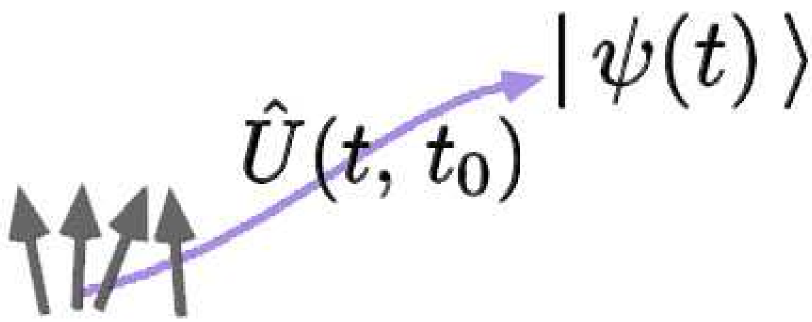
Parallel **Si**mulation of **Q**uantum **Co**herent Dynamics by
Parameterized **A**rbitrary **T**ime-dependent **H**amiltonians

PSiQCoPATH

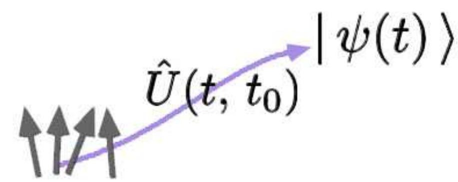
Eric Fellheimer and Mark Rudner

MIT 6.338J, Spring 2005



Overview of Presentation

1. Quantum Mechanics in 30 Seconds
2. How to Simulate Quantum Evolution
3. PSiQCoPATH: A look inside the mind of a serial killer
4. Sanity Check: Spin Dynamics
5. Adiabatic Quantum Computation in 20 seconds
6. Results II: PSiQCoPATH attacks Satisfiability
7. Conclusions

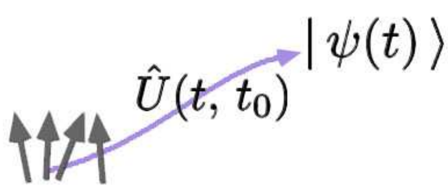


Quantum Mechanics in 30 Seconds

State of quantum system completely specified by a unit vector in complex vector (Hilbert) Space

Observable properties of the system are described by the action of Hermitian operators on the Hilbert space

Time evolution of quantum states is **unitary** (must preserve the norm)



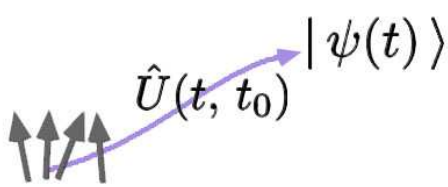
Quantum Mechanics in 30 (More) Seconds

The Hamiltonian or "energy" operator of the system is the generator of time evolution

All dynamics are governed by the Schrödinger equation

$$i \frac{d}{dt}|\psi(t)\rangle = \widehat{H}|\psi(t)\rangle$$

The Hamiltonian can be time-independent (easy) or time-dependent (hard)



How to Simulate Quantum Evolution

Check this out: Let

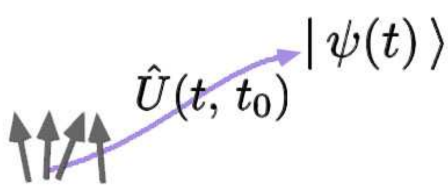
$$|\psi(t)\rangle = \widehat{U}(t,\, t_0)|\psi(t_0)\rangle$$

This “time evolution operator” satisfies the Schrödinger equation

$$i\frac{d}{dt}\widehat{U}(t,\, t_0) = \widehat{H}(t)\,\widehat{U}(t,\, t_0)$$

The following property is also satisfied:

$$\widehat{U}(t,\, t_0) = \widehat{U}(t,\, t_1)\,\widehat{U}(t_1,\, t_0)$$



How to Simulate Quantum Evolution

Now PSiQCoPATH has a natural way to hack up the problem

$$\widehat{U}(t,\, t_0) = \widehat{U}(t,\, t-\varepsilon) \cdots \widehat{U}(t_0+\varepsilon,\, t_0)$$

Expand each time step’s evolution operator in Taylor Series

$$\widehat{U}(t+\varepsilon,\; t) = 1 \;+\; \varepsilon\, \frac{d\widehat{U}}{dt}\bigg|_t \;+\; \frac{\varepsilon^2}{2!}\, \frac{d^2\widehat{U}}{dt^2}\bigg|_t + \cdots$$

Derivative terms related to Hamiltonian via Schrödinger Equation

For sufficiently small ε it will be a good approximation to truncate the series after relatively few terms

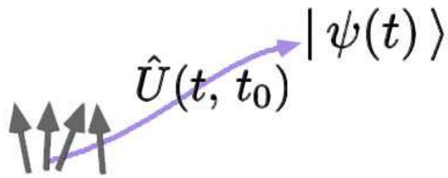
PSiQCoPATH: A look inside the mind of a serial killer

First order sucks, second order works ok, third order was too annoying to bother coding up for now

Time evolution operator of each step calculated independently

Total time evolution operator found by parallel prefix (order matters!)

Alternative approach to parallelization: evolve a single pure state through a series of row-distributed matrix-vector multiplications



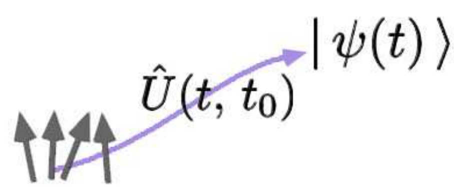
PSiQCoPATH: Multiple Personality Disorder

A Pair of Paths to (almost) Perfect Parallelization

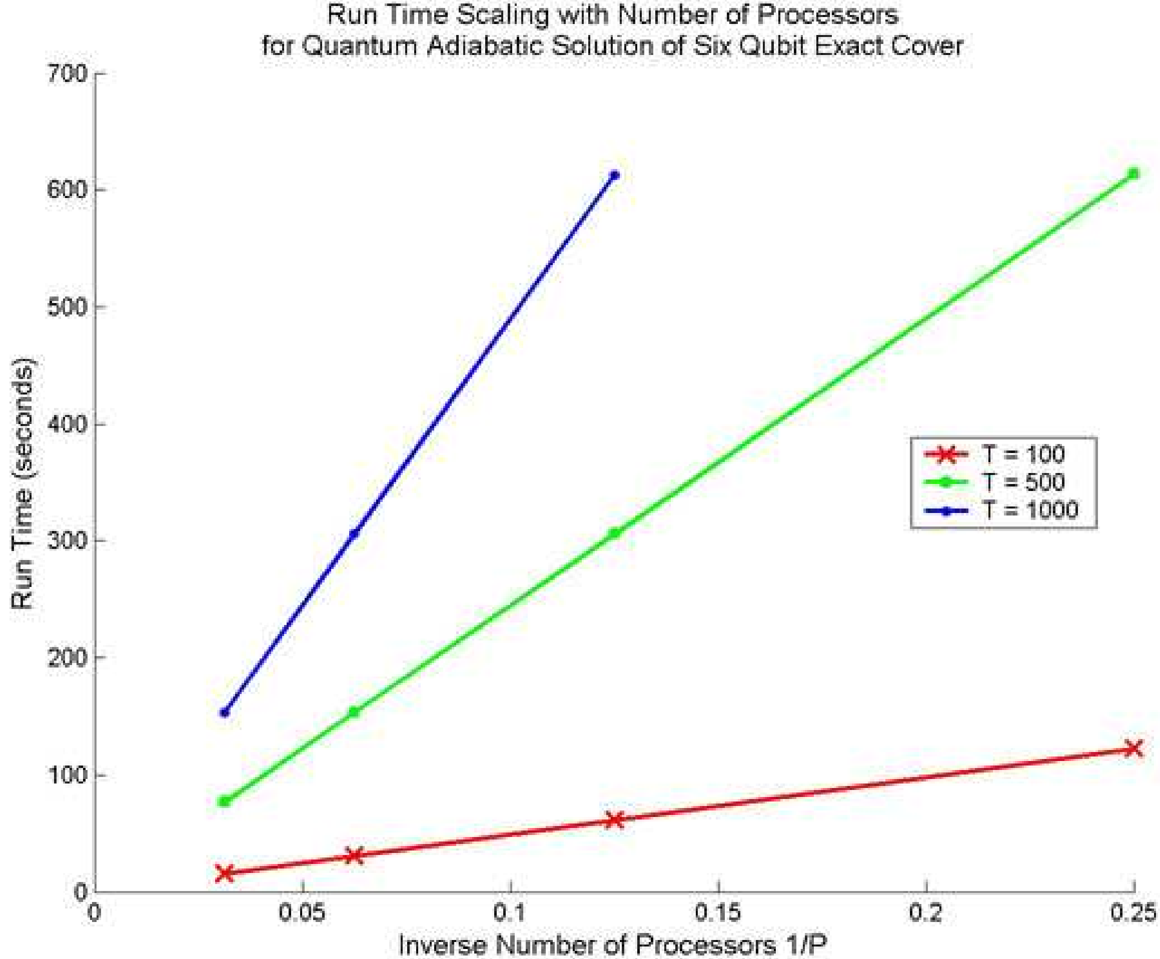
	Row-Distributed S.S.	Parallel Prefix Full U
Output	Time evolution of a single (pure) initial state	Full time evolution operator of the system
Scaling (time)	$T \cdot \left(\frac{N^2}{p} + C(N, p) \right)$	$\frac{2T}{p} \cdot N^3$
Scaling (space)	$N^2 + T \cdot N$	$T \cdot N^2$
Uses	Appropriate when interested in specific evolution of some state	Appropriate for general evolution of mixed states or many initial states

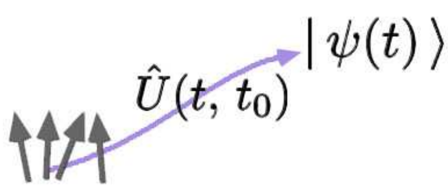
N = dimension, T = length of simulation, p = number of processors, C = communication cost

$$C(N, p) \leq \mathcal{O}\left[N \cdot (p - 1)\right]$$



PSiQCoPATH: Performance (Prefix Mode)





PSiQCoPATH: Getting to Know the Victim

Problem: How to deal with arbitrary time-dependent Hamiltonians

Solution: Parameterize

$$\widehat{H}(t) = \sum_k \alpha_k(t)\, \widehat{H}_k$$

Here, $\{\widehat{H}_k\}$ is a set of time-independent Hermitian matrices

For an N-dimensional Hilbert space, $N^2 - 1$ terms are sufficient to construct **any** time-dependent Hamiltonian

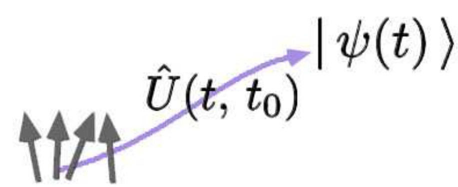
PSiQCoPATH: Premeditating the Attack

Terms in time evolution operator involve $\left[\widehat{H}(t)\right]^2$, etc

With parameterized form, pre-compute products

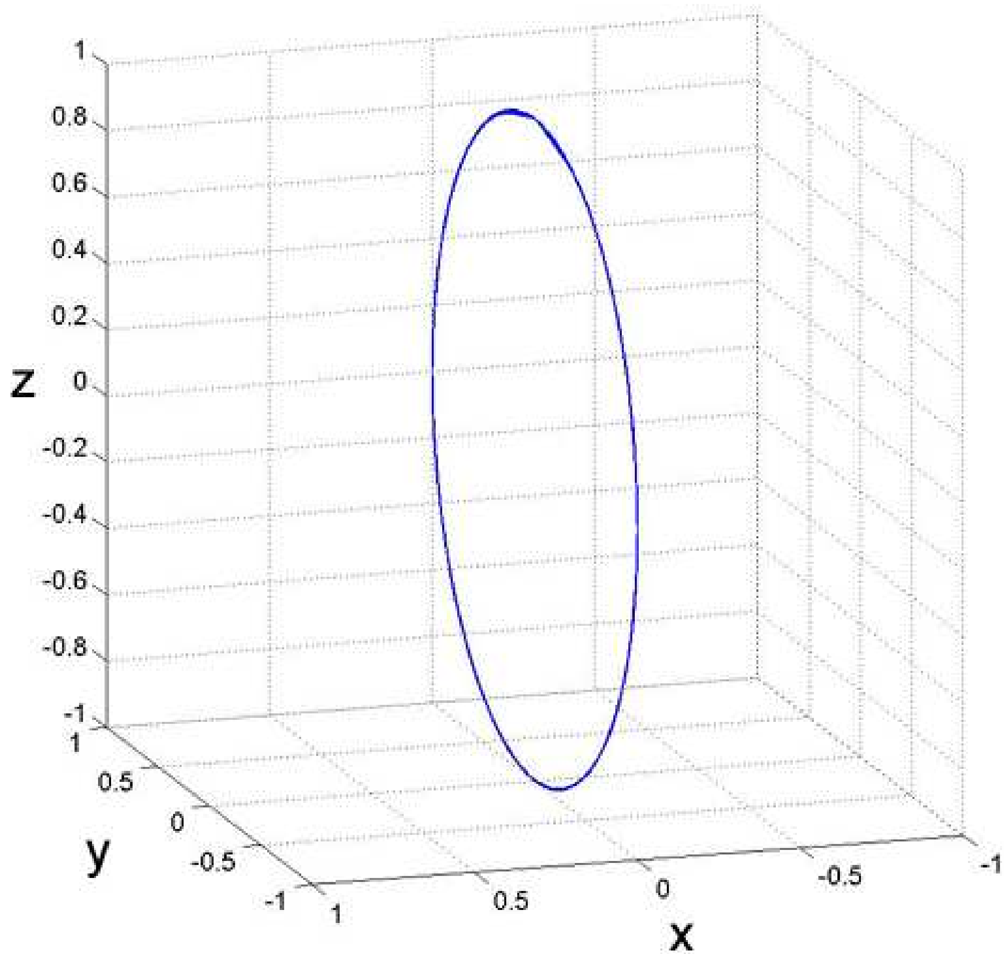
$$\widehat{M}_{ij} = \widehat{H}_i \widehat{H}_j$$

Then calculation of $\widehat{U}(t+\varepsilon,\, t)$ involves only matrix addition
with time-dependent coefficients $\alpha_k(t),\ \dot{\alpha}_k(t), \ldots$



PSiQCoPATH: Sanity Check I

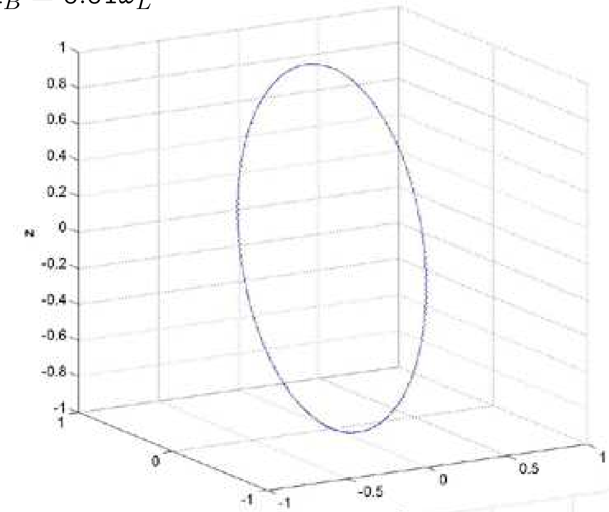
Spin-1/2 Moment in Static Magnetic Field



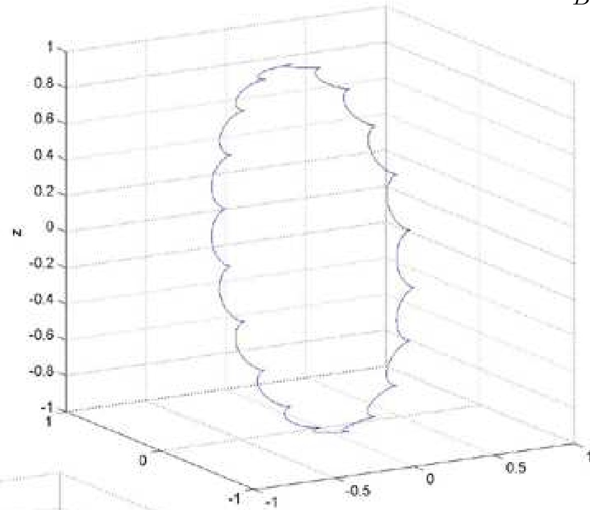
PSiQCoPATH: Sanity Check II

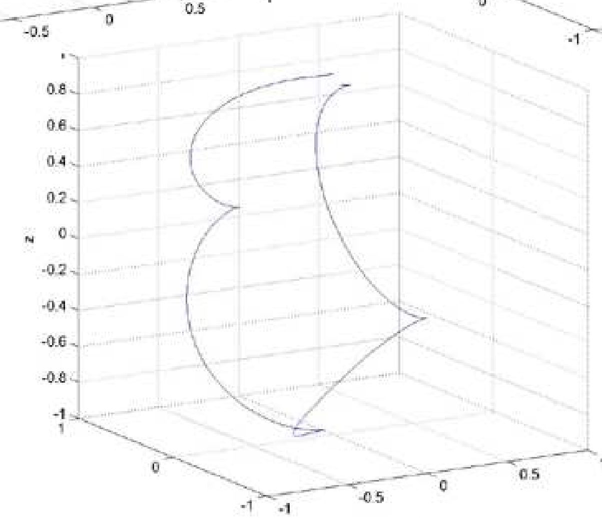
Spin-1/2 Moment in Slowly Rotating Field

$\Omega_B = 0.01\omega_L$



$\Omega_B = 0.1\omega_L$

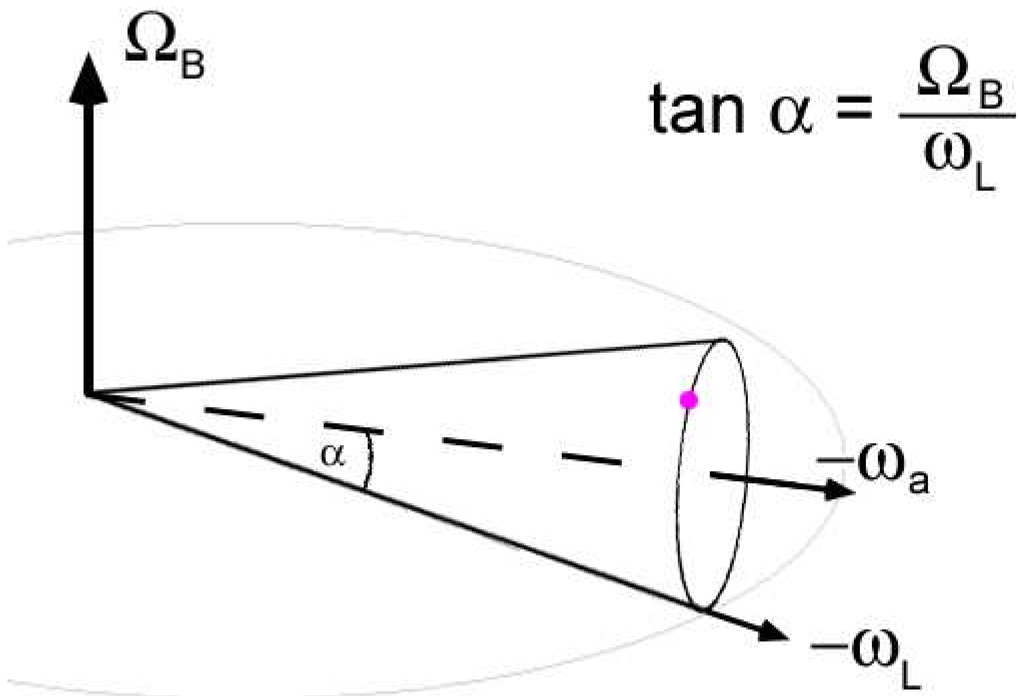


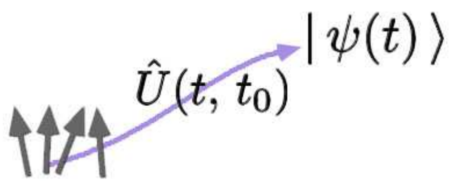


$\Omega_B = 0.5\omega_L$

PSiQCoPATH: Sanity Check II

Spin-1/2 Moment in Slowly Rotating Field





PSiQCoPATH: Sanity Check II

Spin-1/2 Moment in Slowly Rotating Field

Lesson 1: Time step should be small compared to shortest natural timescale $(\omega_L)^{-1}$

Lesson 2: Time ordering is important (already known, but fixed the bug)

Lesson 3: Physics is fun!

Cycloid behavior unexpected, but fully understood in retrospect

Problem maps to that of a point on rim of a cone rolling on flat surface

Adiabatic Quantum Computation in 20 Seconds

Adiabatic Theorem: If Hamiltonian varies *slowly enough*, system initially in instantaneous ground state of $\widehat{H}(t)$ will remain close to instantaneous ground state of $\widehat{H}(t)$ for all time

Idea (Farhi et al.): Encode solution to NP-complete problem as ground state of some Hamiltonian

Start in ground state of a different Hamiltonian with easy to prepare ground state

Adiabatically morph to "problem Hamiltonian" and read out answer

For the whole story see arXiv:quant-ph/0104129

PSiQCoPATH Attacks Satisfiability

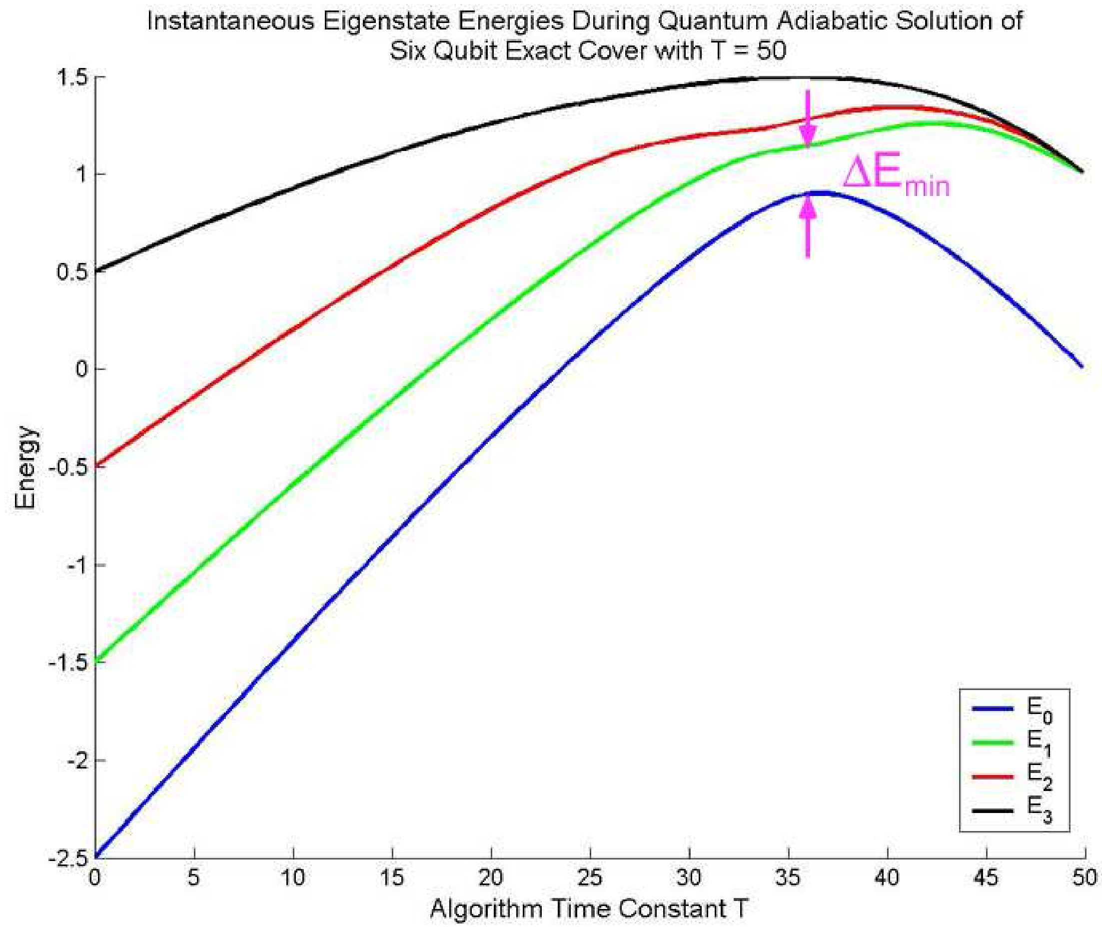
Exact Cover is an NP-Complete version of the satisfiability problem

N bits, m 3-bit constraint clauses of the form

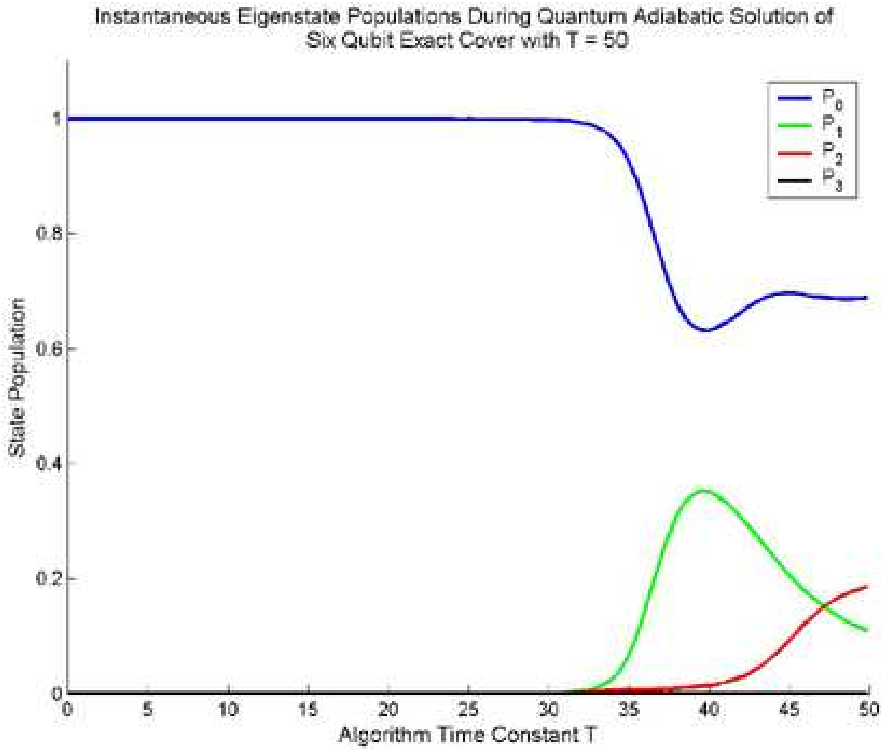
$$z_i + z_j + z_k = 1$$

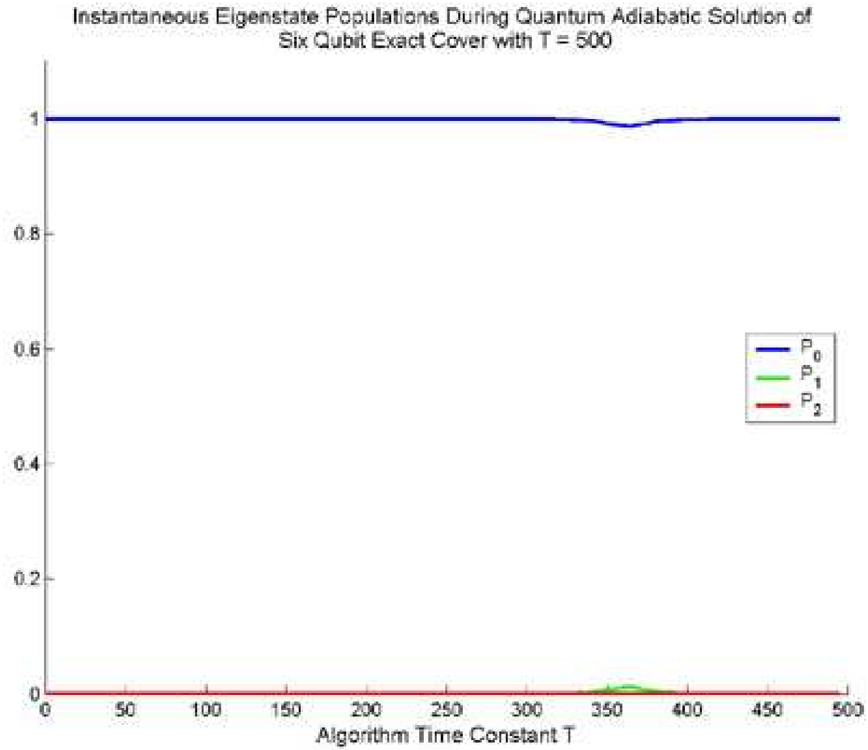
PSiQCoPATH has simulated the quantum adiabatic solution to randomly generated instances of Exact Cover for 4, 6, and 8 qubits

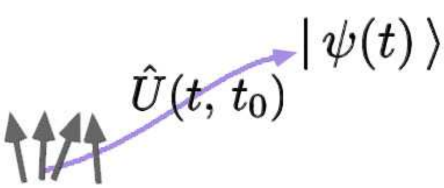
PSiQCoPATH Attacks Satisfiability



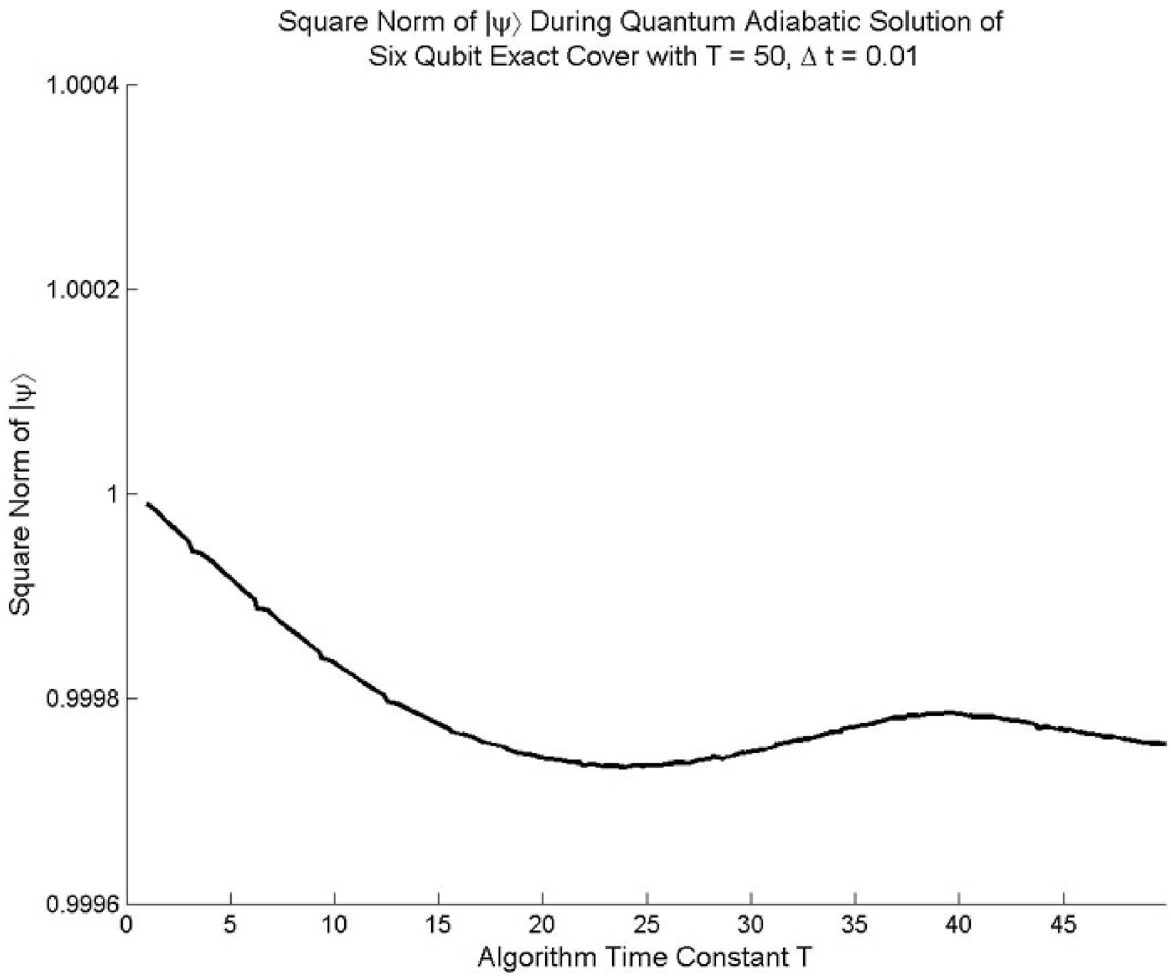
PSiQCoPATH Attacks Satisfiability

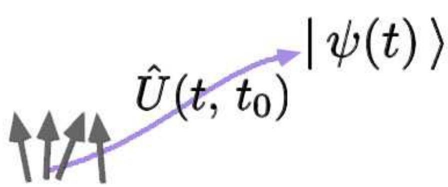






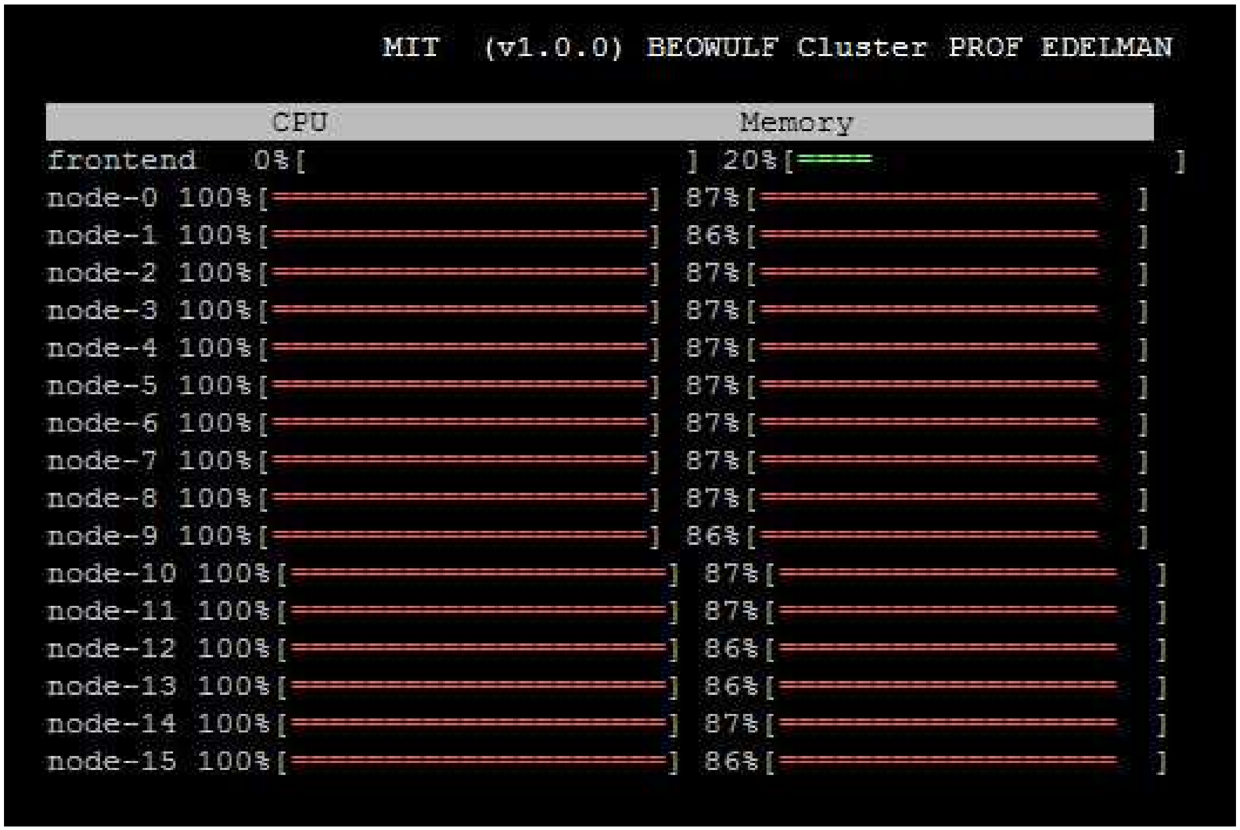
PSiQCoPATH Attacks Satisfiability

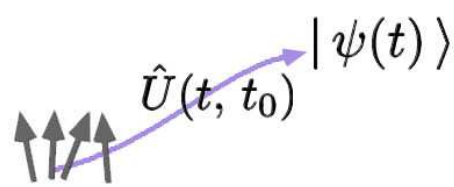




PSiQCoPATH is to the Limit

Eight Qubit Exact Cover, T = 250, Δt = 0.02





Conclusions

Proof of concept – showed that we could create efficient parallel algorithms for evolution of large quantum systems

Better memory management, i.e. storing only as many operators as desired for output, would allow larger systems/longer runs on parallel prefix code

However, parallelization allows simulation of larger quantum systems in less time than serial code