

ReaxFF

Contents

- 1 ReaxFF potential functions
 - 1.1 0. Naming
 - 1.2 1. Overall system energy
 - 1.3 2. Bond Order and Bond Energy
 - 1.3.1 2.1 Uncorrected BO
 - 1.3.2 2.2 Corrected BO
 - 1.3.3 2.3 Fe dimer correction
 - 1.4 3. Lone pair energy
 - 1.5 4. Overcoordination
 - 1.6 5. Undercoordination
 - 1.7 6. Valence Angle Terms
 - 1.7.1 6.1 Angle energy.
 - 1.7.2 6.2 Penalty energy.
 - 1.7.3 6.3 Three-body conjugation term.
 - 1.8 7. Torsion angle terms
 - 1.8.1 7.1 Torsion rotation barriers.
 - 1.8.2 7.2 Four body conjugation term.
 - 1.9 8. Hydrogen bond interactions
 - 1.10 9. Correction for
 - 1.11 10. Triple bond energy correction.
 - 1.12 11. Nonbonded interactions
 - 1.12.1 11.1 Taper correction
 - 1.12.2 11.2 van der Waals interactions
 - 1.12.3 11.3 lg - dispersion correction
 - 1.12.4 11.4 Energy due to charges (includes Coulomb)
 - 1.12.4.1 11.4.1 EEM
 - 1.12.4.2 11.4.2 ACKS2
 - 1.12.4.3 11.4.3 Remark on the "Coulomb" and "Charge" energy
 - 1.13 12. eReaxFF

ReaxFF potential functions

Based on the **Supporting information** for the manuscript "*A ReaxFF Reactive Force Field for Molecular Dynamics Simulations of Hydrocarbon Oxidation*" by Kimberly Chenoweth, Adri C.T. van Duin and William A. Goddard, III

This document contains all the general ReaxFF-potential functions. In the current ReaxFF code all the energy contributions in this document are calculated regardless of system composition. All parameters that do not bear a direct physical meaning are named after the partial energy contribution that they appear in. For example, P_{val1} and P_{val2} are parameters in the valence angle potential function. Parameters with a more direct physical meaning, like the torsional rotational barriers (V_1, V_2, V_3) bear their more recognizable names.

Modifications to the original document contain descriptions of the parameters, taken from the force fields and references to the files and subroutines of ADF implementation of ReaxFF. Several energy terms were added, based on the implementation.

See also Reaxff energy derivatives

0. Naming

Class	Number of params	Name in code
General	41	vpar
Atomic	32	ratomparam
Bond	16	rbondparam
Off-diagonal	6	roffdiagparam
Valence	7	rvalangparam
Torsion	7	rtorangparam
Hydrogen bond	4	rbondparam

Recently ReaxFF model was extended to **eReaxFF** within the same framework. Concept of **explicit electron** was introduced:

- new energy functionals to compute pairwise electrostatic interactions for explicit electrons was added;
- over- and under-coordination energy terms were modified;

Remarks would be implicitly specified below for each energy term where needed.

Full list of force field parameters is described in the List of parameters.

1. Overall system energy

Equation (1) describes the ReaxFF overall system energy.

$$\begin{aligned}
 E_{system} = & E_{bond} + E_{lp} + E_{over} + E_{under} + E_{val} + E_{pen} \\
 & + E_{coa} + E_{C2} + E_{tors} + E_{conj} + E_{H-bond} + E_{vdWaal} \\
 & + \left\{ \begin{array}{l} E_{EEM} \\ E_{ACKS2} \end{array} \right. + \\
 & + E_{Fe-dim} + E_{lg} + E_{trip} \quad (1)
 \end{aligned}$$

Below follows a description of the partial energies introduced in equation (1).

2. Bond Order and Bond Energy

2.1 Uncorrected BO

A fundamental assumption of ReaxFF is that the **bond order** BO'_{ij} between a pair of atoms can be obtained directly from the interatomic distance r_{ij} as given in Equation (2). In calculating the bond orders, ReaxFF distinguishes between contributions from σ -bonds, π -bonds and double π -bonds.

$$\begin{aligned}
 BO'_{ij} = & BO'_{ij}{}^{\sigma} + BO'_{ij}{}^{\pi} + BO'_{ij}{}^{\pi\pi} \\
 = & \exp \left[p_{bo1} \cdot \left(\frac{r_{ij}}{r_o^{\sigma}} \right)^{p_{bo2}} \right] + \exp \left[p_{bo3} \cdot \left(\frac{r_{ij}}{r_o^{\pi}} \right)^{p_{bo4}} \right] + \exp \left[p_{bo5} \cdot \left(\frac{r_{ij}}{r_o^{\pi\pi}} \right)^{p_{bo6}} \right] \quad (2)
 \end{aligned}$$

Formula	Code	Definition	Condition	Comment
$\frac{1}{r_o^\sigma}$	rob1(i, j)	1 / roffdiagparam(i, 4)	if defined and roffdiagparam(i, 4) > 0	rat(i) = ratomparam(i, 1)
		0.5 *(rat(i) + rat(j))	if rob1 ≤ 1.0e-15	
		2 / (rat(i) + rat(j))	otherwise	
$\frac{1}{r_o^\pi}$	rob2(i, j)	roffdiagparam(i, 5)	if defined and roffdiagparam(i, 5) > 0	rapt(i) = ratomparam(i, 7)
		0.5 *(rapt(i) + rapt(j))	if rob2 ≤ 1.0e-15	
		2 / (rapt(i) + rapt(j))	otherwise	
$\frac{1}{r_o^{\pi\pi}}$	rob1(i, j)	roffdiagparam(i, 6)	if defined and roffdiagparam(i, 6) > 0	vnq(i) = ratomparam(i, 17)
		0.5 *(vnq(i) + vnq(j))	if rob3 ≤ 1.0e-15	
		2 / (vnq(i) + vnq(j))	otherwise	
P_{bo1}	bop1	rbondparam(13)	σ - bond parameters	
P_{bo2}	bop2	rbondparam(14)		
P_{bo3}	pdp	rbondparam(10)	π - bond parameters	
P_{bo4}	ptp	rbondparam(11)		
P_{bo5}	pdo	rbondparam(5)	$\pi\pi$ - bond parameters	
P_{bo6}	popi	rbondparam(7)		

Based on the uncorrected bond orders BO' , derived from Equation (1), an uncorrected **overcoordination** Δ' can be defined for the atoms as the difference between the total bond order around the atom and the number of its bonding electrons Val .

$$\Delta'_i = -Val_i + \sum_{j=1}^{neighbours(i)} BO'_{ij} \quad (3a)$$

ReaxFF then uses these uncorrected overcoordination definitions to correct the bond orders BO'_{ij} using the scheme described in Equations (4a-f). To soften the correction for atoms bearing lone electron pairs a second overcoordination definition Δ'^{boc} (Equation (3b)) is used in Equations (4e) and (4f). This allows atoms like nitrogen and oxygen, which bear lone electron pairs after filling their valence, to break up these electron pairs and involve them in bonding without obtaining a full bond order correction.

$$\Delta_i'^{boc} = -Val_i^{boc} + \sum_{j=1}^{neighbours(i)} BO'_{ij} \quad (3b)$$

Formula	Code	Definition	Condition	Comment
Δ'_i	ovi		if ovc(i) = rbondparam(i, 15) > 0.001	uncorrected overcoordination
$\Delta_i'^{boc}$	ovi2			second overcoordination definition; to soften the correction for atoms bearing lone electron pairs
Δ_i	ovi4			corrected overcoordination
Val_i	aval(i)	ratomparam(i, 2)		number of bonding electrons of an atom
$Val_i'^{boc}$	vval3(:)	ratomparam(28)		= valf(:) = ratomparam(11) value changes for the first row elements
$\sum_{j=1}^{neighbours(i)} BO'_{ij}$	aboi	abo(i) + bo(nbon)	in CheckAndAddBond and CorrectBondOrders	
		abo(i) + bor - cutoff	in UpdateBondOrders	cutoff = 0.01 * vpar(30) formula is a bit different and might hold an error; bor = borsi + borpi + borpi2 (4a); total bond order

2.2 Corrected BO

$$\begin{aligned}
 BO_{ij} &= BO_{ij}^{\sigma} + BO_{ij}^{\pi} + BO_{ij}^{\pi\pi} \quad (4a) \\
 BO_{ij}^{\sigma} &= BO_{ij}^{\sigma'} \cdot f_1(\Delta'_i, \Delta'_j) \cdot f_4(\Delta'_i, BO'_{ij}) \cdot f_5(\Delta'_j, BO'_{ij}) \\
 BO_{ij}^{\pi} &= BO_{ij}^{\pi'} \cdot f_1(\Delta'_i, \Delta'_j)^2 \cdot f_4(\Delta'_i, BO'_{ij}) \cdot f_5(\Delta'_j, BO'_{ij}) \\
 BO_{ij}^{\pi\pi} &= BO_{ij}^{\pi\pi'} \cdot f_1(\Delta'_i, \Delta'_j)^2 \cdot f_4(\Delta'_i, BO'_{ij}) \cdot f_5(\Delta'_j, BO'_{ij})
 \end{aligned}$$

Formulas	Code	Definition	Subroutine	Comment
$BO'_{ij}{}^{\sigma}$	borsi	uncorrected bond orders	CheckAndAddBond	
$BO'_{ij}{}^{\pi}$	bopi(:)=borpi			
$BO'_{ij}{}^{\pi\pi}$	bopi2(:)=borpi2			
BO'_{ij}	bor= borsi + borpi + borpi2			
BO_{ij}^{σ}	bo(:)=bor - cutoff	corrected bond orders	CorrectBondOrders	cutoff = 0.01 · vpar(30)
BO_{ij}^{π}	bopi(:)			
$BO_{ij}^{\pi\pi}$	bopi2(:)			
BO_{ij}	abo(:)			

$$f_1(\Delta'_i, \Delta'_j) = \frac{1}{2} \cdot \left(\frac{Val_i + f_2(\Delta'_i, \Delta'_j)}{Val_i + f_2(\Delta'_i, \Delta'_j) + f_3(\Delta'_i, \Delta'_j)} + \frac{Val_j + f_2(\Delta'_i, \Delta'_j)}{Val_j + f_2(\Delta'_i, \Delta'_j) + f_3(\Delta'_i, \Delta'_j)} \right) \quad (4b)$$

$$f_2(\Delta'_i, \Delta'_j) = \exp(-p_{boc1} \cdot \Delta'_i) + \exp(-p_{boc1} \cdot \Delta'_j) \quad (4c)$$

$$f_3(\Delta'_i, \Delta'_j) = -\frac{1}{p_{boc2}} \cdot \ln \left\{ \frac{1}{2} \cdot [\exp(-p_{boc2} \cdot \Delta'_i) + \exp(-p_{boc2} \cdot \Delta'_j)] \right\} \quad (4d)$$

$$f_4(\Delta'_i, BO'_{ij}) = \frac{1}{1 + \exp(-p_{boc3} \cdot (p_{boc4} \cdot BO'_{ij} \cdot BO'_{ij} - \Delta_i^{boc}) + p_{boc5})} \quad (4e)$$

$$f_5(\Delta'_j, BO'_{ij}) \equiv f_4(\Delta'_j, BO'_{ij}) \quad (4f)$$

Formula	Code	Comment
$f_1(\Delta'_i, \Delta'_j)$	corrtot	
$f_2(\Delta'_i, \Delta'_j)$	exp11 + exp12	
$f_3(\Delta'_i, \Delta'_j)$	ovcor	
$f_4(\Delta'_i, BO'_{ij})$	bocor1	
$f_5(\Delta'_j, BO'_{ij})$	bocor2	
p_{boc1}	vpar(1)	
p_{boc2}	vpar(2)	
p_{boc3}	vp132 = $\sqrt{bo132(i) \cdot bo132(j)}$	bo132(:) = ratomparam(21)
p_{boc4}	vp131 = $\sqrt{bo131(i) \cdot bo131(j)}$	bo131(:) = ratomparam(20)
p_{boc5}	vp133 = $\sqrt{bo133(i) \cdot bo133(j)}$	bo133(:) = ratomparam(22)

A corrected overcoordination Δ_i can be derived from the corrected bond orders using equation (5).

$$\Delta_i = -Val_i + \sum_{j=1}^{neighbours(i)} BO_{ij} \quad (5)$$

Equation (6) is used to calculate the bond energies from the corrected bond orders BO_{ij} .

$$E_{bond} = -D_e^\sigma \cdot BO_{ij}^\sigma \cdot \exp [p_{be1} (1 - (BO_{ij}^\sigma)^{p_{be2}})] - D_e^\pi \cdot BO_{ij}^\pi - D_e^{\pi\pi} \cdot BO_{ij}^{\pi\pi} \quad (6)$$

Symbol	Code
E_{bond}	ebh
D_e^σ	de1h = vsymm * de1(:), de1(:) = rbondparam(:,1)
D_e^π	de2h = vsymm * de2(:), de2(:) = rbondparam(:,2)
$D_e^{\pi\pi}$	de3h = vsymm * de3(:), de3(:) = rbondparam(:,3)
p_{be1}	psi(:) = rbondparam(:,4)
p_{be2}	psp(:) = rbondparam(:,9)

Implementation: RxfBondsModule.f90, subroutine: CalcCovalentBonds

2.3 Fe dimer correction

It was added in 2012. Paper: [1] (<http://link.springer.com/article/10.1007%2Fs11837-012-0463-5>) In order to turn it on - fedime should be set to **1** in **control** file.

$$E_{Fe-dim} = p_{fe1} \cdot (BO_{ij})^{p_{fe2}} \cdot \exp \left(1 - \left(\frac{BO'_i + BO'_j}{2 \cdot BO_{ij}} \right)^{p_{fe3}} \right) \quad (6a)$$

Symbol	Code
p_{fe1}	vpar(14)
p_{fe2}	vpar(19)
p_{fe3}	vpar(23)

3. Lone pair energy

Equation (8) is used to determine the number of lone pairs around an atom. Δ_i^e is determined in Equation (7) and describes the difference between the total number of outer shell electrons (6 for oxygen, 4 for silicon, 1 for hydrogen) and the sum of bond orders around an atomic center.

$$\Delta_i^e = -Val_i^e + \sum_{j=1}^{neighbours(i)} BO_{ij} \quad (7)$$

$$n_{lp,i} = -int \left(\frac{\Delta_i^e}{2} \right) + \exp \left[-p_{lp1} \cdot \left(2 + \Delta_i^e - 2 \cdot int \left\{ \frac{\Delta_i^e}{2} \right\} \right)^2 \right] \quad (8)$$

Implementation: file RxfBondsModule.f90, subroutine: vlpFunc & CalcValenceAngles in RxfValAngModule.f90;

Formulas	Code
Δ_i^e	delta_e [in vlpFunc] = exlp1 = atoms%abo(:) - stlp(atoms%iatype(:))
Val_i^e	stlp(:) = ratomparam(8)
BO_{ij}	boa = bonds%bo(:) - bocutoff f; bocutoff = config%bondOrderCutoff
p_{lp1}	vpar(16)
$n_{lp,i}$	vlp, - in vlpFunc;

For oxygen with normal coordination (total bond order = 2, $\Delta_i^e = 4$), Equation (8) leads to 2 lone pairs. As

the total bond order associated with a particular O starts to exceed 2, Equation (8) causes a lone pair to gradually break up, causing a deviation Δ_i^{lp} , defined in Equation (9), from the optimal number of lone pairs $n_{lp,opt}$ (e.g. 2 for oxygen, 0 for silicon and hydrogen).

$$\Delta_i^{lp} = n_{lp,opt} - n_{lp,i} \quad (9)$$

$$n_{lp,opt} = \frac{1}{2} \cdot (Val_i^e - Val_i) \quad (9a)$$

This is accompanied by an energy penalty, as calculated by equation (10).

$$E_{lp} = \frac{p_{lp2} \cdot \Delta_i^{lp}}{1 + \exp(-75 \cdot \Delta_i^{lp})} \quad (10)$$

Implementation: file RxfBondsModule.f90, subroutine: preCompute, lonpa r;

Formulas	Code
E_{lp}	elph
Δ_i^{lp}	deltaLP(i) - in preCompute, deltaLP - in lonpar
p_{lp2}	vlp1(:) = ratomparam(18)
$n_{lp,i}$	vlp, - number of lone pairs; for heavier elements $amas(:) = ratomparam(3) > 21, = n_{lp,opt}$
$n_{lp,opt}$	nlpopt = 0.5 * (atoms%stlpc(:) - atoms%avalc(:));
Val_i^e	stlpc(i) = gRxFFld%stlp(i) = ratomparam(8), - number of valence electrons
Val_i	avalc(i) = gRxFFld%aval(i) = ratomparam(2), - valence

4. Overcoordination

For an overcoordinated atom ($\Delta_i > 0$), Equations (11a-b) impose an energy penalty on the system. The degree of overcoordination Δ is decreased if the atom contains a broken-up lone electron pair. This is done by calculating a corrected overcoordination (equation 11b), taking the deviation from the optimal number of lone pairs, as calculated in Equation (9), into account.

$$E_{over,i} = \frac{\sum_{j=1}^{nbond} p_{ovun1} \cdot D_e^\sigma \cdot BO_{ij}}{\Delta_i^{lpcorr} + Val_i} \cdot \Delta_i^{lpcorr} \cdot \left[\frac{1}{1 + \exp(p_{ovun2} \cdot \Delta_i^{lpcorr})} \right] \quad (11a)$$

$$\Delta_i^{lpcorr} = \Delta_i - \frac{\Delta_i^{lp}}{1 + p_{ovun3} \cdot \exp(p_{ovun4} \cdot \left\{ \sum_{j=1}^{neighbours(i)} (\Delta_j - \Delta_j^{lp}) \cdot (BO_{ij}^\pi + BO_{ij}^{\pi\pi}) \right\})} \quad (11b)$$

Implementation: file RxfBondsModule.f90, subroutine: ovco r;

Formulas	Code
E_{over}	Eover
Δ_j^{lp}	deltaLP(j) = nlpopt - nlp - mentioned before e; compare
Δ_j	deltaXel(j) = delta = atoms%abo(:) - atoms%avalc(:) - mentioned before e; compare e;
Δ_i^{lpcorr}	deltaLPcor
P_{ovun1}	vover(:) = rbondparam(8)
P_{ovun2}	vovun(:) = ratomparam(25)
P_{ovun3}	vpar(33)
P_{ovun4}	vpar(32)
D_e^σ	del(:) = rbondparam(1)
Val_i	avalc(i) = gRxFfld%aval(i) = ratomparam(2), avalc(:) = avalc(:) + helec(:)*eshare, eshare = exp(-vpar(37) * nvlr(:)^2)

5. Undercoordination

For an undercoordinated atom ($\Delta_i < 0$), we want to take into account the energy contribution for the resonance of the p-electron between attached under-coordinated atomic centers. This is done by Equations (12) where E_{under} is only important if the bonds between under-coordinated atom i and its under-coordinated neighbors j partly have π -bond character.

$$E_{under} = -P_{ovun5} \cdot \frac{1 - \exp(P_{ovun6} \cdot \Delta_i^{lpcorr})}{1 + \exp(-P_{ovun2} \cdot \Delta_i^{lpcorr})} \cdot \frac{1}{1 + P_{ovun7} \cdot \exp\left[P_{ovun8} \cdot \left\{ \sum_{j=1}^{neighbours(i)} (\Delta_j - \Delta_j^{lp}) \cdot (BO_{ij}^\pi + BO_{ij}^{\pi\pi}) \right\}\right]} \quad (12)$$

Implementation: file RxfBondsModule.f90, subroutine: ovco r;

Formulas	Code
E_{under}	Eunder
$\Delta_j^{lp}, \Delta_j, \Delta_i^{lpcorr}$	the same as in previous section
P_{ovun2}	vovun = ratomparam(25)
P_{ovun5}	valp1 = ratomparam(12)
P_{ovun6}	vpar(7)
P_{ovun7}	vpar(9)
P_{ovun8}	vpar(10)

6. Valence Angle Terms

6.1 Angle energy.

Just as for bond terms, it is important that the energy contribution from valence angle terms goes to zero as the bond orders in the valence angle goes to zero. Equations (13a-g) are used to calculate the **valence angle energy** contribution. The equilibrium angle Θ_o for Θ_{ijk} depends on the sum of p-bond orders (SBO) around the central atom j as described in Equation (13d). Thus, the equilibrium angle changes from around 109.47 for sp^3 hybridization (π -bond = 0) to 120 for sp^2 (π -bond=1) to 180 for sp (π -bond=2) based on the geometry of the central atom j and its neighbors. In addition to including the effects of π -bonds on the central atom j , Equation (13d) also takes into account the effects of over- and under-coordination in central atom j , as determined by Equation (13e), on the equilibrium valency angle, including the influence of a lone electron pair. Val^{angle} is the valency of the atom used in the valency and torsion angle evaluation. Val^{angle} is the same as Val^{boc} used in equation (3c) for non-metals. The functional form of Equation (13f) is designed to avoid singularities when $SBO = 0$ and $SBO = 2$.

- The angles in Equations (13a-g) are in radians.
- Remark: Equation (13a) has several forms, depending on the sign of p_{val1} . For $p_{val1} < 0$ the angular function is inverted to avoid linear Me-H-Me angle. Metal distance is large (> 2 Angstrom) \Rightarrow H-atom can come in between the Me atoms, making 2 H-Me bonds, without breaking the Me-Me bond. Inverted angular function gives maximum repulsion at 0 degrees and then tapers off exponentially, not affecting a normal, off-center bridging.

$$E_{val} = \begin{cases} f_7(BO_{ij}) \cdot f_7(BO_{jk}) \cdot f_8(\Delta_j) \cdot \{p_{val1} - p_{val1} \exp[-p_{val2}(\Theta_0(BO) - \Theta_{ijk})^2]\}, & \text{if } p_{val1} \geq 0 \\ f_7(BO_{ij}) \cdot f_7(BO_{jk}) \cdot f_8(\Delta_j) \cdot \{-p_{val1} \exp[-p_{val2}(\Theta_0(BO) - \Theta_{ijk})^2]\}, & \text{if } p_{val1} < 0 \end{cases} \quad (13a)$$

$$f_7(BO_{ij}) = 1 - \exp(-p_{val3} \cdot BO_{ij}^{p_{val4}}) \quad (13b)$$

$$f_8(\Delta_j) = p_{val5} - (p_{val5} - 1) \cdot \frac{2 + \exp(p_{val6} \cdot \Delta_j^{angle})}{1 + \exp(p_{val6} \cdot \Delta_j^{angle}) + \exp(-p_{val7} \cdot \Delta_j^{angle})} \quad (13c)$$

$$SBO = \sum_{n=1}^{neighbours(j)} (BO_{jn}^\pi + BO_{jn}^\pi) + [1 - \prod_{n=1}^{neighbours(j)} \exp(-BO_{jn}^8)] \cdot (-\Delta_j^{angle} - p_{val8} \cdot n_{lp,j}) \quad (13d)$$

$$\Delta_j^{angle} = -Val_j^{angle} + \sum_{n=1}^{neighbours(j)} BO_{jn} \quad (13e)$$

$$SBO2 = \begin{cases} 0, & \text{if } SBO \leq 0 \\ SBO^{p_{val9}}, & \text{if } 0 < SBO < 1 \\ 2 - (2 - SBO)^{p_{val9}}, & \text{if } 1 < SBO < 2 \\ 2, & \text{if } SBO > 2 \end{cases} \quad (13f)$$

$$\Theta_0(BO) = \pi - \Theta_{0,0} \cdot \{1 - \exp[-p_{val10} \cdot (2 - SBO2)]\} \quad (13g)$$

Implementation: file RxfValAngModule.f90, subroutine: CalcValenceAngle s;

Formulas	Code
E_{val}	epen
BO_{ij}, BO_{jk}	boa = bonds%bo(:) - bocutoff, bo b; bocutoff = config%bondOrderCutoff
$f_7(BO_{ij}), f_7(BO_{jk})$	exa2, exb2
$f_8(\Delta_j)$	evboardj2
Δ_j	ovb = atoms%abo(:) - aval(atoms%iatype(:)), aval = ratomparam(2)
Δ_j^{angle}	exbo = atoms%abo(:) - gRxFFld%valf(:) 0, if vpar(38) != 0 for non-C-C-C angles
Val_j^{angle}	valf = ratomparam (11)
$n_{lp,j}$	vlpadj
P_{val1}	vka = rvalangparam (2)
P_{val2}	vka3 = rvalangparam(3)
P_{val3}	vval1 = ratomparam(26)
P_{val4}	vval2 = rvalangparam(7)
P_{val5}	vval4 = ratomparam(29)
P_{val6}	vpar(15)
P_{val7}	vkac = rvalangparam (5)
P_{val8}	vpar(34)
P_{val9}	vpar(17)
P_{val10}	vpar(18)
$\Theta_0(BO)$	thetao
$\Theta_{0,0}$	th0(jkl) = rvalangparam(jkl,1)
Θ_{jkl}	hl, most likely calculated based on geometries

6.2 Penalty energy.

To reproduce the stability of systems with two double bonds sharing an atom in a valency angle, like allene, an additional energy penalty, as described in Equations (14a-b), is imposed for such systems. Equation (14b) deals with the effects of over/undercoordination in central atom j on the penalty energy.

$$E_{pen} = p_{pen1} \cdot f_9(\Delta_j) \cdot \exp[-p_{pen2} \cdot (BO_{ij} - 2)^2] \cdot \exp[-p_{pen2} \cdot (BO_{jk} - 2)^2] \quad (14a)$$

$$f_9(\Delta_j) = \frac{2 + \exp(-p_{pen3} \cdot \Delta_j)}{1 + \exp(-p_{pen3} \cdot \Delta_j) + \exp(p_{pen4} \cdot \Delta_j)} \quad (14b)$$

Implementation: file RxfValAngModule.f90, subroutine: CalcValenceAngle s;

Formulas	Code
E_{pen}	epen
$f_9(\Delta_j)$	ecsboardj
Δ_j	ovb = atoms%abo(:) - aval(atoms%iatype(:)), aval = ratomparam(2)
BO_{ij}, BO_{jk}	boa = bonds%bo(:) - bocutoff, bo b; bocutoff = config%bondOrderCutoff
P_{pen1}	vkap = rvalangparam(6)
P_{pen2}	vpar(20)
P_{pen3}	vpar(21)
P_{pen4}	vpar(22)

6.3 Three-body conjugation term.

The hydrocarbon ReaxFF potential contained only a four-body conjugation term (see section 7.2), which was sufficient to describe most conjugated hydrocarbon systems. However, this term failed to describe the stability obtained from conjugation by the NO_2 -group. To describe the stability of such groups a three-body conjugation term is included (Equation (15)).

$$\begin{aligned}
 E_{coa} = & p_{coa1} \cdot \frac{1}{1 + \exp(p_{coa2} \cdot \Delta_j^{val})} \cdot \exp \left[-p_{coa3} \cdot \left(-BO_{ij} + \sum_{n=1}^{neighbours(i)} BO_{in} \right)^2 \right] \cdot \\
 & \cdot \exp \left[-p_{coa3} \cdot \left(-BO_{jk} + \sum_{n=1}^{neighbours(i)} BO_{kn} \right)^2 \right] \cdot \\
 & \cdot \exp \left[-p_{coa4} \cdot (BO_{ij} - 1.5)^2 \right] \cdot \exp \left[-p_{coa4} \cdot (BO_{jk} - 1.5)^2 \right]
 \end{aligned} \tag{15}$$

Implementation: file RxfValAngModule.f90, subroutine: CalcValenceAngles;

Formulas	Code
E_{coa}	ecoah
BO_{ij}, BO_{jk}	boa = bonds%bo(:) - bocutoff, bo b; bocutoff = config%bondOrderCutoff
p_{coa1}	vka8(:) = rvalangparam(:,4)
p_{coa2}	-vpar(3)
p_{coa3}	vpar(39)
p_{coa4}	vpar(31)
Δ_j^{val}	ovb = atoms%abo(:) - vval3(atoms%iatype(:)), vval3 = ratomparam(27)

7. Torsion angle terms

7.1 Torsion rotation barriers.

Just as with angle terms we need to ensure that dependence of the energy of torsion angle w_{ijkl} accounts properly for $BO \rightarrow 0$ and for BO greater than 1. This is done by Equations (16a)-(16c).

$$E_{tors} = f_{10}(BO_{ij}, BO_{jk}, BO_{kl}) \cdot h_{sin}(\Theta_{ijk}, \Theta_{jkl}) \tag{16a}$$

$$f_{10}(BO_{ij}, BO_{jk}, BO_{kl}) = \begin{cases} \left[\frac{1}{2} V_1 \cdot (1 + \cos \omega_{ijkl}) + V_2 \cdot \exp \left\{ p_{tor1} \cdot \left[1 - (BO_{jk}^2 - 1 + f_{11}(\Delta_j, \Delta_k)) \right]^2 \right\} \cdot (1 - \cos 2\omega_{ijkl}) + V_3 \cdot (1 + \cos 3\omega_{ijkl}) \right] & \text{if tors13} \\ \left[1 - \exp(-2 \cdot p_{tor2} \cdot BO_{ij}^2) \right] \cdot \left[1 - \exp(-2 \cdot p_{tor2} \cdot BO_{jk}^2) \right] \cdot \left[1 - \exp(-2 \cdot p_{tor2} \cdot BO_{kl}^2) \right], & \text{otherwise} \\ \left[1 - \exp(-p_{tor2} \cdot BO_{ij}) \right] \cdot \left[1 - \exp(-p_{tor2} \cdot BO_{jk}) \right] \cdot \left[1 - \exp(-p_{tor2} \cdot BO_{kl}) \right], & \end{cases} \tag{16b}$$

$$f_{11}(\Delta_j, \Delta_k) = \frac{2 + \exp \left[-p_{tor3} \cdot \left(\Delta_j^{angle} + \Delta_k^{angle} \right) \right]}{1 + \exp \left[-p_{tor3} \cdot \left(\Delta_j^{angle} + \Delta_k^{angle} \right) \right] + \exp \left[p_{tor4} \cdot \left(\Delta_j^{angle} + \Delta_k^{angle} \right) \right]} \tag{16c}$$

$$h_{sin}(\Theta_{ijk}, \Theta_{jkl}) = \begin{cases} (\sin \Theta_{ijk} \cdot \sin \Theta_{jkl})^3, & \text{if tors19} \\ \sin \Theta_{ijk} \cdot \sin \Theta_{jkl}, & \text{otherwise} \end{cases} \tag{16d}$$

Implementation: file RxfTorsionModule.f90, subroutine: CalcTorsionAngles;

Formulas	Code
E_{tors}	eth
$f_{10}(BO_{ij}, BO_{jk}, BO_{kl})$	bocor4
$f_{11}(\Delta_j, \Delta_k)$	etboadj
$BO_{ij}, BO_{jk}, BO_{kl}$	boa = bonds%bo(:) - bocutoff, bob, boc; bocutoff = config%bondOrderCutoff
BO_{jk}^{π}	bopi(:)
P_{tor1}	v4(:) = rtorangparam(:,4)
P_{tor2}	vpar(24)
P_{tor3}	vpar(25)
P_{tor4}	vpar(26)
V_1	v1 = rtorangparam(1)
V_2	v2 = rtorangparam(2)
V_3	v3 = rtorangparam(3)
ω_{ijkl}	th
$\sin^2(\omega_{ijkl})$	sinh2
$\Theta_{ijk}, \Theta_{jkl}$	ht1, ht2
$\sin(\Theta_{ijk}), \sin(\Theta_{jkl})$	sinhd = sin(ht1), sinhe = sin(ht2)
$h_{\sin}(\Theta_{ijk}, \Theta_{jkl})$	hsin
$\Delta_j^{angle}, \Delta_k^{angle}$	exbo1 = abo(:) - valf(iatype(:)), exbo2; valf = ratomparam(11)
$\Delta_j^{angle} + \Delta_k^{angle}$	htovt

version is set in the **control** file.

According to the old / original document in Equation (16a) were made few changes:

- 1) instead of the term: $\left[1 - \left(BO_{jk}^{\pi} - 1 + f_{11}(\Delta_j, \Delta_k)\right)\right] = \left(2 - BO_{jk}^{\pi} - f_{11}(\Delta_j, \Delta_k)\right)$ was used $\left(BO_{jk}^{\pi} - 1 + f_{11}(\Delta_j, \Delta_k)\right)$;
- 2) part of the sum was missing: $\frac{1}{2}V_1 \cdot (1 + \cos \omega_{ijkl})$.
- 3) there was constant $\frac{1}{2}$ in front of V_2 and V_3 . It no longer corresponds to current implementation.

7.2 Four body conjugation term.

Equations (17a-b) describe the contribution of conjugation effects to the molecular energy. A maximum contribution of conjugation energy is obtained when successive bonds have bond order values of 1.5 as in benzene and other aromatics.

$$E_{conj} = f_{12}(BO_{ij}, BO_{jk}, BO_{kl}) \cdot p_{cot} \cdot 1 \cdot \left[1 + (\cos^2 \omega_{ijkl} - 1) \cdot \sin \Theta_{ijk} \cdot \sin \Theta_{jkl}\right] \quad (17a)$$

$$f_{12}(BO_{ij}, BO_{jk}, BO_{kl}) = \begin{cases} \left[\sin(BO_{ij} \cdot \frac{\pi}{3}) \cdot \sin(BO_{jk} \cdot \frac{\pi}{3}) \cdot \sin(BO_{kl} \cdot \frac{\pi}{3})\right]^4, & \text{if tors13} \\ \exp[-p_{cot} \cdot 2 \cdot (BO_{ij} - 1 \frac{1}{2})^2] \cdot \exp[-p_{cot} \cdot 2 \cdot (BO_{jk} - 1 \frac{1}{2})^2] \cdot \exp[-p_{cot} \cdot 2 \cdot (BO_{kl} - 1 \frac{1}{2})^2], & \text{otherwise} \end{cases} \quad (17b)$$

Implementation: file RxfTorsionModule.f90, subroutine: CalcTorsionAngle s;

Formulas	Code
E_{conj}	ecoh
$f_{12}(BO_{ij}, BO_{jk}, BO_{kl})$	sbo = sbo(boa, bob, boc)
$BO_{ij}, BO_{jk}, BO_{kl}$	boa = bonds%bo(:) - bocutoff, bob, bo c; bocutoff = config%bondOrderCutoff
P_{cot1}	vconj(:) = rtorangparam(:,5)
P_{cot2}	vpar(28)
ω_{ijkl}	th
$\sin^2(\omega_{ijkl})$	sinth2
$\Theta_{ijk}, \Theta_{jkl}$	ht1, ht2
$\sin(\Theta_{ijk}), \sin(\Theta_{jkl})$	sinhd = sin(ht1), sinhe = sin(ht2)

config - refers to control file.

8. Hydrogen bond interactions

Equation (18) described the bond-order dependent hydrogen bond term for a X-H---Z system as incorporated in ReaxFF.

$$E_{Hbond} = p_{hb1} \cdot [1 - \exp(p_{hb2} \cdot BO_{XH})] \cdot \exp \left[p_{hb3} \left(\frac{r_{hb}^o}{r_{HZ}} + \frac{r_{HZ}}{r_{hb}^o} - 2 \right) \right] \cdot \sin^4 \left(\frac{\Theta_{XHZ}}{2} \right) \quad (18)$$

Implementation: file RxfHBondModule.f90, subroutine CalcHBonds;

Formulas	Code	Code
E_{Hbond}	ehbh	
P_{hb1}	dehb(:)	rhbondparam(:,2)
P_{hb2}	- vhb1(:)	- rhbondparam(:,3)
P_{hb3}	- vhb2(:)	- rhbondparam(:,4)
BO_{XH}	boa	bonds%bo(:)
r_{HZ}	distance, calculated from coordinates	
r_{hb}^o	rhb(:)	rhbondparam(:,1)
$\frac{r_{hb}^o}{r_{HZ}}$	rhu1	
Θ_{XHZ}	hhb	

Originally in 2008 formula (18) contained $\sin^8 \left(\frac{\Theta_{XHZ}}{2} \right)$ instead of $\sin^4 \left(\frac{\Theta_{XHZ}}{2} \right)$. This was changed after the development of water-branch force fields.

9. Correction for C_2

ReaxFF erroneously predicts that two carbons in the C_2 -molecule form a very strong (triple) bond, while in fact the triple bond would get de-stabilized by terminal radical electrons, and for that reason the carbon-carbon bond is not any stronger than a double bond. To capture the stability of C_2 we introduced a new partial energy contribution (E_{C2}). Equation (19) shows the potential function used to de-stabilize the C_2 molecule:

$$E_{C2} = \begin{cases} k_{e2} \cdot (BO_{ij} - \Delta_i - 0.04 \cdot \Delta_i^4 - 3)^2, & \text{if } BO_{ij} - \Delta_i - 0.04 \cdot \Delta_i^4 > 3 \\ 0, & \text{if } BO_{ij} - \Delta_i - 0.04 \cdot \Delta_i^4 \leq 3 \end{cases} \quad (19)$$

where Δ_i is the level of under/overcoordination on atom i as obtained from subtracting the valency of the atom (4 for carbon) from the sum of the bond orders around that atom and k_{c2} - the force field parameter associated with this partial energy contribution.

Implementation: file RxfBondsModule.f90, subroutine lonpa r;

Formulas	Code
E_{C2}	elph
k_{c2}	vpar(6)

Extra condition: $\text{abs}(\text{vpar}(6)) > 0.001$

10. Triple bond energy correction.

To describe the triple bond in carbon monoxide a triple bond stabilization energy is used, making CO both stable and inert. This energy term only affects C-O bonded pairs. Equation (20) shows the energy function used to describe the triple bond stabilization energy. Extra condition to turn on **tripleBondOption** is:

$\text{vpar}(40)=1$

$$E_{trip} = p_{trip1} \cdot \exp \left[-p_{trip2} (BO_{ij} - 2.5)^2 \right] \cdot \frac{\exp \left[-p_{trip4} \cdot \left(\sum_{k=1}^{neighbours(i)} BO_{ik} - BO_{ij} \right) \right] + \exp \left[-p_{trip4} \cdot \left(\sum_{k=1}^{neighbours(i)} BO_{jk} - BO_{ij} \right) \right]}{1 + 25 \cdot \exp \left[p_{trip3} \cdot (\Delta_i + \Delta_j) \right]}$$

(20)

Implementation: file RxfBondsModule.f90, subroutine CalcCovalentBonds;

Formulas	Code
E_{trip}	estriph
p_{trip1}	vpar(11)
p_{trip2}	vpar(8)
p_{trip3}	vpar(5)
p_{trip4}	vpar(4)

11. Nonbonded interactions

In addition to valence interactions, which model the interactions between overlapping orbitals, there are **non-covalent interactions**:

1. Repulsion at short interatomic distances due to the Pauli principle. (Strictly speaking, this is also due to the interaction of overlapping orbitals, yet without transferring electrons from an anti-bonding to bonding orbital.)
2. Long-range attraction energies due to dispersion.
3. Electrostatic and polarization interactions.

The first two effects are described by a van der Waals potential in ReaxFF, while the last point is described with a fluctuating charge model.

The non-covalent interactions are included between *all* atom pairs, thus avoiding discontinuities in the energy when bonds are broken or formed. Instead, the repulsive short-range interaction is smoothly switched off when bonds are formed.

11.1 Taper correction

Nonbonded interactions are only computed when the distance between two particles is below the nonbonded cutoff radius, R_{cut} . To avoid energy discontinuities when particles move in and out of the nonbonded cutoff radius, ReaxFF employs a Taper correction, as developed by de Vos Burchart (1995). Each nonbonded energy and derivative is multiplied by a Taper-term, which is taken from a distance-dependent 7th order polynomial (equation 20).

$$Tap(r_{ij}) = \begin{cases} Tap_7 \cdot r_{ij}^7 + Tap_6 \cdot r_{ij}^6 + Tap_5 \cdot r_{ij}^5 + Tap_4 \cdot r_{ij}^4 + Tap_3 \cdot r_{ij}^3 + Tap_2 \cdot r_{ij}^2 + Tap_1 \cdot r_{ij} + Tap_0 & \text{when } r_{ij} < R_{cut} \\ 0 & \text{when } r_{ij} \geq R_{cut} \end{cases} \quad (21)$$

The terms in this polynomial are chosen to ensure that all 1st, 2nd and 3rd derivatives of the nonbonded interactions to the distance are continuous and go to zero at the cutoff boundary. To that end, the terms Tap_0 to Tap_7 in equation (20) are calculated by the scheme in equation (21):

Variable	Code	Formula
$Tap(r_{ij})$	gTaper%sw	
Tap_7	swc7	$20/D^7$
Tap_6	swc6	$-70 \cdot (swa + swb)/D^7$
Tap_5	swc5	$84 \cdot (swa^2 + 3 \cdot swa \cdot swb + swb^2)/D^7$
Tap_4	swc4	$-35 \cdot (swa^3 + 9 \cdot swa^2 \cdot swb + 9 \cdot swa \cdot swb^2 + swb^3)/D^7$
Tap_3	swc3	$140 \cdot (swa^3 \cdot swb + 3 \cdot swa^2 \cdot swb^2 + swa \cdot swb^3)/D^7$
Tap_2	swc2	$-210 \cdot (swa^3 \cdot swb^2 + swa^2 \cdot swb^3)/D^7$
Tap_1	swc1	$140 \cdot swa^3 \cdot swb^3/D^7$
Tap_0	swc0	$(-35 \cdot swa^3 \cdot swb^4 + 21 \cdot swa^2 \cdot swb^5 + 7 \cdot swa \cdot swb^6 + swb^7)/D^7$
swa	vpar(12)	
swb	vpar(13)	R_{cut}
D	D	$swb - swa$

File: Taper.f90

11.2 van der Waals interactions

There are several types of van der Waals interaction [**nonbontype**], depending on the presence of Inner Wall or Shielding:

3: Inner wall, shielding
 2: Inner wall, no shielding
 1: No inner wall, shielding
 0: No inner wall, no shielding

This conditions are switched on or off based on the following criteria:

Inner Wall: ratomparam(30)>0.01 .and. ratomparam(32)>0.01

Shielding: ratomparam(10)>0.5

Van der Waals interactions are modeled with a distance-corrected Morse-potential (Eq.23a-b). By including a **shielded** interaction (Eq.23b) excessively high repulsions between bonded atoms (1-2 interactions) and atoms sharing a valence angle (1-3 interactions) are avoided. Alternatively, without shielding (23b) is not used and in (23a) instead of $f_{13}(r_{ij})$ used r_{ij} - distance between 2 atoms.

$$E_{vdWaal} = Tap(r_{ij}) \cdot D_{ij} \cdot \left\{ \exp \left[\alpha_{ij} \cdot \left(1 - \frac{f_{13}(r_{ij})}{r_{vdW}} \right) \right] - 2 \cdot \exp \left[\frac{1}{2} \cdot \alpha_{ij} \cdot \left(1 - \frac{f_{13}(r_{ij})}{r_{vdW}} \right) \right] \right\} \quad (23a)$$

$$f_{13}(r_{ij}) = \begin{cases} \left[r_{ij}^{p_{vdW1}} + \left(\frac{1}{\gamma_w} \right)^{p_{vdW1}} \right]^{\frac{1}{p_{vdW1}}}, & \text{shielding} \\ r_{ij} & \text{no shielding} \end{cases} \quad (23b)$$

Corresponding parameters [files: RxfNonbond.f90, vvNonBon.fh]:

Formula	Code	Definition	Condition
D_{ij}	p2co(i,j)	roffdiagparam(i,j,1)	if defined
		$\text{sqrt}(\text{ratomparam}(i,5) \cdot \text{ratomparam}(j,5))$	otherwise
α_{ij}	p3co(i,j)	roffdiagparam(i,j,3)	if defined
		$\text{sqrt}(\text{ratomparam}(i,9) \cdot \text{ratomparam}(j,9))$	otherwise
r_{vdW}	p1co(i,j)	roffdiagparam(i,j,2)	if defined
		$\text{sqrt}(\text{ratomparam}(i,4) \cdot \text{ratomparam}(j,4))$	otherwise
P_{vdW1}	vpar(29)		
γ_w		ratomparam(i,10) · ratomparam(j,10)	
$\left(\frac{1}{\gamma_w}\right)^{P_{vdW1}}$	gamwco(i,j)		
$Tap(r_{ij})$		taper correction	
r_{ij}		distance between 2 atoms: i and j	

In case **inner wall** is present, next energy term is added to E_{vdWaal} s and thus to total Energy:

$$E_{coreh} = pc2 \cdot \exp(pc3 \cdot (1 - \frac{r_{ij}}{pc1})) \quad (23c)$$

Formula	Code	Definition
$pc1$	rcore(i,j)	$\text{sqrt}(\text{ratomparam}(i,30) \cdot \text{ratomparam}(j,30))$
$pc2$	ecore(i,j)	$\text{sqrt}(\text{ratomparam}(i,31) \cdot \text{ratomparam}(j,31))$
$pc3$	acore(i,j)	$\text{sqrt}(\text{ratomparam}(i,32) \cdot \text{ratomparam}(j,32))$

11.3 lg - dispersion correction

The long-range correction was added [2] (<http://pubs.acs.org/doi/abs/10.1021/jp201599t>). The low-gradient model is used.

$$E_{lg} = \sum_{ij, i < j}^{natom} \frac{C_{lg,ij}}{r_{ij}^6 + d \cdot R_{eij}^6} \quad (23d)$$

Formulas	Code
E_{lg}	eDisp
$C_{lg,ij}$	$\frac{2 \cdot C_i \cdot C_j}{C_i + C_j}$, if clgDispOffDiag(:, :) < -0.5
	clgDispOffDiag(:, :)
clgDispOffDiag(:, :)	roffdiagparam(:, 7) , if dispersionType = 1 -1, else
C_i	clgDispOnDiag(:)
d	scaling, scaling factor
r_{ij}	rr
R_{eij}	vdWr, vdW Radii in the UFF

Subroutine: LgDispersionCorrection

11.4 Energy due to charges (includes Coulomb)

ReaxFF includes an energy term for fluctuating charges, which are re-optimized for every new geometry,

e.g. at every time step in a molecular dynamics simulation. ReaxFF can be used with two fluctuating charge modes: the electronegativity equalization method (**EEM**) and the atom-condensed Kohn-Sham DFT approximated to 2nd order (**ACKS2**), for which the equations will be specified in more detail below.

11.4.1 EEM

EEM was originally proposed by Mortier et al. in 1986 [3] (<http://dx.doi.org/10.1021/ja00275a013>). Another common name for this type of model is Charge Equilibration (QEq), a variant introduced by Rappe and Goddard, replacing the electrostatic point-charge interactions by those of 1s Slater-type densities. (A second modification in QEq is a third-order term for the self-energy of the hydrogen atoms, which is not used in ReaxFF.) [4] (<http://dx.doi.org/10.1021/j100161a070>) Later Njo et al proposed a simple analytic approximation for the interaction between 1s Slater-type densities [5] ([http://dx.doi.org/10.1016/s1381-1169\(98\)00024-7](http://dx.doi.org/10.1016/s1381-1169(98)00024-7)), which is used in ReaxFF.

$$E_{EEM} = \min_{q_i: \sum q_i = q_{tot}} \left[\sum_{i=1}^{natom} \chi_i q_i + \frac{1}{2} \sum_{i=1, j=1}^{natom} \eta_{ij} q_i q_j \right] \quad (24)$$

The EEM energy contains the following ingredients:

Variables (implicit geometry dependence)	Unit	Code	Description
q_i	e	gRxfAt%ch(i), gRxfAt%chgold(i), gRxfAt%chgold2(i)	Atomic charge
Quantities with explicit geometry dependence	Unit	Code	Description
$\eta_{ij} = Tap(r_{ij}) \frac{1}{4\pi\epsilon_0} \frac{q_i \cdot q_j}{[r_{ij}^3 + (\gamma_i\gamma_j)^{-3/2}]^{1/3}} \quad \forall i \neq j$	kcal mol ⁻¹	xmorr, xmorr1, xmorr2, xmorr	Coulomb interaction between a pair of atoms, with short-range damping
$Tap(r_{ij})$	1	Taper0()	Taper correction
r_{ij}	Å	dis2, distance	Interatomic distance
Constant quantities	Unit	Code	Description
q_{tot}	e	config%totalCharge	Total charge
χ_i	eV e ⁻¹	ratomparam(i,14), gRxFFld%chi(i)	Intrinsic atomic electronegativity, negative of the intrinsic atomic chemical potential
$\eta_{ii}/2 = \eta_i$	eV e ⁻²	ratomparam(i,15), gRxFFld%eta(i)	Atomic hardness, using the factor-two convention introduced by Parr and Pearson. [6] (http://dx.doi.org/10.1021/ja00364a005)
γ_i	Å ⁻¹	ratomparam(i,6), gRxFFld%gam(i)	Atomic short-range damping constant for electrostatic interactions
$(\gamma_i\gamma_j)^{-3/2}$	Å ³	gRxFFld%gamcco(i,j)	Auxiliary variable to avoid recomputation
$\frac{1}{4\pi\epsilon_0}$	kcal mol ⁻¹ e ⁻² Å	(hard-wired in code, implicit in conversion between eV and kcal mol ⁻¹)	332.0638

The polarization catastrophe can be avoided, for any geometry, by imposing the following inequality constraint during the parameter calibration:

$$\eta_{ii} > \frac{\gamma_i}{4\pi\epsilon_0} \quad \text{or} \quad \eta_i > \frac{\gamma_i}{8\pi\epsilon_0}$$

When this condition is satisfied for all pairs of atomic hardness and short-range damping parameters, the

hardness matrix, η_{ij} , is always positive definite.

Instead of constraining the total charge of the system, one may also divide the system into fragments and constrain the charge of each fragment to a fixed value. This will result in one Lagrange multiplier per fragment in the EEM equations instead of just one for the total charge.

11.4.2 ACKS2

The ACKS2 model is an extension of EEM with extra terms due to the Kohn-Sham kinetic energy: [7] (<http://dx.doi.org/10.1063/1.4791569>)

$$E_{ACKS2} = \min_{q_i: \sum q_i = \sum q_i^0} \left[\sum_{i=1}^{natom} \chi_i q_i + \frac{1}{2} \sum_{i=1, j=1}^{natom} \eta_{ij} q_i q_j + \max_{u_i: \sum u_i = 0} \left[\sum_{i=1}^{natom} u_i (q_i - q_i^0) + \frac{1}{2} \sum_{i=1, j=1}^{natom} X_{ij} u_i u_j \right] \right] \quad (25)$$

ACKS2 is a generalization of the SQE model. [8] (<http://dx.doi.org/10.1103/physrevb.79.104303>) and it fixes the following weaknesses of EEM:

- When molecular fragments dissociate, the net charge of the fragments becomes equal to the sum of the reference charges (of the atoms and other particles in the fragment). This can be used to force isolated fragments to bear an integer charge, whereas EEM allows for fractional charges in the dissociation limit.
- Materials can have a finite dielectric constant when using the ACKS2 model. EEM suffers from the so-called metallic polarizability scaling [9] (<http://dx.doi.org/10.1063/1.2872603>), which is unrelated to the polarization catastrophe. This artifact of EEM may lead to large amounts of charge-transfer between two phases in one simulation, which differ in intrinsic electronegativity. Another manifestation of the EEM artifact was observed in eReaxFF: explicit electron or hole particles are almost completely compensated by excessive charge flow. These artifacts are not present in ACKS2.
- The extra parameter (bond softness, inverse of bond hardness in SQE) has a sound physical interpretation: it is a condensed property of the non-interacting response kernel in Kohn-Sham DFT [10] (<http://dx.doi.org/10.1063/1.4791569>) and it is directly related to the dielectric constant. [11] (<http://dx.doi.org/10.1103/physrevb.79.104303>)

ACKS2 also imposes a few limitations:

- Atomic reference charges must be set. (In EEM, such reference charges can always be eliminated by a transformation of variables, not so in ACKS2 or SQE.)
- The set of equations to be solved is twice as large as those of EEM. However, the additional equations are more sparse.

The ACKS2 energy contains the following *new* ingredients (in addition to those already present in EEM as discussed above):

Variables (implicit geometry dependence)	Unit	Code	Description
u_i	V	gRxfAt%effpot(i)	Change in atomic Kohn-Sham (or effective) potential (relative to that of a isolated atom)
Quantities with explicitly geometry dependence	Unit	Code	Description
$X_{ij} = \begin{cases} X_{soft} \left(\frac{2r_{ij}}{C_i+C_j} \right)^3 \left(1 - \frac{2r_{ij}}{C_i+C_j} \right)^6 & \text{if } r_{ij} < (C_i + C_j)/2 \\ 0 & \text{otherwise} \end{cases}$	eV ⁻¹ &nbs p;e ²	xmotr, xmotr1, xmotr2, xmotr	Bond or atom-pair softness
$X_{ii} = - \sum_{j=1, j \neq i}^{natom} X_{ij}$	eV ⁻¹ e ²	xmotr, xmotr1, xmotr2, xmotr	Sum rule imposed on the X matrix
Constant quantities	Unit	Code	Description
$q_i^0 = \begin{cases} \text{Oxidation state, usually 0} & \text{atom} \\ -1 & \text{electron} \\ +1 & \text{hole} \end{cases}$	e	vmcha(mol)/N(mol), where N(mol) is the number of atoms in the molecule	The reference charge of an atom or a particle.
C_i	Å	gRxFFld%softcut(i) = ratomparam(i, 23)	Atomic softness cutoff parameter
X_{soft}	eV ⁻¹ e ²	vpar(35)	Universal softness parameter

The polarization catastrophe can be avoided, for any geometry, by imposing the following inequality during the parameterization (in addition to the requirements for EEM given above):

$$X_{soft} > 0$$

This will guarantee that the X matrix is always negative semi-definite, which is the expected behavior for a response kernel.

Instead of constraining the total charge of the system, one may also divide the system into fragments and constrain the charge of each fragment to a fixed value. This can be done by zeroing all off-diagonal blocks of the X matrix that correspond to atom pairs in different fragments. This will force all fragments to bear a charge equal to the sum of the reference charges. **This is now implemented differently, i.e. with additional Lagrange multipliers as in EEM, which is not the most efficient way.**

11.4.3 Remark on the "Coulomb" and "Charge" energy

The "Coulomb" energy is a part of the EEM or ACKS2 energy. As with the van der Waals-interactions, Coulomb interactions are taken into account between all atom pairs:

$$E_{coulomb} = \frac{1}{2} \sum_{i \neq j}^{natom} \eta_{ij}$$

Note that no Ewald summation is implemented in ReaxFF, nor any of its more advanced alternatives such as SPME. Electrostatic interactions are truncated at the cutoff radius (R_{cut}) using the Taper function, similarly to the Wolf summation. [12] (<http://dx.doi.org/10.1063/1.478738>) In periodic systems, the electrostatic interactions with atoms in periodic images are also taken into account, if they are within the cutoff. For small periodic systems, this may imply that Coulomb contributions are present to the diagonal terms of the Coulomb energy, i.e. between an atom and its own periodic image (not shown in equation). Such diagonal contributions contribute to the total energy and the virial tensor but not the forces.

The "Charge" energy in the ReaxFF output is the EEM or ACKS2 energy minus the Coulomb energy.

12. eReaxFF

Limited pseudo-classical explicit electron/holes degrees-of-freedom scheme, complementary to implicit treatment of electrons in the bonded interactions of ReaxFF. The electron or hole is represented as an additional particle that carries a -1 (electron) or +1 (hole) charge:

- nuclei are treated as point charges
- electrons - Gaussian-type function $\psi = \exp(-a * (r - r')^2)$

Following energy terms are added to Equation (1), when **eReaxFF** is turned on. The pairwise electrostatic interaction between the electron and core-charge:

$$E_{nucl(i)-elec(j)} = -\frac{Z_i}{4\pi\epsilon_0 R_{ij}} \beta_i \operatorname{erf}(\sqrt{\alpha_i} R_{ij}) \operatorname{Tap}(R_{ij})$$

$$E_{elec(i)-hole(j)} = \frac{1}{4\pi\epsilon_0 R_{ij}} \left(p_{elho} \cdot \operatorname{erf}(\sqrt{2\sqrt{\alpha_i\alpha_j}} R_{ij}) - \beta_i \cdot \operatorname{erf}(\sqrt{2\alpha_i} R_{ij}) - \beta_j \cdot \operatorname{erf}(\sqrt{2\alpha_j} R_{ij}) \right) \operatorname{Tap}(R_{ij})$$

$$E_{elec(i)-elec(j)} = \frac{1}{4\pi\epsilon_0 R_{ij}} \beta_e \operatorname{erf}(\sqrt{\alpha_e} R_{ij}) \operatorname{Tap}(R_{ij})$$

Value	Code	Description
$E_{nucl(i)-elec(j)}$		the pairwise electrostatic interaction between the electron and core-charge
$\operatorname{erf} x = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$	DERF(:)	error function
R_{ij}		distance between electron and nucleus
Z_i		nuclear charge
α	ratomparam(24)	constants, dependent on atom type
β	ratomparam(27)	
p_{elho}	vpar(27)	electron-hole interaction
$\operatorname{Tap}(R_{ij}, R_{cut,e})$	$R_{cut,e} = \text{vpar}(41)$	taper correction with a separate radius

Electron - electron interactions are treated through Coulomb and short-range Gaussian repulsion function. Exponential function is used to determine number of electrons in the host atom, thus electron can virtually split itself among its neighbouring atoms.

$$n_{el,ij} = \frac{\exp(-p_{val} \cdot R_{ij}^2)}{\text{electrons}} - \text{fraction of the } j\text{-th electron on the } i\text{-th atom}$$

$$n_{el,i} = \sum_{j=1} n_{el,ij} \text{ number of electrons on the } i\text{-th atom}$$

Value	Code	Description
$n_{el,ij}$	eshare	fraction of the j -th electron on the i -th atom
$n_{el,i}$		number of electrons on the i -th atom
R_{ij}	nvlr(:)	distance between the atom-center and the electron/hole
p_{val}	vpar(37)	general parameter in the force field

Changes in over- and under-coordination energy terms are minimal: Δ_i^{xel} is used instead of Δ_i in formulas (11a)-(11b) and (12).

$$\Delta_i^{xel} = \Delta_i \cdot \exp\left(-p_i^{xel2} \cdot n_{el,i} \cdot \frac{\sum_{j=1}^{neighbours(i)} BO_{ij} \cdot p_{ij}^{xel1}}{\sum_{j=1}^{neighbours(i)} BO_{ij}}\right) \quad (27)$$

Variable valency will be extended to the 3- and 4-body terms: valence angle and torsion energy expression.

Value	Code	Description
Δ_i^{xel}	deltaXel	corrected degree of over-coordination
Δ_i	delta	degree of over-coordination
p_{ij}^{xel1}	rbondparam(16)	bond type parameter, for adjusting number of electrons available to host atom
p_i^{xel2}	ratomparam(13)	atom type parameter
$n_{el,i}$	nel(i)	number of electrons on the i - th
$\sum_{j=1}^{neighbours(i)} BO_{ij} \cdot p_{ij}^{xel1}$	sumpB0(i)	sum of scaled bond orders of atom i
$\sum_{j=1}^{neighbours(i)} BO_{ij}$	abo(i)	sum of bond orders of atom i