf{M}}_q(t_n) = \mathrm{e}^{-\xi_q h} \widehat{\mathbf{M}}_q(t_{n-1}) + \mathrm{e}^{-\xi_q \delta t} \frac{h}{2} \left[\mathbf{E}(t_{n-n_d}) + \mathrm{e}^{-\xi_q h} \mathbf{E}(t_{n-n_d-1}) \right],$$

- Reduced storage need: proportional to $Q \ll N$ and not to N
- Reduced computational cost: proportional to QN and not $N(\log_2 N)^2$
- Known as *Kernel compression scheme* or *Sum-of-exponentials*

Approximation of the memory term

$$\mathbf{M}(t_n) = \sum_{q=1}^{Q} W_q \mathbf{M}_q(t_n) \qquad \mathbf{M}_q(t_n) = \int_0^{t_n - \delta t} \mathrm{e}^{-\xi_q (t_n - t')} \mathbf{E}(t') \, \mathrm{d}t'$$

Exponential kernel allows a step-by-step procedure

$$\mathbf{M}_q(t_n) = \mathrm{e}^{-\xi_q h} \mathbf{M}_q(t_{n-1}) + \int_{t_n - \delta t - h}^{t_n - \delta t} \mathrm{e}^{-\xi_q (t_n - t')} \mathbf{E}(t') \, \mathrm{d}t'$$

Example: the trapezoidal rule

$$\widehat{\mathbf{M}}_q(t_n) = \mathrm{e}^{-\xi_q h} \widehat{\mathbf{M}}_q(t_{n-1}) + \mathrm{e}^{-\xi_q \delta t} \frac{h}{2} \left[\mathbf{E}(t_{n-n_d}) + \mathrm{e}^{-\xi_q h} \mathbf{E}(t_{n-n_d-1}) \right],$$

- Reduced storage need: proportional to $Q \ll N$ and not to N
- Reduced computational cost: proportional to QN and not $N(\log_2 N)^2$
- Known as *Kernel compression scheme* or *Sum-of-exponentials*

Speed-up of the fast algorithm

Performance for $\alpha = 0.9$, $\gamma = 0.95$, $\tau = 10^{-3}$, $h = 2^{-17} \approx 10^{-5}$

		CPU Time (sec.)		
T	N	Standard	Fast	Speed-up
0.003906	512	0.020	1.099	0.02
0.007813	1024	0.079	0.841	0.09
0.015625	2048	0.301	1.0543	0.29
0.031250	4096	1.027	1.389	0.74
0.062500	8192	3.972	2.067	1.92
0.125000	16384	15.891	3.973	3.99
0.250000	32768	64.375	11.059	5.82
0.500000	65536	255.405	35.479	7.20
1.000000	131072	1022.006	129.750	7.88
2.000000	262144	4080.596	503.150	8.11

A test problem

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \mathbf{E}(t) = \sum_{k=1}^{3} a_k \cos \Omega_k t + \sum_{k=1}^{3} b_k \sin \Omega_k t$$

$\alpha = 0.9$ $\gamma = 0.95$

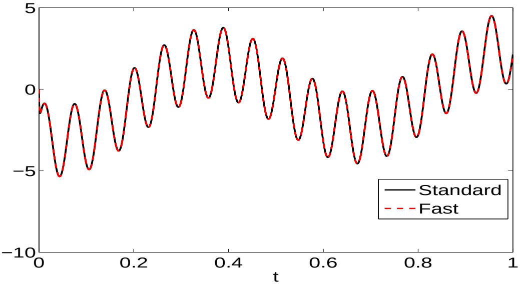
$\tau = 10^{-3}$

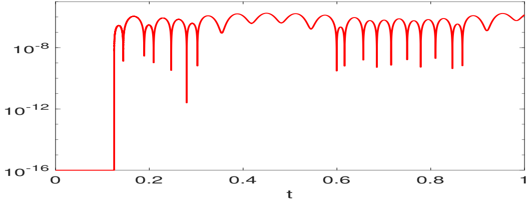
$\delta t = 0.125$

$a = [-1, -2, 1]$

$b = [1, -1, 2]$

$\Omega = [1, 10, 100]$

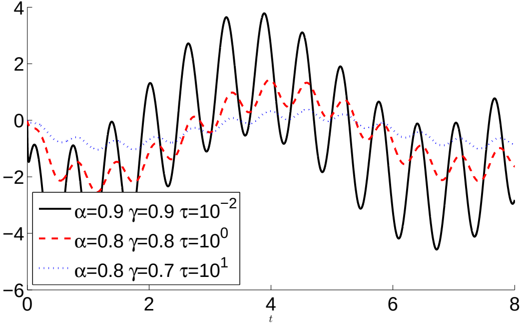




A test problem: errors and EOC

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \mathbf{E}(t) = \sum_{k=1}^{3} a_k \cos \omega_k t + \sum_{k=1}^{3} b_k \sin \omega_k t$$

	α	γ	τ
Set 1	0.8	0.9	10^{-2}
Set 2	0.8	0.8	10^{0}
Set 3	0.8	0.7	10^{1}



Numerical experiments: errors and EOC

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \mathbf{E}(t) = \sum_{k=0}^{N} a_k \cos\omega_k t + \sum_{k=0}^{N} b_k \sin\omega_k t$$

	$\alpha=0.9$ $\gamma=0.9$ $\tau=10^{-2}$		$\alpha=0.8$ $\gamma=0.8$ $\tau=10^{0}$		$\alpha=0.8$ $\gamma=0.7$ $\tau=10^{1}$	
h	Error	EOC	Error	EOC	Error	EOC
2^{-4}	5.10(−3)		6.21(−3)		2.04(−3)	
2^{-5}	1.18(−3)	2.118	1.52(−3)	2.030	4.83(−4)	2.080
2^{-6}	2.85(−4)	2.043	3.80(−4)	2.000	1.21(−4)	1.999
2^{-7}	7.02(−5)	2.021	9.51(−5)	2.000	3.02(−5)	1.998
2^{-8}	1.73(−5)	2.022	2.35(−5)	2.014	7.50(−6)	2.013
2^{-9}	4.12(−6)	2.071	5.61(−6)	2.069	1.79(−6)	2.068

$$\mathrm{EOC} = \log_2\left(E(h)/E(\tfrac{h}{2})\right)$$

Numerical experiments: errors and EOC

$$\mathbf{P}(t) = \int_0^t \chi(t-t')\mathbf{E}(t')\,\mathrm{d}t' \qquad \mathbf{E}(t) = \sum_{k=0}^{N} a_k t^{\alpha k} + \sum_{k=0}^{N} b_k t^{\gamma k} + \sum_{k=0}^{N} c_k t^{k}$$

	$\alpha = 0.9 \quad \gamma = 0.9$ $\tau = 10^{-2}$		$\alpha = 0.8 \quad \gamma = 0.8$ $\tau = 10^{0}$		$\alpha = 0.8 \quad \gamma = 0.7$ $\tau = 10^{1}$	
h	Error	EOC	Error	EOC	Error	EOC
2^{-4}	$1.28(-4)$		$1.26(-3)$		$1.43(-4)$	
2^{-5}	$3.11(-5)$	2.041	$3.15(-4)$	1.996	$3.68(-5)$	1.960
2^{-6}	$7.69(-6)$	2.015	$7.88(-5)$	1.999	$9.29(-6)$	1.987
2^{-7}	$1.91(-6)$	2.009	$1.97(-5)$	2.003	$2.32(-6)$	2.001
2^{-8}	$4.72(-7)$	2.018	$4.86(-6)$	2.017	$5.73(-7)$	2.019
2^{-9}	$1.12(-7)$	2.071	$1.16(-6)$	2.070	$1.36(-7)$	2.075

$$\mathrm{EOC} = \log_2\left(E(h)/E(\tfrac{h}{2})\right)$$

Concluding remarks

- Models incorporating fractional-order operators are obtained in a data-oriented way
- New and more complex operators can be involved
- Non-locality: storage and computational needs can be very demanding
- Specific strategies must be adopted to handle specific problems

Further developments

- Validation with experimental data (Technical University of Bari)
- Releasing robust software for solving Maxwell's systems with anomalous dielectric relaxation

Some references

- Causley M.F., Petropoulos P.G., and Jiang S., "Incorporating the Havriliak-Negami dielectric model in the FD-TD method". In: *J. Comput. Phys.*, 230 (2011)
- Lubich C. "Convolution quadrature and discretized operational calculus (I and II)", In: *Numer. Math.*, 52 (1998)
- Garrappa R., Mainardi F. and Maione G., "Models of dielectric relaxation based on completely monotone functions". In: *FCAA* 19(5) (2016)
- Giusti A., Colombaro I., Garra R., Garrappa R., Polito F., Popolizio M., Mainardi F., "A practical guide to Prabhakar fractional calculus". In *FCAA* 23(1) (2020)
- Yang Y., Wang L.L, Zeng F., "Analysis of a backward Euler-type scheme for Maxwell's equations in a Havriliak-Negami dispersive medium". In *M2AN* 55 (2021)