EMSL

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

U.S. DEPARTMENT OF ENERGY

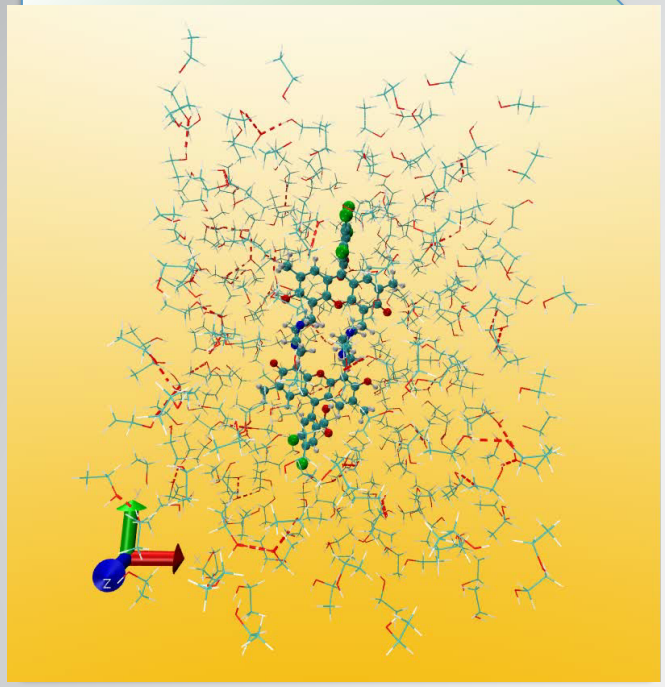
Improving the performance of ab initio molecular dynamics simulations and band structure calculations for actinide and geochemical systems with new algorithms and new machines

Eric J. Bylaska (PNNL)

Outline

- NWChem
- Analyzing EXAFS with ab initio molecular dynamics simulations (AIMD)
- Exact Exchange
  - Brillouin zone integration
  - Fast Localization
- Parallel in time

Plane-Wave Density Functional Theory (NWPW module) in NWChem



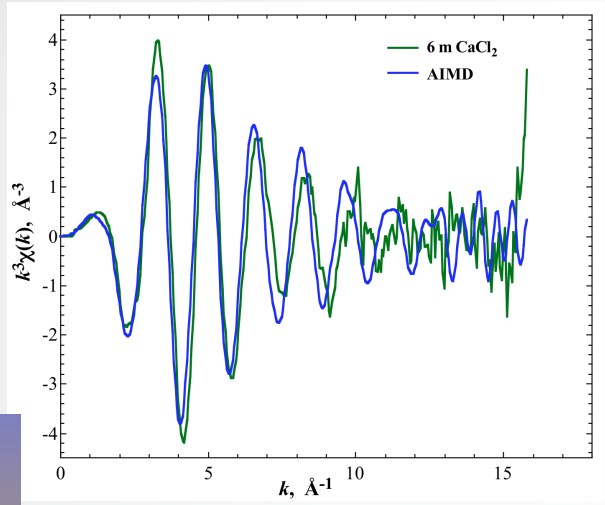
- Highly scalable
- CG, limited memory BFGS, and RMM-DIIS minimization
- Gamma and Band structure capabilities
- Car-Parrinello (extended Lagrangian dynamics) and Born-Oppenheimer
- Constant energy and constant temperature Car-Parrinello
- Metropolis NVT, NPT
- Fixed atoms in cartesian, SHAKE constraints, translation contraints, and rotation constraints, Metadynamics, TAMD, PMF, equation parser-ecode
- Hamann, Troullier-Martins, and HGH norm-conserving pseudopotentials with optional semicore corrections
  - Interface for CPI and TETER formats
  - Spin-orbit
- PAW (full integration in next release, most functionality already working in development branch)
- FEFF6 integration

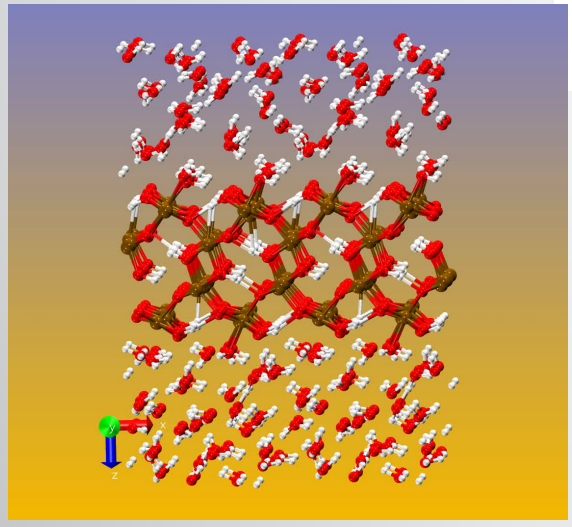
- LDA and GGA exchange-correlation potentials (spin-restricted and unrestricted) SIC, pert-OEP, Hartree-Fock and Hybrid Functionals (restricted and unrestricted), DFT+U, Grimme1, Grimme2, Grimme3…complete set of functionals in next release
- Fractional occupation,
- Geometry/unitcell optimization, frequency, transition-state searches, phonon spectra, NEB, String
- AIMD/MM
- Wannier analysis
- Wavefunction, density, electrostatic, Wannier, density matrix, ELF plotting
- …..

Molecular simulation of XAFS analysis

Objective

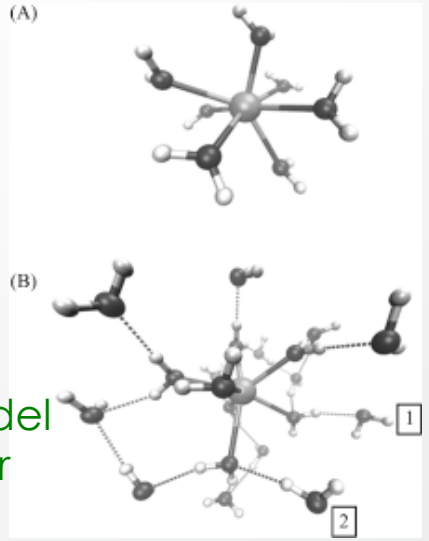
- To provide molecular simulation analysis that will support the application of XAFS (XANES) to complex aqueous systems in extreme chemical/ geochemical environments.





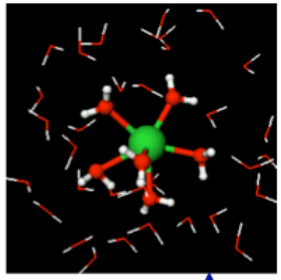
???

1. FEFF
2. Fit molecular structural model
3. Use molecular modeling



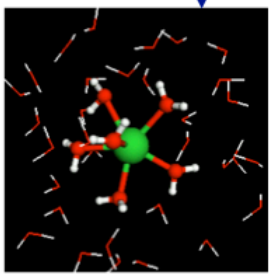
Molecular Dynamics-XAFS (MD-XAFS)

Snapshot 1



Δt = 0.2 ps

Snapshot 2



Snapshot 1000

FEFF 9.03 ($F_i(k)_{effective}$) scattering function

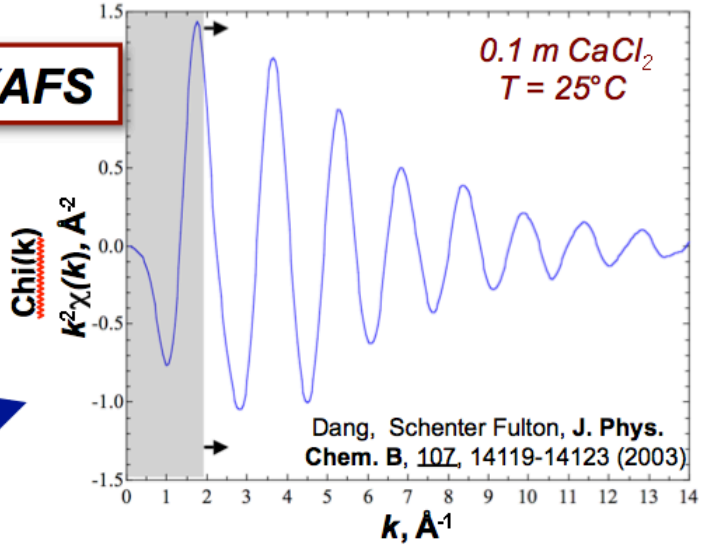
- Ab initio scattering code
- SCF level
- **John Rehr**, Univ. of Washington

$$\bar{\chi}(E) = \frac{\bar{\mu}(E) - \mu_0(E)}{\Delta\mu_0(E_0)}$$

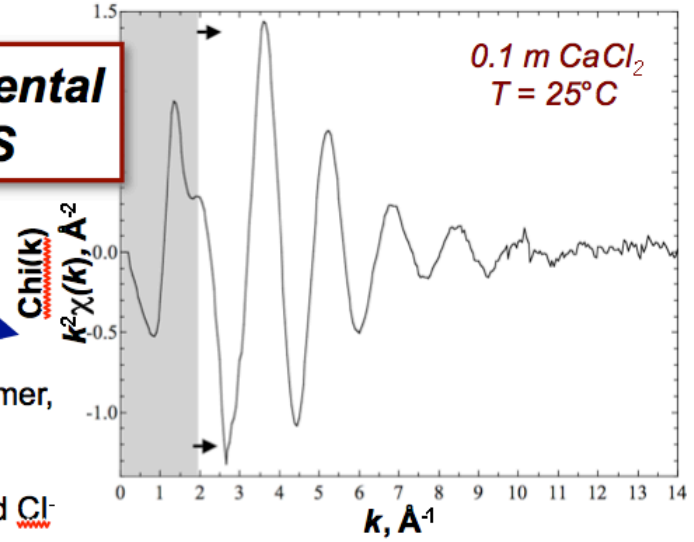
No assumptions
Exact representations

$$\chi(E) = \frac{\mu(E) - \mu_0(E)}{\Delta\mu_0(E_0)}$$

MD-XAFS



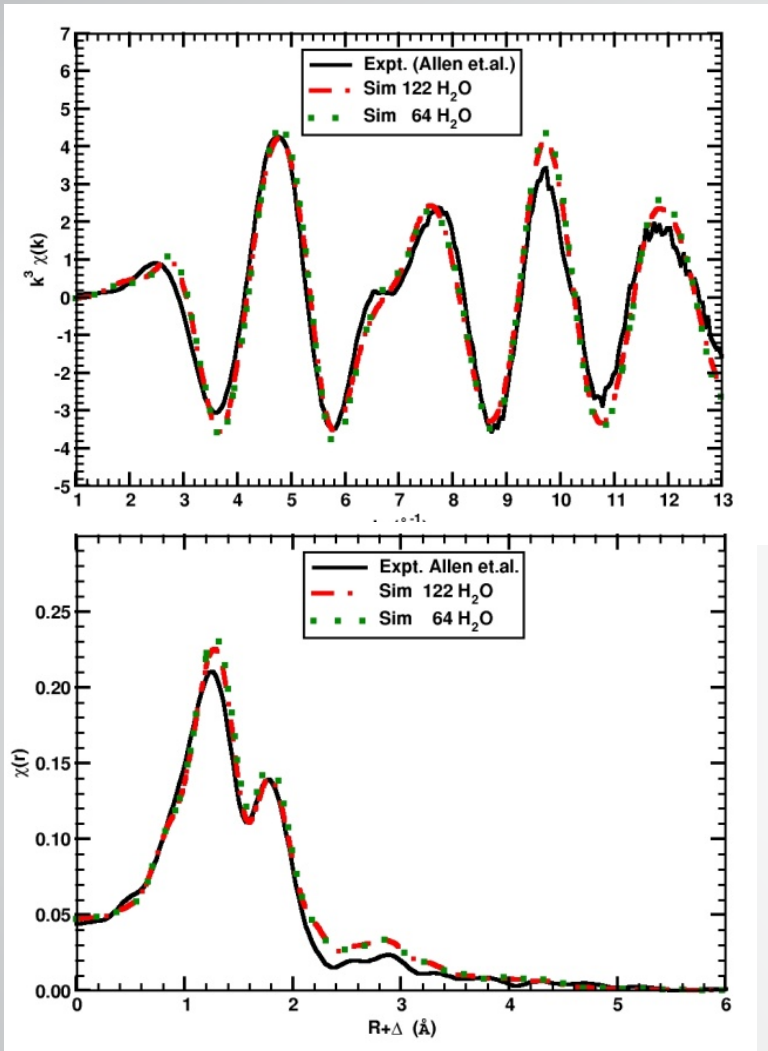
Experimental XAFS



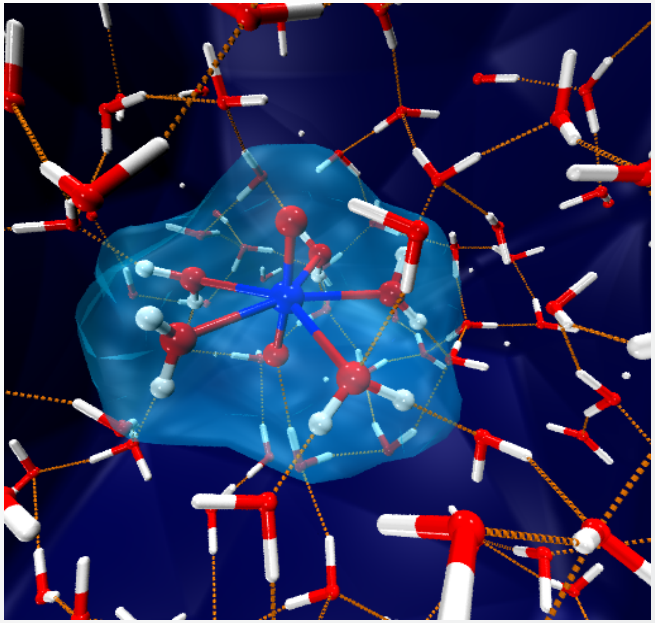
"Direct Modeling of XAFS Spectra from Molecular Dynamics Simulations", B. J. Palmer, D. M. Pfund, J. L. Fulton, **J. Phys. Chem.**, 100, 13393-13398, (1996)

"Molecular Simulation Analysis and X-ray Absorption Measurement of Ca^{2+}, K+, and Cl^- Ions in Solution," L. X. Dang, G.K. Schenter, V. A. Glezakou, J. L. Fulton, **J. Phys. Chem. B** 110, 23644, (2006)

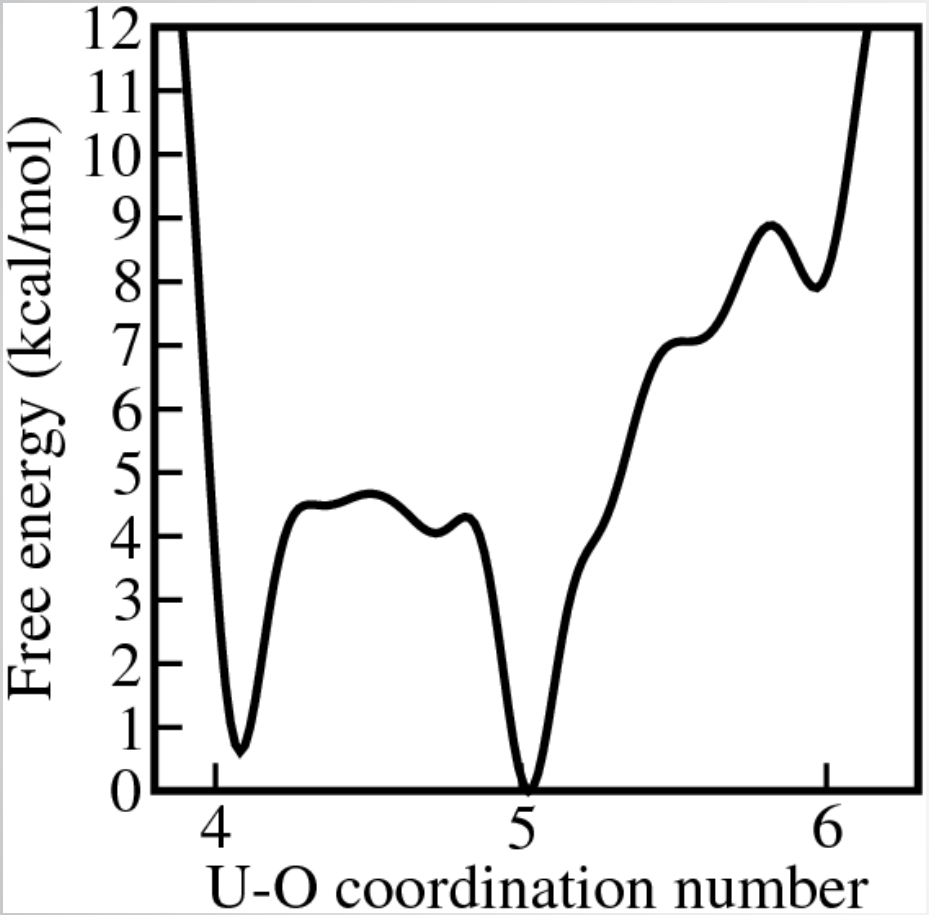
EXAFS and coordination for $UO_2^{2+}(aq)$



- Good EXAFS agreement but recent HEXS experiments suggest a 4-fold state is energetically nearby

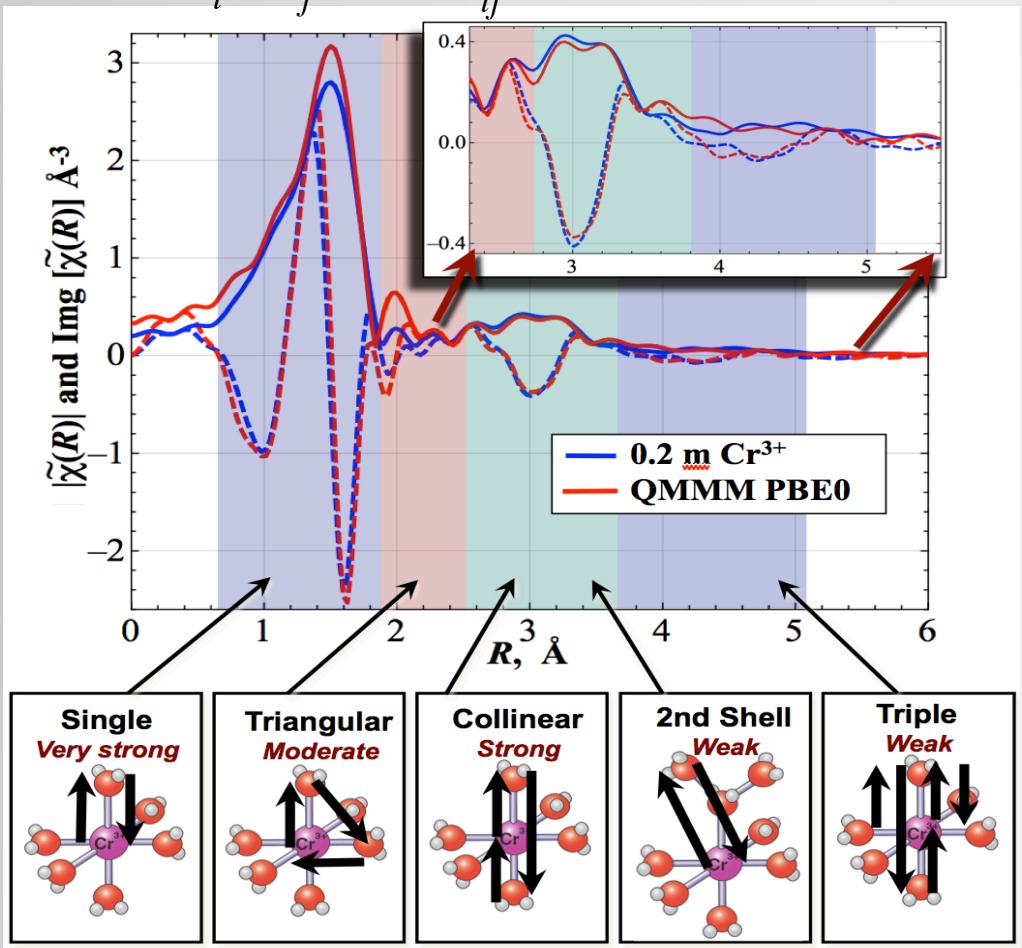


EXAFS and coordination for $UO_2^{2+}(aq)$



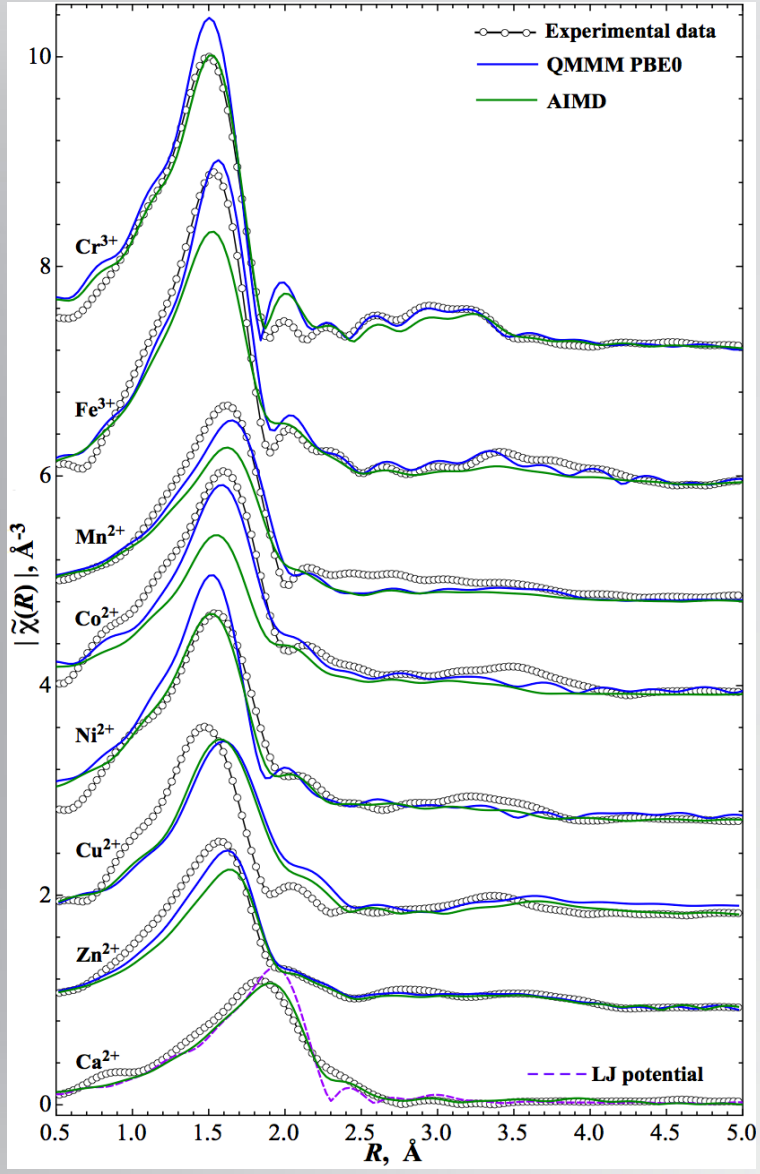
- Good EXAFS agreement but recent HEXS experiments suggest a 4-fold state is energetically nearby
- Results from Metadynamics
  - Coordination number collective variable
  - 5-fold state is favored over 4-fold state by $\Delta A_{5\rightarrow4}$=0.7 kcal/mol
  - Agrees with ΔG_{expt}=1.2 kcal/mol
  - Predicted associative barrier $\Delta A_{5\rightarrow4}^{\ddagger}\approx$4.7 kcal/mol
  - Prediction: 6-fold state has short lifetime in solution; not stable relative to 5-fold state.
  - $\Delta A_{5\rightarrow6}$=8 kcal/mol; $\Delta A_{5\rightarrow6}^{\ddagger}\approx$9 kcal/mol

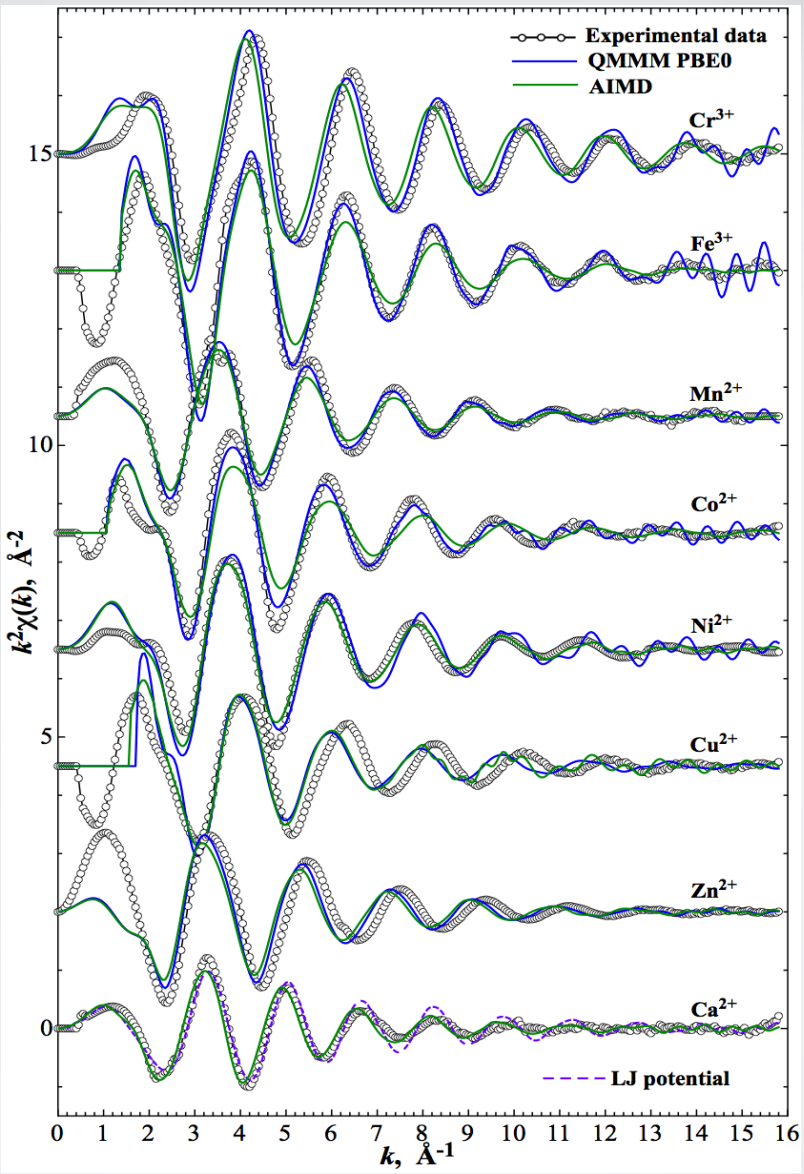
$$\chi(k) = \frac{1}{N} \sum_{i}^{struc.\,paths} \sum_{j} S_0^2 \frac{N_j}{kR_{ij}^2} |F(k)| \sin\left[2R_{ij} - \varphi_j(k)\right] e^{-2R_{ij}/\lambda}$$



Illustrates the agreement with data obtained using the full 1st principle MD-XAFS method. That is direct simulation of spectra using ab-initio MD. All scattering paths are used. Slide illustrates the contributions from the various scattering paths. **The parameter free 1st principle simulation will very accurately reproduce both the structural parameters and disorder effects.**

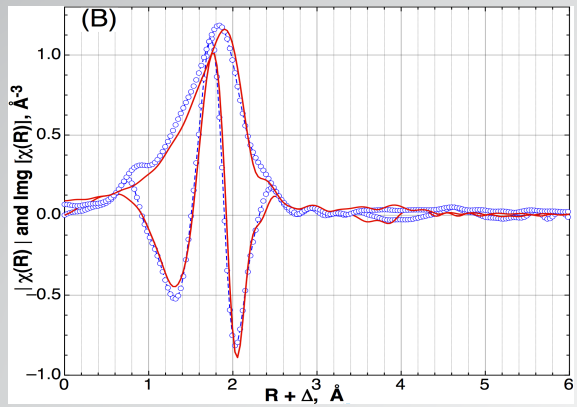
EMSL



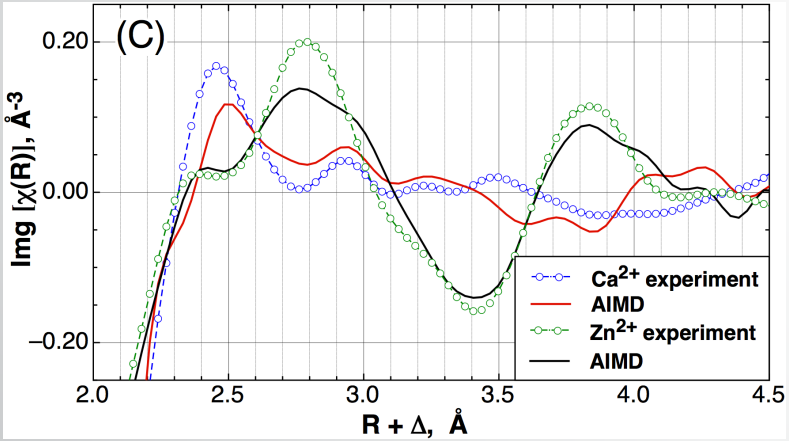


The generally excellent agreement of the 1st principle MD-XAFS simulation with the data. The scans are calculated by a parameter free method which can be implemented more efficiently than the use of empirical interactions suggesting that this **method can be used to interpret more XAFS spectra in more complex environments**.

Second Shell and Defect Structures



1st principle MD XAFS simulation of Ca^{2+}, a relatively weakly interacting ion. Disorder results in no second shell structure.

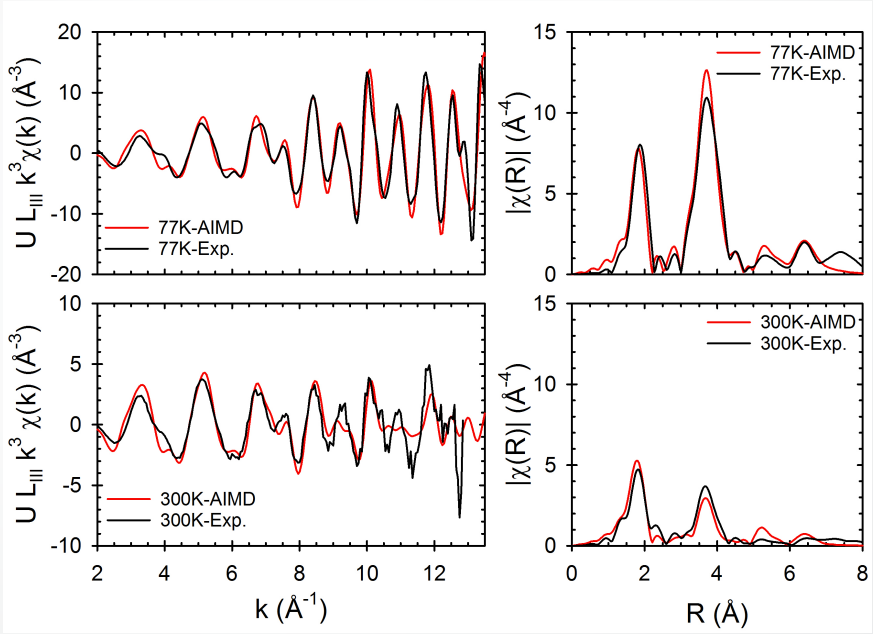


Scattering for the more structured Zn^{2+} ion in the 2nd shell scattering region. Note the difference between the Ca^{2+} and Zn^{2+} features.

Incorporation of incommensurate metals in Iron (oxyhydr)oxides (Ilton, Kerisit)

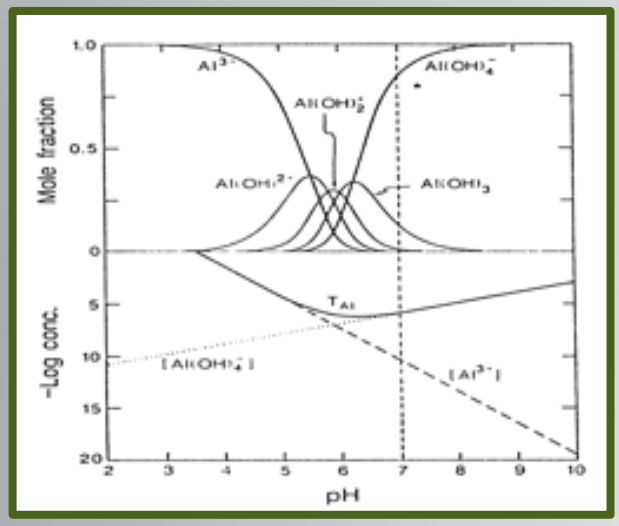
Strong constraint on thermal disorder

Proton dynamics likely important



U(IV) results

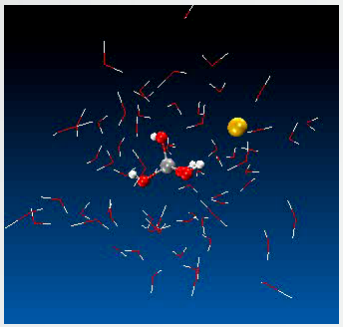
Determine the solute structure of environmentally important species (*e.g.*, Al^{3+}, Mg^{2+}, Cl^{-}, Fe^{3+}) in aqueous solutions as a function of TPX through synergistic simulation and X-ray observation

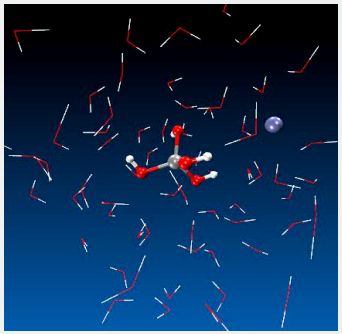


Old Way – The diagram for the distribution of aluminum species was determined primarily from fitting thermodynamic data using an **assumed speciation scheme.**

- By combining AIMD free energy simulations with XAS spectra can provide a transformative new approach to the development of chemically and thermodynamically highly accurate solution models with exceptional extrapolation properties in TPX.
  - Strategies to search configuration space must be developed.
  - $MgCO_3$-H_2O system
    - Ion pairs in this system control the stability of Mg containing minerals.
  - Nature of ion association of Na^+(aq) ion with the aluminate ion to form $NaAl(OH)_4$(aq) ion pair.

$NaAlOH_4$





$LiAlOH_4$

EMSL

Calculating exact exchange

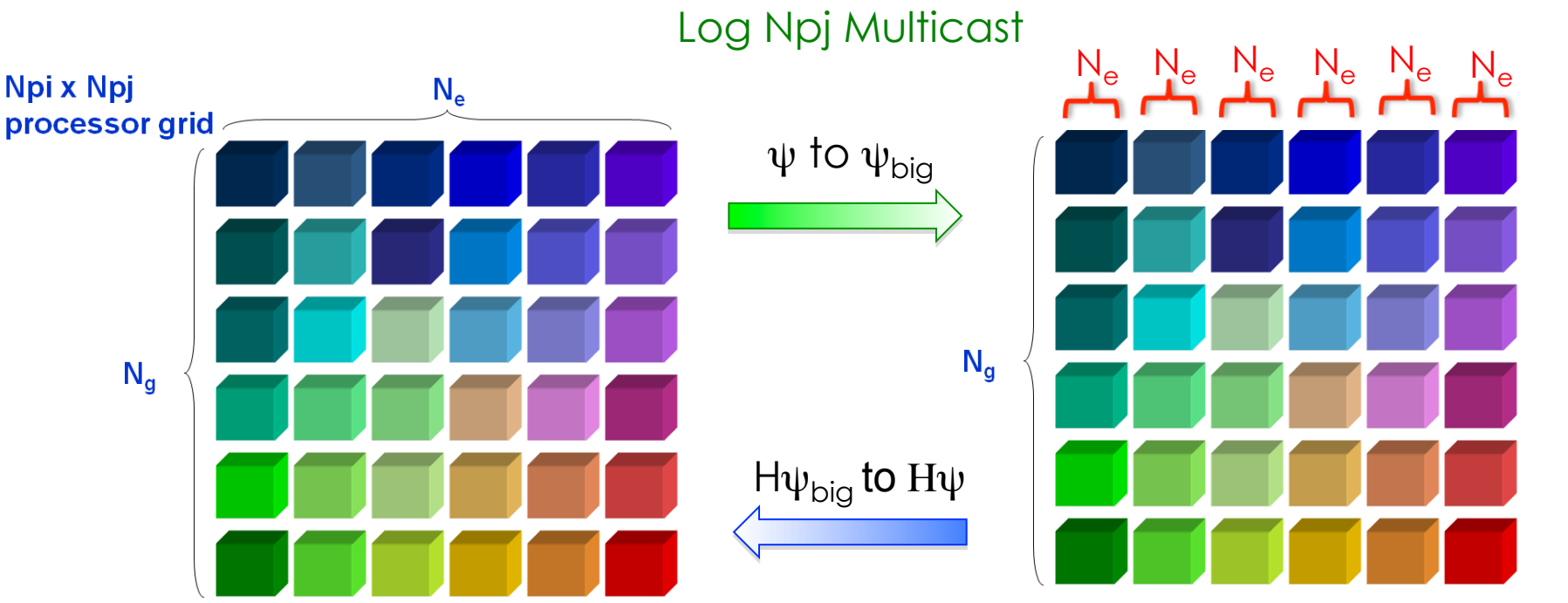
Algorithm 1: Serial algorithm for calculating exact exchange in a plane-wave basis

Input: ψ - N_g x N_e *array*
Output: $K\psi$ - N_g x N_e *array*
for $m=1,N_e$
  for $n=1,m$
    $\rho(:) \leftarrow \mathbf{FFT\_rc}(\psi(:,m)*.\psi(:,n))$
    $V(:) \leftarrow \mathbf{FFT\_cr}(f_{cutoff}(:)*.\rho(:))$
    $K\psi(:,m)$ -= $V(:)*.\psi(:,n)$; *if* $m<>n$ $K\psi(:,n)$ -= $V(:)*.\psi(:,m)$
  end for
end for

For Ne=500 → 500(500+1) = 250,500 three-dimension FFTs per step

For Ng=200x200x200 calculation can readily be run on leadership class machines (e.g. 38640 cores, 4 to each of 9660 processing nodes)

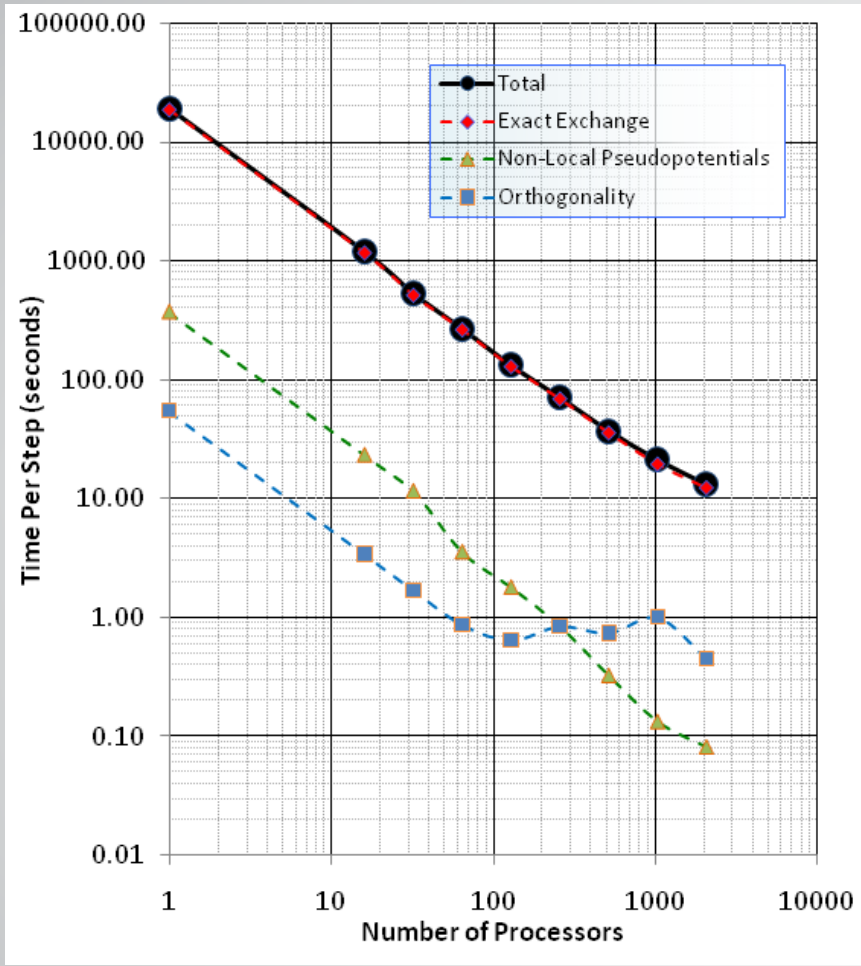
Simple parallel algorithm for exact exchange



Use replicated space to compute exchange using 3d parallel FFTs along columns (load balanced)

The algorithm is scalable because:
→Computation ~Npj^2,
→Message passing ~Npj

Exact exchange timings



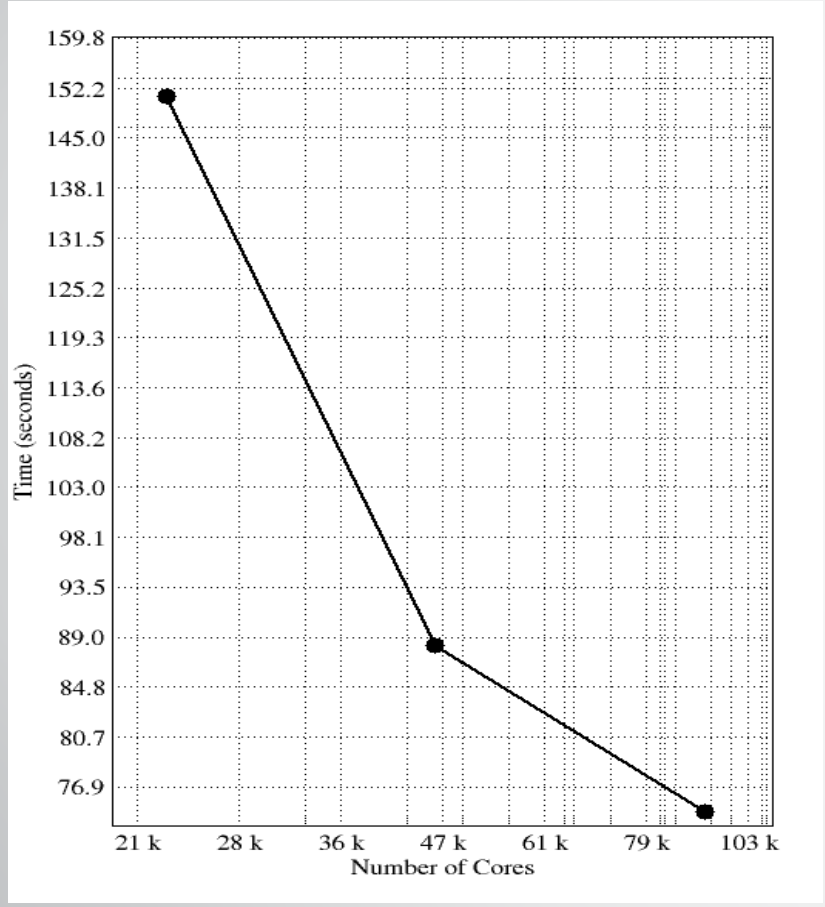
Basic algorithm works fairly well (80 atoms cell of hematite on Cray XT4 - Franklin system at NERSC)
- stalls at 7 seconds by 4096 cpus

- Exchange term is dominant
- Previous algorithm simple to implement
- Requires lots of workspace
- Sends approximately twice as much data as necessary

- **Can be improved**

Exact exchange timings

Incomplete butterfly algorithm results works fairly well (576 atoms cell of water on Cray XE6 - Hopper system at NERSC)



- Exchange term is dominant
- Previous algorithm simple to implement
- Requires lots of workspace
- Sends approximately twice as much data as necessary

- **Can be improved**
  - Developed new parallel algorithm for hybrid DFT (Incomplete butterfly) which reduced communication costs by ½
  - Scaling to at least ~100k
  - MP2 algorithm (easily 10^9 3d FFTs per step)

Problems with simple implementations of exact exchange – the ugly

$$E_{x-exact}[\{\psi\}] =$$

$$-\frac{1}{2\Omega}\sum_{\sigma=\uparrow,\downarrow}\left(\frac{\Omega}{8\pi^3}\right)^2\int_{BZ}d\mathbf{k}\int_{BZ}d\mathbf{l}\left[\sum_{n=1}^{N_{occ}^{\sigma}}\sum_{m=1}^{N_{occ}^{\sigma}}\sum_{\mathbf{G}}\frac{4\pi}{|\mathbf{G}-\mathbf{k}+\mathbf{l}|^2}\rho^{\sigma}_{ml;nk}(-\mathbf{G})\rho^{\sigma}_{nk;ml}(\mathbf{G})\right]$$

- Integral has singularity at G=0, k=l
- Common (bad strategies?)
  - Set $4\pi/G^2 \rightarrow 0$
  - Compute dielectric ε for screening?
  - Refine G==0 term by increasing k-points, and using analytic formulas to handle integrals, e.g.

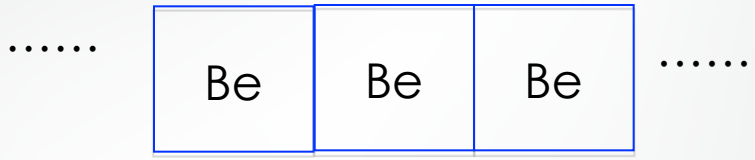
Singular terms - - >

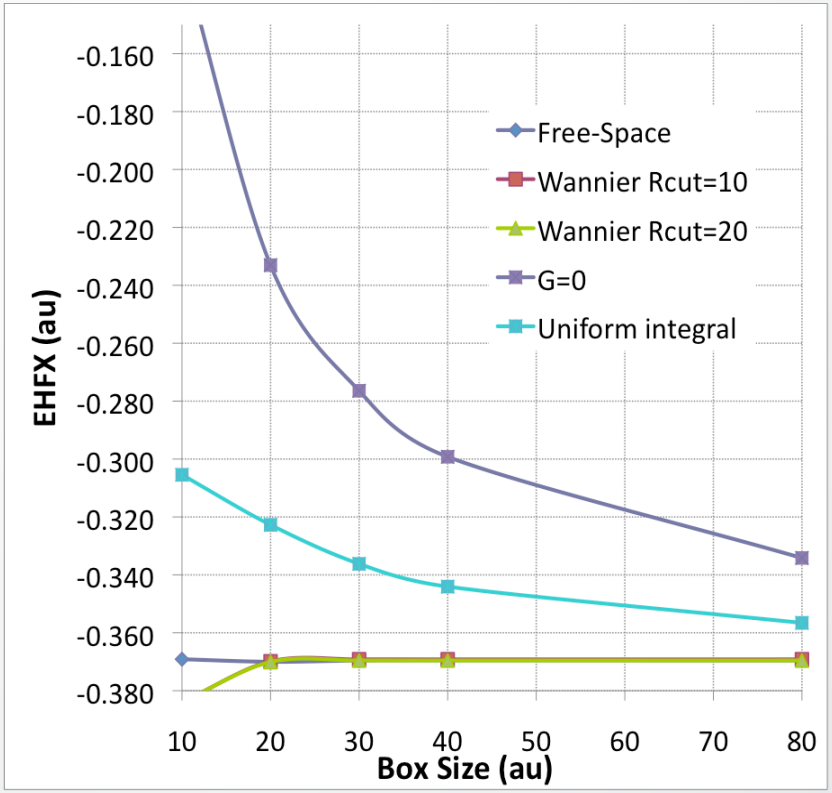
$$-\frac{1}{2\Omega}\sum_{\sigma=\uparrow,\downarrow}\left(\frac{\Omega}{8\pi^3}\right)^2\rho^{\sigma}_{ml;nk}(-\mathbf{G}=0)\rho^{\sigma}_{nk;ml}(\mathbf{G}=0)\int_{BZ}d\mathbf{k}\int_{BZ}d\mathbf{l}\frac{4\pi}{|-\mathbf{k}+\mathbf{l}|^2}$$

- Wannier integration

$$\Rightarrow V_{screened}(\mathbf{r}+\mathbf{R}) = \frac{1}{|\mathbf{r}|} \quad \text{for } (\mathbf{R}-L/2) \le \mathbf{r}+\mathbf{R} \le (\mathbf{R}+L/2)$$

Results for Be atom(s)





Another (Derivation?) Justification for 1/r Kernel

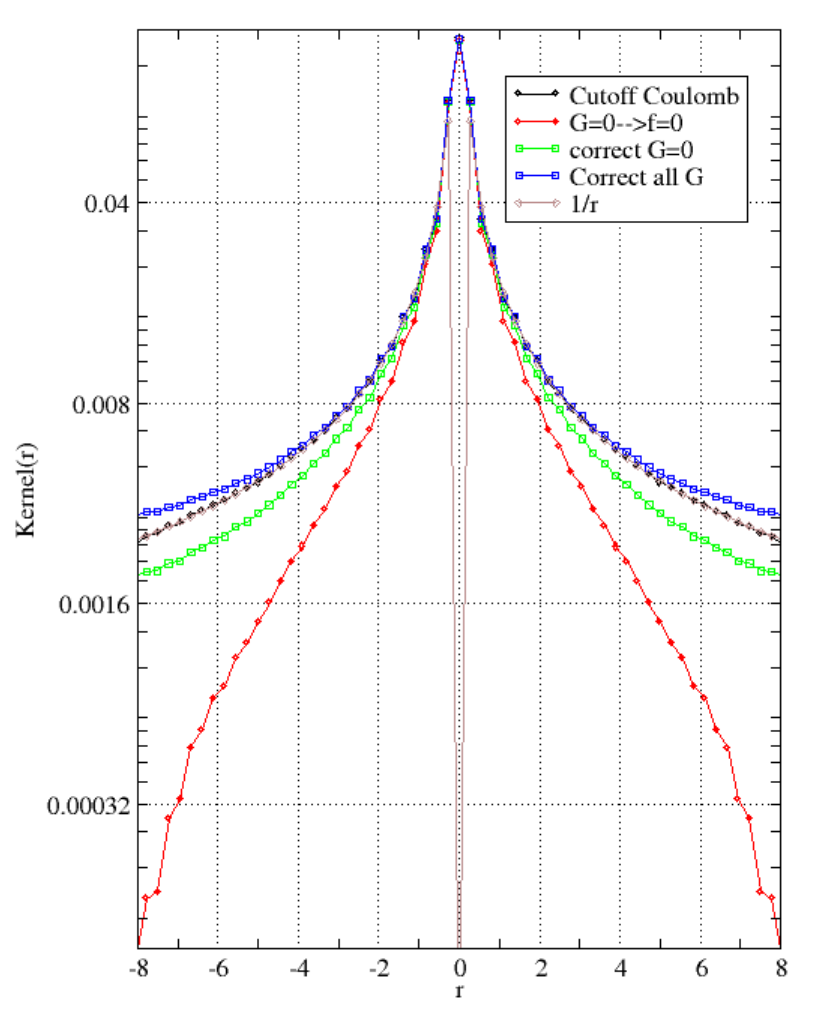
If there is no Brillioun zone dependency in the overlap densities then

$$E_{x-exact}\left[\{\psi\}\right] - - >$$

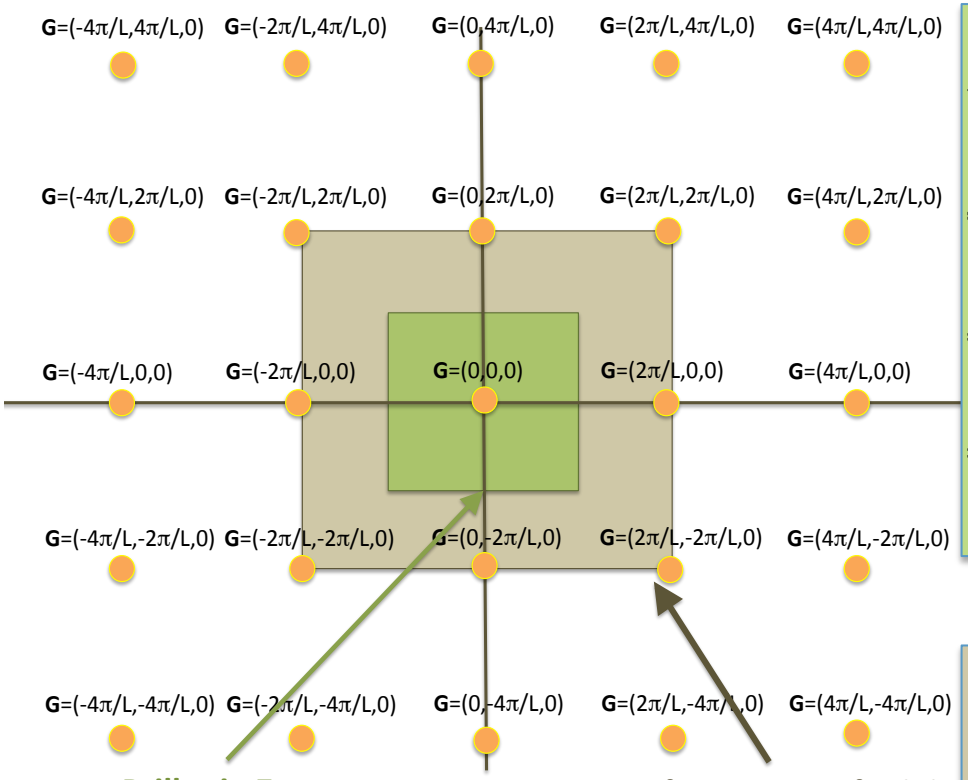
$$-\frac{1}{2\Omega}\sum_{\sigma=\uparrow,\downarrow}\left[\sum_{n=1}^{N_{occ}^{\sigma}}\sum_{m=1}^{N_{occ}^{\sigma}}\sum_{\mathbf{G}}\left[\rho_{ml=0;nk=0}^{\sigma}(-\mathbf{G})\rho_{nk=0;ml=0}^{\sigma}(\mathbf{G})\left(\frac{\Omega}{8\pi^3}\right)^2\int_{BZ}d\mathbf{k}\int_{BZ}d\mathbf{l}\frac{4\pi}{|\mathbf{G}-\mathbf{k}+\mathbf{l}|^2}\right]\right]$$

Thus the screened potential is

$$V_{ScreenedCoulomb}\left[\mathbf{G}\right]=\left(\frac{\Omega}{8\pi^3}\right)^2\int_{BZ}d\mathbf{k}\int_{BZ}d\mathbf{l}\frac{4\pi}{|\mathbf{G}-\mathbf{k}+\mathbf{l}|^2}$$



Evaluation of Exchange for Γ Calculation



$$\frac{-1}{2\Omega}\left(\frac{\Omega}{8\pi^3}\right)\sum_{\mathbf{G}}\iiint_{BZ} d\mathbf{l}\iiint_{BZ} d\mathbf{k}\frac{4\pi}{\left|\mathbf{G}-\mathbf{k}+\mathbf{l}\right|^2}\rho_{m\mathbf{l};n\mathbf{k}}(-\mathbf{G})\rho_{n\mathbf{k};m\mathbf{l}}(\mathbf{G})$$

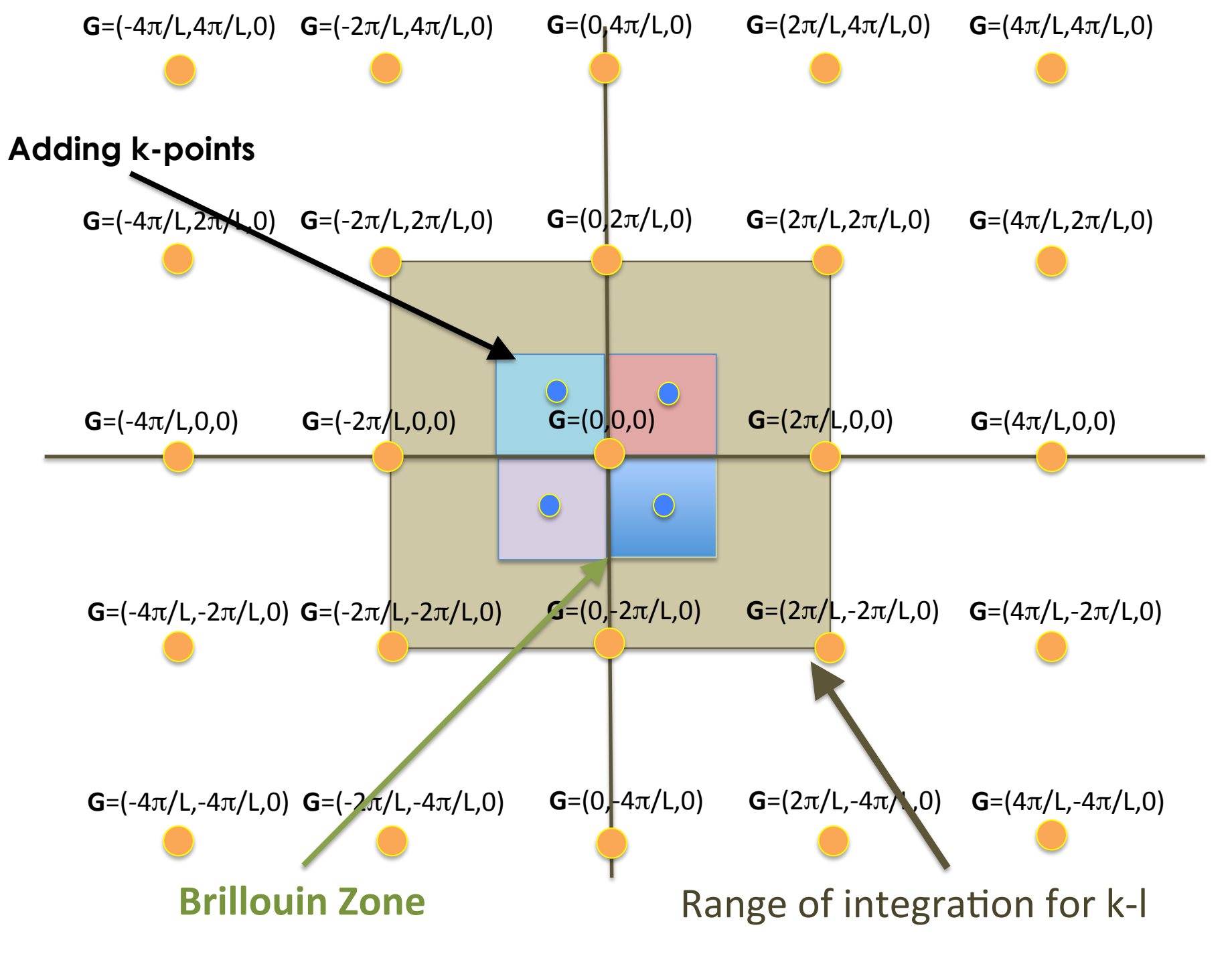
$$\approx\frac{-1}{2\Omega}\left(\frac{\Omega}{8\pi^3}\right)\sum_{\mathbf{G}}\rho_{m\mathbf{l}=0;n\mathbf{k}=0}(-\mathbf{G})\rho_{n\mathbf{k}=0;m\mathbf{l}=0}(\mathbf{G})\iiint_{BZ} d\mathbf{l}\iiint_{BZ} d\mathbf{k}\frac{4\pi}{\left|\mathbf{G}-\mathbf{k}+\mathbf{l}\right|^2}$$

$$=\frac{-1}{2\Omega}\sum_{\mathbf{G}}\rho_{m\mathbf{l}=0;n\mathbf{k}=0}(-\mathbf{G})\rho_{n\mathbf{k}=0;m\mathbf{l}=0}(\mathbf{G})V_{screened}(\mathbf{G})$$

$$\Rightarrow V_{screened}(\mathbf{G})=\left(\frac{\Omega}{8\pi^3}\right)\iiint_{BZ} d\mathbf{l}\iiint_{BZ} d\mathbf{k}\frac{4\pi}{\left|\mathbf{G}-\mathbf{k}+\mathbf{l}\right|^2}$$

For large **G** expansion and **R** Bravais lattice vector

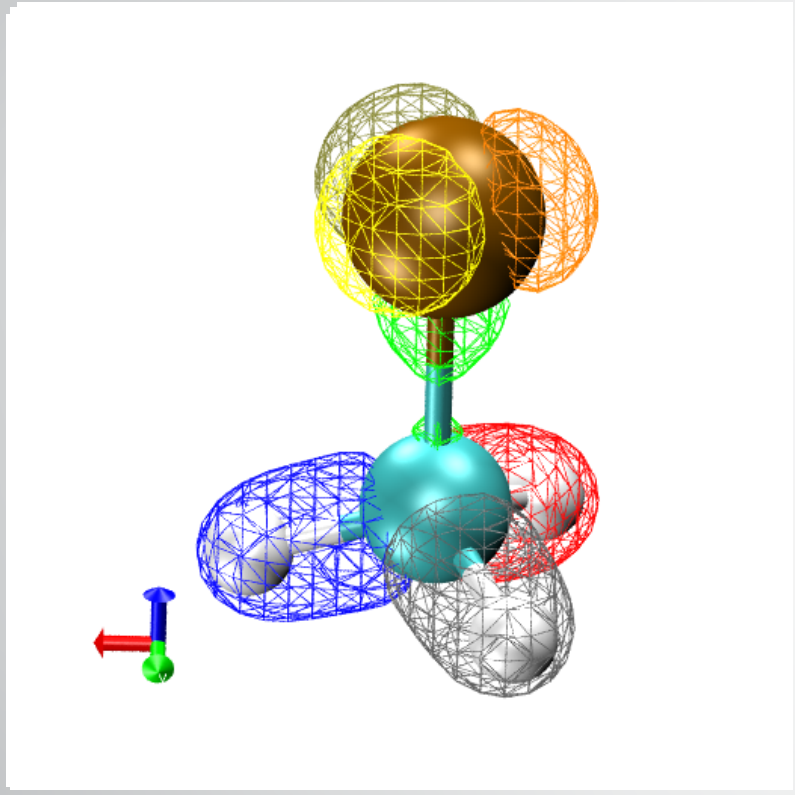
$$\Rightarrow V_{screened}(\mathbf{r}+\mathbf{R})=\frac{1}{|\mathbf{r}|}\quad \text{for } (\mathbf{R}-L/2)\le\mathbf{r}+\mathbf{R}\le(\mathbf{R}+L/2)$$


G=(-4π/L,4π/L,0)
G=(-2π/L,4π/L,0)
G=(0,4π/L,0)
G=(2π/L,4π/L,0)
G=(4π/L,4π/L,0)
Adding k-points
G=(-4π/L,2π/L,0)
G=(-2π/L,2π/L,0)
G=(0,2π/L,0)
G=(2π/L,2π/L,0)
G=(4π/L,2π/L,0)
G=(-4π/L,0,0)
G=(-2π/L,0,0)
G=(0,0,0)
G=(2π/L,0,0)
G=(4π/L,0,0)
G=(-4π/L,-2π/L,0)
G=(-2π/L,-2π/L,0)
G=(0,-2π/L,0)
G=(2π/L,-2π/L,0)
G=(4π/L,-2π/L,0)
G=(-4π/L,-4π/L,0)
G=(-2π/L,-4π/L,0)
G=(0,-4π/L,0)
G=(2π/L,-4π/L,0)
G=(4π/L,-4π/L,0)
Brillouin Zone
Range of integration for k-l

Localization Algorithm of Damle, Lin, and Ying

Plotting

$$\rho\left(\mathbf{r},\mathbf{r}'=\mathbf{r}_i\right) \quad i=\Pi_{1,Ne}$$



- The density matrix is
  - Severely rank deficient, i.e. a density matrix for Ne orbitals has a rank=Ne
  - Localized (real-space) orbitals by selecting certain columns(or **r**')
  - In principle just Ne columns are needed to regenerate the density matrix
- Currently testing algorithm on HPC systems

$$\rho\left(\mathbf{r},\mathbf{r}'\right)=\rho\left(\mathbf{r},\mathbf{r}'=\mathbf{r}_1\cdots\mathbf{r}_{Ne}\right)\left[\rho\left(\mathbf{r}_i,\mathbf{r}_j\right)\right]^{-1}\rho\left(\mathbf{r}=\mathbf{r}_1\cdots\mathbf{r}_{Ne},\mathbf{r}'\right)$$

AIMD and MD Still Need Faster Time To Solutions: The Irrational Pursuit Of Solving The Schrödinger Equation In less than a Second Per Step

- Current *ab-initio* molecular dynamics simulations for 10 to 100 picoseconds can take several months to complete
- The step length in ab initio molecular dynamics simulation is on the order of 0.1…0.2 fs/step
  - 20 ps of simulation time → 200,000 steps
    - At 1 second per step → 2-3 days
    - At 10 seconds per step → 23 days
    - At 30 seconds per step →70 days
  - 1 ns of simulation time → 10,000,000 steps
    - at 1 second per step → 115 days of computing time
    - At 10 seconds per step → 3 years
    - At 30 seconds per step → 9 years
    - At 0.1 seconds per step → 11.5 days

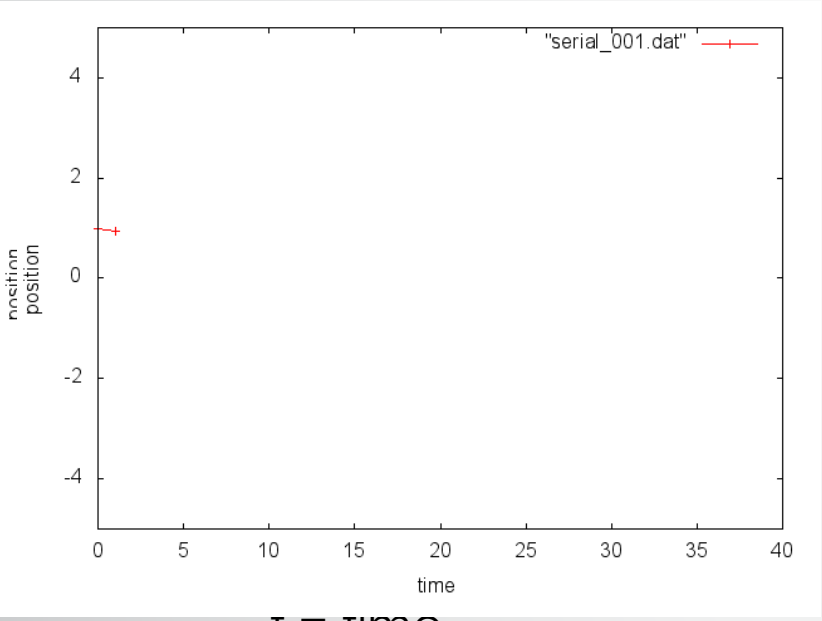
$HCl+4H_2O$ MP2/6-311++G(2d, 2p) force evaluation takes 32 seconds!

- For classical molecular dynamics, time step ~= 1fs/step
  - 1μs of simulation time → 1,000,000,000 steps
    - 1 millisecond per step → 11.6 days of computing time
    - 1 second per step → 31 years of computing time
    - 10 seconds per step → 310 years of computing time
    - 1 μs per step → 16.6 minutes

Parallel in Time

- Increasing the time step (Δt) in time integration quickly becomes unstable
- One approach to bridging these temporal scales is the development of algorithms which parallelize over time, i.e. parallel in time algorithms
- The central philosophy of parallel in time integration is to start with a guess for the trajectory over some fixed time interval and then attempt to relax it until it approximates the "true" trajectory.

x(t)



T = Time

Trajectory for a simple spring (K=1,x0=1,v0=0)

Increasing time step

Can this be parallelized????

New Parallel In Time Algorithms Without Using Approximate Models (W/ J.Q. Weare): Fixed Point Parallel in Time Algorithms

These algorithms transform standard forward substitution time integration solvers, i.e. $x_{i+1} \leftarrow f(x_i)$, into fixed-point root problems

$$\mathbf{F(X) = 0} \quad \text{or} \quad \begin{pmatrix} x_1 - f(x_0) \\ x_2 - f(x_1) \\ \vdots \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \end{pmatrix}$$

Can be solved using a variety of optimization techniques, including preconditioned fixed-point, quasi-Newton, and preconditioned quasi-Newton optimization methods. These algorithms can be parallelized since the evaluation of the trial root function **F(X)** can be done in parallel.

EMSL

Parallel in Time: Fixed Point Iteration

The serial solution to time integration,

$$x_{i+1} = f(x_i)$$

with initial condition $x_0 = x_0$

is $$X_{trajectory} = \begin{bmatrix} f(x_0) \\ f(f(x_0)) \\ f(f(f(x_0))) \\ f(f(f(f(x_0)))) \end{bmatrix}$$

Using column vector to store each step in the time iteration from i=1,4

This equation can also be solved by a fixed point iteration over the whole path or trajectory

$$X^{(k+1)} = X^{(k)} - F\left(X^{(k)}\right) \quad \text{or}$$

$$\begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \\ x_4^{(k+1)} \end{bmatrix} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \\ x_4^{(k)} \end{bmatrix} - \begin{bmatrix} x_1^{(k)} - f(x_0) \\ x_2^{(k)} - f(x_1^{(k)}) \\ x_3^{(k)} - f(x_2^{(k)}) \\ x_4^{(k)} - f(x_3^{(k)}) \end{bmatrix}$$

Parallelized by distributing work over rows

Is Fixed Point Iteration Stable?

Solving $X^{(k+1)} = X^{(k)} - F\left(X^{(k)}\right)$

Guess the Initial Path

$$X^{(1)} = \begin{bmatrix} b \\ c \\ d \\ e \end{bmatrix}$$

Is Fixed Point Iteration Stable?

Solving $X^{(k+1)} = X^{(k)} - F\left(X^{(k)}\right)$

k=1

$$\begin{bmatrix} f(x_0) \\ f(b) \\ f(c) \\ f(d) \end{bmatrix} = \begin{bmatrix} b \\ c \\ d \\ e \end{bmatrix} - \begin{bmatrix} b - f\left(x_0\right) \\ c - f(b) \\ d - f(c) \\ e - f(d) \end{bmatrix}$$

Each step of this global iteration can be parallelized by evaluating each row on a different cpu

Is Fixed Point Iteration Stable?

Solving $X^{(k+1)} = X^{(k)} - F\left(X^{(k)}\right)$

k=2

$$\begin{bmatrix} f(x_0) \\ f(f(x_0)) \\ f(f(b)) \\ f(f(c)) \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(b) \\ f(c) \\ f(d) \end{bmatrix} - \begin{bmatrix} f(x_0) - f\left(x_0\right) \\ f(b) - f(f(x_0)) \\ f(c) - f(f(b)) \\ f(d) - f(f(c)) \end{bmatrix}$$

Each step of this global iteration can be parallelized by evaluating each row on a different cpu

Is Fixed Point Iteration Stable?

Solving $X^{(k+1)} = X^{(k)} - F\left(X^{(k)}\right)$

k=3

$$\begin{bmatrix} f(x_0) \\ f(f(x_0)) \\ f(f(f(x_0))) \\ f(f(f(b))) \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(f(x_0)) \\ f(f(b)) \\ f(f(c)) \end{bmatrix} - \begin{bmatrix} f(x_0) - f\left(x_0\right) \\ f(f(x_0)) - f(f(x_0)) \\ f(f(b)) - f(f(f(x_0))) \\ f(f(c)) - f(f(f(b))) \end{bmatrix}$$

Each step of this global iteration can be parallelized by evaluating each row on a different cpu

Is Fixed Point Iteration Fast?

Solving $X^{(k+1)} = X^{(k)} - F\left(X^{(k)}\right)$

k=4

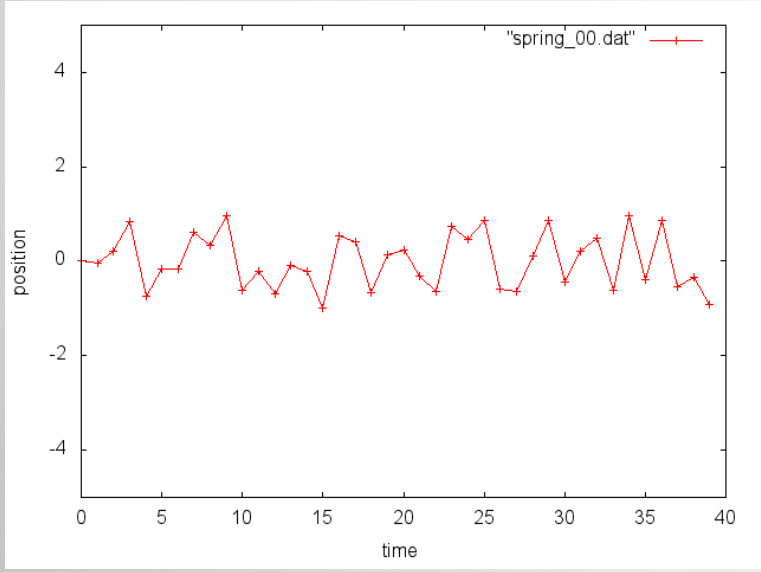
$$\begin{bmatrix} f(x_0) \\ f(f(x_0)) \\ f(f(f(x_0))) \\ f(f(f(f(x_0)))) \end{bmatrix} = \begin{bmatrix} f(x_0) \\ f(f(x_0)) \\ f(f(f(x_0))) \\ f(f(f(b))) \end{bmatrix} - \begin{bmatrix} f(x_0) - f(x_0) \\ f(f(x_0)) - f(f(x_0)) \\ f(f(f(x_0))) - f(f(f(x_0))) \\ f(f(f(b))) - f(f(f(f(x_0)))) \end{bmatrix}$$

Exact solution in 4 steps, but no speedup!

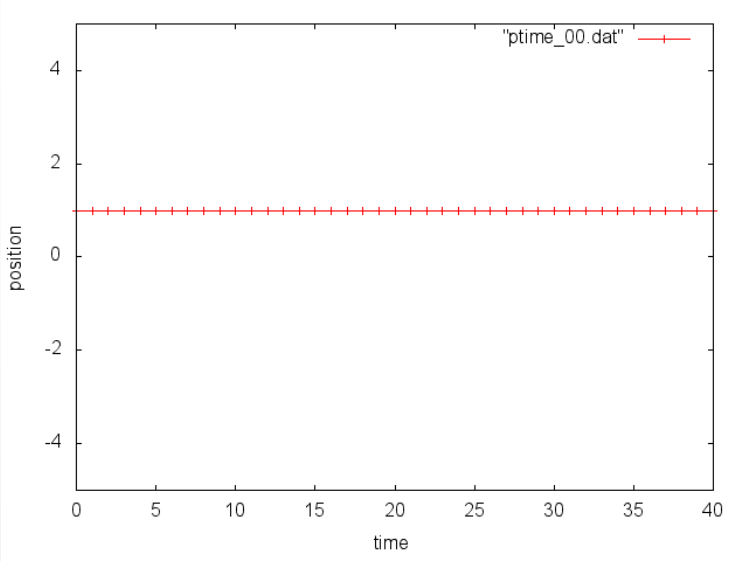
Improving Fixed Point Parallel in Time Algorithm

- For speedup to occur the root finding iterations needs to converge in less than M iterations, where M is the path length!
- Our strategy is to use a variety of optimization techniques, including preconditioned fixed-point, quasi-Newton, and preconditioned quasi-Newton optimization methods to obtain speedup
- Currently working on FAS methods

Fixed Point



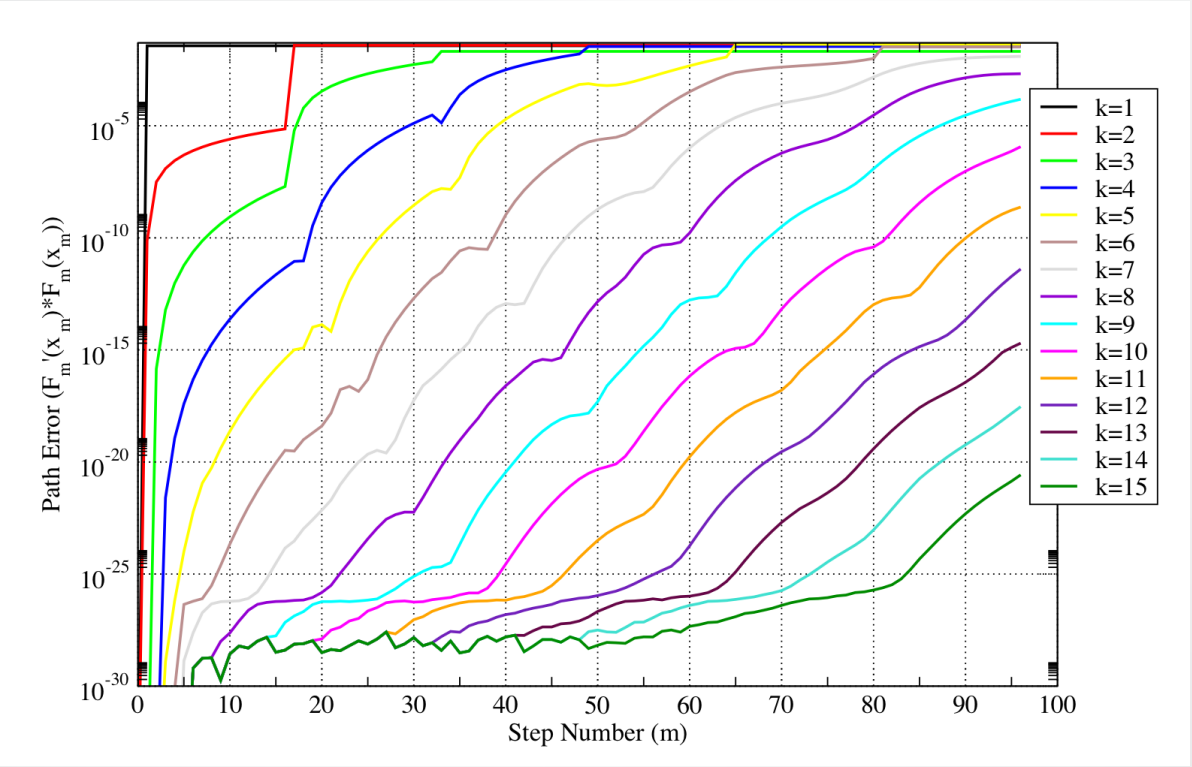
Quasi-Newton



Real Example: 1000 atom Stillinger-Weber MD Simulation

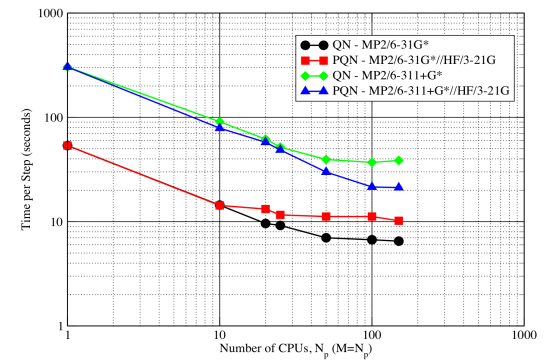
- The maximum ideal speedup (M/K) observed was 8.7 for M=96 and timestep=5.0, and N_{cpu}=M=96
- A true speedup of 5.2 was be obtained by parallelizing over time alone.

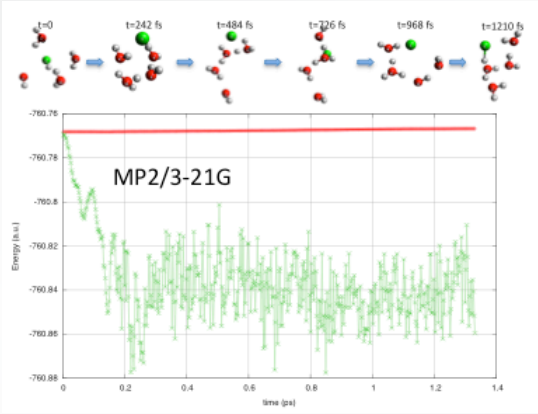
Path error for each sweep of a quasi-Newton parallel in time algorithm



Real Example: $HCl+4H_2O$ MP2 AIMD Simulations

- True speedup of 8.9 seen by parallelizing over time alone
- Energy conservation maintained



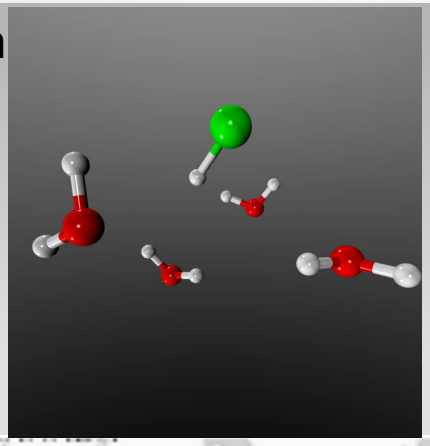


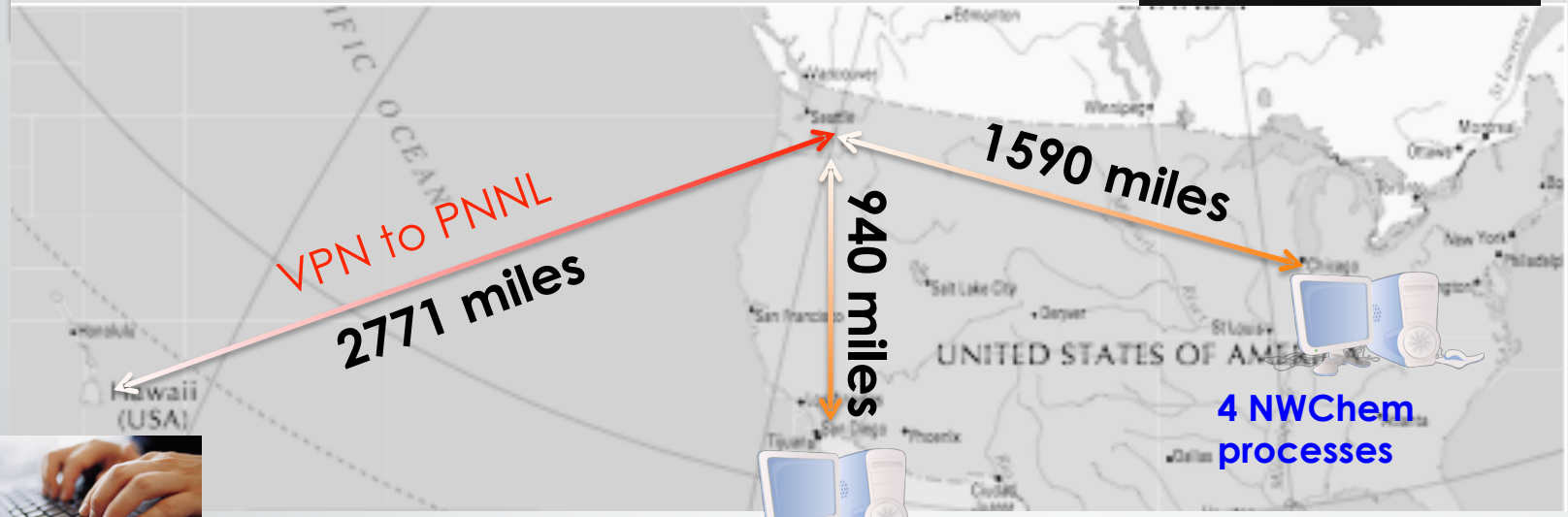
- Parallelization can also be done over forces and time. For a MP2/6-311++G(2d,2p)
  - **Parallelizing over force alone → 32 seconds per step**
    - 16 cpus → 93 seconds per step
    - 32 cpus → 54 seconds per step
    - 64 cpus → 32 seconds per step
  - **Paralleling forces and time → 6.9 seconds per step**
    - 16x40 cpus = 640 cpus → 16 seconds per step
    - 32x40 cpus = 1280 cpus → 11 seconds per step
    - 64x40 cpus = 2560 cpus → 6.9 seconds per step

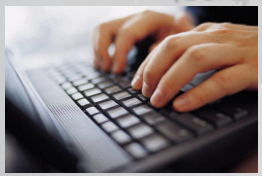
Useful Across Slow Networks: From Hawaii To Chicago/ San Diego And Back In Less Than 5 Seconds

$HCl+4H_2O$ MP2/3-21G Parallel-in-Time Simulation

Algorithm = stabilized quasi-Newton
Number of cpus = 10
Parallel time per step = 4.7 seconds
Serial time per step = 13.2 seconds
Pathlength = 20
Mathematical speedup= 20/3 = 6.67
Speedup with 10 cpus = 2.8
Parallel efficiency = 28%







Ptime driver, 2 NWChem processes

Summary

- Significant progress has been made in terms of accuracy, efficiency, and scalability of AIMD methods in recent years.
  - Demonstrated that it can be used to interpret EXAFS experiments
  - Hybrid-DFT, higher-levels methods, and Free energy methods now feasible for many systems
- New localization algorithms show promise for speeding up exact exchange calculations
- Parallel in time algorithms show promise in quantum chemistry and molecular dynamics
  - Newly developed quasi-Newton parallel in time algorithms are able to give factor of ~10 speedups, even without preconditioning
  - Suitable for cloud computing

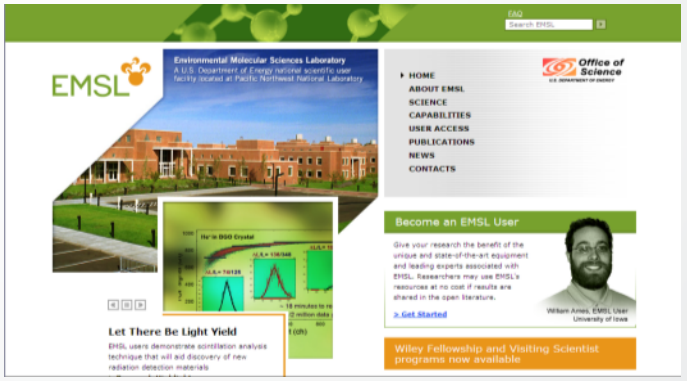
Acknowledgements BES Geosciences program of the U.S. Department of Energy, Office of Science ~ DE-AC06-76RLO 1830. Additional support from BES Heavy element program and EMSL operations. EMSL operations are supported by the DOE's Office of Biological and Environmental Research. We wish to thank the Scientific Computing Staff, Office of Energy Research, and the U. S. Department of Energy for a grant of computer time at the National Energy Research Scientific Computing Center (Berkeley, CA). Some of the calculations were performed on the Chinook computing systems at the Molecular Science Computing Facility in the William R. Wiley Environmental Molecular Sciences Laboratory (EMSL) at PNNL.
PNNL extreme scale LDRD for support for parallel implementations.

EMSL: A national scientific user facility

Mission

EMSL, a national scientific user facility at Pacific Northwest National Laboratory, provides ***integrated experimental and computational resources*** for ***discovery and technological innovation*** in the environmental molecular sciences to ***support the needs of DOE and the nation***.

- Funded by DOE **Office of Science's Office of Biological and Environmental Research (BER)**
- Located in **Richland, Washington**



www.emsl.pnl.gov