



Scientific Computing & Modelling

ReaxFF Manual

**ADF Program System
Release 2014**

Scientific Computing & Modelling NV
Vrije Universiteit, Theoretical Chemistry
De Boelelaan 1083; 1081 HV Amsterdam; The Netherlands
WWW: www.scm.com
E-mail: support@scm.com

Copyright © 1993-2014: SCM / Vrije Universiteit, Theoretical Chemistry, Amsterdam, The Netherlands
All rights reserved

Table of Contents

ReaxFF Manual	1
Table of Contents	2
Preface	3
MCFFOptimizer: Monte Carlo force-field parameter optimizer	4
Input files.....	4
Control parameters	4
Results.....	5
Run-time control.....	5
Grand Canonical Monte Carlo (GCMC)	7
General info.....	7
Input.....	7
Output.....	11
Code Details.....	13
Included Forcefields.....	15

Preface

This manual describes additions and modifications by SCM.

MCFFOptimizer: Monte Carlo force-field parameter optimizer

The MCFFOptimizer uses a Monte Carlo approach to finding the best-fit force-field for the given training set. It is based on the paper by [E. Iype et. al.](#) In the following sections, the input and output of the MCFFOptimizer are described.

Input files

In order to optimize a ReaxFF forcefield, the files listed below must be present in the directory where the reaxff program is executed.

- *iopt* – file containing a single text line with 4 on it, written in the (i3) format. This will instruct reaxff to perform a Monte-Carlo force-field optimization.
- *ffield* – the initial force-field file.
- *ffield_min* – minimum values for each force-field value (this file has the same format as *ffield*).
- *ffield_max* – maximum values for each force-field value (this file has the same format as *ffield*).
- *ffield_bool* – contains 1.0 or 0.0 as a flag whether the corresponding value is to be optimized or not (this file has the same format as *ffield*).
- *trainset.in* – file with test values, the same as in the original reaxff force-field optimization, see page 27 of the ReaxFF User Manual.
- *geo* – file with test geometries in the BGF format, the same as in the original reaxff force-field optimization. Geometries of different molecules in this file must be concatenated.
- *control* – in addition to general ReaxFF control parameters it also contains the MCFFOptimizer-related ones explained below.

The *mcffopt_water* example in the *examples/reaxff* directory demonstrates the use of the MCFFOptimizer. Note however that this example is not physically meaningful. For example, many atomic force-field parameters are allowed to vary in a very broad range, which is the same for all elements. In practice, you will want to set the range for each element separately.

Control parameters

The following control parameters are related to MCFFOptimizer. The default value for each parameter is given in parentheses.

- *mcffit* – number of MC iterations (10000). Since the Monte-Carlo method does not have any notion of convergence the optimization is stopped after *mcffit* iterations.
- *mcbeta* – initial MC beta parameter in the acceptance probability calculation $P = \exp(-\beta \cdot \Delta E)$ (1.0). Here, ΔE is a difference in the error function between the current and the best step so far. If the current step is the best, it is always accepted. Otherwise, the acceptance probability is calculated using the formula above and it is calculated with a random number from the (0,1) range. The optimal value of beta depends on values of the error function.
- *mcdbet* – simulated annealing: increase the beta parameter by this value at each step (0.0). A positive *mcdbet* value means that the probability to take a step that increases the error function will decrease over time. This has the same effect as decreasing the temperature in the classical molecular Monte-Carlo method.
- *mcbzca* – simulated annealing: divide the beta parameter by this value at each step (1.0). A value of *mcbzca* < 1.0 has an effect similar to positive *mcdbet*.
- *mcacpf* – probability to vary a variable at each step (0.2). To avoid taking very large steps only some of the variables are varied at each step (selected randomly).
- *mcrxdd* – number of steps to divide the parameter range between *ffield_min* and *ffield_max* into (100).

- *mcstep* – initial max step size in units of range/mcrxdd, where range = difference between ffield_max and ffield_min values (1.0).
- *mcmxst* – maximum allowed value of max step size (100);
- *mcscps* – factor to scale max step size to satisfy acceptance tolerance (1.1);
- *mctart* – target acceptance rate, percent (30.0);
- *mcmart* – max acceptance rate, percent (70.0);
- *mcmmini* – if not 0, minimize the best force-field parameter set after so many iterations (0). The optimization is performed only if the best set has changed since the previous minimization. The minimization employed here is gradient-free and relatively slow so it should not be used too frequently.
- *replic* – number of replicas to try at each step (1). At each step, replic Monte-Carlo steps are done and the one with the lowest error is selected for the next iteration.

Relation between mcrxdd, mcstep, mcmxst, and mcscps. The allowed range for each parameter is divided into mcrxdd steps. At each optimization step, a number of force-field parameters is changed by the $\Delta x = mcstep \cdot (Xmax - Xmin) / mcrxdd$, where Xmax and Xmin are values of this parameter from ffield_max and ffield_min, respectively. When performing optimization, the program keeps track of the average acceptance rate and adjusts mcstep up or down by the mcscps factor to keep the acceptance rate close to mctart. If the acceptance rate is too low the step size is decreased to allow searching for a smaller parameter space. The mcstep value can never be larger than mcmxst.

It should be noted that the value of the MC step size (and thus all the parameters discussed in this section) applies to all force-field parameters to the same extent, which means that it is very important to select the min and max parameter values very carefully. The rule of thumb here is that the range should be as small as possible covering only the physically meaningful values.

Results

Main results of the MCFFOptimizer are saved in the following files:

- ffield_best - force-field file corresponding to the lowest error value
- ffield_last - the most recent accepted force-field
- MCFFOptimizer.log - summary of iterations including the error function value, number of changed and bounded force-field parameters, cumulative number of accepted and rejected steps at each step. Also the current MC parameters such as the β value and the acceptance rate, are shown, as well as the elapsed time in seconds.
- fort.99 - the error function breakdown for the latest step

Run-time control

The progress of the force-field optimization can be controlled by changing parameters in the *istop* file present in the calculation directory. The file is read every 10 iterations. The parameters are explained below:

- StopKey - replace 0 with 1 to stop the calculation
- Beta - the current β value corresponding to the *mcbeta* control parameter
- Command - one of: NONE, WALK, JUMPWALK, MINIMIZE, JUMPMINI:
 - NONE - no change in the procedure
 - WALK - switch to Monte-Carlo steps (the default);
 - MINIMIZE - switch to gradient-free minimization of the latest accepted force-field;
 - JUMPWALK - take the best force-field so far and start the Monte-Carlo procedure from there;
 - JUMPMINI - take the best force-field so far and minimize it.
- ScaleFactor - change the current step size corresponding to the *mcstep* control parameter

- ActiveParameterFraction - set the fraction of force-field parameters changed at each step (*mcacpf*)
- deltaBeta - change the current value of the *mcdbet* control parameter
- BetaScaling - change the current value of the *mcbzca* control parameter
- ScaleParSpace - change the current value of the *mcszps* control parameter

Grand Canonical Monte Carlo (GCMC)

General info

About Monte Carlo / the Grand Canonical Ensemble

It is best to read a bit about Monte Carlo and ensembles before working with the GCMC code. Almost every book or review text on molecular simulations will do, for example: Frenkel D, Smit B. Understanding molecular simulation: from algorithms to applications. Academic Press; 2002. 672 p.

Wikipedia also has some pages of interest:

- http://en.wikipedia.org/wiki/Monte_Carlo_method
- http://en.wikipedia.org/wiki/Grand_canonical_ensemble

It is important to note that this method heavily relies on random numbers, and simulations are thus non-repeatable in detail, but should converge to the same answer.

About the Reaxff GCMC code

The GCMC code for reaxff was originally developed by Thomas Senftle, working as a Graduate Student at Penn State University under the supervision of Dr. Adri van Duin. The original version was a wrapper code that called an external executable to perform the reaxff minimization step and energy calculation, and relied on file modification and parsing to steer the reaxff code and get the results back.

A rewrite of the code, made by Hans van Schoot (SCM) in close collaboration with Thomas Senftle, is now available in the ADF package. The rewrite directly integrates into the ADF-ReaxFF code, solving performance issues of the original code by removing the calling overhead of the reaxff executable and the relatively slow file management. It also merged several modifications of the original code to support the usage of whole molecules for Monte Carlo moves, and supports the usage of multiple atom/molecule types during the simulation. Other improvements were made on the input options, the accessible volume calculation, the MC acceptance prefactor calculation and the writing of logfiles.

The relevant papers are:

Thomas P. Senftle, Randall J. Meyer, Michael J. Janik and Adri C.T. van Duin, *Development of a ReaxFF potential for Pd/O and application to palladium oxide formation*, *J. Chem. Phys.* **139**, 044109 (2013)

Thomas P. Senftle, Adri C.T. van Duin, Michael J. Janik, *Determining in situ phases of a nanoparticle catalyst via grand canonical Monte Carlo simulations with the ReaxFF potential*, *Catalysis Communications* **volume 52**, 5 July 2014, Pages 72–77

Input

Overview

The GCMC code in ADF-ReaxFF needs the following input files to run:

- control_MC : The GCMC control file, which holds MC settings and the atoms/molecules to insert/move/delete
- control : A reaxff control file, in which only a small number of parameters is of interest
- ffield : A standard reaxff forcefield file
- geo : A geometry file, preferably in biograph format (code not yet tested with xyz)

- iopt : Text file that should only contain a "5" (without the quotes)
- insertData_MC : (optional file) Table used when restarting GCMC simulations

Also, the current version of the GCMC code can only run in serial, so please set the NSCM environment variable to 1 (type "export NSCM=1" (without quotes) in the shell before starting a GCMC reaxff run).

the control_MC file

Lines in the control_MC file that start with ! or # will be ignored, so those can be used for comments. Empty lines are also ignored, so feel free to leave some in the file. Lines with keywords should have their value in the first 8 columns, followed by a couple of spaces (at least 1), followed by the 6-character keyword. The order is free, except for the nmols keyword, which should be the last one. The nmols keyword signals the parser that the next section of control_MC will define X new MC Molecule Types.

This is an example for the control_MC file:

```
! GCMC control file example
  0  iensmb !select MC ensemble (0=Mu-NVT with fixed volume,
1=Mu-NPT with variable volume)
 5000 niter !number of MC iterations to do this simulation
  0  nstart !start the iteration counter with an offset, usefull for
restarts to avoid double files
300.0 mctemp !Temperature of the simulation, affects acceptance rate
for steps that increase the energy
  0.0 mcpres !NPT pressure in GPa (set to zero for incompressible
solid systems unless at very high pressures)
  3.0 rmaxpl !Max radius for atom placement on insert/displace move
  1.2 rminpl !Min radius for atom placement on insert/displace move
 2000 nmctry !Maximum number of trials allowed when inserting or
moving a molecule. If the
!
!          ! rmaxpl and rminpl variables are very strict, this
number needs to be large
  1  igcfac !include GC prefactor in probabilities? 0 = no 1 = yes
  0  ivol  !select MC volume calculation technique:
!
!          ! 0: vvacu needed! volume = total volume - occupied
volume - specified vacuum volume (vvacu)
!
!          ! 1:
!          !          volume = total cell volume
!
!          ! 2: vacc needed! volume = specified accessible volume
(vacc)
!
!          ! 3:
!          !          volume = total cell volume -
occupied volume
!
!          ! 4: vacc needed! volume = specified accessible volume
(vacc) - occupied volume
435.0 vacc  !if ivol=2 or ivol=4, specify Vacc in angrstoms^3
  0.0 vvacu !if ivol=0 specify non-accessible (vacuum) volume Vvacu
in angrstoms^3
  0.25 ivlim !volume change limit (value between 0 and 1,
Vnew = ((1+ivlim)*V1)
  1  resopt !write restart files: 0=no, 1=yes
  1  resfrq !frequency of writing restart files (MC code only writes
files if the MC move is accepted)
  0  debug !print debug output if set to 1, print even more debug
output when set to 2
  5  nmols !Number of MC molecule types, must match the number of
molecule blocks that follow!
!
! Molecule Specific Data: C2H2 example
! This part is fixed format!
```

```

! We need cmpot on line 1,
! possibly followed by the noinsr on line two,
! and forced to be ended with nmatom on line 2 or 3, followed by
nmatom lines of coordinates.
! the coordinates are FIXED FORMAT! (24d.15,1x,A2) x,y,z (24 wide, 15
after decimal), 1 space, 2chars symbol)
-75.00  cmpot  !chemical potential of molecule
      4  nmatom !number of atoms in molecule
      12.180480000000000      0.421696000000000      1.316689000000000
C
      13.124731000000000      0.376902000000000      0.568360000000000
C
      11.349475000000000      0.459560000000000      1.988208000000000
H
      13.957314000000000      0.335843000000000      -0.101000000000000
H

!Molecule Specific Data: C2H4 example
-75.00  cmpot  !chemical potential of molecule
      6  nmatom !number of atoms in molecule
      13.989222000000000      0.405391000000000      1.000150000000000
C
      13.316784000000000      0.399646000000000      0.885795000000000
C
      11.494513000000000      0.461837000000000      1.970612000000000
H
      11.335219000000000      0.353577000000000      0.129581000000000
H
      13.811701000000000      0.340224000000000      -0.084000000000000
H
      13.970561000000000      0.453325000000000      1.756236000000000
H

!Molecule Specific Data: H2O Example
-75.00  cmpot  !chemical potential of molecule
      1  noinsr !setting this to 1 disables insert/deletion moves. If
it is set to 1 for all types, the ensamble becomes NVT/NPT
      3  nmatom !number of atoms in molecule
      39.996720000000000      40.747660000000000      40.512210000000000
H
      40.000210000000000      39.999520000000000      39.934730000000000
O
      40.000030000000000      39.259880000000000      40.523700000000000
H

!Molecule Specific Data: H2 Example
-75.00  cmpot  !chemical potential of molecule
      2  nmatom !number of atoms in molecule
      5.025812000000000      0.000000000000000      0.000000000000000
H
      5.774188000000000      0.000000000000000      0.000000000000000
H

!Molecule Specific Data: Single atom example
-75.00  cmpot  !chemical potential of molecule
      1  nmatom !number of atoms in molecule

```

H	0.0000000000000000	0.0000000000000000	0.0000000000000000
---	--------------------	--------------------	--------------------

The Molecule Specific Data blocks define the molecules (or atoms) that can be inserted/moved/deleted with the MC code. The atoms named here should of course be in the forcefield files, and the coordinates should form a reasonable structure. The MC code uses these coordinates during the insertion step by giving them a random rotation, followed by a random translation to generate a random position of the molecule inside the box. Currently, there is no check to make sure the molecule stays inside the boundaries of the box, the code only checks that the rmaxpl/rminpl values are satisfied. If you plan on inserting large molecules, make sure there is enough room in the rmaxpl value, otherwise the code will stop with an error message.

The chemical potential (cmpot) keyword

The cmpot keyword sets the chemical potential of the molecule (or atom) reservoir, and is employed when calculating the Boltzmann accept/reject criteria after a MC move is executed. This value can be derived from first principles using statistical mechanics, or equivalently, it can be determined from thermochemical tables available in literature sources.

For example, the proper chemical potential for a GCMC simulation in which single oxygen atoms are exchanged with a reservoir of O2 gas, should equal 1/2 the chemical potential of O2 at the temperature and pressure of the reservoir:

$$\text{cmpot} = \mu_{\text{O}}(T,P) = 1/2 * \mu_{\text{O}_2}(T,P) = 1/2 * [\mu_{\text{ref}}(T,P_{\text{ref}}) + kT * \log(P/P_{\text{ref}}) - E_{\text{diss}}]$$

where the reference chemical potential [$\mu_{\text{ref}}(T,P_{\text{ref}})$] is the experimentally determined chemical potential of O2 at T and Pref, $kT * \log(P/P_{\text{ref}})$ is the pressure correction to the free energy, and E_{diss} is the dissociation energy of the O2 molecule.

The no insert (noinstr) keyword

The noinstr keyword tells the GCMC code to keep the number of molecules/atoms of this type fixed. It will thus disable Insert/Delete moves on this type, meaning it can only do a displacement move, or volume change move (if the iensmb keyword is set to 1).

the control file

The control file is a regular reaxff control file and it influences the minimization step after an MC trial move. Because of this, only a small number of the reaxff keywords are used during the GCMC simulation.

An example of the control file:

```
# some of the parameters that influence the minimization step in the GCMC
code
    1 icentr      Put the center of mass at the center of the cube
    1 igeofo      0:xyz-input geometry 1: Biograf input geometry 2:
xmol-input geometry
2.50000 endmm    End point criterium for MM energy minimisation
    500 imaxit    Maximum number of iterations
    0 icelop      Optimize cell parameters 0=no 1=yes
1.00050 celopt   Cell parameter change
    0 imaxmo      In this case: 0: POLAK_RIBIERE Conj.Grad method, 1:
Limited-memory BFGS method
```

The code has been tested with various imaxit and endmm values, the other options have not been fully tested. Other reaxff keywords might also influence the minimization procedure, but those are best left to their default settings

theffield file

The ffield file should be a normal reaxff forcefield file, as described in the reaxff documentation by A. van Duin (visit the documentation section on the SCM website to obtain this document).

the geo file

The GCMC code has been tested with biograph input files, but other input formats might work. The details of this file are also described in the original reaxff documentation by van Duin.

the iopt file

The iopt file is a text file with a single digit inside that selects the execution mode of the reaxff code. To run the GCMC code, this file should contain a "5" (without the quotes).

the insertData_MC file

The GCMC code can insert multiple atom/molecule types in a single simulation, so it needs to keep track of what atom belongs to which insert. This information is automatically stored and updated when insertion/deletion/moving of atoms or molecules during the simulation, but is by default unknown of the atoms of the starting geometry. The GCMC code will therefore by default not modify the atoms in the original input in the MC trial moves (keep in mind that they can move around during the minimization step). The insertData_MC file can be used to tell the GCMC code what atoms in the geo file belong to which molecule.

An example of the insertData_MC file:

#	atomNumber	MCInsMolType	MCInsertNmbr
	30	1	1
	40	2	1
	46	2	1
	47	1	2
	48	1	3

This example specifies 4 molecules/atoms that are modifiable by the GCMC code, belonging to 2 different GCMC molecules/atoms that are defined in the control_MC file. The first "molecule" in the control_MC file should thus consist of a single atom (if this doesn't match, the code will most likely crash!). It was inserted three times (atom 30, 47 and 48) The second molecule has two atoms, and was inserted once.

The atoms do not have a fixed order, and not all atoms have to be defined. If an atom is not appointed to a certain MCInsMolType and MCInsertNmbr, it will simply not be modified during the MC moves. The insertData_MCXXXXXX files generated by the restart option of the code can be directly used as valid insertData_MC files, just remove the digits from the filename and replace the geo file with the corresponding geo_MCXXXXXX file.

Output

Overview

The GCMC code writes a couple of output files, each described in this section. It also produces a number of reaxff output files, and some of these are described in the original reaxff documentation by van Duin. Keep in mind that these files might not provide a complete or correct picture of the simulation, as they could also contain data originating from rejected MC trial moves.

geo_MCXXXXXX

This file is generated every X accepted MC moves and contains the current geometry of the system in biograph format (X is set with the resfrq keyword in the control_MC file).

insertData_MCXXXXXX

This file contains a table of all the atoms in the system with their MC Molecule Type and MC Insert Number. This data can be used to map atoms to an inserted molecule, and is needed if you want to restart your calculation from an accepted MC step. The table contains -1 values for atoms that were in the original input and did not get a manually assigned MCInsert Molecule Type and MC Insert Number, the GCMC code will not modify these atoms during the MC steps.

Also see the section on insertData_MC file.

MCstats

The MCstats file is a logfile that contains the statistics of the MC simulation. The GCMC code writes a single line to it after every MC step, containing the number of: Tried MC moves (tried), Accepted MC moves (accept), Rejected MC moves (reject), Accepted Insertion/Deletion/Moving/Volume change MC moves (addAcc/delAcc/mvAcc/volAcc), Rejected Insertion/Deletion/Moving/Volume change MC moves (addRej/delRej/mvRej/volRej)

An example of the MCstats file:

	tried	accept	reject	addAcc	delAcc	mvAcc	volAcc	addRej	delRej
mvRej	0	1	0	1	0	0	0	0	
volRej	0	0	0						
0	1	1	1	1	0	0	0	0	
0	1	0							
0	2	1	2	1	0	0	0	0	
0	2	0							
1	3	1	3	1	0	0	0	0	
1	2	0							
1	4	1	4	1	0	0	0	1	
1	2	0							
1	5	1	5	1	0	0	0	1	
1	3	0							
2	6	1	6	1	0	0	0	1	
2	3	0							
3	7	1	7	1	0	0	0	1	
3	3	0							
3	8	1	8	1	0	0	0	1	
3	4	0							
4	9	1	9	1	0	0	0	1	
4	4	0							
4	10	1	10	1	0	0	0	1	
4	5	0							
4	11	2	10	2	0	0	0	1	
4	5	0							
4	12	3	10	3	0	0	0	1	
4	5	0							

Elog

The Elog file contains the Volume and energies of the accepted MC steps. The energies in this logfile are the pure ReaxFF energy of the system (RxFFEnergy) and the MC corrected energy, which is used in determining if the step should be accepted or not (see the section on calculating energies for details).

An example of the Elog file:

Iteration	Naccepted	Volume	MC Energy	RxFFEnergy
0	1	15625.00	-3098.88	-3179.88
2	2	15625.00	-3107.92	-3269.92

3	3	15625.00	-3130.13	-3373.13
4	4	15625.00	-3160.05	-3484.05
6	5	15625.00	-3169.77	-3493.77
11	6	15625.00	-3200.13	-3605.13

reaxout.kf

This is the binary logfile generated by the GCMC code. Its contents can be viewed with the KFBrowser utility, or it can be loaded into ADFMovie to view the geometries in the file. Only the data of successful MC moves is written to this file.

Code Details

Overview

The GCMC code will perform niter (control_MC file option) Grand Canonical Monte Carlo trial moves, and accept or reject them based on the Energy produced by the ReaxFF minimization step of the trial geometry. The Monte Carlo algorithm will always accept a step if it results in a decrease of the energy, and accept steps that go up in energy with a probability. This section will give some details about how the code works.

MC Moves (Insert/Delete/Move/Volume)

The GCMC code currently supports 4 types of MC Moves: Insert, Delete, Move (displace), Volume change. The first three moves always change a whole "molecule" of the system, as defined in the control_MC file (a molecule can of course contain only a single atom). Every MC iteration selects one MC Molecule Type from the defined molecules in control_MC at random, followed by a random MC Move (unless there are no molecules of the type in the system, in that case it will do an insert move).

The Insert and Displace (move) MC Moves will generate a random rotation and position for the molecule, and then check if the random positions are within the "RminPI" and "RmaxPI" boundaries (this means no atom in the molecule can be closer to any atom currently in the system than "RminPI", and it should be within "RmaxPI" distance to an atom in the system). If the conditions are not satisfied, a new set of coordinates is generated and the code checks again. This is repeated a maximum number of "nmctry" times before stopping with an error.

The volume change is controlled by the "ivlim" variable in control_MC. The ivlim sets the volume change limit, and it should be a value between 0 and 1. The new volume will be calculated like this: $V_{new} = (1 + ivlim) * V_{old}$.

Calculating energies

Because the GCMC simulation adds and deletes atoms or molecules during the runtime, it cannot directly compare the ReaxFF energies for the MC acceptance criteria: inserting a molecule will usually lower the total energy of the system, causing the MC to always accept it, and always reject a deletion. To balance this out, the GCMC code calculates a "corrected" MC energy to compare the trial reaxFF energy with, consisting of the previously accepted ReaxFF energy + the chemical potential (cmpot in control)MC) for the inserted molecule, or the ReaxFF energy - the chemical potential for the deleted molecule. The volume change energy is also corrected, using the following formula:

$$E_{MC_Corr} = E_{reaxff_last_accept} - (Pressure * 0.1439 * (newV - oldV)) + ((1.0/beta) * n_{InsertedMols} * \log(newV_{avail}/oldV_{avail}))$$

where newVavail and oldVavail are calculated from the MC available volume (see the section calculating volumes).

Calculating volumes

The GCMC code can calculate the available volume in a couple of different ways, depending on the `ivol` setting in `control_MC`:

- `ivol = 0`: volume = total volume - occupied volume - specified vacuum volume (`vvacu`)
- `ivol = 1`: volume = total cell volume
- `ivol = 2`: volume = specified accessible volume (`vacc`)
- `ivol = 3`: volume = total cell volume - occupied volume
- `ivol = 4`: volume = specified accessible volume (`vacc`) - occupied volume

Where the occupied volume is calculated by summing up the volumes of the atoms in the `geo` file that are not specified to be part of an MC type molecule. The volume of an atom is calculated using the average of the covalent atomic radius and the vd Waals radius of the atom, which are found in the `reaxff` forcefield file (`ffield`).

the `vacc` and `vvacu` options can be specified in the `control_MC` file to get a more accurate available volume.

Acceptance criteria

An MC move is always accepted if the `reaxff` energy is lower than the corrected MC energy of the last accepted MC move, or if the energy increase is small enough. If the new energy is higher, the code generates a random number between 0 and 1, and accepts the move if the random number is bigger than:

$$| \text{prob} = \text{preFactor} * \exp(-\text{Beta} * \text{deltaE})$$

Where the `prefactor` is calculated (for insert and delete moves) using the deBroglie wavelength of the inserted molecules, the number of inserted molecules and the available MC volume of the system.

Included Forcefields

Description of ReaxFF force fields

Disclaimer: Using these force fields for systems they have not been explicitly trained against may produce unrealistic results. Please see the full manuscripts for more detailed information.

The force field files used by the SCM version of ReaxFF are compatible with those used by the original ReaxFF code. So if you have the force field information from somewhere else you can just use it (save it in a text file, and select it in ADFinput via the Other... option).

AB.ff: (H/O/N/B) Ammonia Borane

-
- M.R. Weismiller, A.C.T. van Duin, J. Lee, R.A. Yetter
ReaxFF Reactive Force Field Development and Applications for Molecular Dynamics Simulations of Ammonia Borane Dehydrogenation and Combustion
J. Phys. Chem. A 2010, 114, 5485-5492
<http://dx.doi.org/10.1021/jp100136c>
- QM data were generated describing the single and (if relevant) double and triple bond dissociation for all B/N/O/H combinations. These data were used to derive initial ReaxFF bond parameters, and all calculations were performed using DFT with the B3LYP functional and the Pople 6-311G** basis set.
 - The training set was then extended with QM data describing angular distortions in a set of small AB-related (AB = H3N-BH3) molecules. These data were used to derive the initial ReaxFF angular parameters.
 - The training set was extended with reaction barriers for key reaction steps such as H2 release from AB, dimerization of H2B-NH2 and reaction energies associated with H2 release from AB and with AB oxidation.

AuCSOH.ff: (Au/C/S/O/H)

-
- J. A. Keith, D. Fantauzzi, T. Jacob, and A.C.T. van Duin
Reactive forcefield for simulating gold surfaces and nanoparticles
Physical Review B (2010) 81, 235404-1/235404-8
<http://dx.doi.org/10.1103/PhysRevB.81.235404>
- The original Au-Au parameters were extended by three publications:
 - Au/O: K. Joshi, A.C.T. van Duin, and T. Jacob
Development of a ReaxFF description of gold oxides and initial application to cold welding of partially oxidized gold surfaces
Journal of Materials Chemistry 20, (2010), 10431-10437
<http://dx.doi.org/10.1039/C0JM01556C>
 - Au/C/S/H: T.T. Jarvi, A.C.T. van Duin, K. Nordlund, and W.A. Goddard
Development of interatomic ReaxFF potentials for Au-S-C-H systems
Journal of Physical Chemistry C 115, (2011), 10315-10322
<http://dx.doi.org/10.1021/jp201496x>
 - C/O/H/S: Rahaman, O., van Duin, A. C. T., Goddard, W. A., III, and Doren, D. J.
Development of a ReaxFF reactive force field for glycine and application to solvent effect and tautomerization
Journal of Physical Chemistry B 115 (2011), 249-261
<http://dx.doi.org/10.1021/jp108642r>
 - The forcefield does not include Au/N parameters

CHO.ff: (C/H/O) Hydrocarbon oxidation

-
- K. Chenoweth, A.C.T. van Duin, W. A. Goddard
ReaxFF Reactive Force Field for Molecular Dynamics Simulations of Hydrocarbon Oxidation
J. Phys. Chem. A 2008, 112, 1040-1053
<http://dx.doi.org/10.1021/jp709896w>
- To obtain the H/C/O compound data required to extend the hydrocarbon-training set, DFT calculations were performed on the following systems: (a) dissociation energies for various bonds containing carbon, oxygen, and hydrogen. The ground state structure was obtained through full geometry optimization. Dissociation curves were calculated by constraining only the bond length of interest and re-optimization of the remaining internal coordinates. Optimization was also performed for the various angles and torsions associated with C/H/O interactions.

HCONSB.ff: (H/C/O/N/S/B)

-
- M.R. Weismiller, A.C.T. van Duin, J. Lee, and R.A. Yetter,
ReaxFF Reactive Force Field Development and Applications for Molecular Dynamics Simulations of Ammonia Borane Dehydrogenation and Combustion.
J. Phys. Chem. A (2010), 114, 5485-5492.
<http://dx.doi.org/10.1021/jp100136c>
- The parameters in this forcefield were extended/improved by two other publications:
 - A. M. Kamat, A.C.T. van Duin, and A. Yakovlev
Molecular Dynamics Simulations of Laser-Induced Incandescence of Soot Using an Extended ReaxFF Reactive Force Field.
Journal of Physical Chemistry A (2010), 114, 12561-1257
<http://dx.doi.org/10.1021/jp1080302>
 - F. Castro-Marciano, A.M. Kamat, M.F. Russo, A.C.T. van Duin, and J.P. Mathews
Combustion of an Illinois No. 6 Coal Char Simulated Using an Atomistic Char Representation and the ReaxFF Reactive Force Field.

- Combustion and Flame (2012), 159, 23273-1285
<http://dx.doi.org/10.1016/j.combustflame.2011.10.022>
- The C/H/O parameters are the same as in the CHO forcefield, with added S/C, S/H and S/O descriptions. This force field was used in Castro et al, Combustion and Flame 2011
 - The Boron and Nitrogen parameters are based on (but not identical to) the parameters used in Weismiller et al, JPC-A 2010.

CuCl-H2O.ff: (Cu/Cl/H/O)

- O. Rahaman, A.C.T. van Duin, V.S. Bryantsev, J.E. Mueller, S.D. Solares, W.A. Goddard III, and D.J. Doren
Development of a ReaxFF Reactive Force Field for Aqueous Chloride and Copper Chloride
J. Phys. Chem. A 114 (2010), 3556-3568
<http://dx.doi.org/10.1021/jp9090415>
- This forcefield is an extension of:
A.C.T. van Duin, V.S. Bryantsev, M.S. Diallo, W.A. Goddard, O. Rahaman, D.J. Doren, D. Raymand, and K. Hermansson
Development and validation of a ReaxFF reactive force field for Cu cation/water interactions and copper metal/metal oxide/metal hydroxide condensed phases
Journal of Physical Chemistry A 2010, 114, 9507-9514
<http://dx.doi.org/10.1021/jp102272z>

FeOCHCl.ff: (Fe/O/C/H/Cl)

- M. Aryanpour, A.C.T. van Duin, J.D. Kubicki
Development of a Reactive Force Field for Iron-Oxyhydroxide Systems
J. Phys. Chem. A 2010, 114, 6298-6307
<http://dx.doi.org/10.1021/jp101332k>
- The Cl parameters were published by:
O. Rahaman, A.C.T. van Duin, V.S. Bryantsev, J.E. Mueller, S.D. Solares, W.A. Goddard III, and D.J. Doren
Development of a ReaxFF Reactive Force Field for Aqueous Chloride and Copper Chloride
J. Phys. Chem. A 114 (2010), 3556-3568
<http://dx.doi.org/10.1021/jp9090415>
 - The initial force field parameters for the Fe-Fe parameters were taken from an earlier force field development project on bulk-iron metal, based on DFT-calculations on antiferromagnetic BCC and FCC. The ReaxFF parameters have not been published yet, however the DFT data can be found in ref 31 of the above mentioned manuscript. The O/H parameters were taken from the ReaxFF bulk water description. The Fe/Fe and O/H parameters were kept fixed to these initial values, whereas the Fe/O parameters were reoptimized against the quantum mechanical results presented in the above mentioned manuscript.
 - Detailed information on the force field parameters is given in the supporting information of the above mentioned manuscript.

HE.ff: (C/H/O/N) RDX/High Energy

- L.Z. Zhang, A.C.T. van Duin, S.V. Zybin, and W.A. Goddard
Thermal Decomposition of Hydrazines from Reactive Dynamics Using the ReaxFF Reactive Force Field
Journal of Physical Chemistry B (2009) 113, 10770-10778
<http://dx.doi.org/10.1021/jp900194d>
- Part of this forcefield is also published in:
L.Z. Zhang, S.V. Zybin, A.C.T. van Duin, S. Dasgupta, W.A. Goddard, and E.M. Kober
Carbon Cluster Formation during Thermal Decomposition of Octahydro-1,3,5,7-tetranitro-1,3,5,7-tetrazocine and 1,3,5-Triamino-2,4,6-trinitrobenzene
High Explosives from ReaxFF Reactive Molecular Dynamics Simulations
Journal of Physical Chemistry A (2009) 113, 10619-10640
<http://dx.doi.org/10.1021/jp901353a>
 - The parameters of the nitramine ReaxFF are based on a large number of ab initio QM calculations. Over 40 reactions and over 1600 equilibrated molecules have been used; they are designed to characterize the atomic interactions under various environments likely and unlikely high energy each atom can encounter. The training set contains bond breaking and compression curves for all possible bonds, angle and torsion bending data for all possible cases, as well as crystal data.
 - Please see the supplemental material from Phys. Rev. Lett. 2003, 91, 098301 for a detailed description of the parameterization of this force field.

NaH.ff: (Na/H)

- J.G.O. Ojwang, R. Van Santen, G.J. Kramer, A.C.T. van Duin, and W.A. Goddard
Modeling the sorption dynamics of NaH using a reactive force field
Journal of Chemical Physics 2008, 128, 164714
<http://dx.doi.org/10.1063/1.2908737>

NiCH.ff: (Ni/C/H/O/N/S/F/Pt/Cl)

- J.E. Mueller, A.C.T. van Duin, and W.A. Goddard III
Development and Validation of ReaxFF Reactive Force Field for Hydrocarbon Chemistry Catalyzed by Nickel
J. Phys. Chem. C 2010, 114, 4939-4949
<http://dx.doi.org/10.1021/jp9035056>

SiOH.ff: (Si/O/H)

- J.C. Fogarty, H.M. Aktulga, A.Y. Grama, A.C.T. van Duin, S.A. Pandit
A reactive molecular dynamics simulation of the silica-water interface
J. Chem. Phys. 2010, 132, 174704
<http://dx.doi.org/10.1063/1.3407433>
- This force field was trained to model the interaction of water at the SiO₂ surface, with specific emphasis on proton-transfer reactions. Updated parameters were fitted for all Si/O/H bond, angle, and torsion interactions as well, in addition to the dissociation of a water molecule from a single Si(OH)₄ molecule and reaction energies for the polymerization of Si(OH)₄

SiC.ff: (Si/C/O/H/N/S)

- D. Newsome, D. Sengupta, H. Foroutan, M.F. Russo, and A.C.T. van Duin
Oxidation of Silicon Carbide by O₂ and H₂O: A ReaxFF Reactive Molecular Dynamics Study, Part I
Journal of Physical Chemistry 2012 116, 16111-16121
<http://dx.doi.org/10.1021/jp306391p>
- The included forcefield is based on the Newsome reference, with slightly improved parameters by van Duin et al.

VOCH.ff: (V/O/C/H)

- K. Chenoweth, A.C.T. van Duin, P. Persson, M.J. Cheng, J. Oxgaard, W.A. Goddard
Development and Application of a ReaxFF Reactive Force Field for Oxidative Dehydrogenation on Vanadium Oxide Catalysts
J. Phys. Chem. C, 2008, 112, 14645-14654
<http://dx.doi.org/10.1021/jp802134x>
- The ReaxFF force field parameters have been fit to a large quantum mechanics (QM) training set containing over 700 structures and energetics related to bond dissociations, angle and dihedral distortions, and reactions between hydrocarbons and vanadium oxide clusters.
In addition, the training set contains charge distributions for small vanadium oxide clusters and the stabilities of condensed-phase systems including V₂O₅, VO₂, and V₂O₃ in addition to metallic V (V₀).

ZnOH.ff: (Zn/O/H)

- D. Raymand, A.C.T. van Duin, M. Baudin, K. Hermansson
A reactive force field (ReaxFF) for zinc oxide
Surface Science 2008, 602, 1020-1031
<http://dx.doi.org/10.1016/j.susc.2007.12.023>
- updated version published by:
D. Raymand, A.C.T. van Duin, D. Spangberg, W.A. Goddard, K. Hermansson
Water adsorption on stepped ZnO surfaces from MD simulation
Surface Science 2010, 604, 9-10, 741-752
<http://dx.doi.org/10.1016/j.susc.2009.12.012>
- Based on QM calculations for Zn(s), ZnO(s), and Zn hydroxide clusters [Zn(OH)₂ and O(ZnOH)₂], ReaxFF parameters were generated for Zn-O and Zn-Zn bond energies and for Zn-O-Zn, O-Zn-O, O-Zn-Zn and Zn-O-H valence angle energies.
- QM calculations were performed for the four crystal polymorphs of the wurtzite, zincblende, rocksalt and caesium chloride structures (the structures are also referred to as h-ZnS, c-ZnS, NaCl and CsCl, respectively).

Al-H₂O.ff: (Al/H/O)

- M. Russo, R. Li, M. Mench, and A.C.T. van Duin
Molecular Dynamic Simulation of Aluminum-Water Reactions Using the ReaxFF Reactive Force Field
International Journal of Hydrogen Energy 36 (2011), 5828-5835
<http://dx.doi.org/10.1016/j.ijhydene.2011.02.035>

CaSiAlO.ff: (C/H/O/Fe/Cl/Si/Al/Ca)

- M.C. Pitman and A.C.T. van Duin
Dynamics of Confined Reactive Water in Smectite Clay-Zeolite Composites
J. Am. Chem. Soc., 2012, 134 (6), 3042-3053
<http://dx.doi.org/10.1021/ja208894m>

dispersion/CHONSSi-lg.ff: (C/H/O/N/S/Si)

- L. Liu, Y. Liu, S.V. Zybin, H. Sun, and W.A. Goddard III
ReaxFF-lg: Correction of the ReaxFF Reactive Force Field for London Dispersion, with Applications to the Equations of State for Energetic Materials
The Journal of Physical Chemistry A, 2011, 115(40), 11016-11022
<http://dx.doi.org/10.1021/jp201599t>
- This forcefield adds London dispersion correction terms to reaxFF, and is optimized for the energetic materials RDX, PETN, TATB, and NM plus graphite, polyethylene, solid carbon dioxide, and solid N₂, using the low temperature crystal structures to determine the lg correction parameters.

CHOFeAlNiCuS.ff: (C/H/O/Fe/Al/Ni/Cu/S)

-
- O. Rahaman, A.C.T. van Duin, W.A. Goddard III, and D.J. Doren
Development of a ReaxFF reactive force field for glycine and application to solvent effect and tautomerization.
Journal of Physical Chemistry B 115 (2011), 249-261.
<http://dx.doi.org/10.1021/jp204894m>
- C/O/H parameters only. The Cu/Fe/Al/Ni parameters are from:
Y.K. Shin, H. Kwak, C. Zou, A.V. Vasenkov, and A.C.T. van Duin
Development and Validation of a ReaxFF Reactive Force Field for Fe/Al/Ni Alloys:
Molecular Dynamics Study of Elastic Constants, Diffusion, and Segregation
J. Phys. Chem. A, 2012, 116 (49), pp 12163--12174
<http://dx.doi.org/10.1021/jp308507x>
 - Not all cross-terms between the two forcefield files are defined, which might cause problems if the system has (for example) C-Cu interactions.

AuSCH_2011.ff: (Au/S/C/H)

-
- T.T. Jarvi, A.C.T. van Duin, K. Nordlund, and W.A. Goddard III,
Development of Interatomic ReaxFF Potentials for Au-S-C-H Systems
J. Phys. Chem. A, 115, 10315-10322 (2011)
<http://dx.doi.org/10.1021/jp201496x>

AuSCH_2013.ff: (Au/S/C/H)

-
- Gyun-Tack Bae and Christine M. Aikens
Improved ReaxFF Force Field Parameters for Au-S-C-H Systems
Journal of Physical Chemistry A 2013 117 (40), 10438-10446
<http://dx.doi.org/10.1021/jp405992m>
- Based upon: T.T. Jarvi, A.C.T. van Duin, K. Nordlund, and W.A. Goddard
Development of interatomic ReaxFF potentials for Au-S-C-H systems.
Journal of Physical Chemistry C 115, (2011), 10315-10322.
 - yields improvements for bond bending potential energy surfaces
 - aims to agree with DFT geometries of small clusters and gold-thiolate nanoparticles

PDMSDecomp.ff: (C/H/O/Si)

-
- K. Chenoweth, S. Cheung, A.C.T. van Duin, W.A. Goddard III, and E.M. Kober
Simulations on the Thermal Decomposition of a Poly(dimethylsiloxane) Polymer
Using the ReaxFF Reactive Force Field
J. Am. Chem. Soc., 2005, 127 (19), pp 7192-7202
<http://dx.doi.org/10.1021/ja050980t>
- Specialized forcefield, designed to "investigate the failure of the poly(dimethylsiloxane) polymer (PDMS) at high temperatures and pressures and in the presence of various additives"

TiOCHNCl.ff: (C/H/O/N/S/Mg/P/Na/Ti/Cl/F)

-
- S.Y. Kim, A.C.T. van Duin, and J.D. Kubicki
Molecular dynamics simulations of the interactions between TiO2 nanoparticles and water with Na+ and Cl-, methanol, and formic acid using a reactive force field
Journal of Materials Research / Volume / Issue 03 / 2013, pp 513-520
<http://dx.doi.org/10.1557/jmr.2012.367>
- used for simulating TiO2 (both rutile and anatase) nanoparticles with water, methanol, and formic acid
 - The force field was validated by comparing water dissociative adsorption percentage and bond length between Na-O with density functional theory (DFT) and experimental results

PtCH.ff: (C/H/Pt)

-
- C.F. Sanz-Navarro, P. Astrand, De Chen, M. Ronning, A.C.T. van Duin, T. Jacob, and W.A. Goddard III
Molecular Dynamics Simulations of the Interactions between Platinum Clusters and Carbon Platelets
J. Phys. Chem. A 112, 1392-1402 (2008)
<http://dx.doi.org/10.1021/jp074806y>

BaYZrCHO.ff: (C/H/O/Ba/Zr/Y)

-
- A.C.T. van Duin, B.V. Merinov, S.S. Jang, and W.A. Goddard III
ReaxFF Reactive Force Field for Solid Oxide Fuel Cell Systems with Application to Oxygen Ion Transport in Yttria-Stabilized Zirconia
J. Phys. Chem. A, 112, 3133-3140 (2008)
<http://dx.doi.org/10.1021/jp076775c>

CHONSSiPtZrNiCuCo.ff: (C/H/O/N/S/Si/Pt/Zr/Ni/Cu/Co)

-
- K.D. Nielson, A.C.T. van Duin, J. Oxgaard, W.Q. Deng, and W.A. Goddard III
Development of the ReaxFF Reactive Force Field for Describing Transition Metal Catalyzed Reactions, with Application to the Initial Stages of the Catalytic Formation of

Carbon Nanotubes
J. Phys. Chem. A, 109, 493-499 (2005)
<http://dx.doi.org/10.1021/jp046244d>

Glycine.ff: (C/H/O/N)

O. Rahaman, A.C.T. van Duin, W.A. Goddard III, and D.J. Doren,
Development of a ReaxFF Reactive Force Field for Glycine and
Application to Solvent Effect and Tautomerization
J. Phys. Chem. B, 115, 249-261 (2011)
<http://dx.doi.org/10.1021/jp108642r>

SiONH.ff: (C/H/O/N/Si/S)

A.D. Kulkarni, D.G. Truhlar, S.G. Srinivasan, A.C.T. van Duin, P. Norman,
and T.E. Schwartzentruber
Oxygen Interactions with Silica Surfaces: Coupled Cluster and Density Functional
Investigation and the Development of a New ReaxFF Potential
J. Phys. Chem. C, 2013, 117 (1), pp 258-269
<http://dx.doi.org/10.1021/jp3086649>
- Aimed at oxygen interactions with realistic silica surfaces

CHONPt.ff: (C/H/O/N/Pt)

M.F. Russo, D. Bedrov, S. Singhai, A.C.T. van Duin
Combustion of 1,5-Dinitrobiuret (DNB) in the Presence of Nitric Acid Using
ReaxFF Molecular Dynamics Simulations
J. Phys. Chem. A, 2013, 117 (38), pp 9216-9223
<http://dx.doi.org/10.1021/jp403511q>
- specialized combustion forcefield for 1,5-dinitrobiuret (DNB) and nitric acid

CHOFe.ff: (C/H/O/Fe/Cl/Si/Al)

Chenyu Zou, A.C.T. Van Duin
Investigation of Complex Iron Surface Catalytic Chemistry Using the ReaxFF Reactive
Force Field Method
JOM, December 2012, Volume 64, Issue 12, pp 1426-1437
<http://dx.doi.org/10.1007/s11837-012-0463-5>
- only the parameters for Fe (and crossterms) differ from the CHOAlSi.ff forcefield

CHOAlSi.ff: (C/H/O/Fe/Cl/Si/Al)

F. Castro-Marcanao, A.C.T. van Duin
Comparison of thermal and catalytic cracking of 1-heptene from ReaxFF reactive
molecular dynamics simulations
Combustion and Flame, Volume 160, Issue 4, April 2013, Pages 766-775
<http://dx.doi.org/10.1016/j.combustflame.2012.12.007>
- only the parameters for Fe (and crossterms) differ from the CHOFe.ff forcefield

CHOLi.ff: (C/H/O/N/S/Mg/P/Na/Li)

D. Bedrov, G.D. Smith, A.C.T. van Duin
Reactions of Singly-Reduced Ethylene Carbonate in Lithium Battery Electrolytes: A
Molecular Dynamics Simulation Study Using the ReaxFF
Journal of Physical Chemistry A, 2012, 116 (11), pp 2978-2985
<http://dx.doi.org/10.1021/jp210345b>
- specifically generated for simulating Lithium battery electrolytes
- must be used in combination with the MOLCHARGE keyword to set a charge restraint on
Li and CO3!

SiOAlLi.ff: (H/O/Si/Al/Li)

B. Narayanan, A.C.T. van Duin, B.B. Kappes, I.E. Reimanis and C.V. Ciobanu
A reactive force field for lithium-aluminum silicates with applications to
eucryptite phases
Modelling and Simulation in Materials Science and Engineering 2012 20 015002
<http://dx.doi.org/10.1088/0965-0393/20/1/015002>

PdO.ff: (Pd/O)

T.P. Senftle, R.J. Meyer, M.J. Janik and A.C.T. van Duin
Development of a ReaxFF potential for Pd/O and application to palladium oxide formation
The Journal of Chemical Physics 139, 044109 (2013)
<http://dx.doi.org/10.1063/1.4815820>
- used for studying Oxidation states of Pd nanoparticles, surfaces and bulk configurations
with a GCMC method

PdH.ff: (Pd/H)

T.P. Senftle, M.J. Janik and A.C.T. van Duin

A ReaxFF Investigation of Hydride Formation in Palladium Nanoclusters via Monte Carlo and Molecular Dynamics Simulations
The Journal of Physical Chemistry C, 2014, 118 (9), pp 4967-4981
<http://dx.doi.org/10.1021/jp411015a>
- used in combination with a GCMC method

Co.ff: (Co)

Xue-Qing Zhang, E. Iype, S.V. Nedea, A.P.J. Jansen, B.M. Szyja, E.J.M. Hensen, and R.A. van Santen
Site Stability on Cobalt Nanoparticles: A Molecular Dynamics ReaxFF Reactive Force Field Study
The Journal of Physical Chemistry C, 2014, 118 (13), pp 6882-6886
<http://dx.doi.org/10.1021/jp500053u>
- forcefield was generated using a recently developed Monte Carlo algorithm with simulated annealing.

CHONSMgPNaCuCl.ff: (C/H/O/N/S/Mg/P/Na/Cu/Cl)

Susanna Monti, Cui Li, and Vincenzo Caravetta
Reactive Dynamics Simulation of Monolayer and Multilayer Adsorption of Glycine on Cu(110)
J. Phys. Chem. C, 2013, 117 (10), pp 5221-5228
<http://dx.doi.org/10.1021/jp312828d>
- Reactive MD-force field for amino acids on copper

CHOSMoNiLiBFPN.ff: (C/H/O/S/Mo/Ni/Li/B/F/P/N)

Md M. Islama, V.S. Bryantsevb, A.C.T. van Duin
ReaxFF Reactive Force Field Simulations on the Influence of Teflon on Electrolyte Decomposition during Li/SWCNT Anode Discharge in Lithium-Sulfur Batteries
J. Electrochem. Soc. 2014 volume 161, issue 8, E3009-E3014
<http://dx.doi.org/10.1149/2.005408jes>
- forcefield for Electrochemistry in Li-S batteries

CHONSSiNaFZr.ff: (C/H/O/N/S/Si/Na/F/Zr)

A. Rahnamoun and A.C.T. van Duin
Reactive Molecular Dynamics Simulation on the Disintegration of Kapton, POSS Polyimide, Amorphous Silica, and Teflon during Atomic Oxygen Impact Using the Reaxff Reactive Force-Field Method
J. Phys. Chem. A, 2014, 118 (15), pp 2780-2787
<http://dx.doi.org/10.1021/jp4121029>
- comments in the forcefield file: interactions with water and Na+ Fogarty et al. JCP-2010 ; with glycine + C/H/F parameters; Si-F bond/offdiag/angle parameters; Si-S dummy parameters + S-O-H parameters(Yun 2012 Oct8) + H-F bond/offdiag (Jan14 2013 Joon) Jan31: added Zr/O/H/C

TiClOH.ff: (C/H/O/N/S/Mg/P/Na/Ti/Cl/F)

Sung-Yup Kim and A.C.T. van Duin
Simulation of Titanium Metal/Titanium Dioxide Etching with Chlorine and Hydrogen Chloride Gases Using the ReaxFF Reactive Force Field
J. Phys. Chem. A, 2013, 117 (27), pp 5655-5663
<http://dx.doi.org/10.1021/jp4031943>
- adaptation/evolution of the TiOCHNCl.ff forcefield by Kim, S.-Y. et al.

See also

- [Included Forcefields \(development version\)](#)