Magnus Gustafsson
Parallel Lanczos-based exponential integrators for quantum dynamics

Division of Scientific Computing
Uppsala University
751 05 Uppsala Sweden
magnus.gustafsson@it.uu.se
Katharina Kormann
Sverker Holmgren

The time-dependent Schrödinger equation (TDSE) describes the quantum dynamical nature of molecular processes. Simulations are, however, computationally very demanding due to the curse of dimensionality. With our MPI and OpenMP parallelized code, HAParaNDA (cf.[1]), we are able to accurately solve the full Schrödinger equation, currently in up to five dimensions using a medium-size cluster.

The d-dimensional TDSE reads

$$
i\hbar\psi(x,t) = \left(-\sum_{i=1}^{d} \frac{\hbar^2}{2m_i} \frac{\partial^2}{\partial x_i^2} + V(x,t) \right) \psi(x,t), \quad \psi(x,0), x \in \mathbb{R}^d = \psi_0,
$$

where $\psi(x,t)$ denotes the wave packet and the potential $V(x,t)$ describes interactions within the system. Currently, we only consider localized molecules on sufficiently large d-orthotopes and close the system by periodic boundary conditions. We introduce a spatial grid and compute the derivatives with an pth order finite difference (FD) method (8th order in the reported experiments). For a time-independent potential, the solution of the semi-discrete system is given by

$$
u(t) = e^{-\frac{i}{\hbar}H}u(0),
$$

where H is the spatially discretized differential operator and u is the discrete wave packet. For time-dependent potentials, the exponential form can still be exploited on small time intervals but the Hamiltonian has to be time-averaged using e.g. the Magnus expansion (see [3] for details). The matrix exponential can be computed efficiently in serial using the Lanczos algorithm. In a parallel environment, the fact that two inner products have to be computed in each iteration hampers scalability. Kim and Chronopoulos [2] discuss two modifications of the Lanczos algorithm with reduced communication. Rearranging the computations makes it possible to compute both inner products at the same point, reducing the number of synchronizations to one per iteration. In the s-step Lanczos method, blocks of s consecutive steps are executed with only one synchronization step required for each block. We have transferred these ideas to

the Lanczos propagator method. A second hassle with using the Lanczos procedure is that the number of iterations has to be chosen with care. Numerical round-off errors cause instabilities. We therefore propose to choose the number of steps adaptively and to make sure that the iterations are stopped once the residual is small enough. For a discussion of error estimation in the propagation of the TDSE, we refer to [4].

In order to demonstrate the performance of a massively parallel simulation of the TDSE based on a FD-Lanczos discretization, we have conducted several simulations on a medium-size cluster, consisting of 316 nodes. Each node is equipped with dual quad-core Intel Nehalem CPU and 24 GB of DRAM. The nodes are interconnected by an InfiniBand fabric. For the experiments, we have considered the harmonic oscillator and the Henon–Heiles potential. The analytical solution is known for the harmonic oscillator, so we are able to verify the correctness and the accuracy of our numerical results. Table 1 shows the scalability of the results for the 4D harmonic oscillator. Comparing the variants of the Lanczos algorithm, we see that the performance can indeed be improved by reducing the communication. We are currently working on an implementation of the s-step method and hope to further improve the scalability in that way. The Henon–Heiles potential is a common test case for high-dimensional simulations. We have simulated this problem in five dimensions on a grid with 60^5 points over 10000 time steps. We used the standard Lanczos algorithm and have chosen the size of the Krylov space adaptively. The simulation was performed on 1024 cores in 19 hours.

Table 1: Scalability of standard Lanczos (L1) and the few-synchronizations variant (L2). The wall time is reported in units of 1000 s. Each node solves a 60^4 problem and the largest problem has $240 \times 240 \times 120 \times 120$ unknowns.

# Cores	8	16	32	64	128	256	512
L1	3.92	4.31	4.37	4.55	4.72	4.87	5.16
L2	3.79	4.06	4.26	4.33	4.43	4.72	4.83

REFERENCES

[1] M. Gustafsson and S. Holmgren. An implementation framework for solving high-dimensional PDEs on massively parallel computers, to appear in: *Proceedings of ENUMATH 2009*, Uppsala, Sweden

[2] S.K. Kim and A.T. Chronopoulos. A class of Lanczos-like algorithms implemented on parallel computers, *Parallel Comput.* **17** (1991)

[3] K. Kormann, S. Holmgren, and H.O. Karlsson. Accurate time propagation for the Schrödinger equation with an explicitly time-dependent Hamiltonian, *J. Chem. Phys.* **128** (2008)

[4] K. Kormann and A. Nissen. Error Control for Simulations of a Dissociative Quantum System, to appear in: *Proceedings of ENUMATH 2009*, Uppsala, Sweden