Numerical simulations of spin dynamics

Zdeněk Tošner

Charles University in Prague
Faculty of Science
Institute of Computer Science

Spin dynamics

behavior of spins
nuclear spin
magnetic moment
in magnetic field
quantum mechanical problem
\[
\frac{d}{dt} \rho = -i[H, \rho]
\]
density matrix Hamiltonian
Nuclear magnetic resonance

- local microscopic method with atomic resolution
- structure, chemical analysis, dynamics
- in solution as well as solid materials
- noninvasive imaging of living objects
- and much more...

Numerical simulations

... when analytical tools fail or become too complicated

- Extracting structural and dynamical properties from spectra fitting
- Insight into the underlying spin dynamics magnetization flow in TROSY-ST2-PT
- Performance tests of new pulse sequences
- Design of new pulse sequences
Optimal Control
Spin-state-selective coherence transfer driven by dipolar interactions

\[ I_x \rightarrow 2I_x, I_y \leftarrow I_y \]

\[ J_{\text{GdC}} \approx 55 \text{ Hz} \]

different samples different probes


Numerical simulations of spin dynamics

SIMPSON – virtual NMR spectrometer

SIMPSON version 3.1
Zdeněk Tošner, Rasmus Andeen, Malte Otte Nielsen, and Thomas Vosegaard
Released March 25, 2011

Downloads:
- SIMPSON 3.1 for Mac: Read the installation instructions for help with the installation.
- SIMPSON 3.1 for Linux: Read the installation instructions for help with the installation.
- SIMPSON 3.1 for Windows: Read the installation instructions for help with the installation.
- Source code: Note that the source code has been reorganized and omitting the configure script. Now it is just a simple makefile.

SIMPSON:

SIMPSON with Optimal Control:
**NMR interactions and Hamiltonian**

- molecule in solution
- in solid phase

![Diagram showing NMR interactions and Hamiltonian](image)

- Hamiltonian usually very sparse
- simple NMR experiments
- long periods of constant $\mathbf{H}$

- Hamiltonian still quite sparse
- complicated NMR experiments
- always time-dependent $\mathbf{H}$

---

**Basic approaches**

- **Hilbert space**
  \[
  \frac{d}{dt} \rho = -i[H, \rho] = -i\mathcal{L}\rho
  \]
  
  \[
  \rho(t) = U(t)\rho(0)U(t)\]

  \[
  U(t) = \prod_{j=0}^{n-1} \exp\{-iH(j\Delta t)\Delta t\}
  \]

  \[
  U \rightarrow 2^N \times 2^N
  \]

- **Liouville space**
  \[
  \rho(t) = P(t)\rho(0)\]

  \[
  P(t) = \prod_{j=0}^{n-1} \exp\{-i\mathcal{L}(j\Delta t)\Delta t\}
  \]

  \[
  \rho \rightarrow 2^N \times 2^N
  \]

  \[
  P \rightarrow 4^N \times 4^N
  \]

**Memory:**
- density matrix for 15 spins 1/2
  \[
  2 \times 8 \times 2^{15} \times 2^{15} = 16 \text{ GB}
  \]

**CPU time:**
- matrix exponential
- matrix – matrix or matrix – vector operations
- algorithms:
  - reuse of propagators
  - $\gamma$-COMPUTE
  - block-diagonalization
- parallelization:
  - powder averages
  - mathematical operations
Propagators

Matrix exponential

\[ U_j = \exp \{-iH_j\Delta t\} \]

diagonalization by eigen-decomposition

*golden standard, works always...*

Padé approximation

requires scaling & squaring
time consuming matrix division

Taylor expansion

requires scaling & squaring
only matrix multiplications, can be efficient when discarding “small” elements

Chebyshev expansion

requires scaling and shifting (eigenvalues between -1 and +1)
only matrix multiplications (possibly sparse x dense)

...?... Lanczos?

---

Constant Hamiltonian

\[ \rho(t) = \rho_0 \quad \text{(Hilbert space)} \]

\[ U = \text{Liouville space} \]

\[ P = \exp \{-i\mathcal{L}\Delta t\} \quad P \to 4^N \times 4^N \]

\[ \rho(t_n) = P^n \rho_0 \quad \text{(similarity transform)} \]

evaluate in propagator eigenbasis

*Krylov propagation* \[ e^{-i\mathcal{L}t} \rho \]
calculate propagator action without evaluating matrix exponential

---
**Constant Hamiltonian**

- Liouville space
- more tricks
- decomposition into non-interacting subspaces
- identify relevant subspaces for simulation
- smaller matrices
- skip empty/non-observable subspaces

**Time dependent Hamiltonian**

\[
\frac{d}{dt} \rho = -i [H, \rho] \\
\rho(t) = U(0, t) \rho(0) U^\dagger(0, t) \\
U(0, t) = U(t_N, t_{N-1}) \ldots U(t_2, t_1) U(t_1, 0) \\
U(t_2, t_1) = \exp \{-i H(t_1)(t_2 - t_1)\}
\]

Hilbert space

piece-wise constant approximation

Magic angle spinning & complicated rf irradiation
Hamiltonian periodicity

No periodicity – no way around re-evaluation of propagators

\[ H(t) \]

propagators are re-used

propagator of period

\[ U(0, T) = U_N \cdots U_2 U_1 \]

\[ \rho(t_j) \quad j = kN + l \]

\[ \rho(t_j) = U_1 \cdots U_1 U^k(0, T) \rho_0 (U^\dagger(0, T))^k U_1^\dagger \cdots U_1^\dagger \]

efficiently evaluated at eigenbasis of period propagator

---

Hamiltonian and MAS

single interaction

\[ H^\lambda = R_{20}^{\lambda, \text{LAB}}(t) X_{20}^{\lambda} = \sum_{n=-2}^{2} \frac{g_n^{(2)} R_{20}^{\lambda, \text{MOL}}(t)}{w_n(t)} \]

total interaction Hamiltonian

\[ H_{\text{TOT}} = \sum_{\lambda} H^\lambda = w_0(t)Q_0 + 2 Re \left\{ w_1(t)Q_1 + w_2(t)Q_2 \right\} \]

can this structure be used?

can this structure be used?

eigenvalues estimation

evolution of eigenvectors

interpolation

combination with rf irradiation

\[ H(t) = H_{\text{TOT}}(t) + H_{rf}(t) \]
Summary

Most time consuming operation is calculation of propagators

how to do it best?

Propagators are most efficiently re-used in their eigenbasis

eigendecomposition is necessary (or?)

Calculations in Liouville space can be faster due to matrix-vector operations,

possible subspace decomposition and exploiting high sparsity

not used due to necessity of eigendecomposition