Physics informed neural networks for learning about dynamics of magnets

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

[1]Télécom Paris, Institut Polytechnique de Paris, F-91123, Palaiseau
[2]Institut Denis Poisson UMR 7013, Université de Tours, F-37200, Tours
[3]CEA, DAM, Le Ripault, F-37260, Monts

July 3, 2023

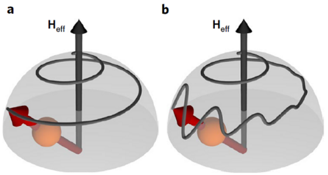
Landau-Lifshitz-Gilbert equation with nutation

- inertial Landau-Lifshitz-Gilbert equation (iLLG) = magnetization dynamics with precession, dampening and nutation

$$\dot{\boldsymbol{M}} = \boldsymbol{M} \times \left(\boldsymbol{\omega} + \lambda \dot{\boldsymbol{M}} + \tau \ddot{\boldsymbol{M}}\right) \tag{1}$$

$\boldsymbol{M} \to \boldsymbol{M}/\|\boldsymbol{M}\|$ reduced magnetization,
$\boldsymbol{\omega} \equiv \gamma\mu_0 \boldsymbol{H}_{eff}$: precession field given by magnetic induction $\mu_0 \boldsymbol{H}_{eff}$,
λ: dimensionless Gilbert "dampening" constant
τ: relaxation time for magnetic inertia
Recast in a dimensionless form $t \to t/\tau$ and $\boldsymbol{\Omega} = \tau\boldsymbol{\omega}$:

$$\dot{\boldsymbol{M}} = \boldsymbol{M} \times \left(\boldsymbol{\Omega} + \lambda \dot{\boldsymbol{M}} + \ddot{\boldsymbol{M}}\right)$$



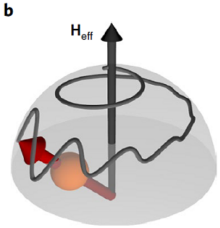
Symplectic integration

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

Let $\boldsymbol{J}$ be defined as $\boldsymbol{J} \equiv \boldsymbol{M} - \boldsymbol{M} \times \dot{\boldsymbol{M}}$; iLLG written in canonical form:

$$\begin{cases} \dot{\boldsymbol{M}} = \boldsymbol{M} \times \boldsymbol{J} \\ \dot{\boldsymbol{J}} = \boldsymbol{M} \times \boldsymbol{\Omega} + \lambda(\boldsymbol{M} - \boldsymbol{J}) \end{cases} \quad \text{with} \quad \begin{matrix} \boldsymbol{M}(0) = \boldsymbol{M}_0 \\ \boldsymbol{J}(0) = \boldsymbol{J}_0 \end{matrix} \tag{2}$$

- $\boldsymbol{J}$ is misaligned with $\boldsymbol{M}$ and only aligned when $\boldsymbol{M} // \dot{\boldsymbol{M}}$
- $\boldsymbol{M}$ is in precession around $\boldsymbol{J}$; $\dot{\boldsymbol{J}}$ is a torque $\boldsymbol{M} \times \boldsymbol{\Omega}$ along with a "damping" $\lambda(\boldsymbol{M} - \boldsymbol{J})$
- $0 \leq t \leq t_f$



Symplectic integration II

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

- Equation on $\dot{\boldsymbol{M}}$ describes precession of $\boldsymbol{M}$ around $\boldsymbol{J}$ and can be solved over an interval Δt (assuming $\boldsymbol{J}$ constant), according to Rodrigue's rotation formula:

$$\boldsymbol{M}(t+\Delta t) = \alpha(\Delta t)\boldsymbol{M}(t) + \beta(\Delta t)\frac{\boldsymbol{J}(t)}{\|\boldsymbol{J}(t)\|} + \gamma(\Delta t)\boldsymbol{M}(t) \times \frac{\boldsymbol{J}(t)}{\|\boldsymbol{J}(t)\|}$$

  with $\alpha^2 + 2\alpha\beta\boldsymbol{M}(t)\cdot\boldsymbol{J}(t) + \beta^2 + \gamma^2(1-(\boldsymbol{M}(t)\cdot\boldsymbol{J}(t))^2) = 1.$

- Since $\boldsymbol{J}$ is not constant, the corrections can be expressed by a 4th-order Runge-Kutta scheme

$$\boldsymbol{J}(t+\Delta t) = RK_4[\boldsymbol{J}(t), \boldsymbol{M}(t)]$$

Neural Network integration

- Compact notation $\boldsymbol{A} \equiv \begin{pmatrix} \boldsymbol{M} \\ \boldsymbol{J} \end{pmatrix}$ defines a 1^{st}-order ODE:

$$\begin{cases} \dot{\boldsymbol{A}} = \boldsymbol{F}(\boldsymbol{A}) \\ \boldsymbol{A}(0) = \boldsymbol{A}_0 \end{cases} \tag{3}$$

- Trial solution

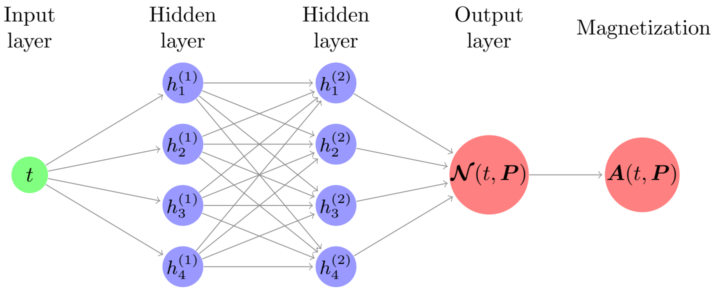
$$\boldsymbol{A}(t, \boldsymbol{P}) = \boldsymbol{A}_0 + t\mathcal{N}(t, \boldsymbol{P}) \tag{4}$$

$\boldsymbol{P}$: set of learning parameters
$\mathcal{N}(t, \boldsymbol{P})$: output of a neural network

- L^2-loss function discretized in time, $t_k = t_f k/N$, to be minimized over $\boldsymbol{P}$

$$\mathrm{loss}[\boldsymbol{P}] = \frac{1}{N} \sum_{k=1}^{N} \left\| \dot{\boldsymbol{A}}(t_k, \boldsymbol{P}) - \boldsymbol{F}(\boldsymbol{A}(t_k, \boldsymbol{P})) \right\|^2 \tag{5}$$

Neural Network Architecture



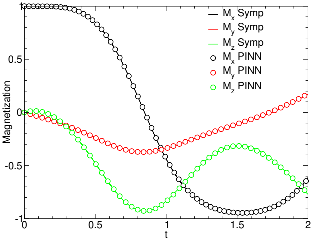
$$\begin{aligned} \boldsymbol{h}^{(1)}(t, \boldsymbol{P}) &= \sigma(t\boldsymbol{W}^{(1)} + \boldsymbol{b}^{(1)}) \\ \boldsymbol{h}^{(k+1)}(t, \boldsymbol{P}) &= \sigma(\boldsymbol{W}^{(k+1)}\boldsymbol{h}^{(k)} + \boldsymbol{b}^{(k+1)}) \\ \mathcal{N}(t, \boldsymbol{P}) &= \boldsymbol{W}^{(L+1)}\boldsymbol{h}^{(L)} + \boldsymbol{b}^{(L+1)} \end{aligned} \tag{6}$$

- Implemented with Tensorflow

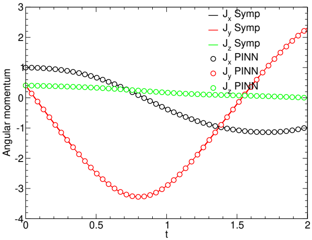
Results

Parameters:

- $N = 32$, $0 \leq t \leq t_f = 2$
- Physical parameters: $\boldsymbol{\Omega} = 2\pi \boldsymbol{z}, \lambda = 0.3$
- Initial conditions: $\boldsymbol{M}_0 = \boldsymbol{x}$, $\boldsymbol{J}_0 = (1, \sqrt{\frac{1}{6}}, \sqrt{\frac{1}{6}})^T$
- Neural network: First layer of 64 neurons, second layer of 32 neurons, sigmoid activation functions
- Optimization: Adam algorithm, 50000 iterations (2 minutes)
- Final loss: $1.0.10^{-4}$



(a) Components of $\boldsymbol{M}$



(b) Components of $\boldsymbol{J}$

Direct iLLG Neural Network integration

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

Direct formulation without any canonical transformation:

$$\begin{cases} \dot{\boldsymbol{M}} = \boldsymbol{M} \times \left(\boldsymbol{\Omega} + \lambda \dot{\boldsymbol{M}} + \ddot{\boldsymbol{M}}\right) \\ \boldsymbol{M}(0) = \boldsymbol{M}_0 \\ \dot{\boldsymbol{M}}(0) = \boldsymbol{V}_0 \end{cases} \tag{7}$$

Trial solution:

$$\boldsymbol{M}(t, \boldsymbol{P}) = \boldsymbol{M}_0 + t\boldsymbol{V}_0 + t^2 \mathcal{N}(t, \boldsymbol{P}) \tag{8}$$

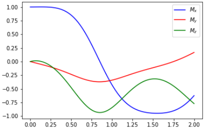
Loss function:

$$\mathrm{loss}[\boldsymbol{P}] = \frac{1}{N} \sum_{k=1}^{N} \left\| \dot{\boldsymbol{M}} - \boldsymbol{M} \times \left(\boldsymbol{\Omega} + \lambda \dot{\boldsymbol{M}} + \ddot{\boldsymbol{M}}\right) \right\|^2 (t_k, \boldsymbol{P}) \tag{9}$$

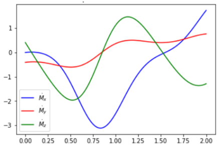
Results

Parameters:

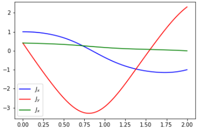
- $N = 32$, $0 \leq t \leq t_f = 2$
- Physical parameters: $\boldsymbol{\Omega} = 2\pi \boldsymbol{z}, \lambda = 0.3$
- Initial conditions: $\boldsymbol{M}_0 = \boldsymbol{x}$, $\boldsymbol{V}_0 = (0, -\sqrt{\frac{1}{6}}, \sqrt{\frac{1}{6}})^T$
- Neural network: First layer of 64 neurons, second layer of 32 neurons, sigmoid activation functions
- Optimization: Adam algorithm, 50000 iterations (2 minutes)
- Final loss: $3.0.10^{-4}$



(a) Components of $\boldsymbol{M}$ for the PINN solution



(b) Components of $\dot{\boldsymbol{M}}$ for the PINN solution



(c) Components of $\boldsymbol{J}$ for the PINN solution

PINNs versus symplectic method

Pros

- Few samples points in the training set are sufficient
- No canonical form of ODE is needed

Cons

- Tracking the error through backpropagation is harder (depends on network architecture)
- Describing the symmetries by the neural network is not as obvious as through the ODEs
- Computation is for the moment more time consuming

The cons represent conceptual challenges

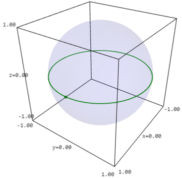
Thank you

Thank you

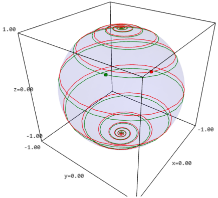
Backup slides

Solution of the LLG equation

- When $\lambda = 0$ and $\tau = 0$; precession in a constant field: exact solution $\forall t$ is given by Rodrigue's rotation formula
- When $\lambda \neq 0$ and $\tau = 0$; dampened motion in a constant field: exact solution is also known (a.k.a Magnus expansion) and here depicted



(a) Solution for $\lambda = 0$ and $\tau = 0$



(b) Solution for $\tau = 0$

Enforcing symmetries: conservation of the magnetization

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

- Symmetries of the system can be enforced by adding a constraint term in the loss function, along with a Lagrange multiplier μ.
- In the case of the conservation of the norm of the magnetization ($\|\boldsymbol{M}\|^2 = 1$):

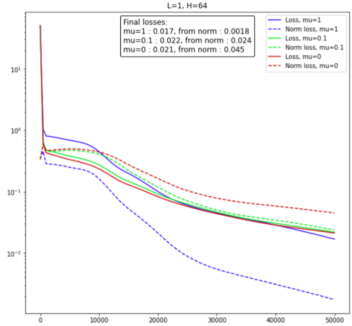
$$\begin{aligned} \mathrm{loss}(\boldsymbol{P}, \mu) = & \frac{1}{N} \sum_{k=1}^{N} \Big(\left\| \dot{\boldsymbol{A}}(t_k, \boldsymbol{P}) - \boldsymbol{F}(\boldsymbol{A}_0 + t_k \mathcal{N}(t_k, \boldsymbol{P})) \right\|^2 \\ & + \mu(\|\boldsymbol{M}\|^2 - 1)^2 \Big)(t_k, \boldsymbol{P}) \end{aligned} \tag{10}$$

Experiments with the Lagrange multiplier

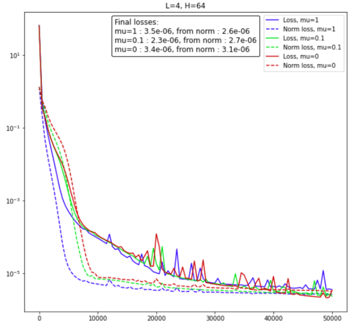
M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

Integrating equation (2) with a constraint term in the loss:

- Neural network: L hidden layers of H neurons, sigmoid activation functions
- Optimization: Adam algorithm, 50000 iterations



(a) Evolution of the loss with a smaller network



(b) Evolution of the loss with a deeper network

Experiments with the number of sample points

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

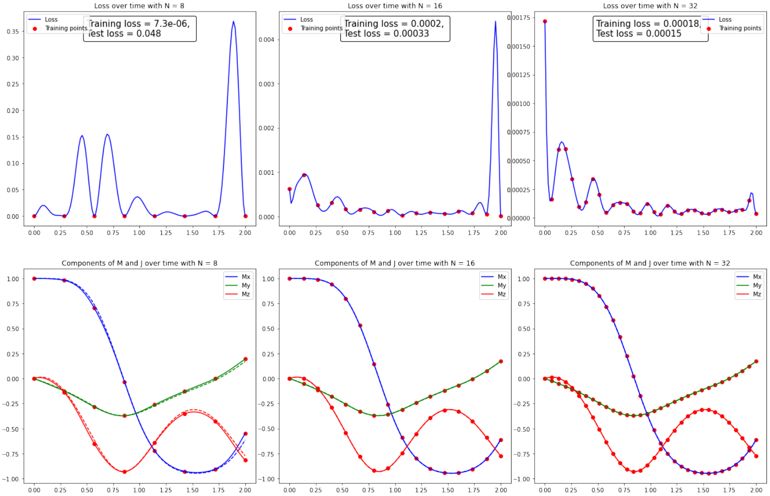


Figure: Experiments with the number of sample points

Rodrigue's formula

M.Carreau[1], S.Nicolis[2], B.Souaille[2] and P.Thibaudeau[3]

when

$$\frac{d\boldsymbol{M}(t)}{dt} = \boldsymbol{M}(t) \times \boldsymbol{\Omega} \tag{11}$$

If $\forall t$

$$\boldsymbol{M}(t) = \alpha(t)\boldsymbol{M}(0) + \beta(t)\frac{\boldsymbol{\Omega}}{\|\boldsymbol{\Omega}\|} + \gamma(t)\boldsymbol{M}(0) \times \frac{\boldsymbol{\Omega}}{\|\boldsymbol{\Omega}\|} \tag{12}$$

then $\chi \equiv \boldsymbol{M}(0) \cdot \frac{\boldsymbol{\Omega}}{\|\boldsymbol{\Omega}\|}$ and

$$\begin{aligned} \alpha(t) &= \cos(\|\boldsymbol{\Omega}\| t) \\ \beta(t) &= \chi(1 - \cos(\|\boldsymbol{\Omega}\| t)) \\ \gamma(t) &= \sin(\|\boldsymbol{\Omega}\| t) \end{aligned} \tag{13}$$