GPU enhanced spin dynamics with Trotter-Suzuki evolutions

Axel D. Dente

dente@famaf.unc.edu.ar
Spin Dynamics

Study Spin dynamics (with Many-Body interactions) using Trotter-Suzuki

Objective: Accelerate Quantum Spin Dynamics simulations.

\[ U(t) = e^{-itH} \]

\[ H = \text{System Hamiltonian} \]
Trotter-Suzuki Approximation

\[ \tilde{U}(t) = e^{-itH} = e^{-it(H_1 + \ldots + H_K)} = \lim_{m \to \infty} \left( \prod_{k=1}^{K} e^{-\frac{itH_k}{m}} \right)^m \]

First order Approximation

\[ \tilde{U}_1(t) = e^{-itH} = e^{-\frac{itH_1}{m}} \ldots e^{-\frac{itH_K}{m}} \]

High orders Approximations

\[ \tilde{U}_2(t) = \tilde{U}_1^\dagger(-t/2)\tilde{U}_1(t/2) \]

\[ \tilde{U}_4(t) = \tilde{U}_2(at)\tilde{U}_2(at)\tilde{U}_2((1-4a)t)\tilde{U}_2(at)\tilde{U}_2(at) \]

\[ a = 1/(4 - 4^{1/3}) \]

H. De Raedt and K. Michielsen, Handbook of Theoretical and Computational Nanotechnology (American Scientific Publishers)
Trotter-Suzuki Approximation

Many-Body spin Hamiltonian

\[ H = \sum_{j=1}^{N} \sum_{\alpha=x,y,z} h_{j}^{\alpha} S_{j}^{\alpha} + \sum_{j,k=1}^{N} \sum_{\alpha=x,y,z} J_{j,k}^{\alpha} S_{j}^{\alpha} S_{k}^{\alpha} \]

Single Spin Operators

\[
\exp \left( -it \left[ \sum_{j=1}^{N} \sum_{\alpha=x,y,z} h_{j}^{\alpha} S_{j}^{\alpha} \right] \right) = \prod_{j=1}^{N} \exp \left( -it \sum_{\alpha=x,y,z} h_{j}^{\alpha} S_{j}^{\alpha} \right)
\]
Trotter-Suzuki Approximation

Two spin Operators

\[ H^z = \sum_{j,k=1}^{N} J_{j,k}^z S_j^z S_k^z \]

It only modify the phase of each states (trivial parallelization)

Using Rotations we can rewrite \( H^x \) and \( H^y \)

\[ Y = \prod_{j=1}^{N} Y_j \quad e^{-it H^x} = \overline{Y} Y e^{-it H^x} \overline{Y} \overline{Y} = \overline{Y} \exp \left( -it \sum_{j,k=1}^{N} J_{j,k}^x S_j^z S_k^z \right) Y \]

\[ X = \prod_{j=1}^{N} X_j \quad e^{-it H^y} = \overline{X} X e^{-it H^y} \overline{X} = X \exp \left( -it \sum_{j,k=1}^{N} J_{j,k}^y S_j^z S_k^z \right) \overline{X} \]
Trotter-Suzuki Algorithm

Two spin Operators

X90 Rotation → Phases in Z → X-90 Rotation

Trivial Parallelization

Need Parallelization

Need Parallelization
Trotter-Suzuki Algorithm

\[ X = \prod_{j=1}^{N} X_j \rightarrow \exp(-itS_j^\alpha) \]

N=4 Spins and select j=2

Individual Rotations are Parallelizable
GPU implementation

Intel DX58SO motherboard with
Intel Core i7-950 3.07GHz 16GB DDR3 1066MHz,

two Tesla C2070 boards
(448 cores, 6GB GDDR5),
donated by NVIDIA.
Max of 28 Spins in one GPU (limited by the GPU memory)
CPU vs GPU

Max of 28 Spins in one GPU (limited by the GPU memory)
CPU vs GPU

Max of 28 Spins in one GPU (limited by the GPU memory)
Conclusions

Parallelized the Trotter-Suzuki CPU algorithm for spin dynamics and implement it on GPUs.

Compared TWO compilers (Fortran and C)

Obtain a maximum Speedup of 90x (compared with a single core) and 36x (compared with 4 cores)

Perspectives

Develop a new code based on this algorithm for NMR simulations
Questions?

Axel D. Dente