SAND2015-9550C

Exceptional service in the national interest

Sandia National Laboratories

Reactive Molecular Dynamics Simulations with LAMMPS

Ray Shan

Sandia National Laboratories

SciMeeting2015

Chicago, IL

November 3-4

U.S. DEPARTMENT OF ENERGY

National Nuclear Security Administration

LDRD LABORATORY DIRECTED RESEARCH & DEVELOPMENT

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2015-XXXXP

Outline

- **LAMMPS**
  - What is it and why use it

- **Reactive interatomic potentials in LAMMPS**
  - Charge optimized many body (COMB) potentials
  - Reactive force fields (ReaxFF)

- **Select new capabilities in LAMMPS**
  - On-the-fly chemical species analysis tool
  - Enabling LAMMPS on advanced computing architectures

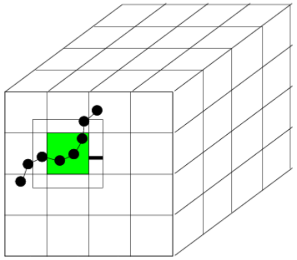
- **Conclusions**

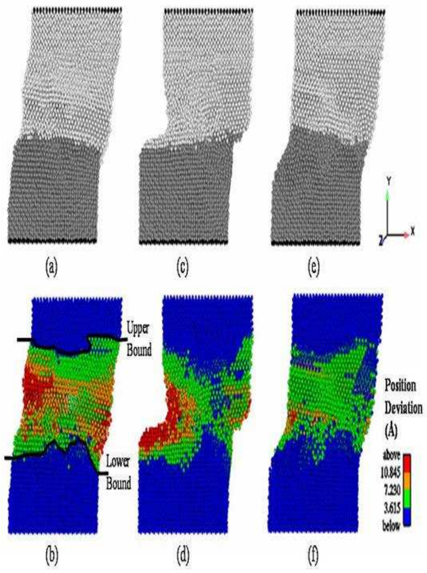
What is LAMMPS?

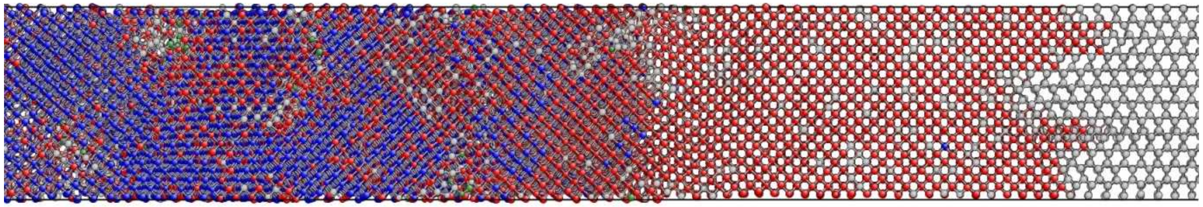
(Large-scale Atomic/Molecular Massively Parallel Simulator)

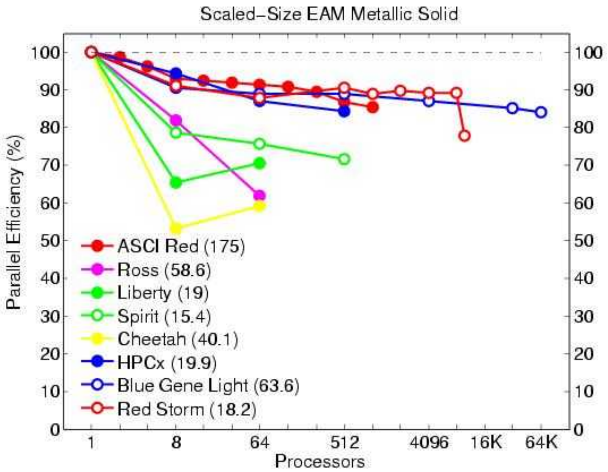
http://lammps.sandia.gov

- Classical MD code
- Open source, highly portable C++
- Freely available for download under GPL
- Easy to download, install, and run
- Well documented
- Easy to modify or extend with new features and functionality
- Active users e-mail list with over **650** subscribers
- Users' workshops: Feb '10, Aug '11, Aug '13, **Aug '15**
- Spatial-decomposition of simulation domain for parallelism
- GPU, Xeon Phi, and OpenMP enhanced
- Energy minimization, dynamics, non-equilibrium MD
- Atomistic, mesoscale, and coarse-grain simulations
- Variety of potentials (including many-body and coarse-grain)
- Variety of boundary conditions, constraints, etc.









Freely available parallel MD codes

- **CHARMM, AMBER: grand-daddies of MD codes, lots of bio features**
- **NAMD: bio, clever decomposition, very scalable**
- **GROMACS: bio, fastest single processor performance, now scalable**
- **DL-POLY: soft-materials**
- **GULP: crystalline inorganic solids**
- **HOOMD: GPU-based code, fastest on single GPUs**

- **LAMMPS**
  - materials oriented, wide range of interatomic potentials
  - many coarse-grained models for mesoscale to continuum
  - scalable for large simulations (1000s of particles/processor)
  - easy to extend

Why use LAMMPS?

- **Answer 1: Good Parallel performance**

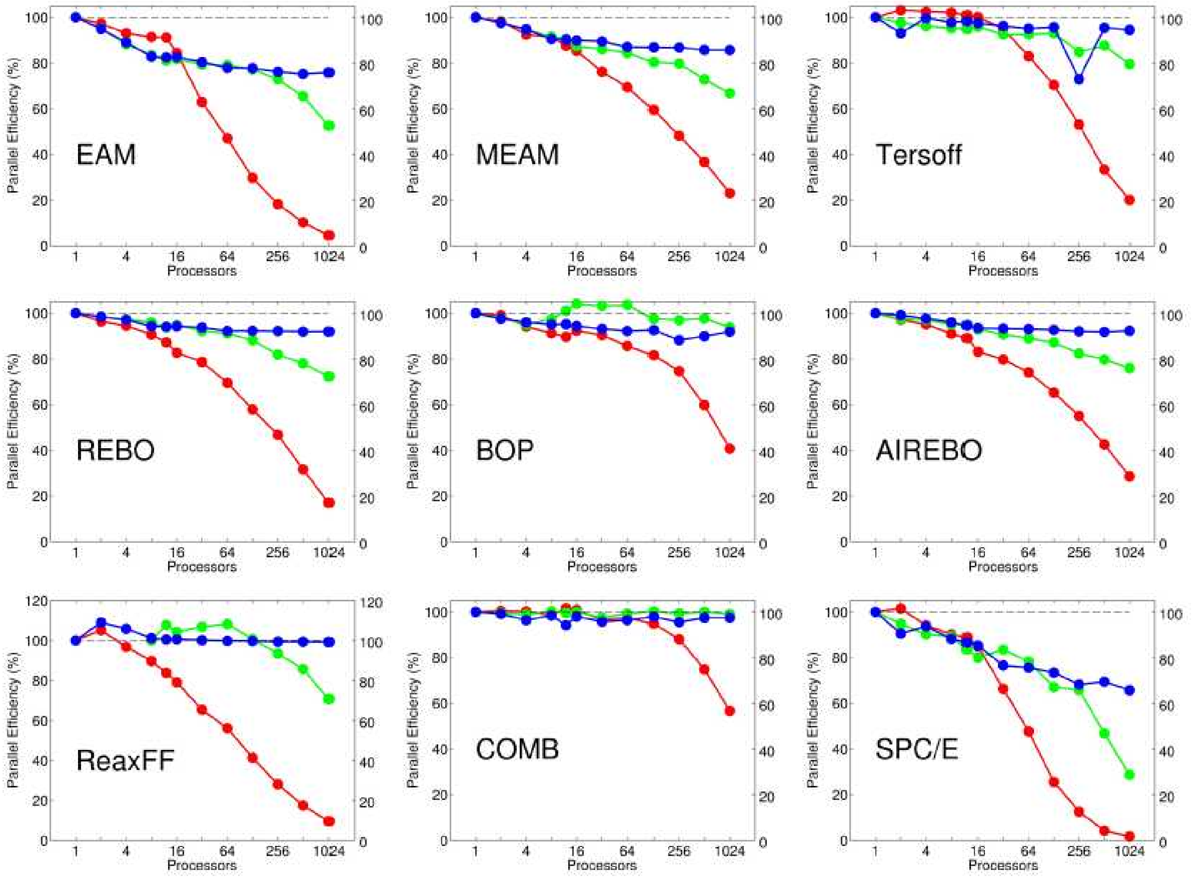


Figure 1: *Performance of 8 many-body potentials and an SPC/E water potential on varying numbers of cores of a Cray XT5 machine, as implemented in LAMMPS. For each potential, efficiency is defined as the one-processor timing divided by the P-processor timing, multiplied by 100/P. The red curves are for 32K atom systems, the green curves are for 1M atom systems; the blue curves are for scaled systems with 32K atoms per processor. The single-core CPU times per-atom per-timestep are listed in Table I.*

Red = 32k atoms
small fixed-size
strong scaling

Green = 1M atoms
large fixed-size
strong scaling

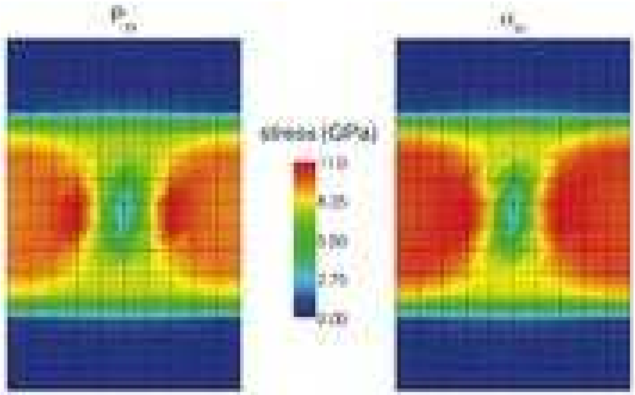
Blue = 32k atoms/proc
scaled size
weak scaling

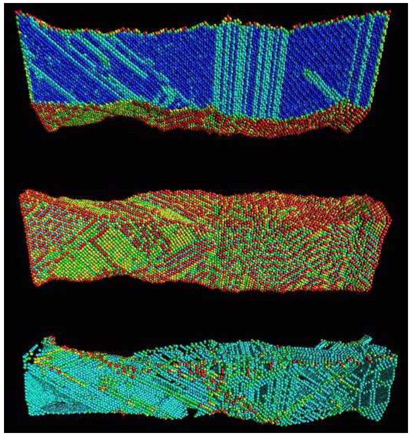
MRS Bulletin,
May 2012,
37, 513-521.

Why use LAMMPS?

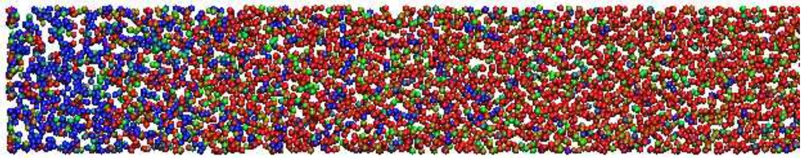
- Answer 2: Versatility

Solid
Mechanics



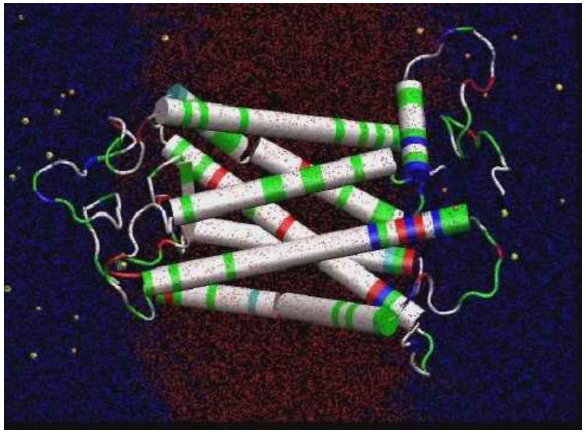


Materials
Science

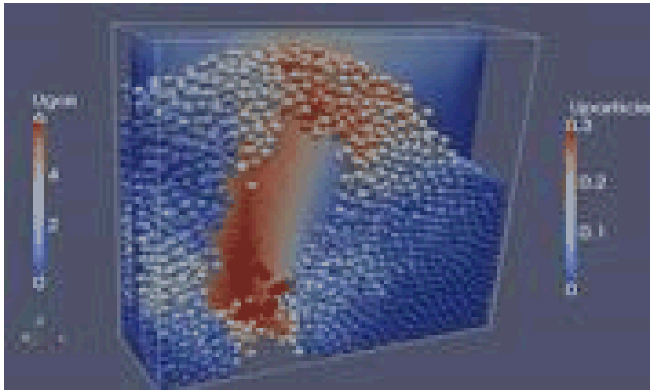


Chemistry

Biophysics



Granular
Flow



Why use LAMMPS?

- **Answer 3: Modularity**

LAMMPS Objects

atom styles: atom, charge, colloid, ellipsoid, point dipole

pair styles: LJ, Coulomb, Tersoff, ReaxFF, AI-REBO, COMB, MEAM, EAM, Stillinger-Weber,

fix styles: NVE dynamics, Nose-Hoover, Berendsen, Langevin, SLLOD, Indentation,...

compute styles: temperatures, pressures, per-atom energy, pair correlation function, mean square displacements, spatial and time averages

Goal: All computes works with all fixes work with all pair styles work with all atom styles

Why use LAMMPS?

- **Answer 4: Potential Coverage**

LAMMPS Potentials by Material

Biomolecules: CHARMM, AMBER, OPLS, COMPASS (class 2), long-range Coulombics via PPPM, point dipoles, ...

Polymers: all-atom, united-atom, coarse-grain (bead-spring FENE), bond-breaking, …

Materials: EAM and MEAM for metals, Buckingham, Morse, Yukawa, Stillinger-Weber, Tersoff, Streitz-Mintmire, COMB, SNAP, ...

Chemistry: REBO, AIREBO, ReaxFF, eFF

Mesoscale: granular, DPD, Gay-Berne, colloidal, peri-dynamics, DSMC...

Hybrid: can use combinations of potentials for hybrid systems: water on metal, polymers/semiconductor interface, colloids in solution, …

Why use LAMMPS?

- **Answer 4: Potential Coverage (cont’d)**

LAMMPS Potentials by Functional Form

pairwise potentials: Lennard-Jones, Buckingham, ...

charged pairwise potentials: Coulombic, point-dipole

manybody potentials: EAM, Finnis/Sinclair, modified EAM (MEAM), embedded ion (EIM), Stillinger-Weber, Tersoff, AI-REBO, ReaxFF, COMB

coarse-grained potentials: DPD, GayBerne, ...

mesoscopic potentials: granular, peridynamics

long-range electrostatics: Ewald, PPPM, MSM

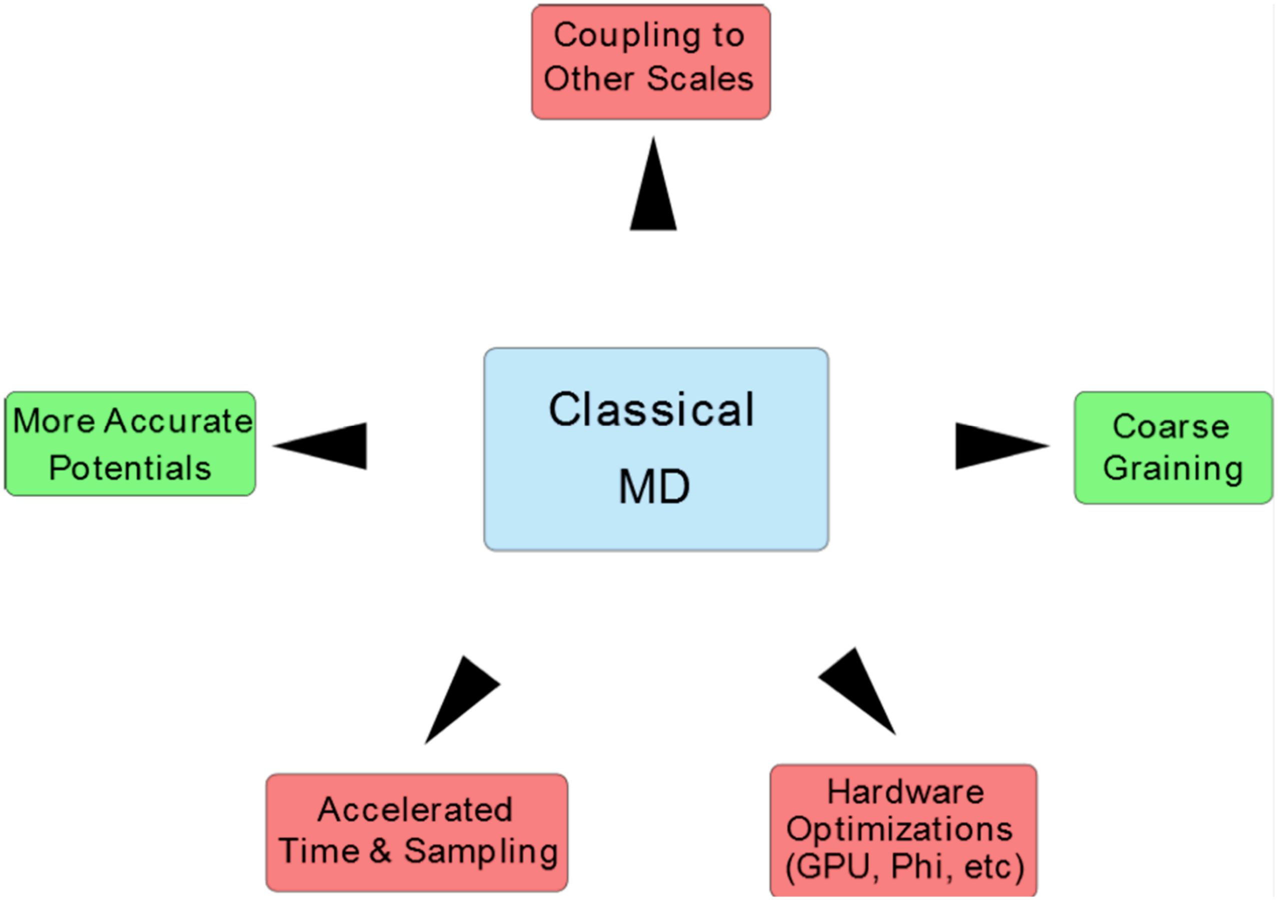
implicit solvent potentials: hydrodynamic lubrication, Debye
force-field compatibility with common CHARMM, AMBER, OPLS, GROMACS options

Why use LAMMPS?

- **Answer 5: Extendibility**

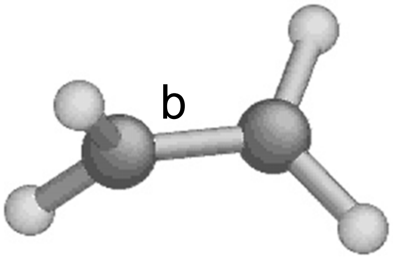
- **One of the best features of LAMMPS**
  - 90% of code is “extensions” via styles
  - only ~35K of 474K lines is core of LAMMPS
- **Easy to add new features via 14 “styles”**
  - new particle types = atom style
  - new force fields = pair style, bond style, angle style, dihedral style, improper style
  - new long range = kspace style
  - new minimizer = min style
  - new geometric region = region style
  - new output = dump style
  - new integrator = integrate style
  - new computations = compute style (global, per-atom, local)
  - new fix = fix style = BC, constraint, time integration, ...
  - new input command = command style = read_data, velocity, run, …
- **Enabled by C++**
  - virtual parent class for all styles, e.g. pair potentials
  - defines interface the feature must provide
  - compute(), init(), coeff(), restart(), etc

Directions for LAMMPS development

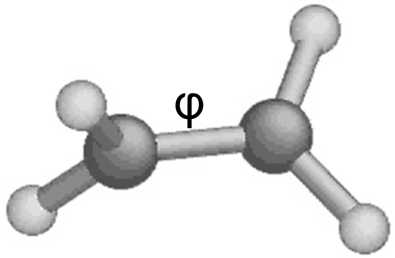


- **Any other suggestions/comments welcome**

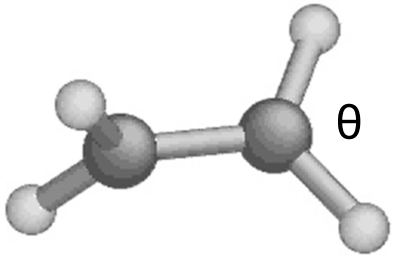
Conventional force fields



$$E_{bond-stretch} = \sum_{1,2\ pairs} K_b \left(b - b_0\right)^2$$



$$E_{rotate-along-bond} = \sum_{1,4\ pairs} K_\phi \left(1 - \cos\left(n\phi\right)\right)$$



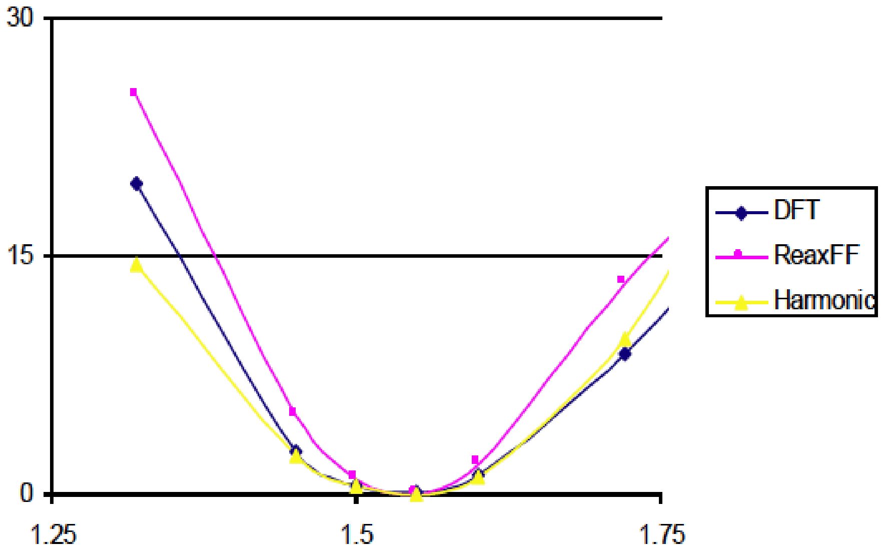
$$E_{bond-bend} = \sum_{angles} K_\theta \left(\theta - \theta_0\right)^2$$

- **Harmonic approximation: 2-body bond stretching + 3-body bond bending + 4-body dihedral**
  - Empirical; force constants fitted to experiments or first principles
  - Simple; MUCH faster than first principles methods; can be applied to MUCH bigger systems

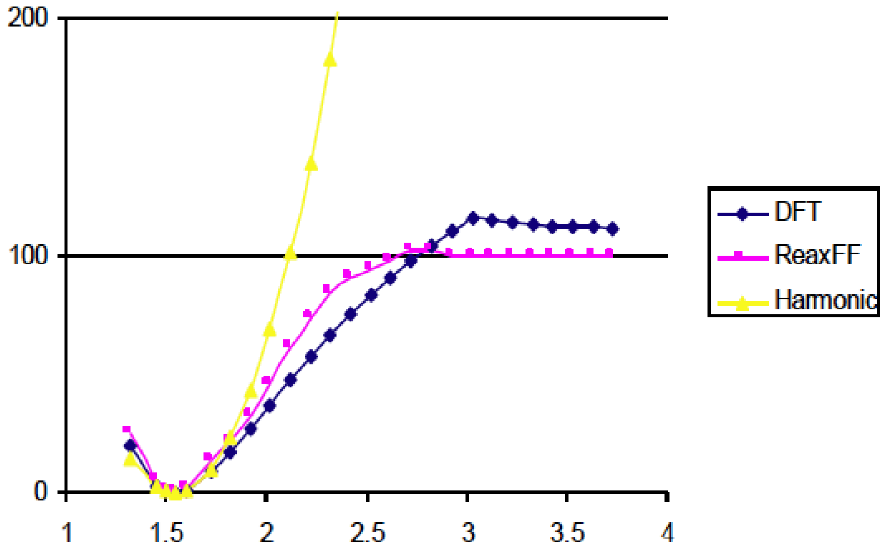
Failure of conventional force fields

- **C-C bond stretching in ethylene**
  - Although the harmonic approximation can describe C-C bond stretching around equilibrium, it cannot describe bond dissociation

Around equilibrium bond length



Full dissociation curve

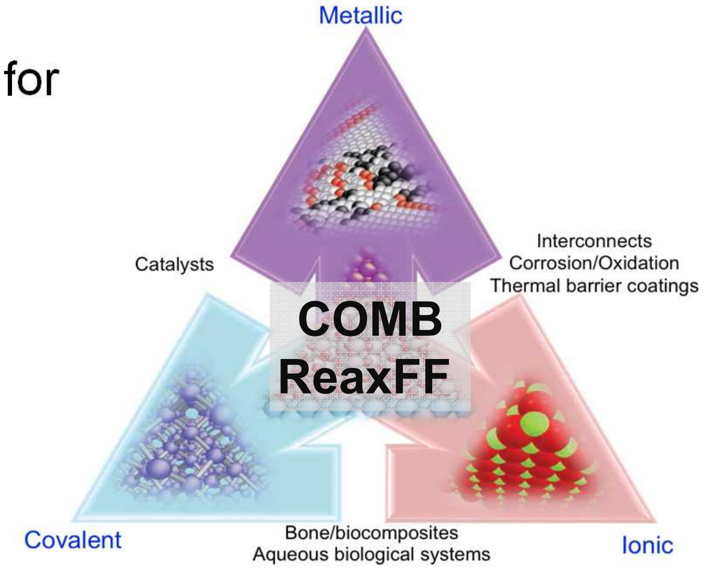


Courtesy of Adri van Duin, Penn St.

  - No or fixed charges; cannot model oxidation and reduction

Reactive interatomic potentials

- **Chemically reactive: REBO [1], AIREBO [2], Tersoff [3]**
  - Bond order; existing bonds can break; new bonds can form
- **Variable charge equilibration: Streitz-Mintmire [4], EIM [5]**
  - Atomic charges vary in response to environment
- **Chemically reactive, variable charge potentials**
  - Charge-optimized many body (COMB) [6]
  - Reactive force field (ReaxFF) [7]
  - Flexible for dissimilar materials; transferrable for varying bonding environments
  - Fitted to large amount of QC/QM calculations
  - Can simulate metallic/covalent/ionic bonds in one cell



[1] D. W. Brenner, *Phys. Rev. Lett.* 63, 1022 (1989)
[2] S. J. Stuart, A. B. Tutein, J. A. Harrison, *J. Chem. Phys.* 112, 6472 (2000)
[3] J. Tersoff, *Phys. Rev. B* 37, 6991 (1988)
[4] F. H. Streitz and J. W. Mintmire, *Phys. Rev. B* 50, 11996 (1994)
[5] X. W. Zhou, F. P. Doty, *Phys. Rev. B* 78, 224307 (2008)
[6] T.-R. Shan et al., *Phys. Rev. B* 81, 125328 (2010)
[7] A. C. T. van Duin et al., *J. Phys. Chem. A* 105, 9396 (2001)

Cost of empirical interatomic potentials

Potential	System	# Atoms	Memory	LJ Ratio
Lennard-Jones	LJ liquid	32000	12 Mb	1.0x
EAM	bulk Cu	32000	13 Mb	2.4x
REBO	polyehylene	32400	30.9 Mb	3.1x
Tersoff	bulk Si	32000	9.2 Mb	4.1x
Stillinger-Weber	bulk Si	32000	11 Mb	4.1x
EIM	crystalline NaCl	32000	14 Mb	6.5x
CHARMM + PPPM	solvated protein	32000	124 Mb	13.6x
AIREBO	polyethylene	32640	101 Mb	13.7x
MEAM	bulk Ni	32000	54 Mb	15.6x
Streitz-Mintmire	Al_2O_3	31500	209 Mb	111x
ReaxFF	PETN crystal	32480	976 Mb	**185x**
COMB	crystalline SiO_2	32400	85 Mb	**214x**
VASP/small*	**water**	**192 (512e$^-$)**	**320 procs**	**17.7×10^6**

*Not from LAMMPS

Charge-optimized many-body (COMB) potential

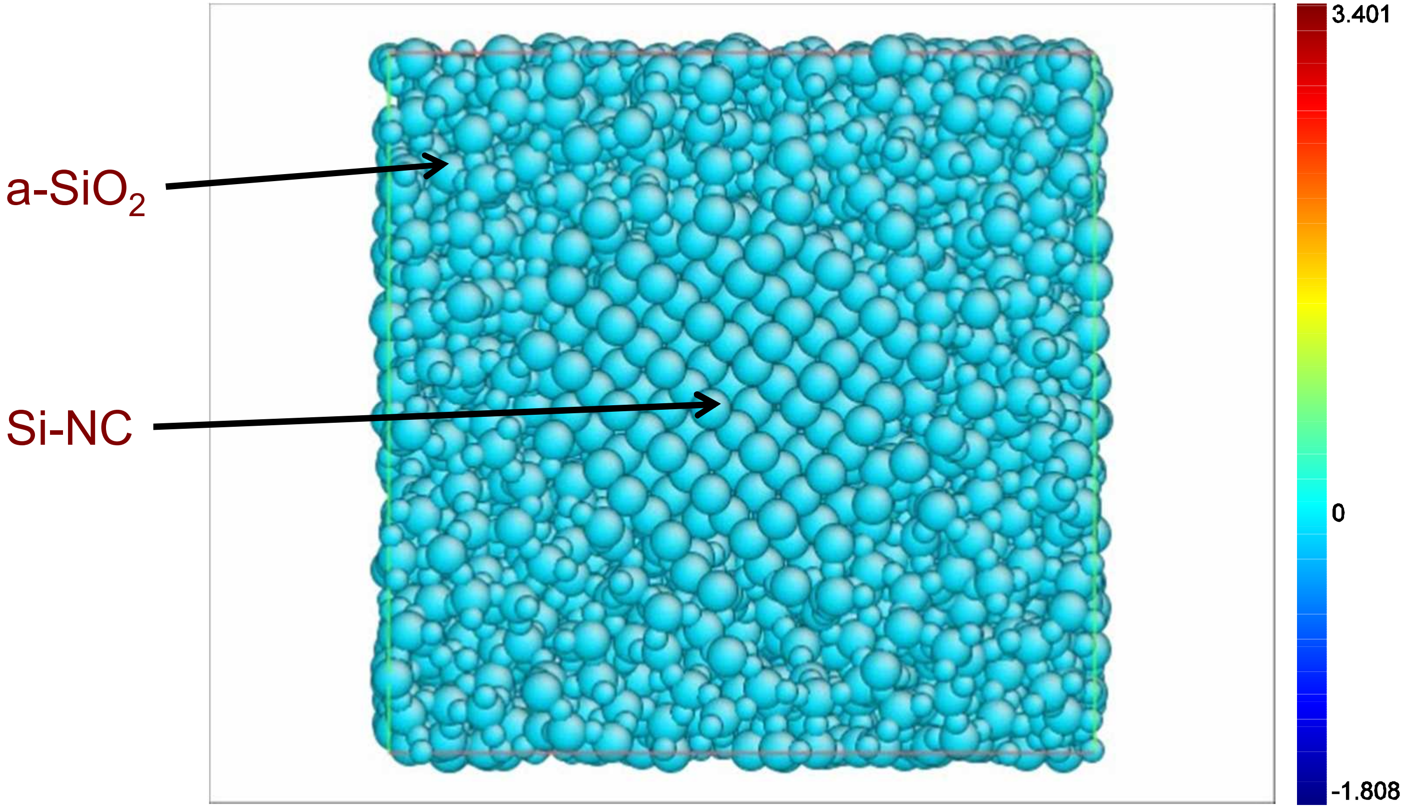
$$E_T = \sum_i \left\{ \begin{array}{l} E_i^{Self}(q_i) + \frac{1}{2}\sum_{j\neq i}\left[V_{ij}^{short}(r_{ij}, q_i, q_j) + V_{ij}^{Coul}(r_{ij}, q_i, q_j)\right] \\ +B_i(q_i) + C_i(r_{ij}, \theta_{ijk}) + E^{polar}(q_i, r_{ij}) + E^{vdW}(r_{ij}) \end{array} \right\}$$

- Self energy: ionization energies and electron affinities; includes penalty function to capture change in self-energy due to the field from the ionic lattice
- Short-range interactions: bond-order REBO potential
- Coulomb interactions: Streitz & Mintmire QeQ scheme
- Angular correction terms
- Polarization: Atomic polarizabililty for organic systems
- van der Waals energy: Lennard Jones

T.-R. Shan et al., *Phys. Rev. B* 81, 125328 (2010)
T. Liang et al., *Annu. Rev. Matter. Res.* 43, 109 (2013)

Variable charge in COMB

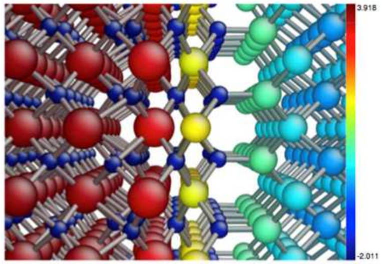
- **Si nanocrystal embedded in a-SiO_2 with COMB potential**



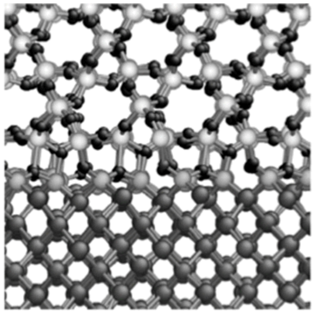
- **Variable charge potentials in LAMMPS include:**
  - EIM, and Streitz-Mintmire, COMB and ReaxFF

COMB potential in LAMMPS

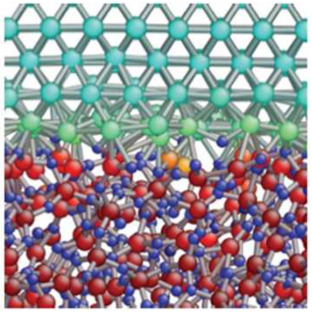
- 2^{nd} Generation: Si, Hf, Cu, Ti and their oxides



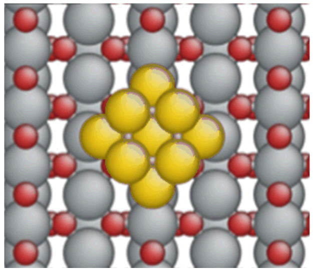
HfO_2, 2010



SiO_2, 2010

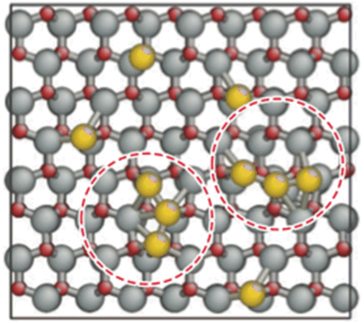


Cu/SiO_2, 2011

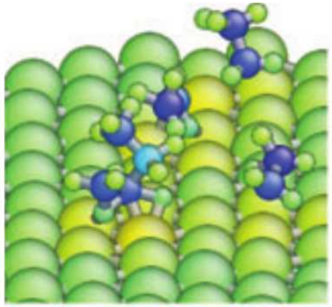


Cu/TiO_2, 2014

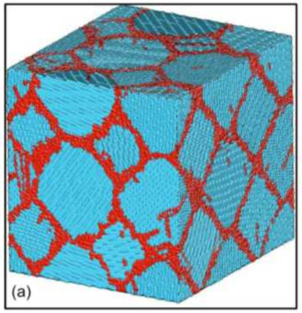
- 3^{rd} Generation: Cu, Zn, Ti, C, H, O, N and their mixtures
  - Also Zr, U, O



Cu/ZnO, 2012



Cu/C/H, 2013



ZrO_2, 2014

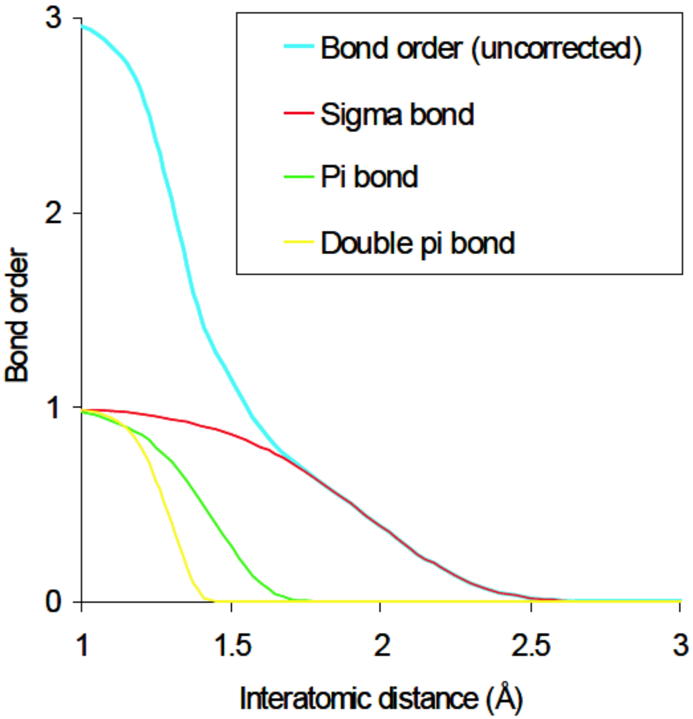
T.-R. Shan et al., *Phys. Rev. B* 81, 125328 (2010)
T. Liang et al., *Annu. Rev. Matter. Res.* 43, 109 (2013)

Reactive force field (ReaxFF)

- $$\begin{aligned} E^{ReaxFF} &= E^{self} + E^{Coul} + E^{vdW} + E^{bond} + E^{angle} + E^{torsion} \\ &+ E^{conjugation} + E^{H-bond} + E^{lone-pair} + E^{over} + E^{under} \\ &+ E^{others} \end{aligned}$$

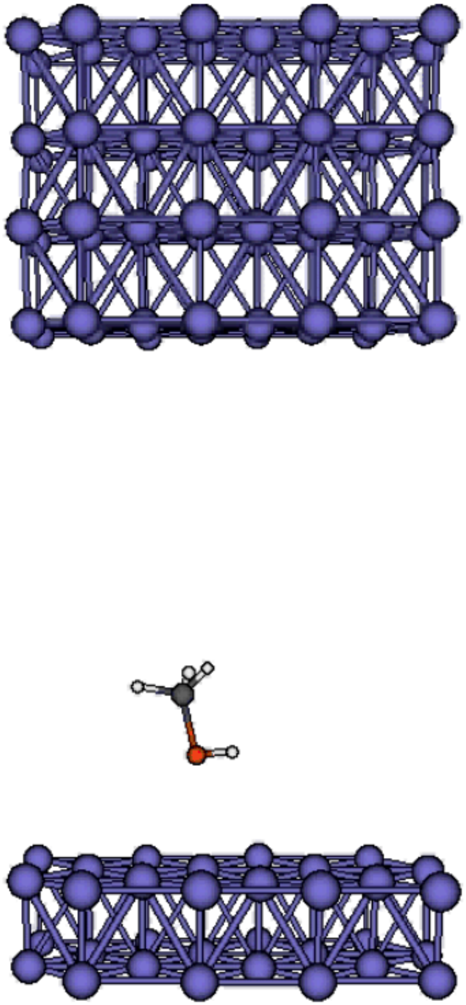
- **Calculation of bond orders from interatomic distances**

$$\begin{aligned} BO'_{ij} &= \exp\left[p_{bo,1} \cdot \left(\frac{r_{ij}}{r_o^{\sigma}}\right)^{p_{bo,2}}\right] \\ &+ \exp\left[p_{bo,3} \cdot \left(\frac{r_{ij}}{r_o^{\pi}}\right)^{p_{bo,4}}\right] \\ &+ \exp\left[p_{bo,5} \cdot \left(\frac{r_{ij}}{r_o^{\pi\pi}}\right)^{p_{bo,6}}\right] \end{aligned}$$

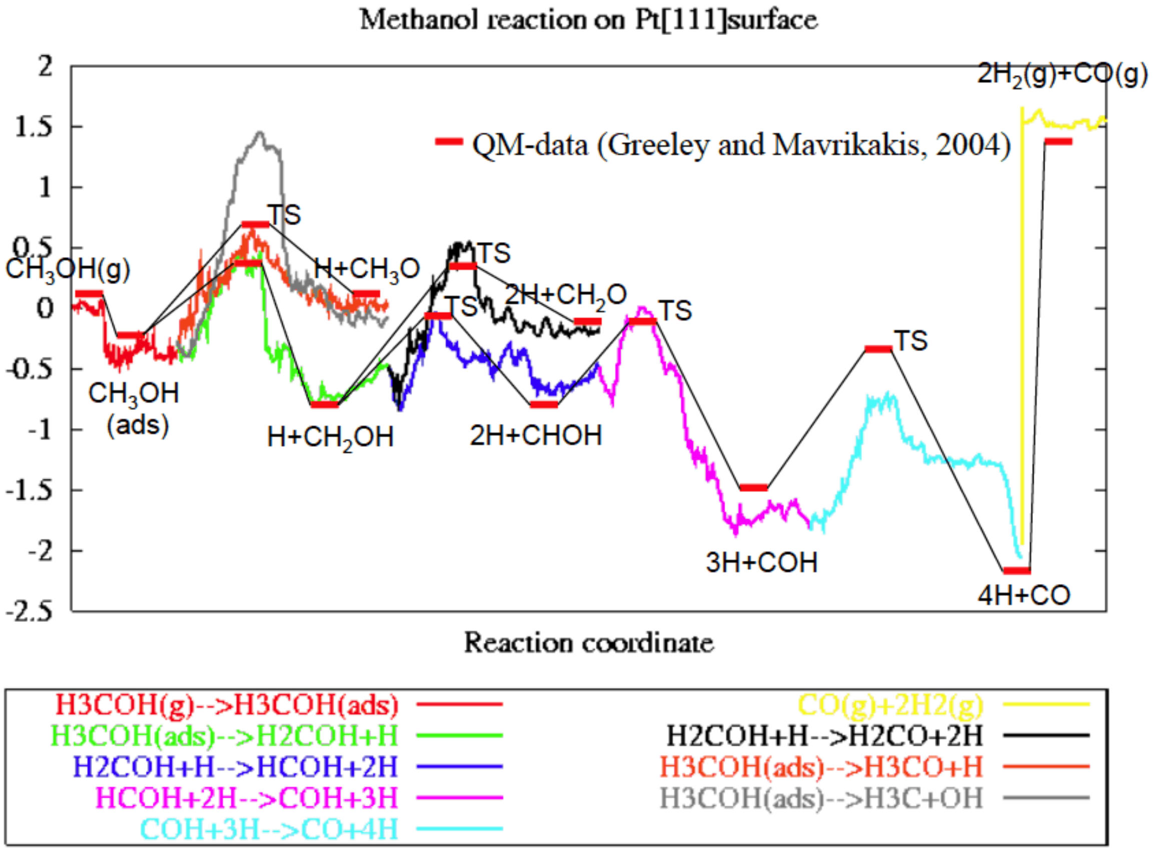


ReaxFF trained against QM data

- **Good agreement between ReaxFF and QM for entire reaction path**

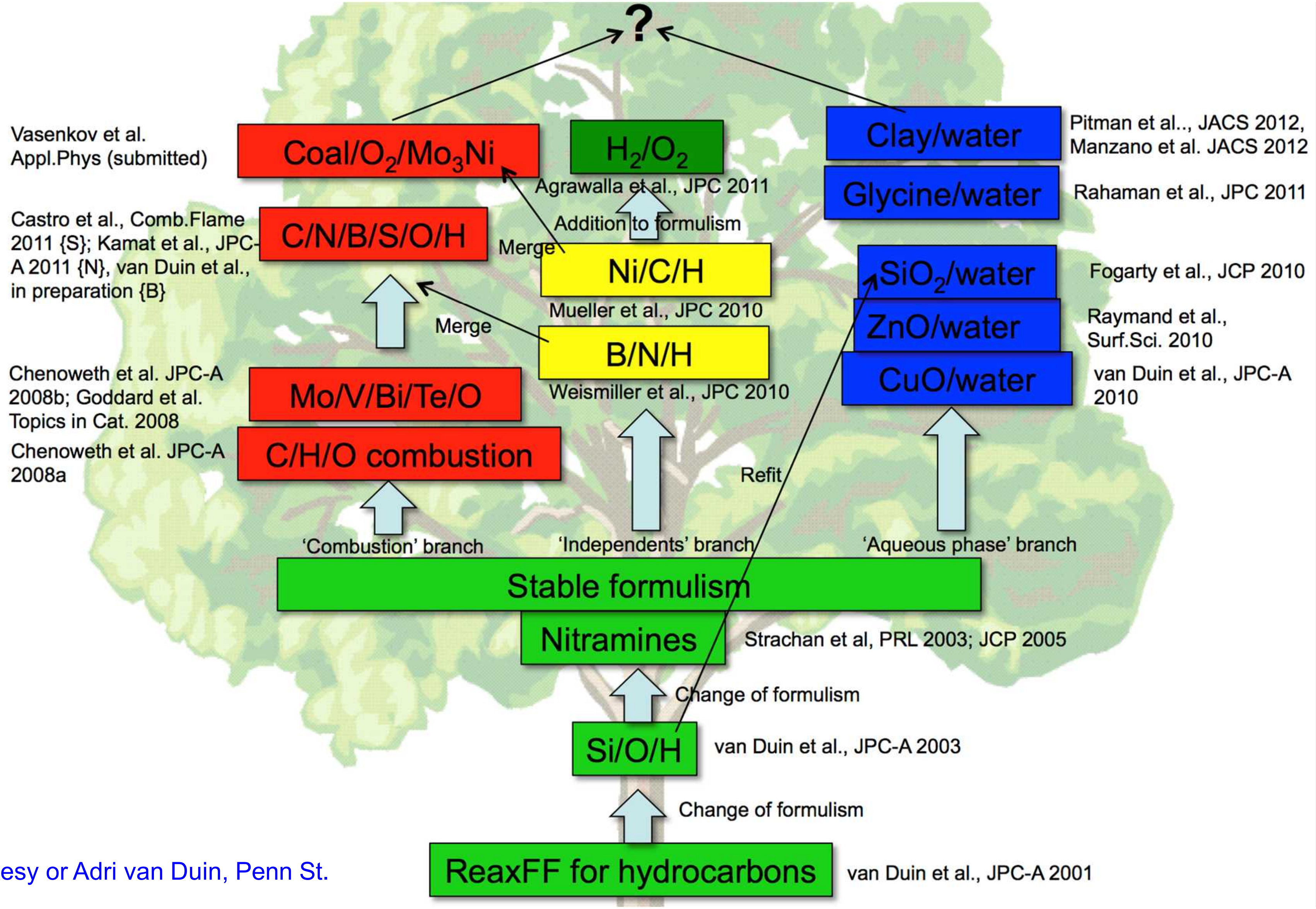


QM/ReaxFF methanol reaction pathways



Courtesy or Adri van Duin, Penn St.

ReaxFF development tree



?

Vasenkov et al. Appl.Phys (submitted)

Coal/O_2/Mo_3Ni

H_2/O_2

Agrawalla et al., JPC 2011

Addition to formulism

Clay/water

Pitman et al.., JACS 2012, Manzano et al. JACS 2012

Glycine/water

Rahaman et al., JPC 2011

Castro et al., Comb.Flame 2011 {S}; Kamat et al., JPC-A 2011 {N}, van Duin et al., in preparation {B}

C/N/B/S/O/H

Merge

Ni/C/H

Mueller et al., JPC 2010

SiO_2/water

Fogarty et al., JCP 2010

Merge

B/N/H

Weismiller et al., JPC 2010

ZnO/water

Raymand et al., Surf.Sci. 2010

CuO/water

van Duin et al., JPC-A 2010

Chenoweth et al. JPC-A 2008b; Goddard et al. Topics in Cat. 2008

Mo/V/Bi/Te/O

Chenoweth et al. JPC-A 2008a

C/H/O combustion

Refit

'Combustion' branch

'Independents' branch

'Aqueous phase' branch

Stable formulism

Nitramines

Strachan et al, PRL 2003; JCP 2005

Change of formulism

Si/O/H

van Duin et al., JPC-A 2003

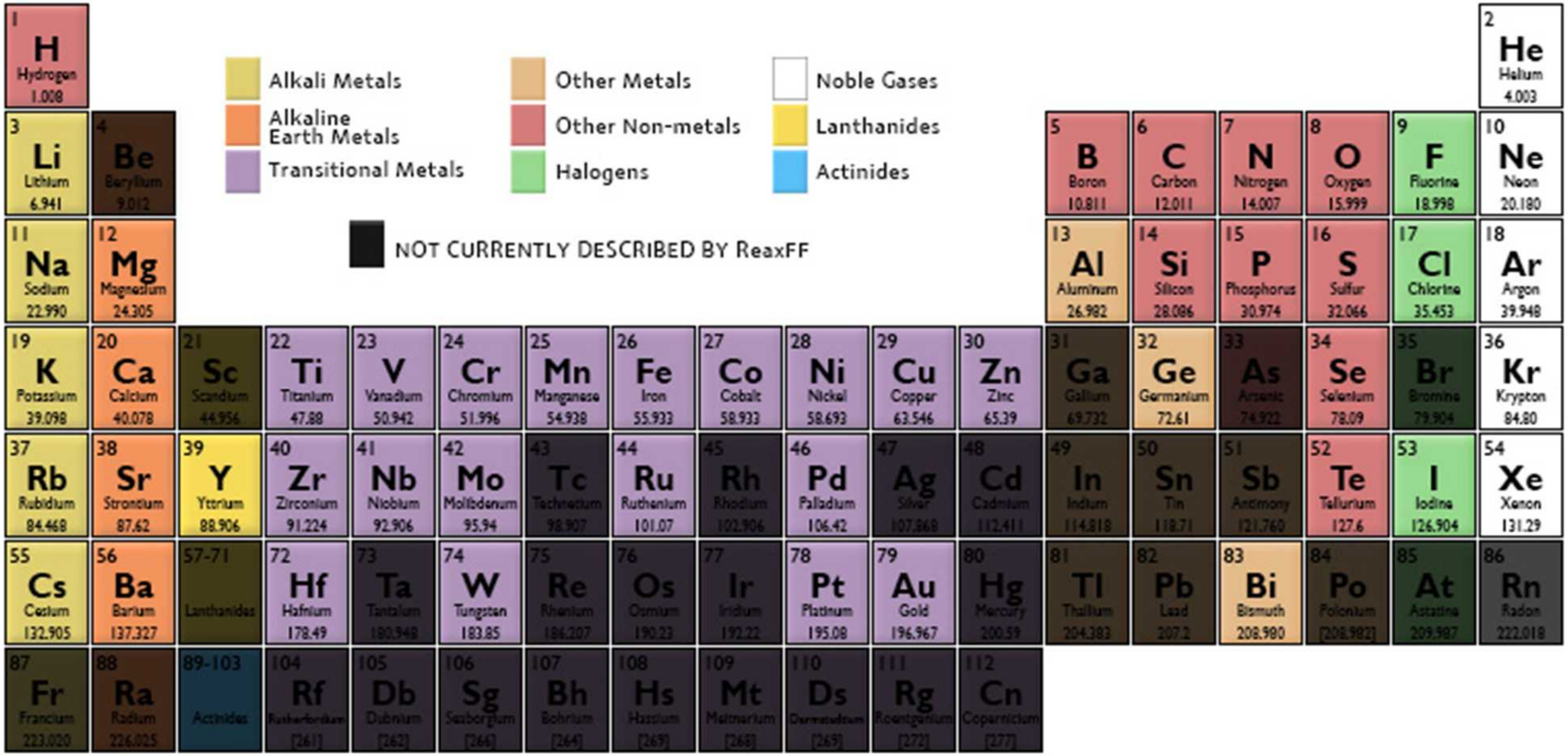
Change of formulism

ReaxFF for hydrocarbons

van Duin et al., JPC-A 2001

Courtesy or Adri van Duin, Penn St.

Current development status of ReaxFF



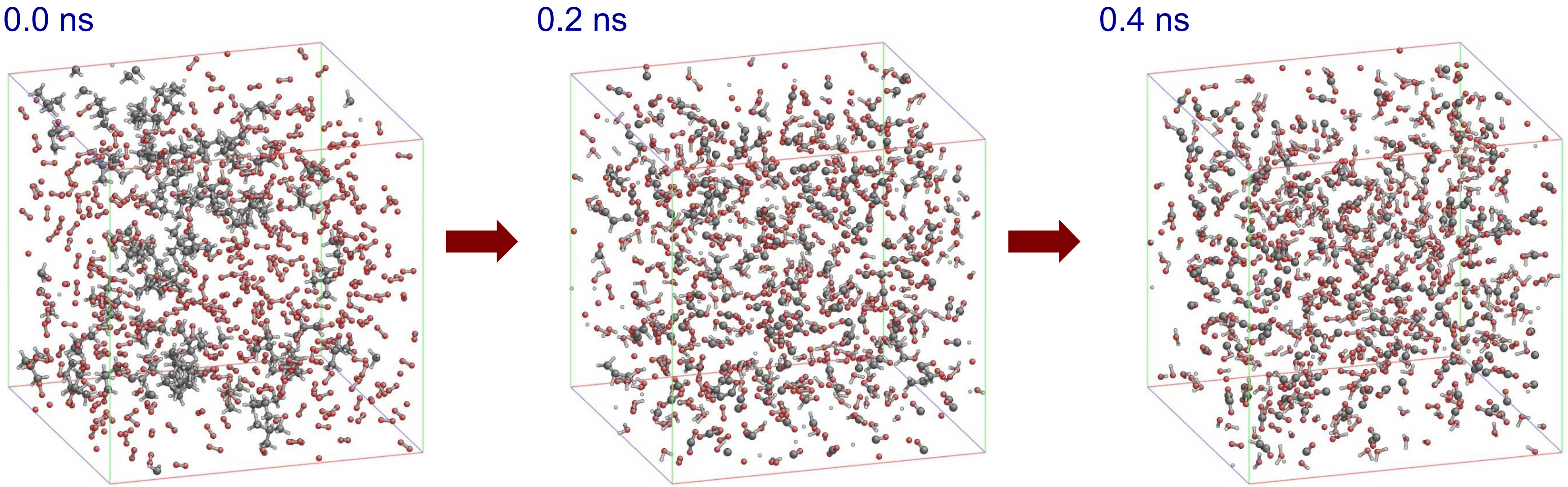
- 🟨 Alkali Metals
- 🟧 Alkaline Earth Metals
- 🟪 Transitional Metals
- 🟫 Other Metals
- 🟥 Other Non-metals
- 🟩 Halogens
- ⬜ Noble Gases
- 🟨 Lanthanides
- 🟦 Actinides
- ■ NOT CURRENTLY DESCRIBED BY ReaxFF

1 H Hydrogen 1.008																	2 He Helium 4.003
3 Li Lithium 6.941	■ 4 Be Beryllium 9.012											5 B Boron 10.811	6 C Carbon 12.011	7 N Nitrogen 14.007	8 O Oxygen 15.999	9 F Fluorine 18.998	10 Ne Neon 20.180
11 Na Sodium 22.990	12 Mg Magnesium 24.305											13 Al Aluminum 26.982	14 Si Silicon 28.086	15 P Phosphorus 30.974	16 S Sulfur 32.066	17 Cl Chlorine 35.453	18 Ar Argon 39.948
19 K Potassium 39.098	20 Ca Calcium 40.078	■ 21 Sc Scandium 44.956	22 Ti Titanium 47.88	23 V Vanadium 50.942	24 Cr Chromium 51.996	25 Mn Manganese 54.938	26 Fe Iron 55.933	27 Co Cobalt 58.933	28 Ni Nickel 58.693	29 Cu Copper 63.546	30 Zn Zinc 65.39	■ 31 Ga Gallium 69.732	32 Ge Germanium 72.61	■ 33 As Arsenic 74.922	34 Se Selenium 78.09	■ 35 Br Bromine 79.904	36 Kr Krypton 84.80
37 Rb Rubidium 84.468	38 Sr Strontium 87.62	39 Y Yttrium 88.906	40 Zr Zirconium 91.224	41 Nb Niobium 92.906	42 Mo Molibdenum 95.94	■ 43 Tc Technetium 98.907	44 Ru Ruthenium 101.07	■ 45 Rh Rhodium 102.906	46 Pd Palladium 106.42	■ 47 Ag Silver 107.868	■ 48 Cd Cadmium 112.411	■ 49 In Indium 114.818	■ 50 Sn Tin 118.71	■ 51 Sb Antimony 121.760	52 Te Tellurium 127.6	53 I Iodine 126.904	54 Xe Xenon 131.29
55 Cs Cesium 132.905	56 Ba Barium 137.327	■ 57-71 Lanthanides	72 Hf Hafnium 178.49	■ 73 Ta Tantalum 180.948	74 W Tungsten 183.85	■ 75 Re Rhenium 186.207	■ 76 Os Osmium 190.23	■ 77 Ir Iridium 192.22	78 Pt Platinum 195.08	79 Au Gold 196.967	■ 80 Hg Mercury 200.59	■ 81 Tl Thallium 204.383	■ 82 Pb Lead 207.2	83 Bi Bismuth 208.980	■ 84 Po Polonium [208.982]	■ 85 At Astatine 209.987	■ 86 Rn Radon 222.018
■ 87 Fr Francium 223.020	■ 88 Ra Radium 226.025	■ 89-103 Actinides	■ 104 Rf Rutherfordium [261]	■ 105 Db Dubnium [262]	■ 106 Sg Seaborgium [266]	■ 107 Bh Bohrium [264]	■ 108 Hs Hassium [269]	■ 109 Mt Meitnerium [268]	■ 110 Ds Darmstadtium [269]	■ 111 Rg Roentgenium [272]	■ 112 Cn Copernicium [277]						

On-the-fly species analysis

- **Motivation**
  - Very difficult to identify molecules and species in large-scale reactive simulations
    - < 100 atoms: naked eye; < 10,000 atoms: serial post process; >>10,000 atoms: ?
    - What we don't want: post-processing with serial codes, writing/storing large trajectory files, etc
- **For example:**
  - Combustion of *i*-butane at 2500K: $2C_4H_{10} + 13O_2 \rightarrow 8CO_2 + 10H_2O$
  - Started with 64 C_4H_{10} and 416 O_2

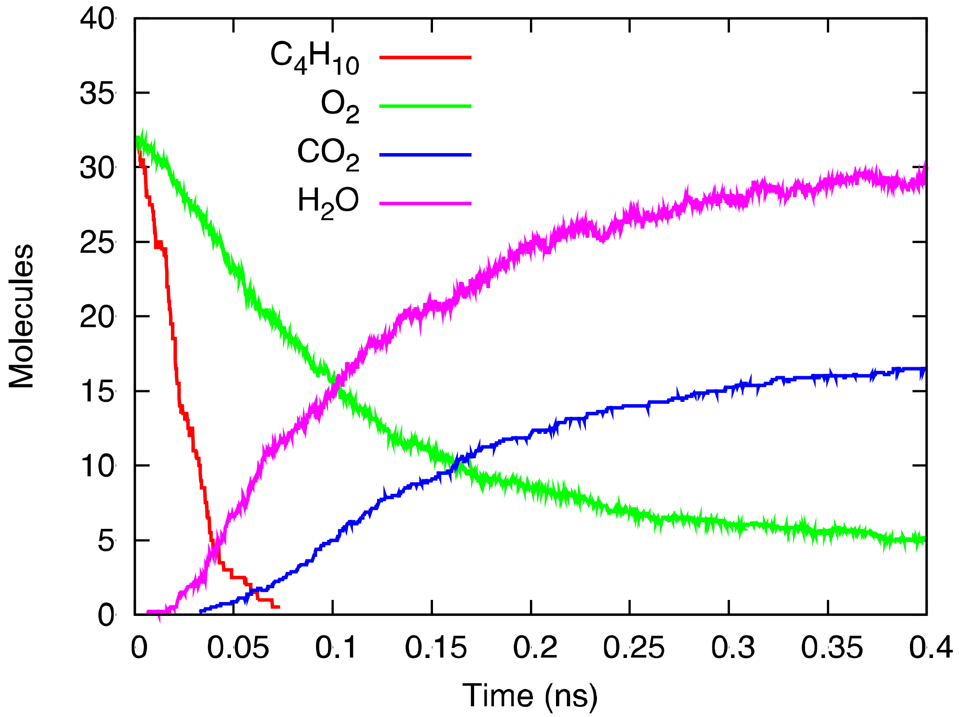
How many of what molecules at later time?



On-the-fly species analysis

- **Identify number of molecules and species *on-the-fly!***
  - Concise report as the simulation runs; in parallel (efficient)
  - No writing/storing unnecessary trajectory files

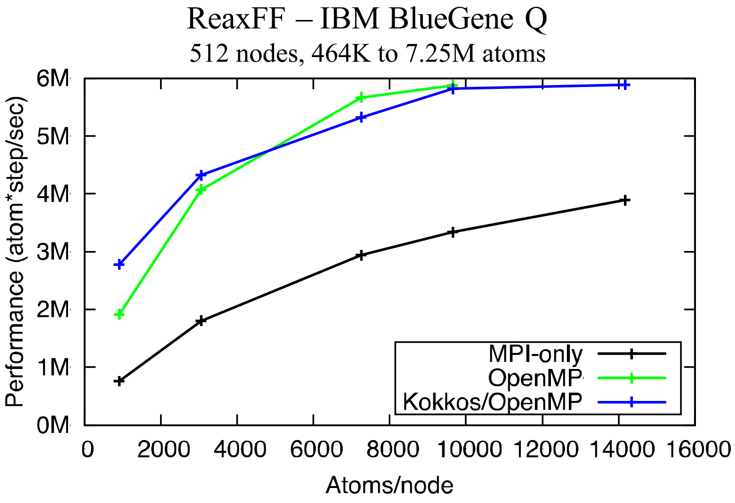
```
# Timestep     No_Moles     No_Specs     C4H10  O2
1000                    480        2          64    416
...
# Timestep     No_Moles     No_Specs     CO2    CO      H2O     CH2O
CH2O2   H3O     C2HO2   HO2     CHO2    C2H4O2  CH4O    C3H4O   HO
H4O2    C2H3O   C3H5    C2H2O   H2      CHO     C2H4    C2H4O   CH2O3
C3H3O   C2H2O2  C2H2    C2O2    O2
1000000       610        27             97    86      248      25      4       1       1
6       3
2          2       1       5       1       1       1
3          3       2       1       3       1       1
4          1       1       111
...
# Timestep     No_Moles     No_Specs     CO     H2O     CH2O    CO2
C2H2O2  HO      CH2O2   CH4O    C2H2O   CHO2    H2      HO2     C2H4O
O2
2000000       628        14         106    300
5         132      3       8       1       1       1
1       1       1       1       67
```

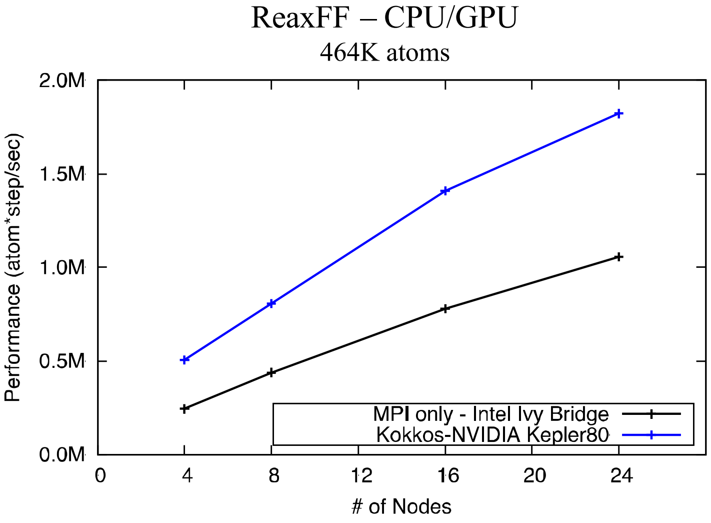


Enabling Kokkos within LAMMPS

- **A collection of C++ libraries developed at Sandia**
  - Used on top of existing MPI parallelization (MPI + X); runs on NVIDIA GPU, Intel Xeon Phi, conventional multi-core CPU (IBM, Intel, AMD, etc)
  - 6 atoms styles, 35 pair styles, 2 fix styles, 2 bond styles, 2 angle styles, 1 dihedral style, 1 improper
  - In progress: PPPM, fix nvt/npt, ReaxFF

1 of the only 3 implementations of ReaxFF that runs on GPU!





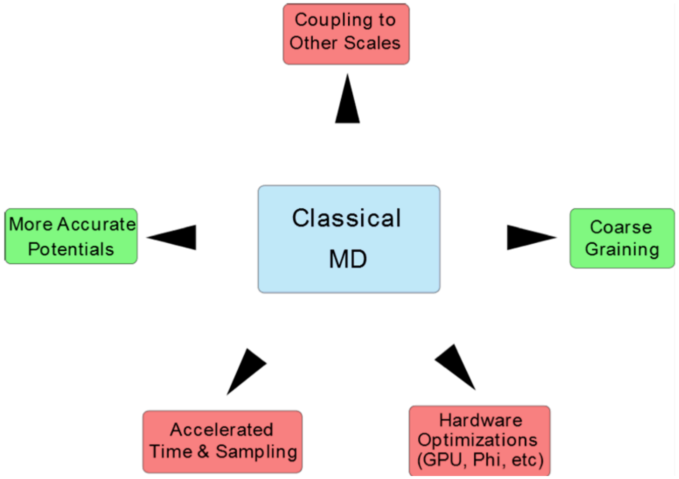
- Impact: **faster** (simulation speed), **greener** (power consumption) and **cheaper** (lower cost computers) simulations!

Other exciting developments/additions

- **SNAP potential**
  - Automated generation of potentials with quantum-level accuracy
- **Coarse-grained models**
  - Particle-particle, stochastic rotation dynamics, fast lubrication dynamics, aspherical particles, dissipative particle dynamics, peridynamics, smoothed particle hydrodynamics, granular models
- **Load balancing**
  - Static and dynamic commands
- **Accelerated MD methods**
  - Parallel replica dynamics, temperature accelerated dynamics
- **Many-core computing architecture support**
  - GPU, Xeon Phi, and OpenMP support via multiple packages
- **Automated regression testing**
  - Test suite includes all examples; driven by Jenkins software
  - Increases the stability and reliability of LAMMPS

Conclusions

- **We welcome feature requests from users**
- **We welcome new LAMMPS developers!**
- **Biennial LAMMPS' workshop and symposium**
  - Next one in August of 2017



- **Acknowledgements**
  - Scienomics!
  - All developers and contributors: Steve Plimpton, Aidan Thompson, Paul Crozier, Axel Kohlmeyer, Ray Shan, Stan Moore, Christian Trott, Reese Jones, Jon Zimmerman, Roy Pollock, Mike Brown, Greg Wagner, Mike Parks, Andres Jaramillo-Botero, Christoph Kloss, Metin Aktulga, Jeremy Lechman, Pieter in't Veld, Jeremy Templeton, Amit Kumar, Stuart Silling, and many many more!