SUMMER INTERNSHIP

Argonne NATIONAL LABORATORY

Implementing the Hellman Feynman theorem for the calculation of molecular dynamics using Qiskit and DiNT.

PRESENTER
Alberto Acevedo

SCHOOL
The University of Arizona
Dept of Applied Mathematics

ARGONNE ADVISOR
Yuri Alexeev

August 1st 2019

Quantum Advantage?

- Quantum Computers will eventually replace classical computers with their superior computational power.
- Nature follows the rules of Quantum Mechanics therefore Quantum Computers will bring forth better understanding of matter.
- Breaking RSA via Shor's algorithm.
- Providing stronger encryption methods.
- Protein folding…etc

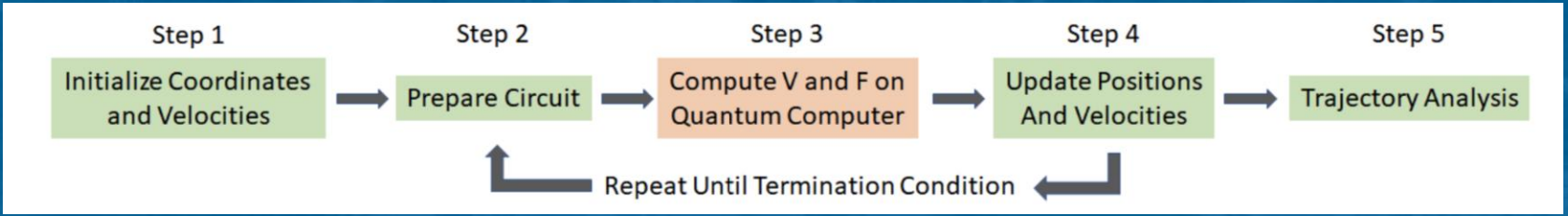
“Chemistry is quantum computing’s killer app”

- IBM looking for a Quantum “Killer App”.
- Killer App: A computer program that is so necessary or desirable that it improves the core value of some larger technology.
- Chemistry is thought to be this “Killer App”.
- Molecular dynamics is the subfield of chemistry that we analyze.
- IBM created Qiskit, an open-source quantum computing framework for leveraging today’s quantum processors in research, education, and business.

Qiskit & DiNT

- We can implement Qiskit framework with python.
- Qiskit has a framework specially designed for chemistry. i.e. Qiskit Chemistry.
- We use Qiskit in conjunction with DiNT (Direct Nonadiabatic Trajectories) a code for non-Born-Oppenheimer molecular dynamics. Developers are Ahren W. Jasper et al (Sandia/Argonne Labs).

The program for computing molecular trajectories using Qiskit and DiNT.



- All steps except 2 & 3 are accomplished by DiNT.

- We use our Qiskit/python program to compute the energies and forces associated with our molecule.

Going more in depth.

Second Quantization.

$$(H_2, (x_1, y_1, z_1, x_2, y_2, z_2))$$
$$\rightarrow \int dr dr' a_r^\dagger \langle r|H_0|r'\rangle a_{r'} + \frac{1}{2}\int dr dr' d'' dr''' a_{r'}^\dagger a_{r''}^\dagger \langle r' r''|V_{int}|r'' r'''\rangle a_{r''} a_{r'''}$$

Mapping to a qubit operator with JW transform for example.

$$a_j = \Pi_{i=1}^{j-1} \sigma_{z_i} \otimes (\sigma_{x_j} + i\sigma_{y_j})$$

UCCSD Ansatz is used to approximate energy using Qiskit's Variational Eigen Solver and the Pauli Hamiltonian produced after JW transform.

$$E = \langle \psi_{opt}|H_{Qiskit}|\psi_{opt}\rangle$$

Approximate gradient, for dynamçics, with Hellmann Feynman theorem.

$$\frac{dE_\lambda}{d\lambda} = \langle \psi_\lambda | \frac{dH_\lambda}{d\lambda} | \psi_\lambda \rangle \approx \langle \psi_\lambda | \frac{\Delta H_\lambda}{\Delta \lambda} | \psi_\lambda \rangle$$

Estimating gradient.

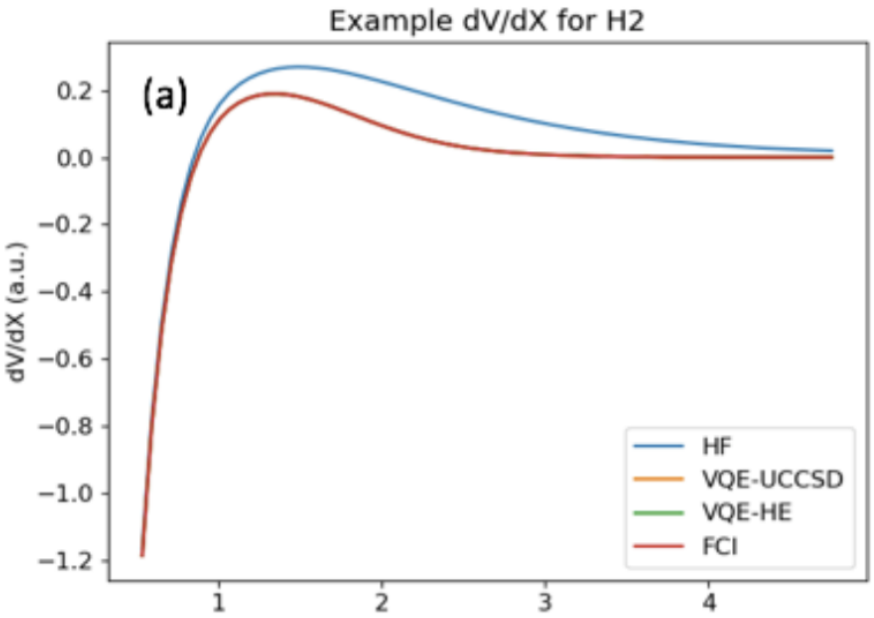
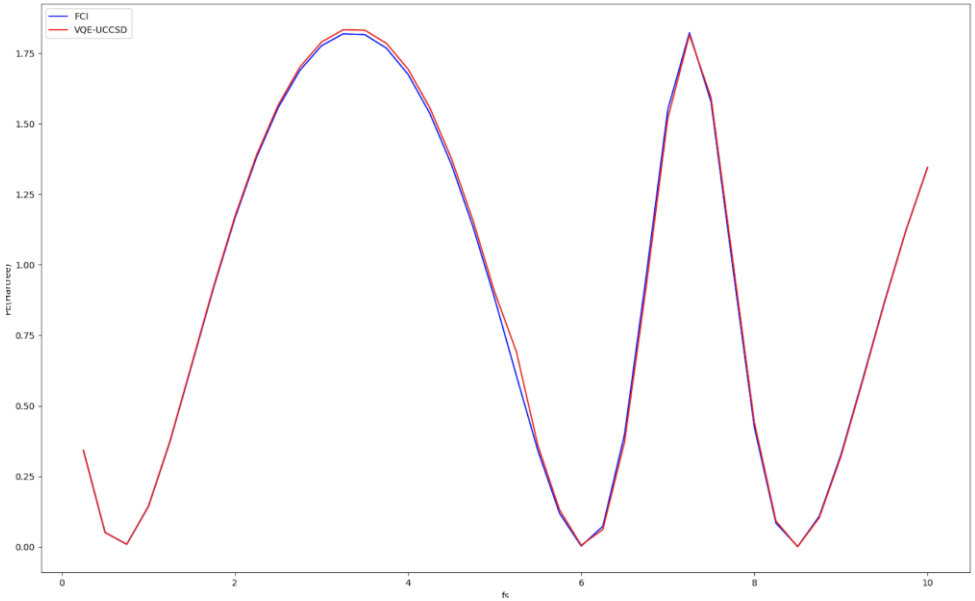


Image by Matt J. Otten

HF: Hartree-Fock
VQE-UCCSD: Unitary coupled clusters.
VQE-HE:IBM's Hardware Efficient
FCI: Full Configuration Interaction

TRAJECTORIES DINT+VQE-UCCSD VS EXACT.

Using simulator



Choosing the proper offset for numerical differentiation, COBYLA optimizer tolerance and time step size we can attain very reliable results. Some cost us more computational time than others so we must pick the optimal set.

PARAMETERS

-Recall that we are estimating derivatives. $\frac{dE_\lambda}{d\lambda} = \langle\psi_\lambda| \frac{dH_\lambda}{d\lambda} |\psi_\lambda\rangle \approx \langle\psi_\lambda| \frac{\Delta H_\lambda}{\Delta\lambda} |\psi_\lambda\rangle$

- VQE uses COBYLA optimizer. Smaller tolerance leads to more COBYLA iterations. Want to minimize iterations.

- The smaller the time step the higher the number of VQE computations necessary.

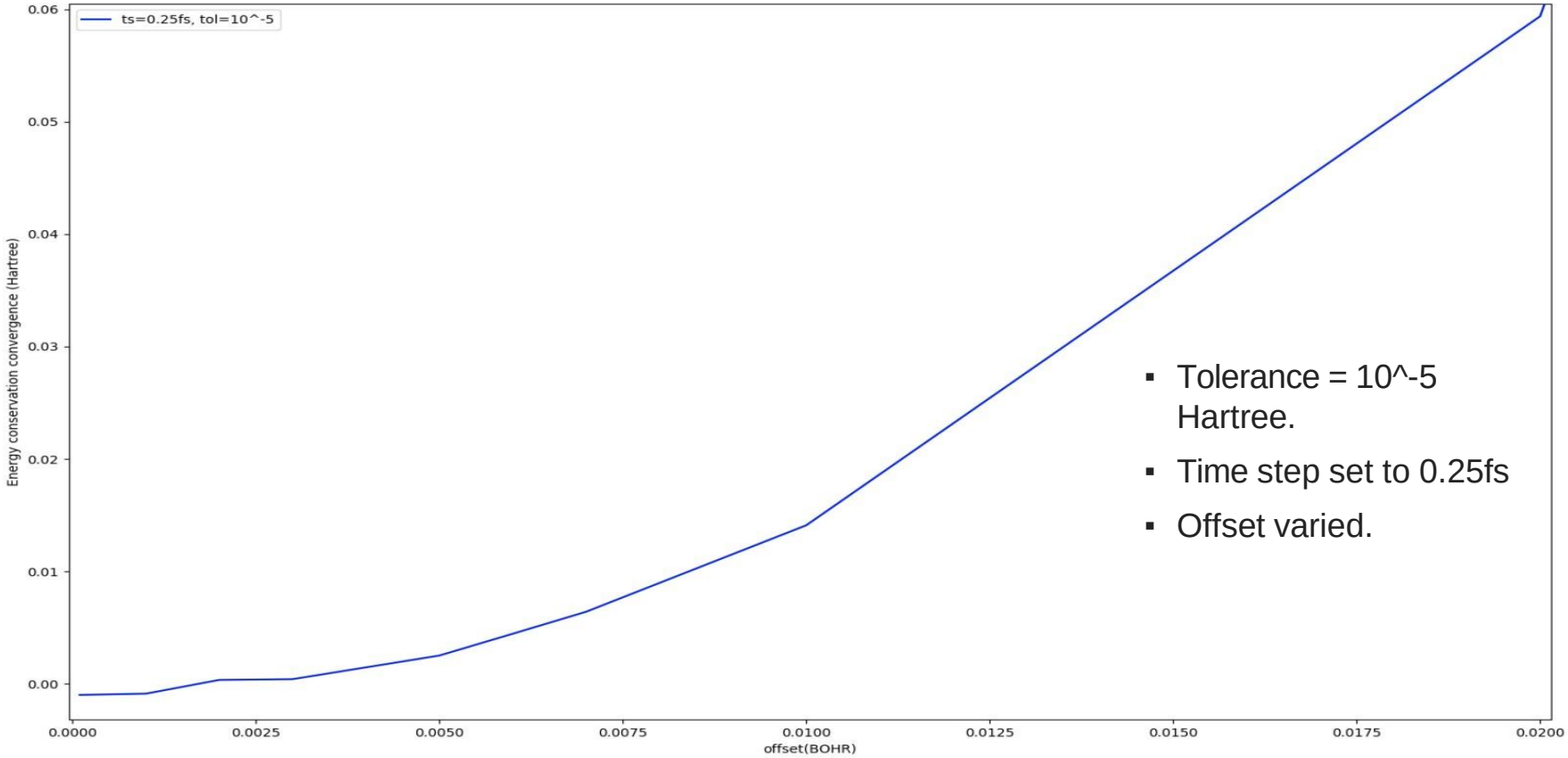
SIEVE FOR FINDING OPTIMAL PARAMETERS.

- Choose Geometry.
- Fix COBYLA MAXITER parameter to a high value.
- Compute gradient with varying values for offset and tolerance. Tolerance varies over [10^-6, 10^-4] and offset varies over [10^-8, 10^-5].
- Choose biggest offset and tolerance pair that yields a relative error the order of 10^-6 in the normed difference of optimal gradient vs approx.
- Compute trajectories with varying time steps. 0.1fs, 0.25fs and 0.5fs for example.

GRADIENT ANALYSIS H2 (1,0,0,0,0,0)

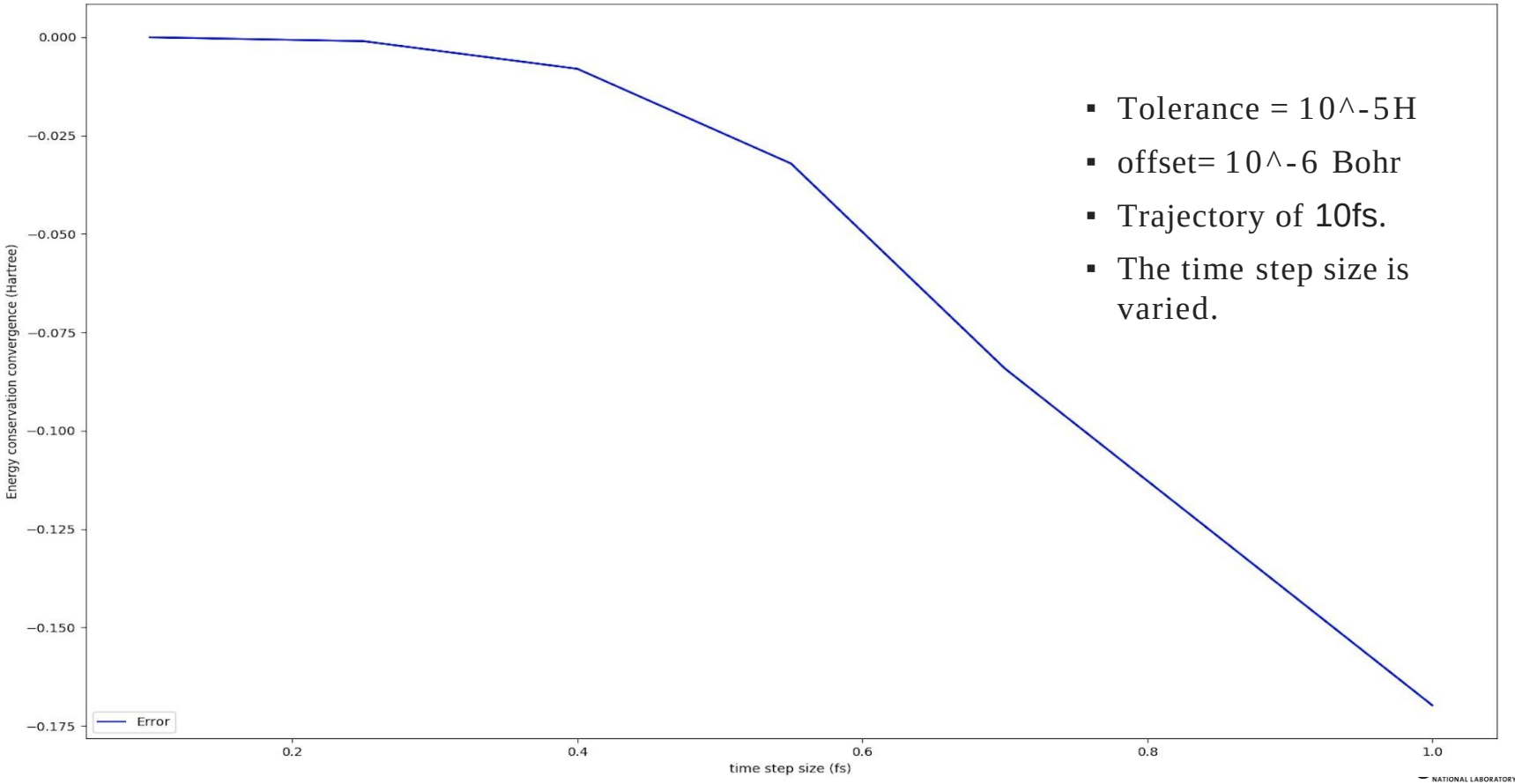
tol/dx	E-05	E-06	E-07	E-08
E-0.04	[-1.9e-06, -1.9e-06, 0.38017, -1.9e-06, -1.9e-06, -0.38016]	[-1.9e-07, -1.9e-07, 0.38017, -1.9e-07, -1.9e-07, -0.38016]	[-1.3e-08, -1.3e-08, 0.38016, -1.3e-08, -1.3e-08, -0.38016]	[0.0, 0.0, 0.38016, 0.0, 0.0, -0.38016]
E-05	[-1.9e-06, -1.9e-06, 0.38017, -1.9e-06, -1.9e-06, -0.38015]	[-1.9e-07, -1.9e-07, 0.38016, -1.9e-07, -1.9e-07, -0.38016]	[-8.9e-09, -8.9e-09, 0.38016, -8.9e-09, -8.9e-09, -0.38016]	[0.0, 0.0, 0.38016, 0.0, 0.0, -0.38016]
E-06	[-1.9e-06, -1.9e-06, 0.38017, -1.9e-06, -1.9e-06, -0.38015]	[-1.9e-07, -1.9e-07, 0.38016, -1.9e-07, -1.9e-07, -0.38016]	[-8.9e-09, -8.9e-09, 0.38016, -8.9e-09, -8.9e-09, -0.38016]	[0.0, 0.0, 0.38016, 0.0, 0.0, -0.38016]

ENERGY CONSERVATION VS OFFSET SIZE.



- Tolerance = 10^-5 Hartree.
- Time step set to 0.25fs
- Offset varied.

ENERGY CONSERVATION VS TIME STEP SIZE.



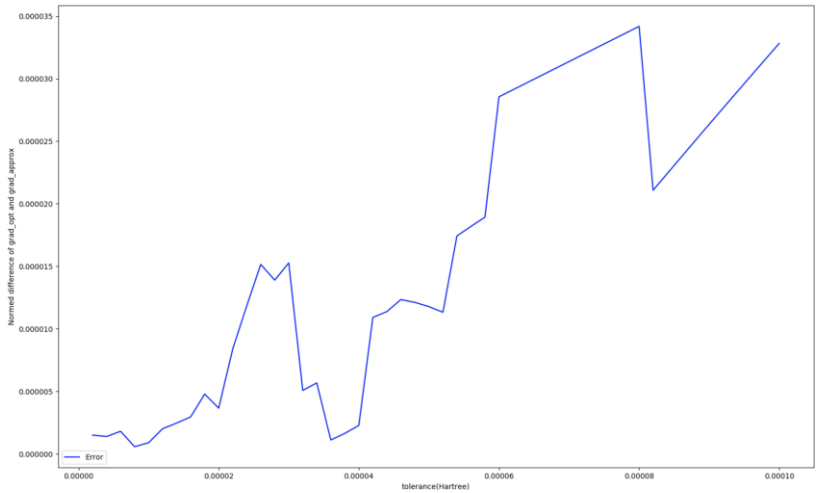
- Tolerance = 10^-5H
- offset= 10^-6 Bohr
- Trajectory of 10fs.
- The time step size is varied.

Gradient Analysis continued

dx/tol	0.0005	0.0002	0.0001	0.00005	0.00001	0.000005	0.000001
0.01	0.0076497782	0.007298202929218627	0.007342802458487744	0.007324394381631327	0.007314793767388214	0.0073156387	0.0073138621
0.008	0.0073630552	0.0069960422	0.0070426882	0.007023438752397946	0.0070133975	0.0070142813	0.007012422949060455
0.005	0.0070393429	0.006653309	0.00670249177670277	0.0066821999	0.0066716124	0.0066725443	0.0066705847
0.004	0.00696251363040878	0.006571690074850403	0.0066215144	0.006600959	0.006590233347067907	0.0065911775	0.0065891922645655516
0.003	0.0069021691	0.006507503940919357	0.0065578437	0.0065370766	0.00652624	0.0065271938	0.0065251881
0.002	0.0068587423						
0.001	0.0068325544	0.0064333675	0.0064843148	0.0064632982	0.006452330724751785	0.0064532962	0.0064512661
0.0001	0.0068238904	0.006424133802260428	0.00647515773118108	0.0064541097	0.006443125735159542	0.0064440926	0.006442059531443544
0.00001	0.0068238036	0.0064240414	0.006475066077579574	0.0064540177	0.006443033574195834	0.0064440005	0.0064419674
0.000001	0.0068238024	0.0064240397	0.006475064135434637	0.0064540159	0.0064430322	0.0064439988	0.006441966

FURTHER ANALYSIS OF SENSITIVITY TO TOLERANCE

For H2 geometry (1.3984,0,0, 0, 0, 0) we vary COBYLA tolerance.



$$lim_{t,d\to 0}||\nabla E_{\infty} - \nabla E_{t,d}||_2 = 0$$

$d = 10^{-6}$ Is small enough here.

$$\nabla E_{\infty} \approx [4.5559,0,0,-4.5548,0,0] * 10^{-3}$$

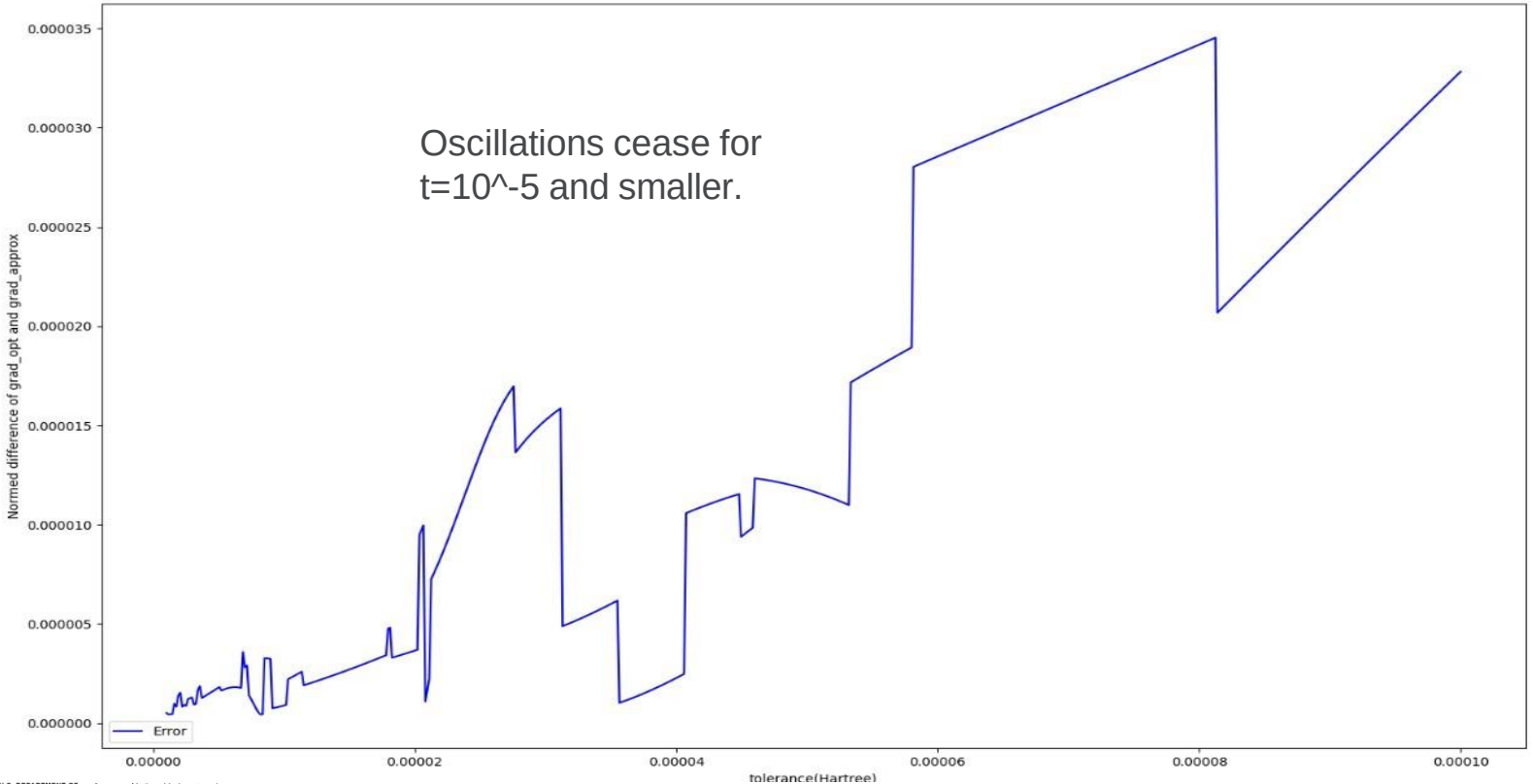
Point-wise convergence is equivalent to norm convergence in R^n so if the above holds then the following holds.

$$lim_{t,d\to 0}\nabla E_{t,d} = \nabla E_{\infty}$$

Oscillations cease for t=10^{-5} and smaller.

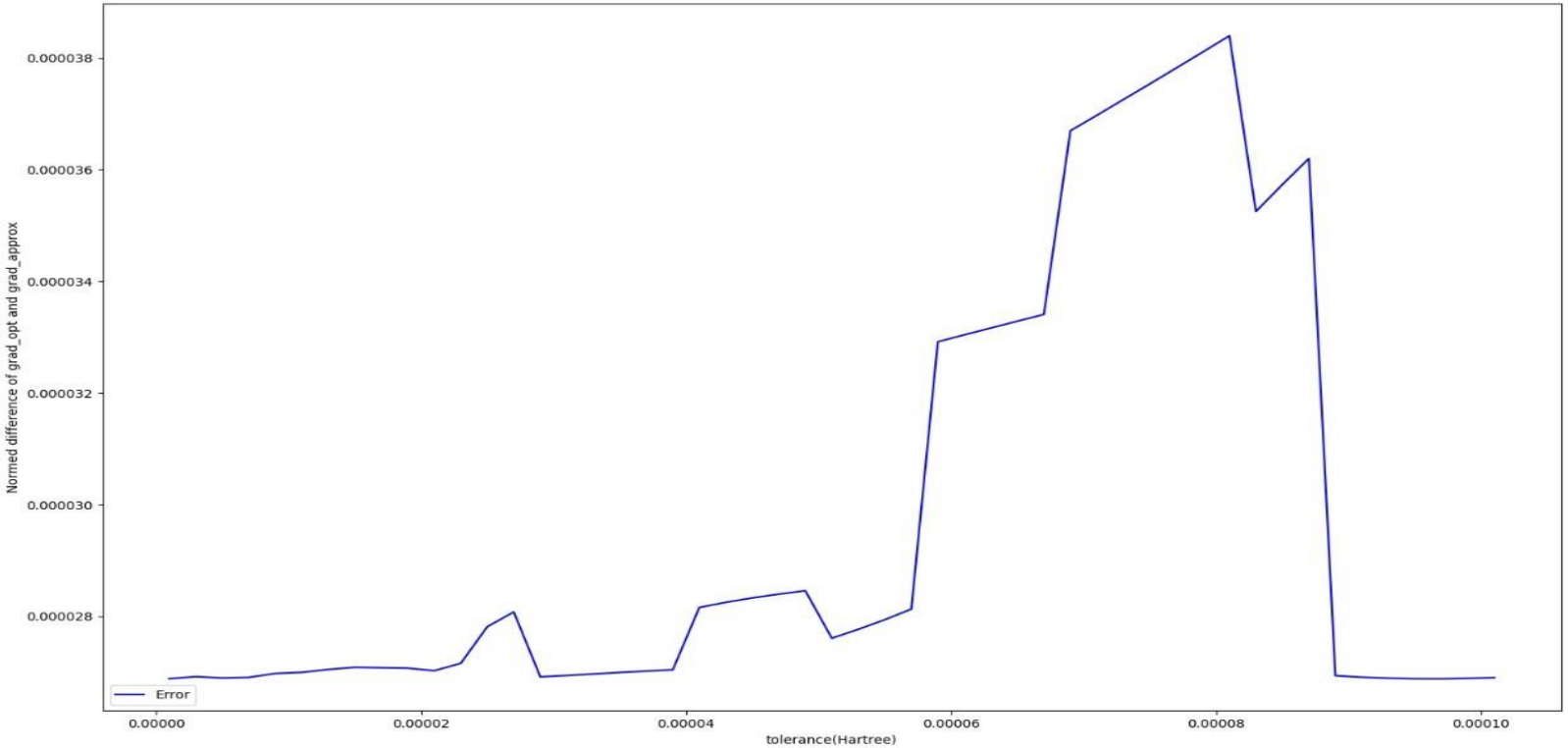
FURTHER ANALYSIS OF SENSITIVITY TO TOLERANCE

FOR H2 GEOMETRY (1.3984,0,0, 0, 0, 0) WE VARY COBYLA TOLERANCE.



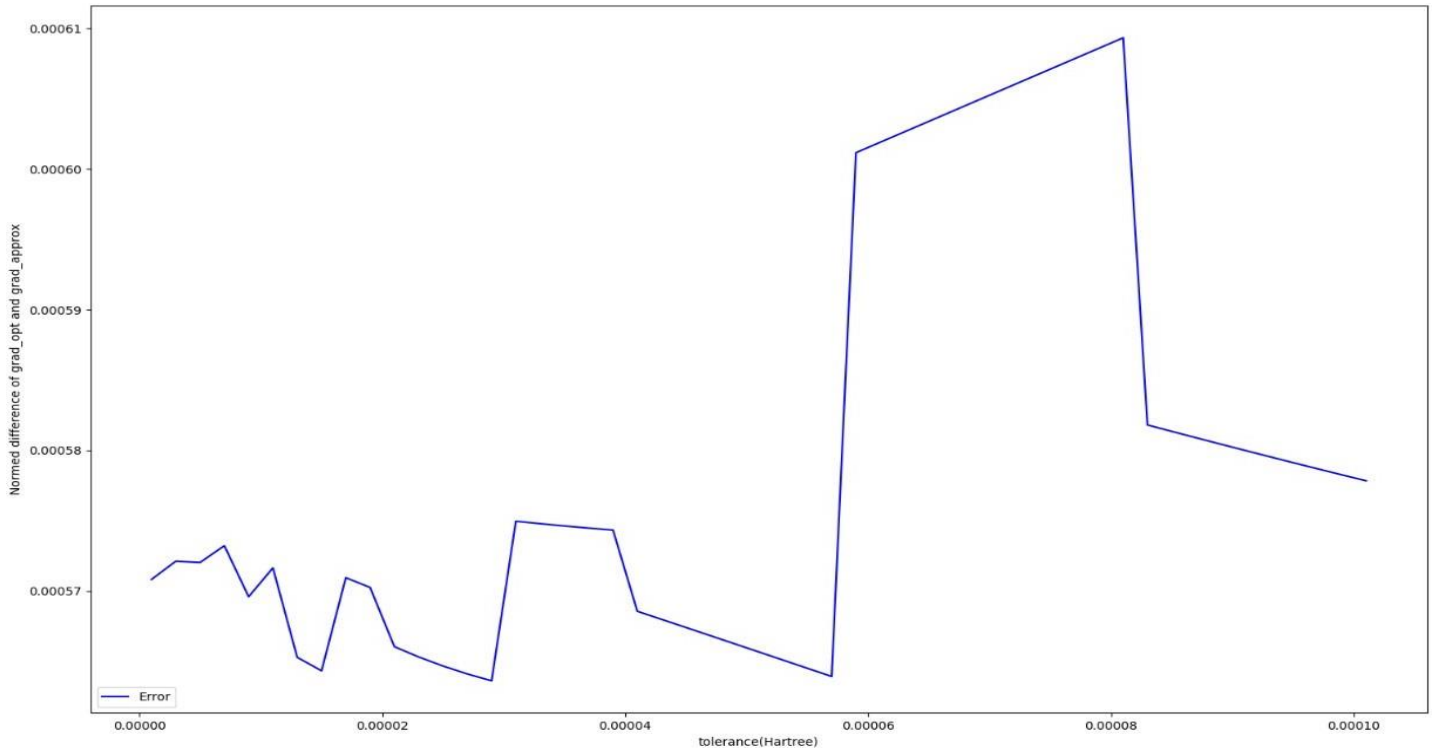
FURTHER ANALYSIS OF SENSITIVITY TO TOLERANCE

FOR H2 GEOMETRY (0.3984,0,0, 0, 0, 0) WE VARY COBYLA TOLERANCE.



FURTHER ANALYSIS OF SENSITIVITY TO TOLERANCE

FOR H2 GEOMETRY (2.3984,0,0, 0, 0, 0) WE VARY COBYLA TOLERANCE.



SIEVE FOR FINDING OPTIMAL PARAMETERS.

- Choose Geometry.
- Fix COBYLA MAXITER parameter to a high value.
- Compute gradient with varying values for offset and tolerance. Tolerance varies over [10^-6, 10^-4] and offset varies over [10^-8, 10^-2].
- Choose biggest offset and tolerance pair that yields an error 10^-6 or smaller in the normed difference of the gradient and its limit.
- Compute trajectories with varying time steps. 0.1fs, 0.25fs and 0.5fs for example.

FUTURE WORK

-REFINE SCHEME FOR FINDING OPTIMAL PARAMETERS.

-IMPLEMENT NOISE INTO TRAJECTORY RUNS.

-RUN OUR PROGRAM ON AN ACTUAL QUANTUM COMPUTER.